Universidad Nacional de Luján

Trabajo Final de Licenciatura en Sistemas de Información

Recuperación de Información Distribuida en Dispositivos Móviles

Autor: David Conde
Director: Mg. Gabriel H. Tolosa

2012

Agradecimientos

A mis padres, quienes me formaron como persona y apoyan incondicionalmente en el camino de la vida que he elegido. A mis hermanas. que están conmigo siempre y que me han ayudando inmensamente, junto con toda mi familia, durante mi formación profesional. A mi novia, quien me ha apoyado durante el desarrollo de este trabajo, en el día a día, a pesar de los sacrificios que que se precisaron para su realización.

A Gabriel Tolosa, quien con su demostración de dedicación y profesionalismo, fue el que más ha influenciado en mi formación académica de la cual me siento orgulloso. A quien agradezco infinitamente por su ayuda y orientación, que me han posibilitado la realización el de esta carrera universitaria y, en instancias finales, el desarrollo de este trabajo.

A Juan Manuel Fernandez y Juan Ignacio Fernandez, mis compañeros durante el curso de esta carrera universitaria, con los que he compartido tantos momentos y de los que he recibido gran apoyo.

A Diego, a quien dedico este trabajo y toda aquella acción que salga de mi corazón. Que tu luz siga iluminando el camino de todos aquellos que te queremos y recordamos.

Resumen

En el transcurso de los últimos años la información ha tomado un valor muy importante para el hombre y la sociedad. El gran desarrollo de la tecnología ha intensificado este fenómeno generando un escenario económico, social y cultural que está altamente vinculado con las tecnologías de la información.

La información más valiosa precisa ser accedida desde distintos medios y formas. Dicha tarea no se presenta para nada fácil teniendo en cuenta los volúmenes y cantidades que se encuentran disponibles en la actualidad. La rama de investigación de la Ciencias de Computación denominada Recuperación de Información (*Information Retrieval*) aborda esta problemática brindando técnicas y soluciones que se aplican mediante software informático.

Tal software tradicionalmente ha sido diseñado para su utilización en equipos informáticos como servidores y computadoras de escritorio. En este trabajo se estudió la posibilidad de utilizar aplicaciones de estas características en dispositivos móviles presentes en el mercado actual. Dichos aparatos están acaparando gran atención en el público y se perfilan para ser en el futuro próxima la herramienta predilecta para el acceso a la información.

Por esto mismo se considera muy importante el análisis y evaluación de funcionalidades de Recuperación de Información en dispositivos de estas características. Dicha tarea se realizó en este escrito desde dos enfoques: el rendimiento de RI de un dispositivo móvil por un lado, y el de un conjunto de estos mismos actuando de manera distribuida por el otro.

Índice de contenidos

Índice de tablas

Índice de gráficos

1 Introducción

En la actualidad, y ya desde fines del siglo 20, la sociedad se encuentra en la denominada “Era de la información” [Lallana, 2003]. En dicho contexto los marcos económico, político y social están altamente determinados por las tecnologías de información y comunicación. El manejo de información en formato digital permite que ésta sea acumulada y generada de una manera más voluminosa y precisa, en comparación al papel, para ser también distribuida instantáneamente a una audiencia . Esto posibilita que los receptores puedan utilizarla para sus propios propósitos con el objeto de generar nuevas ideas que luego serán compartidas y distribuidas por el mismo medio. En consecuencia se genera un impacto en valores económicos, culturales y sociales, ya que depende del acceso y adopción de este nuevo esquema el hacer más o menos competitivo un negocio o determinar la inclusión de un individuo en la sociedad.

Para lograr el acceso a la información a través de las fuentes que la brindan y compartir la propia con otros usuarios, es necesario utilizar una herramienta de soporte la cual ofrezca las funcionalidades básicas para satisfacer dicha necesidad. Tradicionalmente, al hablar de tal herramienta, hacemos referencia a una computadora de escritorio con un sistema operativo tradicional y una conexión a internet. Ésta es, en la actualidad, la entidad primaria para acceso, procesamiento y distribución de la información, utilizada por la mayoría de los usuarios.

La revolución informática, hoy en día, se encuentra en una instancia en la que genera en las personas la necesidad de encontrarse informadas y conectadas prácticamente todo el tiempo. En consecuencia han empezado a tornarse obsoletas las tecnologías que primariamente eran utilizadas debido a su carencia en cuanto a portabilidad impulsando el desarrollo y despliegue de unas nuevas. Es por eso que actualmente se percibe una etapa de cambio, en la cual las tecnologías de dispositivos móviles y de conexiones de datos inalámbricas han presentado un enorme avance y están acaparando gran atención ya que obtuvieron muy buenas respuestas por parte de los usuarios, que son cada vez más numerosos.

El desarrollo más evidente sea quizás el de los teléfonos celulares del tipo *smartphones* los

cuales han presentado una gran evolución en los últimos años como indica [Li et al.,2010], donde se evidencia que los modelos que se exponen en el mercado aumentan al doble su capacidad de procesamiento, memoria RAM y almacenamiento todos los años ajustándose de manera similar a la ley de Moore [Moore, 1965], la cual fue formulada para computadoras y que explica que el poder de cómputo de éstas últimas — medido como el número de semi-conductores por circuito integrado — se duplica cada año reflejando un crecimiento exponencial. Dicha ley dejó de ser válida hace unos años y se tornó obsoleta por restricciones físicas que implica la miniaturización como lo demuestra [Laszlo, 2002], sin embargo el crecimiento en poder de cómputo de los teléfonos móviles se asemeja a la evolución de las computadoras al momento de ser la ley de Moore válida. La gran ventaja de estos dispositivos móviles frente a los equipos informáticos convencionales es la portabilidad, aunque dicha característica trae como desventaja la disminución en tamaño de interfaces gráficas al usuario y la dependencia de baterías eléctricas de un limitado tiempo de carga. Las capacidades de procesamiento y almacenamiento de los móviles son menores en comparación a las PC u otros dispositivos superiores, sin embargo, se puede decir que son suficientes para muchas de las necesidades de un usuario común, ofreciendo un amplio espectro de funcionalidades que en el pasado sólo eran brindadas por una computadora de escritorio o laptop.

Debido a esto, existe una tendencia creciente a que las personas utilicen como herramienta primaria de acceso y obtención de información estos dispositivos móviles, ya que por su característica de portabilidad posibilitan la realización de esta operación prácticamente en cualquier momento de su vida cotidiana. Es por esto que [Claburn, 2009] estima que el futuro de las computadoras se encuentra en los dispositivos móviles inteligentes y los centros de datos, evidenciando que para los próximos años los teléfonos móviles y otros dispositivos similares van a enviar y recibir 14 veces más datos con respecto al 2008, y de los cuales la mayoría de éstos serán provenientes del acceso a Internet. En otro interesante estudio [Anderson et al., 2008] se indica que para 2020 los dispositivos móviles van a sobrepasar a las computadoras convencionales como la herramienta principal que poseerá un usuario para acceder a Internet. Siguiendo la misma línea, el reporte [Ericsson, 2011] indica en sus estadísticas presentadas del año 2010 que los dispositivos móviles *smartphone* están comenzando a sobrepasar a los *laptop* en lo que hace a uso diario para acceso a internet tal como se muestra en el *Gráfico 1* con datos obtenidos de Estados Unidos y Suecia. Por otra parte, [Comscore, 2011] profundiza sobre el uso de de estos dispositivos indicando que la obtención de información es la cuarta actividad más realizada por los usuarios de *smartphones* como se puede apreciar en el *Gráfico 2.*

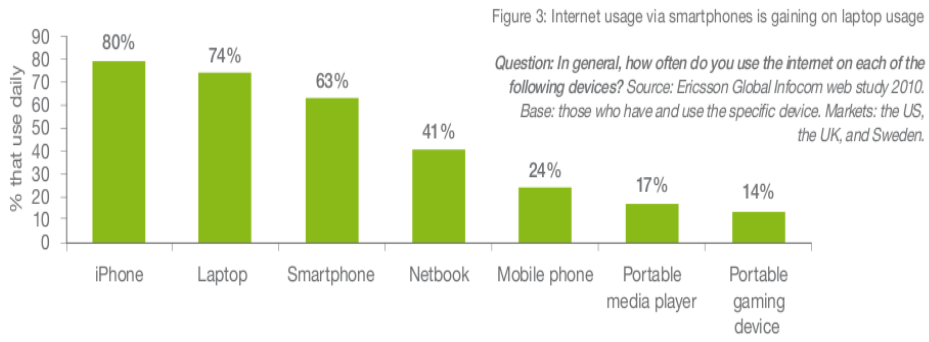


Gráfico 1: Acceso a internet mediante dispositivos [Ericsson, 2011]

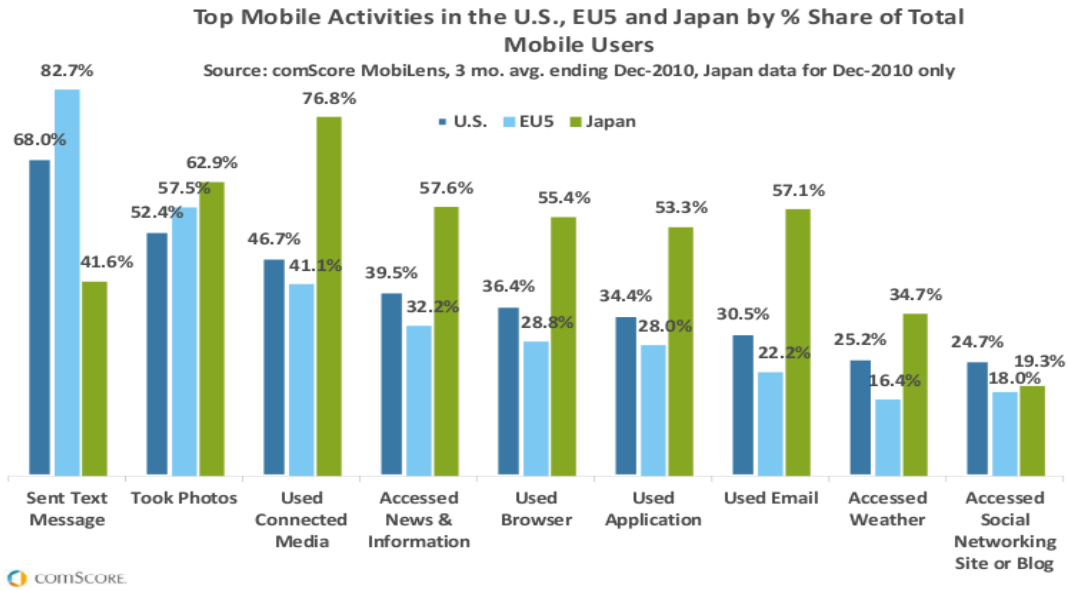


Gráfico 2: Actividades más realizadas mediante dispositivos móviles [Comscore, 2011]

La información que los usuarios usualmente consultan y comparten es, por lo general, de formato textual y de un carácter semi o no estructurado. El estudio y desarrollo de técnicas para la implementación de sistemas que satisfacen necesidades de información representada en medios de este tipo corresponde a la rama de Ciencias de la Computación llamada Recuperación de Información (Information Retrieval)[Manning et al., 2008] [Baeza et al., 1999][Ingwersen, 2002] [Singhal , 2001]. Dichas técnicas han sido fuertemente evaluadas e implementadas sobre equipos informáticos convencionales y mucho tiempo antes de las nuevas tecnologías mencionadas. La aplicación de RI más paradigmática en la actualidad son los motores de búsqueda[1]. Otros tipos de aplicaciones también conocidas son las de Clasificación, Agrupamiento, Reducción de texto, Búsqueda de documentos y Filtrado–Ruteo[Tolosa et al., 2008]. En la actualidad, existe una diversa cantidad de aplicaciones de estas características — con *background* de RI — disponibles para los

1 Sistema informático que busca archivos dentro de servidores Web de Internet.

usuarios[2].

Los modelos de dispositivos móviles del tipo *smartphones* y *tablets*[3] presentes actualmente en el mercado poseen tales capacidades que les permiten realizar tareas que requieren un considerable poder de cómputo y procesamiento local como lo son la captura de vídeo en alta definición. Sin embargo, casi la totalidad de las funcionalidades correspondientes al acceso y recuperación de información son delegadas a centros de datos y servidores en internet, siguiendo el paradigma de *cloud computing* [Vouk, 2008] que es tendencia en los últimos años.

2 Sistemas de recuperación de información: http://www.csc.lsu.edu/~kraft/retrieval.html

3 Computadora de propósito general contenida en un único panel. Su principal característica es el uso de una *touch screen* como principal dispositivo de entrada.

2 Objetivos

En este trabajo se pretende analizar las posibilidades de ciertos dispositivos móviles de uso cotidiano para tareas de organización y búsqueda eficiente de información, y evaluar si las técnicas clásicas del área de Recuperación de Información (RI) son lo suficientemente eficientes. Por contrario, de acuerdo a [Flora et al., 2010], nuevas demandas requieren nuevas tecnologías para la representación, la modelización, indexación y recuperación ya que no se cuenta con las mismas capacidades de cómputo que las computadoras de escritorio. También se procura investigar qué volumen de datos es posible manejar y qué carga de trabajo pueden realizar tales dispositivos manteniendo el resto de las funcionalidades disponibles.

En resumen, debido a que el tratamiento del acceso a la información en grandes colecciones de textos es uno de los objetos de estudio de la Recuperación de Información, y ya que los dispositivos portátiles tienden a ser la herramienta tecnológica más utilizada en actualidad, la primera cuestión consiste en determinar si es factible tener un sistema de recuperación de información clásico en un dispositivo móvil y, en caso afirmativo, cuáles son las prestaciones esperadas. Además, surge la necesidad de evaluar los modelos de representación, indexación y recuperación clásicos [Baeza et al., 1999][Manning et al., 2008] . Para responder a este interrogante, se llevó a cabo en este trabajo el desarrollo y evaluación de un prototipo de sistema de recuperación de información clásico desplegado en una serie de dispositivos móviles de la categoría *smartphones* de distintas características.

Debido a que los dispositivos *smartphones* están diseñados para estar conectados de forma continua a redes datos, se plantea como otro objetivo la implementación y evaluación de una Sistema de Recuperación de Información Distribuido[Callan et al., 2000] el cual realice su funcionamiento mediante la coordinación y cooperación de un conjunto de dispositivos móviles que ejecuten el prototipo nombrado en el párrafo anterior. De esta forma se pretende estudiar el comportamiento de los móviles en un entorno distribuido, que actúa como un único Sistema de Recuperación de Información.

3 Escenario Tecnológico

Los dispositivos móviles *smartphones*[4] son una nueva generación de tecnología de teléfonos celulares que tienen características similares a una computadora portátil. Han adquirido una gran respuesta por el público en el mercado y hoy en día son una gran herramienta para el acceso a la información y la navegación por internet. Con respecto a este fenómeno, se pueden nombrar cuatro factores que están presentes en el marco actual que posibilitaron la concepción del mismo y que forman el fundamento del corriente trabajo: Primero, la considerable evolución de los dispositivos móviles en lo referido a poder de procesamiento y funcionalidades ofrecidas. Segundo, la gran acumulación de información disponible para los usuarios, con una tendencia a continuar en un marcado crecimiento. Tercero, el desarrollo en tecnologías de comunicación inalámbricas. Y cuarto, el surgimiento se los sistemas operativos móviles y la competencia existente entre las empresas de software que los desarrollan con el objetivo de dominar el mercado.

3.1 Evolución de smartphones

El término *smartphone* se refiere a un determinado tipo de teléfono móvil el cual ofrece más características computacionales y de conectividad en comparación a uno convencional. Posee un sistema operativo móvil de propósito general el cual brinda un amplio espectro de funcionalidades, entre las cuales se encuentran la telefonía, agenda personal, envío y recepción de *e-mail*, navegación web, conectividad con redes sociales, reproducción de música y vídeo. El SO también posibilita la instalación y uso de una gran variedad de aplicación brindadas por terceros.

En lo referido a hardware, un *smartphone* posee una arquitectura similar a la de una computadora disponiendo de un microprocesador, memoria primaria — RAM[5] —,

4 Teléfono móvil con capacidades que lo acercan a un pequeño ordenador portátil.
5 Es un tipo de memoria de computadora volátil que provee acceso directo a cualquier ubicación única (un byte) y es accedida por el procesador para leer y escribir datos e instrucciones.

almacenamiento permanente — por lo general una memoria flash[6] y/o memoria del tipo SD[7] — y, como se trata de un dispositivo portátil, una batería como fuente de alimentación. Además, de estos componentes, los modelos existentes en la actualidad poseen también cámara, pantalla táctil, sensores — como el GPS[8] y detectores de movimiento —, WiFi[9] y Bluetooth[10]. Estéticamente son similares a sus antecesores, los "PDA y PocketPC"[11], de un tamaño pequeño característico de un accesorio de mano.

Hoy en día el *smartphone* está marcando una gran presencia en el mercado y por ende en la sociedad. Es por eso que grandes compañías están en constante competencia por ser líder en venta de estos equipos, lanzando en cortos períodos de tiempo nuevos modelos con más prestaciones y capacidades computacionales en comparación a sus antecesores. En relación a esto [Li et al.,2010] explica que la tendencia actual de dispositivos móviles cada vez más potentes ha transformado el mercado de teléfonos celulares convencionales al de teléfonos inteligentes. Las *grandes* compañías de informática están empujando estas tecnologías tan fuertemente que nuevos *smartphones* con mayor velocidad y más características están saliendo a la venta cada año. Los teléfonos inteligentes ahora son equipados con procesadores de alta gama, mayores espacio de almacenamiento, diversos sensores, pantallas táctiles y diferentes capacidades en conexiones de red. La superioridad de estos dispositivos indica que su desarrollo va a continuar en un rápido aumento y con tendencia al dominio del mercado de dispositivos móviles.

En las gráficas mostradas a continuación se puede apreciar la evolución de los dispositivos *smarthphone* a partir tres de los componentes más importantes que los conforman. Los datos son obtenidos de distintos modelos lanzados al mercado en los últimos años. Por un lado se presenta la frecuencia del microprocesador y la capacidad de la memoria RAM, dos componentes vitales de los cuales definen estrictamente el poder computacional del dispositivo. Por otro lado, las baterías que deben acompañar el incremento de capacidad de los dos primeros ya que el mayor poder de cómputo precisa de un mayor consumo eléctrico.

6 Es un chip de almacenamiento de computadora no volátil que puede ser eléctricamente borrad y reprogramado.
7 Modelo de memoria flash, no volátil, utilizada en diversos dispositivos tecnologicos.
8 Global Positioning System: sistema de posicionamiento global
9 (Wireless Fidelity) Es un conjunto de estándares para redes inalámbricas basado en las especificaciones IEEE 802.11.
10 Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPANs).
11 Es un computador de mano originalmente diseñado como agenda electrónica con un sistema de reconocimiento de escritura.

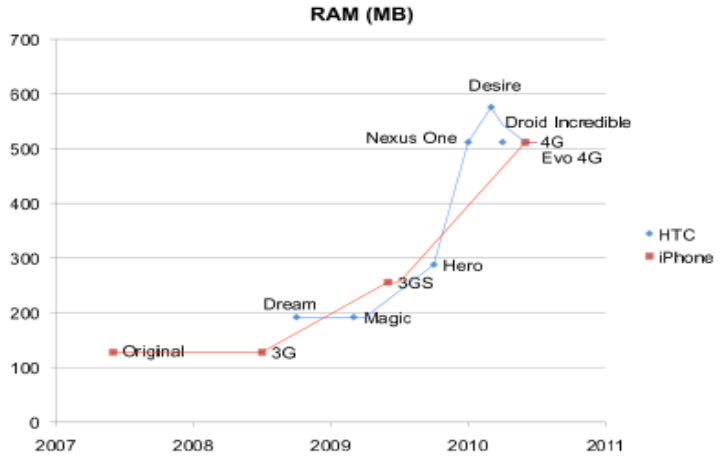


Gráfico 3: Capacidad de Memoria RAM de smartphone para distintas generaciones de HTC y Apple [Li et al.,2010]

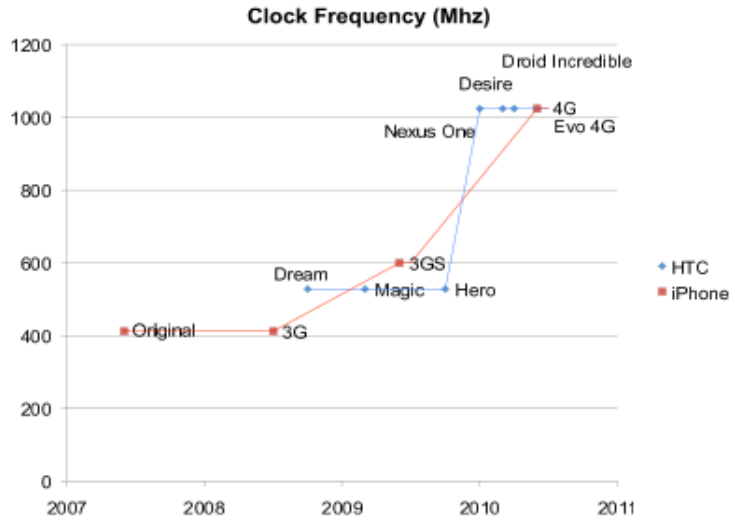


Gráfico 4: Frecuencia de microprocesador de smartphones de distintas generaciones de HTC y Apple [Li et al.,2010]

Es evidente que la velocidad de desarrollo que presentan los modelos de *smartphones* que son lanzados al mercado es frenética con un crecimiento muy alto. Desde este aspecto podemos darnos cuenta la importancia que tienen dichos dispositivos en el marco actual y su tendencia a futuro, ya que se está llevando al *smartphone* a ser una herramienta digital con potencial para llevar a cabo todo tipo de funcionalidades para los usuarios.

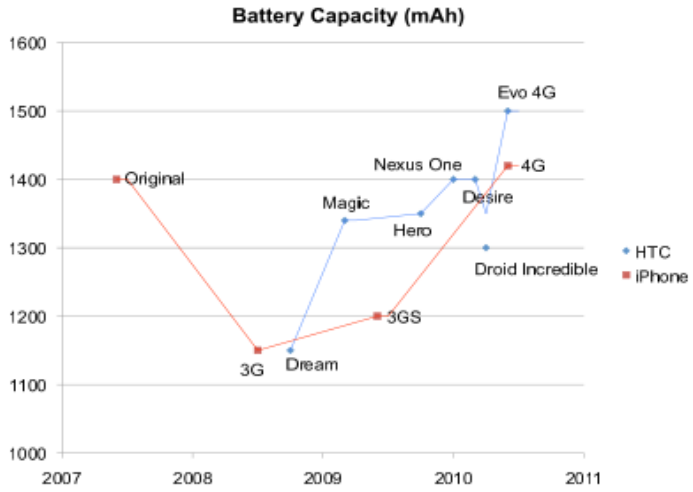


Gráfico 5: Capacidad de batería de smartphone para distintas generaciones de HTC y Apple [Li et al.,2010]

En consecuencia, esto implica un incremento en la aceptación de los nuevos teléfonos móviles por parte de las personas, evidenciando un gran aumento de usuarios de *smartphones*. Según [Nielsen, 2011] ya en el tercer cuarto del año 2010, 28% de los suscritos a planes de telefonía móvil en los Estados Unidos correspondían a usuarios de *smartphones* y 41% de las personas que se suscribieron en ese período optaron por este tipo de dispositivos en comparación de un 35% del período anterior. Siguiendo la misma línea en un análisis un poco más actual [Comscore, 2011b] indica que en el año 2011 la adquisición de *smartphones* se incrementó en un 53% llegando a los 78,5 millones de usuarios en Estados Unidos.

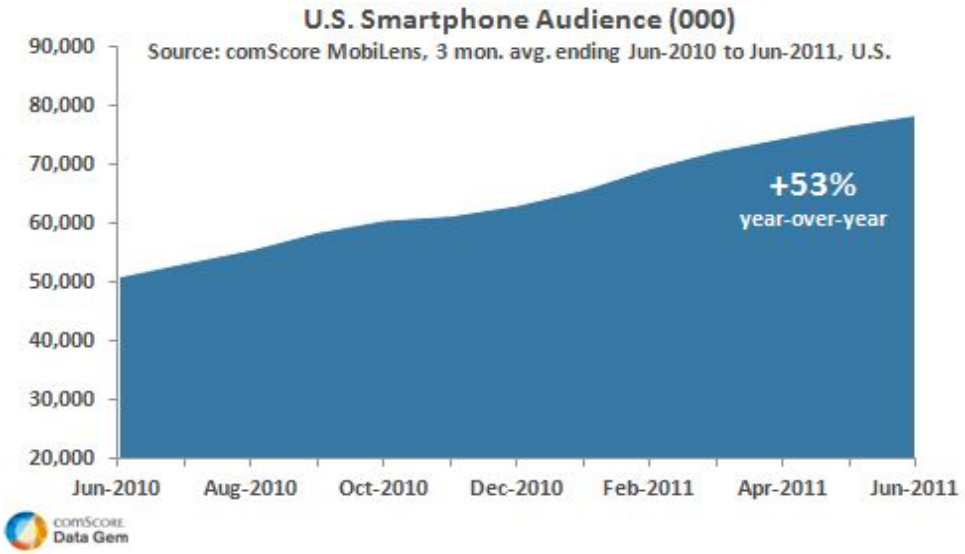


Gráfico 6: Adopción de smartphone en Estados Unidos [Comscore, 2011b]

Con las crecientes capacidades de procesamiento, es esperable que los datos que consume y administra un dispositivo móvil *smartphone* promedio también presenten un incremento. Según [Kellogg, 2011] en el último año la cantidad de datos que un usuario consume de la web por mes a través de su *smartphone* ha aumentado en un 89%, de 230 Mb a 435 Mb. Para aquellos usuarios que más consumen datos, el aumento ha sido de un marcado 109%. La principal razón por las que sucede este fenómeno es la utilización de las diversas aplicaciones a las que el sistema operativo móvil brinda plataforma.

Dado que los celulares del tipo *smartphones* han evolucionado marcadamente y poseen cada vez más características atrayentes para potenciales usuarios se estima que en años futuros éstos sean los predominantes en el mercado de los teléfonos móviles.

3.2 Crecimiento de cantidad de información disponible

Es evidente también la existencia de un crecimiento notable de información en formato digital, tal como evidencia el estudio [Gantz et al., 2011], en el cual explica que la cantidad de información creada, capturada y replicada globalmente, la cual denomina en el artículo como "universo digital", se incrementó en el año 2011 nueve veces en un lapso de 5 años con respecto a un primera estimación publicada en otro estudio anterior [Gantz et al., 2007]. En dichas publicaciones se dedujo un valor de 161 *exabytes*[12] de información digital existente para la primera y de 1,8 *zetabytes*[13] en la más reciente. Los datos obtenidos son estimaciones que, si bien es probable que no se aproximen al valor real que representa la cantidad información digital total existente ya que es una variable imposible de calcular de una manera exacta, no dejan de ser valiosos para evidenciar ciertas cuestiones. Dos aspectos interesantes mostrados en [Gantz et al., 2007] son: primero que los usuarios comunes son productores de un 70% del total de la información digital generada, y segundo que un 95% de la totalidad de la misma es de carácter no estructurado. Dada esta inmensa cantidad información disponible y no estructurada, es evidente que la obtención de ciertos ítems de valor que satisfagan la necesidad de un individuo u organización se torna un gran desafío, ya que para este contexto donde existe un volumen de una magnitud astronómica poder encontrar aquella que es relevante necesita de un gran esfuerzo por parte tecnologías de soporte para cumplir dicho objetivo.

La evolución de Internet es también un indicador del crecimiento de cantidad de información disponible, debido a que es el principal medio de obtención de la misma tal cual indica el artículo [Reuters, 2009]. Desde otro enfoque, se puede asociar el crecimiento de la información disponible con la cantidad de usuarios conectados a la red. En [IWS, 2011] se afirma que en los últimos 10 años el número de usuarios conectados a la red aumentó en un 480.4% a nivel mundial existiendo en el momento de dicho estudio un número de 2,095,006,005 personas en todo el globo con acceso a la web. En el *Gráfico 7* se muestra el número de usuarios de internet a nivel mundial en un determinado período de tiempo, discriminando por países desarrollados y en vías de desarrollo. Se indica que en el año 2010 aproximadamente 69 de cada 100 personas en los países desarrollados son usuarios activos de internet con una tendencia creciente.

Muchas de estas personas interconectadas producen contenidos informativos, los cuales

12 2³º gigabyte
13 1024 exabyte

comparten y transmiten a través de la red. Dichos contenidos son aprovechados por otros individuos que los utilizan para generar unos nuevos colaborando, de esta manera, al gran crecimiento en cantidad de información disponible en la web. Este factor trae aparejado un relativo aumento del tráfico de datos en la red, tal como explica [Coffman et al., 2001] quien afirma que en estadísticas presentadas en dicho trabajo se evidencia que el tráfico en el *backbone*[14] de Internet se duplica aproximadamente en cada año.

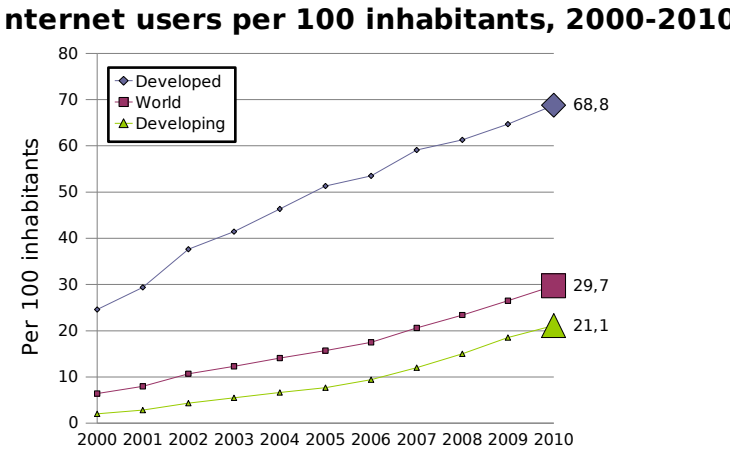


Gráfico 7: Usuarios activos de internet 2000-2010 [ITU, 2011]

Un buen indicador para evidenciar el crecimiento de la información disponible en internet es el tamaño de la web visible mediante la estimación de los tamaños de los índices de los buscadores web más utilizados. Para este caso, en el *Gráfico 8*, se indica el tamaño estimado del índice del conocido buscador *Google*[15]. Se puede apreciar que tiende a un marcado crecimiento en el último tiempo. Claramente esto no es un parámetro de todo el tamaño de la web, ya que existe una gran parte denominada como "web oculta"[16] la cual no se encuentra indexada por los buscadores y es imposible el acceso a dichos recursos mediante tales herramientas. Es por eso que el tamaño de Internet es una variable imposible de indicar y comprobar de forma exacta y los valores que se han logrado hasta la actualidad no son más que estimaciones. No obstante, no deja de ser un buen dato para comprobar que evidentemente la información disponible para los usuarios en internet se encuentra en un importante incremento.

A pesar que el valor de la cantidad de información digital disponible es algo muy difícil de

14 Representa a los principales routers de datos en el núcleo de Internet.
15 http://www.google.com
16 Se refiere al contenido dentro de la red de redes que no se encuentra indexado por los motores de búsqueda.

estimar, está claro que se encuentra accesible un cúmulo de información digital de una gran magnitud, de tal manera que la búsqueda y administración de la misma es una tarea imposible sin la utilización de un software de soporte.

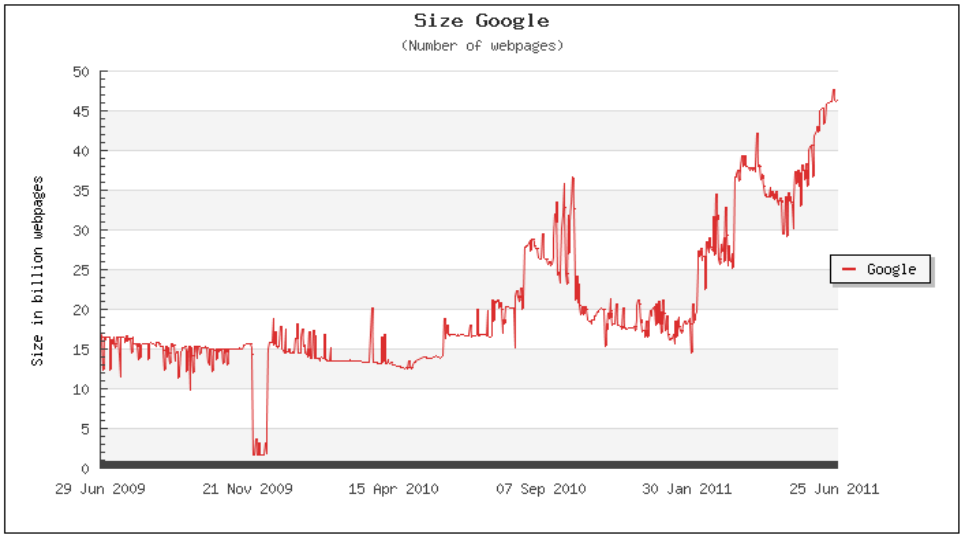


Gráfico 8: El tamaño de la web: tamaño estimado de el índice de Google [TSOTWWW, 2011]

3.3 Desarrollo de tecnologías de conectividad inalámbrica

Otro parámetro de gran influencia en esta revolución informática y que va de la mano tanto con la evolución de *smartphones* como el crecimiento de información disponible es el desarrollo en las tecnologías de conectividad inalámbrica. Sin dicho factor no sería posible la portabilidad, rasgo principal de los tipos de dispositivos que son objeto de estudio del presente trabajo, y el desarrollo de aplicaciones distribuidas, que implican una importante transferencia de datos.

El *smartphone* promedio tiene soporte sobre varias tecnologías de conexión inalámbrica, las cuales poseen distintas características de alcance y capacidad de transmisión. Las tres principales son:

- Conexión de red inalámbrica de área personal (WPAN[17]). El protocolo más

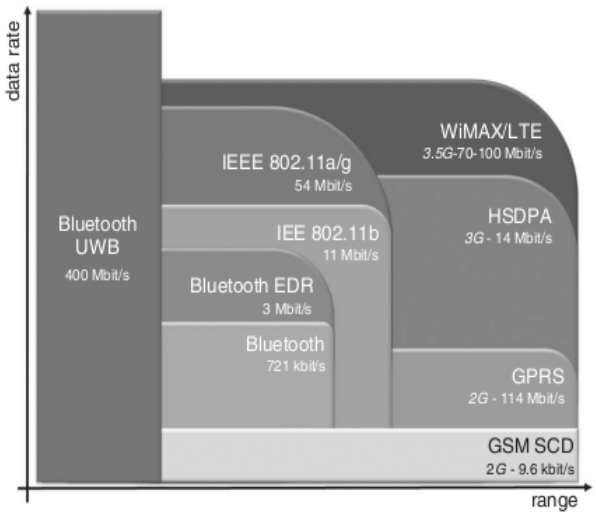
17 Wireless Personal Area Network

utilizado es el denominado *Bluetooth* que se caracteriza por posibilitar una comunicación directa a una distancia cercana entre los dispositivos que transmiten los datos. *Bluetooth* es una tecnología de radio-frecuencia que opera en la banda de 2.4 Ghz, es de corto alcance y en un principio fue diseñada para el reemplazo del cableado para periféricos de computadoras de escritorio como mouse, teclado o impresoras. Posteriormente, con el desarrollo de nuevas versiones, se obtuvo una mayor potencia demostrando ser muy útil para la transferencia de datos. En lo que hace a potencia podemos subdividir a la tecnología Bluetooth en tres clases: la clase 1 que posee un alcance de al menos 100 metros y las clases 2 y 3 con un alcance de 10 metros. Los teléfonos móviles por lo general son clase 2 mientras que la clase 1 corresponde a los puntos de acceso.

- Conexión de red de área local (WLAN[18]), siendo el protocolo más popular el estándar IEEE 802.11[19]. Utilizado para redes de datos de banda ancha para computadoras, permitiendo ahora a los *smartphones* formar parte de las mismas. Opera en la frecuencia 2.4 Ghz la capacidad transferencia de datos es de 54 Mbps, para su versión más utilizada 802.11g.

- Conexión de red inalámbrica de área extensa (WWAN[20]), la cual permite transmisión de voz y de datos en un largo alcance. Actualmente este tipo de conexión se encuentra en la denominada tercera generación (3G[21]), siendo HSDPA[22] — también se denominará en adelante como HSPA — la tecnología principal. Los móviles 3G, de todas formas, mantiene compatibilidad con las tecnologías de segunda generación. Se estima que en el corto plazo se migrará a la emergente cuarta generación de telefonía móvil la cual presentará una notable mejora en alcance y capacidad de transmisión de datos.

18 Wireless Local Area Network
19 http://www.ieee802.org/11
20 Wireless Wide Area Network
21 3G es la abreviación de tercera-generación de transmisión de voz y datos a través de telefonía móvil.
22 High Speed Downlink Packet Access



Dibujo 1: Tecnologías inalámbricas. Alcance y capacidad de transferencia de datos soportados. [Fitzek et al., 2009]

En este trabajo se hizo hincapié en las tecnologías WLAN y WWAN, redes en las que funciona el prototipo de aplicación desarrollado y para las cuales se focalizó en lo que sigue del apartado. Sobre el uso de estas mismas tecnologías y su marcado crecimiento, es [Horrigan, 2009] quien demuestra que en Estados Unidos más de la mitad de las personas adultas – un 56% – acceden a internet mediante alguna red inalámbrica ya sea mediante un celular, *smartphone*, *laptop* o algún otro dispositivo. Por otro lado, en referencia puntual a WWAN, es [GSA, 2011] quien reporta que alrededor del mundo crecieron un 42,4% las a suscripciones a la red de datos móvil 3G — más específicamente las tecnologías WCDMA y HSPA — como indica el *Gráfico 9*.

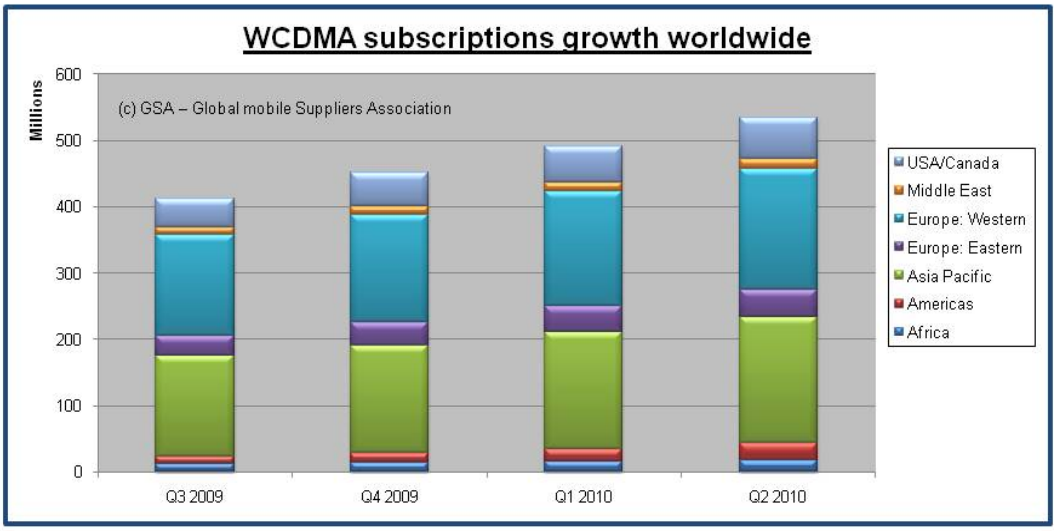


Gráfico 9: Suscripciones mundiales a redes móviles WCDMA/HSPA[GSA, 2011]

A pesar de el gran crecimiento y desarrollo de redes inalámbricas, aún son las conexiones de redes fijas la más utilizadas por un amplio margen, y se estima que en el futuro lo seguirán siendo. El *Gráfico 10* muestra claramente la diferencia entre el uso de estos tipos de conexiones. La banda ancha fija y conexiones de acceso telefónico son calculadas sobre la base de disponibilidad en el hogar, mientras que banda ancha móvil y navegación móvil son calculadas del porcentaje de utilización. Las estadísticas fueron obtenidas de países desarrollados, que son aquellos que tienen mayor penetración de redes móviles, sin embargo se aprecia la predominante presencia de las redes de conexión fija aunque las móviles se encuentran en crecimiento de usuarios y se estima que aumenten las suscripciones en los próximos años.

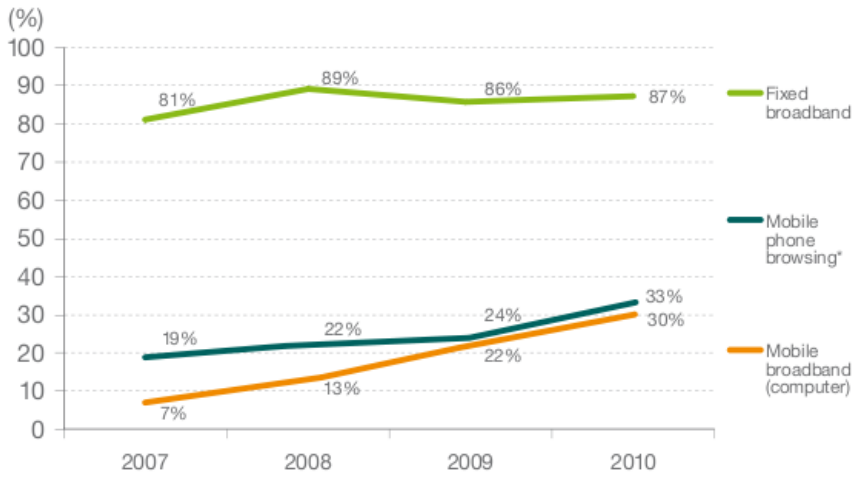


Gráfico 10: Tipos de conexiones [Ericsson, 2011]

Las tecnologías de conectividad móviles e inalámbricas han evolucionado notablemente logrando una capacidad de transferencia de datos similar a las conexiones fijas, de todas formas son estás últimas las predominantes en cuanto a uso en la actualidad. No obstante, los últimos protocolos de tecnologías inalámbricas han presentado grandes mejoras ocasionando el lanzamiento al mercado de nuevos dispositivos compatibles con las mismas impulsando intensamente su utilización en la actualidad.

3.4 Surgimiento de los sistemas operativos móviles

Un sistema operativo móvil es un software que controla un dispositivo móvil. El concepto es similar al de un "sistema operativo estándar para computadoras"[23], a diferencia de que están diseñados con orientación a una mayor simpleza y para operar de forma nativa con tecnologías inalámbricas de banda ancha móvil y de red área local. Los dispositivos que comúnmente son controlados por uno de éstos sistemas son los *smartphones*, PDAs y *tablets*.

Cuando un usuario decide adquirir un *smartphone* no sólo se encuentra obteniendo el dispositivo en sí, sino que además, elige el sistema operativo con el que opera. En el ámbito de los celulares, a diferencia del de computadoras, si el usuario se encuentra insatisfecho con el SO no podrá reemplazarlo por otro distinto. Entonces el SO es de vital importancia ya que es el que debe explotar de forma óptima la pieza de hardware que representa al dispositivo móvil posibilitando, a su vez, el uso de un amplio espectro funcionalidades y aplicaciones que reflejan estilo de vida del usuario.

Existen en la actualidad varias versiones de SO móviles, algunos desarrollados por las mismas empresas fabricantes de dispositivos y otros creados por terceros. Entre los más representativos podemos mencionar: Android OS[24], RIM BlackBerry[25], iOS[26], Symbian[27] y Windows Mobile[28] entre otros. Gracias a la explosión del fenómeno *smartphone* y en base a la elección de los usuarios se ha producido, en el último tiempo, una gran competencia a nivel global entre las empresas involucradas por el liderazgo del universo de SO móviles. Según estadísticas publicadas por [Nielsen, 2011] quien toma la delantera en la actualidad en la competencia de popularidad de sistemas operativos móviles es Android, captando un 40% de los consumidores de smartphones en el mercado de Estados Unidos.

23 Tales como las diferentes distribuciones de Linux, los sistemas Windows de Microsoft o las diferentes versiones de MacOS de Apple entre otros.
24 Propiedad de Google e implementado para productos de diversos fabricantes.
25 Desarrollado por la empresa RIM (http://www.rim.com) para dispositivos del fabricante Blackberry (http://us.blackberry.com).
26 SO desarrollado por la compañía Apple (http://www.apple.com) implementado exclusivamente para sus dispositivos.
27 Operan en dispositivos elaborados por Nokia (http://www.nokia.com), la mismo firma que llevo a cabo su desarrollo.
28 Sistema diseñado por Microsoft (http://www.microsoft.com/windowsphone)

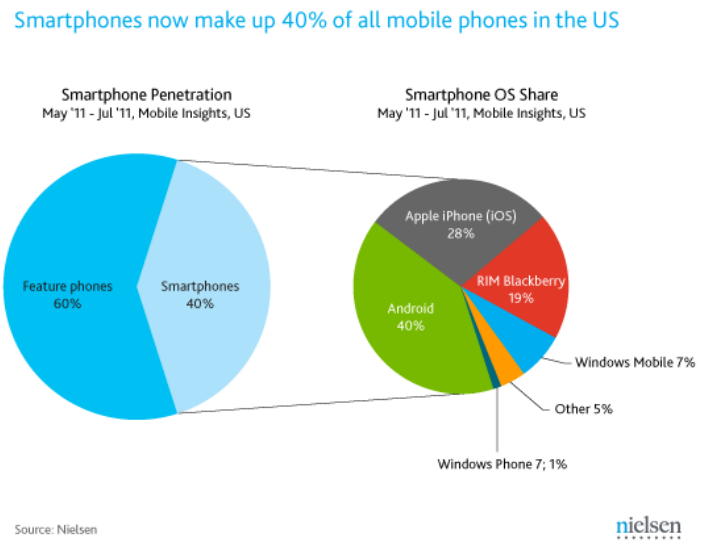


Gráfico 11: Escenario del mercado de smartphones en Estados Unidos [Nielsen, 2011]

Teniendo en cuenta que el mercado estadounidense es de alta importancia ya que suele marcar tendencia global, y debido a ciertos atractivos que brinda la plataforma en comparación a sus competidores como la adaptabilidad a diversos dispositivos y su naturaleza de código abierto, en este trabajo se ha escogido el SO móvil Android como plataforma para la aplicación prototipo. Se considera que dicho sistema es de un gran potencial y se espera, como marcan las tendencias, que sea el sistema operativo móvil más utilizado.

3.4.1 Android

A grandes rasgos, Android es un sistema operativo basado en GNU/Linux[29] para dispositivos móviles desarrollado por Google. Fue presentado el 5 de noviembre de 2007 junto con la fundación Open Handset Alliance[30], un consorcio de 48 compañías de hardware, software y telecomunicaciones comprometidas a la promoción de estándares abiertos para dispositivos móviles.

29 Así se denomina al sistema operativo formado por la combinación del kernel libre llamado Linux junto a las herramientas del proyecto GNU (http://www.gnu.org/gnu/linux-and-gnu.html)
30 http://www.openhandsetalliance.com

Según [Android, 2011] "Android es una pila de software para dispositivos móviles que incluye un sistema operativo, middleware y aplicaciones fundamentales. El SDK de Android[31] proporciona las herramientas y APIs[32] necesarias para comenzar a desarrollar aplicaciones en la plataforma Android usando el lenguaje de programación Java[33]".

Android fue diseñado desde cero para permitir a los desarrolladores crear aplicaciones móviles que aprovechen al máximo todo que un *smartphone* tiene para ofrecer [OHA, 2011]. Fue construido para ser verdaderamente libre. La mayoría del código fuente se encuentra liberado al público bajo la segunda versión de la licencia Apache[34], una licencia de software libre y código abierto.

3.4.1.1 Arquitectura

Android es una pila de software para dispositivos móviles. Esto significa que entre sus mayores prioridades se encuentran la preservación de la batería de energía y la administración eficiente de recursos de memoria. La arquitectura del sistema presenta cinco capas bien definidas:

- Android posee un kernel de linux en su versión 2.6 para procesadores ARM[35] que ofrece los servicios más básicos del sistema. El núcleo actúa como capa de abstracción entre el hardware y la pila de software, siendo la base para el resto de las capas. Linux es una tecnología bien probada que es altamente confiable y la familia de procesadores ARM es conocida por su alta performance en sistemas de requerimientos de bajo poder de cómputo.

- Se incluyen un conjunto de librerías C/C++[36] utilizadas por los diversos componentes del sistema. Estas capacidades están expuestas a los desarrolladores a través del marco de aplicación para Android. Las librerías ofrecen código de bajo nivel reutilizable y compartible para las funciones básicas, de presentación de gráficos, soporte de componentes para navegar por la Web, funcionalidad de base de datos SQL y la

31 http://developer.android.com/sdk
32 Application Programming Interface. Es la interfaz entre distintos softwares facilitando su interacción.
33 Lenguaje de programación del paradigma orientado a objetos (http://java.com),
34 Esta licencia permite innovar en la plataforma sin obligación de contribuir esas innovaciones a la comunidad.
35 Advanced RISC Machine. Procesador simple, de conjunto de instrucciones reducidas, ideal para aplicaciones de bajo consumo de energía.
36 Lenguaje de programación de propósito general.

funcionalidad de la biblioteca estándar C que se necesita en un sistema Linux. Algunas de las librerías incluidas son:

- Librería C: una implementación derivada de BSD de la librería C estándar de sistema (libc), modificada para dispositivos embebidos basados en Linux.

- Librerías de medios: basadas en OpenCORE[37] y PacketVideo[38], soportan la reproducción y grabación de muchos formatos populares de audio y vídeo, así como archivos de imágenes.

- Administrador de pantalla: gestiona el acceso de muchas aplicaciones al subsistema de visualización y a sus compuestos de capas gráficas 2D y 3D.

- LibWebCore: un motor de navegador web moderno.

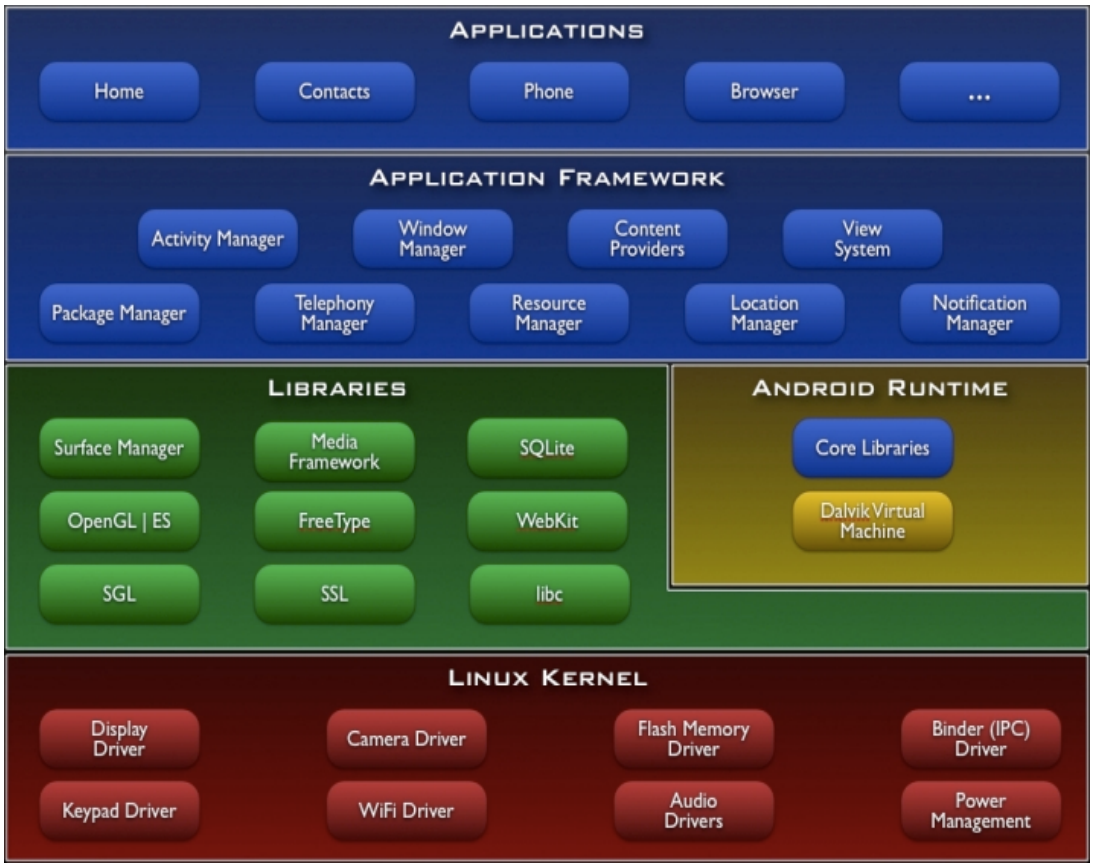
- SGL: el motor de base de gráficos 2D.

- Librerías 3D: una implementación basada en OpenGL[39]

- FreeType: renderizador de mapa de bits y fuentes vectorizadas.

- SQLite: un motor de base de datos relacional potente y ligero disponible para todas las aplicaciones.

37 Producto de código abierto disponible comercialmente en combinación con políticas de código no abierto.
38 Software embebido para servicios multimedia (http://www.packetvideo.com/).
39 API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D (http://www.opengl.org/).



Dibujo 2: Arquitectura de Android [Android, 2011]

- El entorno de ejecución de Android esta formado por un conjunto de librerías centrales que proporcionan la mayor parte de la funcionalidad disponible al núcleo del lenguaje de programación Java. Posee un intérprete de código objeto, o máquina virtual Java, llamado Dalvik que difiere un poco del nativo del lenguaje, mejorando aspectos de seguridad y de preservación de energía. Además, a diferencia de este último, cada aplicación que se está ejecutando lo hace en su propio proceso con un copia del intérprete separando, de esta manera, los procesos estrictamente para lograr mayor seguridad y estabilidad. La máquina virtual Dalvik se apoya en el kernel linux para funcionalidad de bajo nivel como *threading*[40] y administración de memoria.

- El *framework*[41] de Android permite reutilizar y reemplazar los componentes a medida que resulte conveniente. Son clases Java de alto nivel que están estrechamente integradas los componentes que definen la API de Android.

40 Manejo de hilos de ejecución o subprocesos que son planificados por el sistema operativo.
41 Estructuras de soporte conceptual o de software concreto que permiten a un proyecto de software ser más fácilmente organizado y desarrollado

- Las aplicaciones Android están escritas en lenguaje Java. El sistema en sí provee algunas básicas, y además existe la posibilidad de descargar en instalar de forma inmediata un gran cúmulo de otras desde el Android Market[42].

La arquitectura en capas permite que Android pueda ser ejecutado desde una diversa cantidad de dispositivos de distintas características. Tal cual expone [Jantscher et al., 2009]: "Android no es una sola pieza de hardware, es una solución completa, de extremo a extremo de software de plataforma que puede ser adaptado para trabajar en cualquier número de configuraciones de hardware. Todo está ahí, desde el gestor de arranque hasta las aplicaciones. Con los dispositivos Android lanzados ya en el mercado, se ha demostrado que tiene todo lo necesario para competir realmente en el campo móvil".

3.4.1.2 Framework y SDK

Al proporcionar una plataforma de desarrollo abierto, Android ofrece a los desarrolladores la capacidad de crear aplicaciones innovadoras. Los desarrolladores son libres de tomar ventaja de los dispositivos de hardware, acceder a la información de ubicación, ejecutar servicios en segundo plano, establecer alarmas, añadir las notificaciones de la barra de estado, entre muchas otras cosas más.

Los desarrolladores tienen pleno acceso a la misma API del *framework* utilizado por las aplicaciones centrales. La arquitectura de la aplicación está diseñada para simplificar la reutilización de componentes, y cualquier otra aplicación podrá entonces hacer uso de esas capacidades (sujeto a restricciones de seguridad impuestas por el *framework*).

El Kit de Desarrollo de Software[43] de Android provee todas las herramientas necesarias para el desarrollo de aplicaciones en dicho sistema. Para lograr esto el SDK consta de los siguientes elementos:

- Herramientas de desarrollo

42 Tienda de software en línea para dispositivos Android.
43 Software Development Kit (SDK)

Proveen las funcionalidades que facilitan el armado de aplicaciones, tales como el empaquetado del software en los archivos del tipo *.apk los cuales contienen los recursos y la aplicación en sí. También el SDK provee los medios para transferir estos archivos a los dispositivos Android o al emulador para la instalación del software. Entre otras herramientas de desarrollo se puede nombrar el monitoreo de la aplicación en su entorno de ejecución.

- Emulador: el SDK también incluye un emulador que es capaz de simular casi todas las funcionalidades de un dispositivo Android actual en una computadora convencional. Esto se logra cargando las llamadas imágenes de sistema que representan el SO de Android con la pila de software completa.

- Documentación y código de ejemplo: el framework de desarrollo posee una documentación completa la cual se encuentra escrita muy específicamente describiendo las clases Java reutilizables que ofrece. Además de la documentación, y con el objeto de acelerar el aprendizaje del desarrollador, la SDK también ofrece el código fuente de aplicaciones que sirve como ejemplo de implementación y utilización del framework.

- Integración con IDE[44]: se puede desarrollar aplicaciones Android prácticamente desde cualquier IDE. Google recomienda la utilización de Eclipse[45] junto al *plug-in*[46] ADT (Android Development Tools) el cual permite acceder a todas las características de la SDK nombradas en esta sección.

44 Entorno de desarrollo integrado. Es la herramienta que posee la interfaz y las herramientas integradas para facilitar el desarrollo de software.
45 http://www.eclipse.org
46 Es un complemento. Aplicación que se relaciona con otra ofreciendo a la misma una funcionalidad nueva.

4 Antecedentes

4.1 Recuperación de Información

Es evidente que frente a grandes volúmenes de información la obtención de ciertos ítems particulares que resulten de interés para el usuario se convierte en una tarea muy compleja. Si nos referimos específicamente a la información en formato digital carente de estructura o semi-estructurada, son varios los escenarios que se presentan con esta particularidad, entre los cuales se destaca la Internet. El área de Recuperación de información es una rama de la Ciencias de la Computación que tiene como objeto de estudio la problemática presentada.

Según [Baeza et al., 1999], "la Recuperación de la Información trata con el la representación, organización, almacenamiento y acceso de los ítems de información. La representación y la organización de los ítems información debe proveer al usuario una forma de acceso fácil a la que es de su interes". De forma análoga es definido por [Ingwersen, 2002].

"Recuperación de Información trata con la búsqueda de material — usualmente documentos — de una naturaleza carente de estructura — usualmente texto — que satisface una necesidad de información y proviene de grandes colecciones — usualmente almacenadas en computadoras —" [Manning et al., 2008] .

Siguiendo una línea similar, [Tolosa et al., 2008] define el accionar de la Recuperación de Información como "encontrar y rankear documentos relevantes que satisfagan la necesidad de información de un usuario, expresada en un determinado lenguaje de consulta".

Si bien RI es un área que ha logrado un gran protagonismo debido a la aparición e inmediata evolución de la World Wide Web y las soluciones que se lograron aplicar sobre la misma, la disciplina no empezó con la web, sino que tiene sus comienzos un tiempo atrás en aplicaciones

sobre colecciones de publicaciones científicas. Pero su campo de aplicación y sus líneas de investigación se han ampliado en el último tiempo. Es por eso que [Manning et al., 2008] indica que "hoy en día la investigación de RI incluye modelado, clasificación y caracterización de documentos, arquitecturas de sistemas, interfaces de usuarios, visualización de datos, filtrado, etc".

Por definición todos las aplicaciones de RI persiguen el objetivo de satisfacer la necesidad de información de un usuario facilitando aquellos documentos que sean relevantes a su requerimiento, pero existen distintos enfoques que intentan cumplir con dicha meta de distintas formas. Entre los enfoques clásicas de la Recuperación de Información podemos nombrar:

- Clasificación: Los documentos son organizados, según sus características, en distintas categorías definidas por clases. En un primer desarrollo las clases debían ser mutuamente excluyentes y cada documento en la colección podía ser situado en una categoría distintiva. Estos tipos de sistemas debían estar altamente adaptados al dominio y las clases estaban predefinidas. Más adelante se incorporó la clasificación por facetas que tiene la particularidad de poder asignarle a un documento varios aspectos, en vez de uno, permitiendo de esta manera búsquedas más avanzadas u específicas. De manera similar existe el método de agrupamiento pero sin clases predefinidas [Manning et al., 2008] .

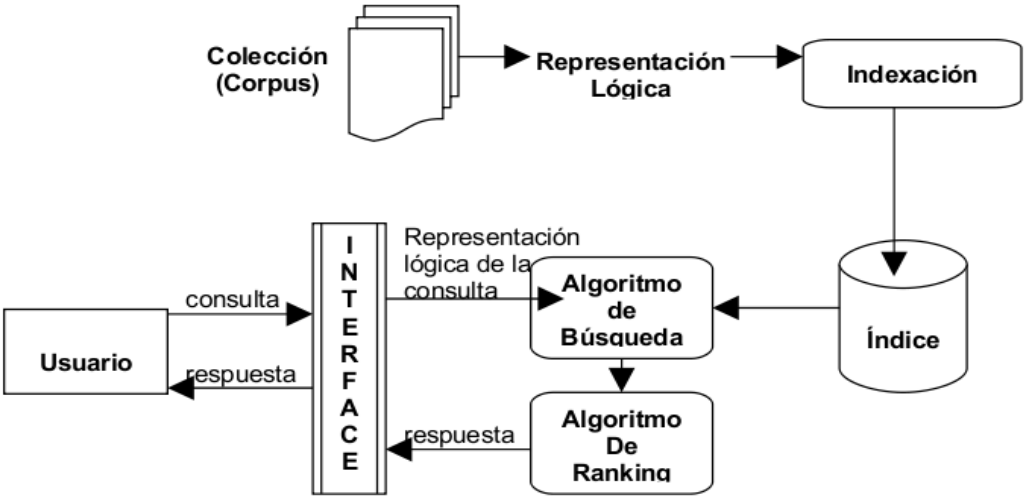
- Indexación y administración de vocabulario: El índice está vinculado con un vocabulario que es una cadena de términos y frases predefinidas que típicamente representan los significados de los documentos. A la hora de la búsqueda de documentos se suele expresar una consulta que comparte y utiliza el mismo vocabulario que los documentos. La combinación de los términos de la consulta es la que discrimina ciertos documentos y conjuntos de documentos. Los documentos de la colección también pueden estar vinculados mediante el uso consistente de términos entre los mismos.

- Representación de lenguaje natural: Se presupone una similitud en el uso de terminologías y relación de conceptos — semántico — entre los autores de los documentos y aquellos que realizan la búsqueda de los textos. A diferencia del uso de un índice que busca ocurrencias de términos entre consultas y documentos, aquí se tiene en cuenta los conceptos y la terminología presentada principalmente en el título y

en el resumen de los textos, y se verifican también citaciones con otros documentos que hacen las veces de relaciones de conceptos. Esto supone un esfuerzo en desarrollo de algoritmos que suele ser más complejo, y por lo general una implementación adaptada a un dominio específico a diferencia de otras aplicaciones.

- Orientado al usuario: Se focaliza en el comportamiento y en los aspectos psicológicos de la comunicación de la información deseada entre aquel que la generó y el usuario. Al contrario de los enfoques clásicos, en éste se apunta a mejoras en la efectividad de RI desde el marco del usuario, a partir de su necesidad de información y la interacción con los procesos.

- Sumarización: "Área que entiende sobre técnicas de extracción de aquellas partes – palabras, frases, oraciones, párrafos – que contienen la semántica que determina la esencia de un documento". [Tolosa et al., 2008]



Dibujo 3: Arquitectura de un SRI [Tolosa et al., 2008]

Un sistema de Recuperación de Información (SRI) es aquel que aplica los enfoques teóricos de RI sobre una colección de documentos, brindando aquellos que son relevantes a la necesidad de un usuario. En este trabajo se hace hincapié en el enfoque de indexación, con respecto al mismo el Dibujo 3 diagrama la arquitectura de un SRI.

- Colección: es el conjunto de documentos que son accesibles mediante el SRI.

- Representación lógica: corresponde a la abstracción utilizada para asociar la colección de documentos a una estructura lógica interpretada por el SRI.

- Indexación: es el proceso mediante el cual se transforma al documento de la colección en una representación lógica. El mismo es profundizado en la *Sección 4.1.3* .

- Índice: es la estructura de datos que representa el índice. Se ven en detalle en la *Sección 4.1.4* .

- Algoritmo de búsqueda: generalmente la consulta del usuario se transforma en una representación lógica del mismo tipo que los documentos y mediante un algoritmo se calculan aquellos documentos los cuales su representación lógica es similar (relevante a la de la consulta). Depende del modelo utilizado para la representación

- Algoritmo de rankeo: ya que el concepto de relevancia es subjetivo y poco preciso, es evidente que entre los documentos retornados por el SRI existan unos más relevantes que otros para el usuario, es necesario que este algoritmo tome los resultados obtenidos por el de búsqueda y forme un ranking de documentos relevantes para el usuario con el objetivo de que el mismo acceda primero a los que se creen de mayor relevancia.

- Interface: es la fachada del SRI por el cual interactúa el usuario que desea satisfacer una necesidad de información.

4.1.1 Modelos de Recuperación

El modelo es el marco teórico mediante el cual se diseña el SRI y por eso es la parte más fundamental del mismo. Existen distintos enfoques de modelado de la tarea de recuperación, por ejemplo la estadística, el álgebra de boole, el álgebra de vectores, la lógica difusa, el procesamiento del lenguaje natural y entre otros.

- Modelo booleano: es el modelado más simple con el cual llevar a cabo una tarea de

RI. Los documentos se representan como conjuntos de términos. Se tiene un vocabulario de términos extraídos de la colección. El índice es un matriz de término-documento donde cada elemento ***{t,d}*** corresponde al valor que representa la existencia el término ***t*** en el documento ***d***. En este modelo las consultas son formadas como a una *expresión booleana*[47] de términos. Un problema que tiene este modelo es que carece de información en su representación lógica para poder presentar un ranking de relevancia al usuario en los resultados obtenidos de una consulta. Por causa de esta carencia que presenta el modelo existe una mejora del mismo bajo el nombre de *Modelo Booleano extendido* [Waller et al., 1979][Salton et al., 1983].

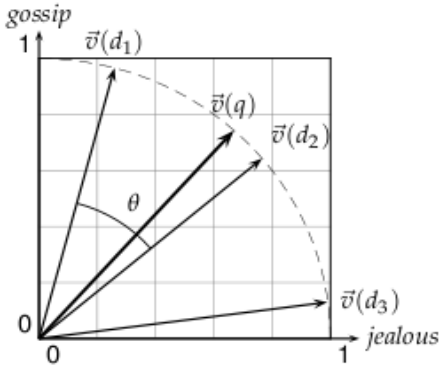
- Modelo vectorial: este modelo fue diseñando por Gerard Salton en su trabajo [Salton, 1971], y trata a cada documento como un vector en un espacio vectorial de n dimensiones, en la cual n es la cantidad de términos del vocabulario. La relevancia teórica se mide por distancia entre vectores en el mismo espacio, ya que las consultas también son tratadas como dichos elementos. Este tema se profundiza en la *Sección 4.1.1.1* .

- Modelo probabilístico: inicialmente ideado por [Robertson, 1977] está basado en el principio presentado en dicho trabajo llamado *El principio probabilístico*. Según [Baeza et al., 1999] dada una consulta *q* expresada por un usuario y un documento d_j en la colección, el modelo probabilístico trata de estimar la probabilidad de que el usuario encuentre al documento d_j interesante. Se asume que esta probabilidad de relevancia depende de la consulta y de la representación del documento solamente. También se considera que existe un subconjunto de documentos el cual el usuario prefiere como el resultado a su consulta, denominando a este conjunto como el ideal. [Tolosa et al., 2008] aporta diciendo que a partir de una expresión de consulta se puede dividir una colección de N documentos en cuatro subconjuntos distintos: REL conjunto de documentos relevantes, REC conjunto de documentos recuperados, RR conjunto de documentos relevantes recuperados y NN el conjunto de documentos no relevantes no recuperados. El resultado ideal de a una consulta se da cuando el conjunto REL = REC. Este modelo requiere varias interacciones con el usuario para lograr un resultado lo más cercano al ideal, es por eso que presenta una performance baja frente al modelo vectorial. Trabajos posteriores como [Fuhr et al., 1990] presentan

47 Las expresiones booleanas se usan para determinar si un conjunto de una o más condiciones es verdadero o falso, y el resultado de su evaluación es un valor de verdad.

una modificación de la teoría con una combinación de técnicas similares a las del modelo vectorial que presentan una mejora en el rendimiento.

4.1.1.1 Modelo Vectorial



Dibujo 4: Ejemplo de similitud en el modelo vectorial en un espacio de 2 dimensiones.[Manning et al., 2008]

El modelo vectorial, inicialmente presentado por [Salton, 1971] en su sistema SMART, mapea tanto a los documentos como a las consultas en un espacio *n*-dimensional, donde *n* es la cardinalidad del conjunto de términos indexados (cantidad de términos del vocabulario). La teoría plantea que cada *i*-coordenanda del vector *d* que representa un documento corresponde al valor del peso del término *i* en el documento *d.* También el modelo define una función que calcule el valor de similitud entre los vectores, en donde similitud se representa como la cercanía entre los mismos. Tanto la métrica de ponderación que asigna los pesos a los términos como la función de similitud pueden ser determinadas entre distintas alternativas y las combinaciones de las mismas que se utilicen influirán en gran medida la tarea de recuperación de información. Se caracterizan las distintas alternativas más adelante en esta sección.

Un SRI que aplica el modelo vectorial administra un vocabulario de términos y un conjunto de documentos representados como vectores, resultado del proceso de indexación de la colección original. Además aplica una metodología de asignación de pesos a los términos dentro de los documentos y aplica una función de similitud. Resumiendo, el proceso de recuperación de información mediante el modelo vectorial, dada una consulta expresada como un conjunto de

términos, implica el mapeo de la misma como un vector y luego una comparación con aquellos vectores que representan los documentos de la colección mediante la función de similitud. Se realiza el cálculo respectivo y se recolectan los resultados obtenidos ordenados por aquellos documentos los cuales la función de similitud determinó como más próximos al vector de la consulta, y por ende considerados como más relevantes.

En el *Dibujo 4* se muestra gráficamente a modo de ejemplo un espacio vectorial de dos dimensiones, el caso más simple. Para éste el vocabulario está formado por sólo dos términos (*'gossip'* y *'jealous'*) cada uno representado por un eje en la gráfica. Se indican tres documentos (*d1*, *d2*, y *d3*) de los cuales sus vectores son comparados con el de una consulta *q* a través de una función de similitud. En este caso la similitud es representada por el ángulo θ dado que la función utilizada es la *métrica del coseno*, la cual es presentada a continuación con otras métricas de similitud.

Al aplicar una representación mediante el modelo vectorial se supone la independencia entre los términos dentro de los documentos, esto se evidencia con el hecho de que los términos que los representan son ortogonales ya que cada uno define un eje.

4.1.1.1.1 Métricas de similitud

Para exponer las métricas de similitud es necesario previamente definir formalmente el modelo vectorial [Baeza et al., 1999]:

> *Para el modelo vectorial, el peso w_{ij} asociado con el par (k_i, d_j) es positivo y no binario. Además, a los términos en el índice presentes en la consulta también se les asigna un peso. Siendo w_{iq} el peso asociado con el par $[k_i, q]$, donde $w_{iq} \geq 0$. Entonces, el vector de la consulta q se define como $\vec{q} = (w_{1q}, w_{2q}, \ldots, w_{tq})$ donde t es el número total de términos en el índice del sistema. De manera similar, el vector para el documento d_j es representado por $\vec{d_j} = (w_{1j}, w_{2j}, \ldots, w_{tj})$.*

En la ***métrica del coseno*** la similitud entre la consulta *q* y el vector d_j se define como:

$$s(d_j, q) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{t} w_{ij}^2 \times \sum_{i=1}^{t} w_{iq}^2}} \qquad \textbf{(1)}$$

La similitud es medida por el coseno del ángulo que forman el vector documento y el vector consulta como indica el *Dibujo 4*. Existen otras medidas de semejanza además de las del coseno aunque no tan utilizadas como esta última.

Producto escalar:

$$\sum_{i=1}^{t} w_{ij} \times w_{iq} \qquad \textbf{(2)}$$

Coeficiente de Dice:

$$\frac{2 \times \sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sum_{i=1}^{t} w_{ij}^2 + \sum_{i=1}^{t} w_{iq}^2} \qquad \textbf{(3)}$$

Coeficiente de Jaccard:

$$\frac{\sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sum_{i=1}^{t} w_{ij}^2 + \sum_{i=1}^{t} w_{iq}^2 - \sum_{i=1}^{t} w_{ij} \times w_{iq}} \qquad \textbf{(4)}$$

4.1.1.1.2 Métricas de ponderación de términos

En el modelo vectorial, como fue indicado anteriormente, los vectores que representan los documentos contienen los pesos de los términos dentro del documento original para cada dimensión del espacio vectorial. La forma más simple de representar este peso o ponderación no es más que la ***frecuencia pura*** del término, es decir, la cantidad de veces que aparece dicho término en el documento original. En principio parece una métrica bastante coherente pero presenta un serio problema cuando se realiza el cálculo de similitud ya que éste último favorecerá, con dicha métrica de ponderación de términos, a los documentos más largos ya que las frecuencias de sus términos son mayores, pero esto no significa que éstos documentos sean realmente más relevantes a la consulta ya que quizás en documentos más cortos los términos de la consulta no aparecen con tanta frecuencia pero debido a la proporción son muchos más determinantes que en los otros casos. Ya que esto afecta en muchos casos la tarea de recuperación de información existen otras

ponderaciones más elaboradas que producen mejores resultados ponderando más alto a aquellos términos que brindan mayor poder de discriminación a los documentos.

Siendo N el número total de documentos indexados en el sistema y n_i el número de documentos en los cuales el término indexado k_i aparece. Siendo $frec_{ij}$ la frecuencia pura del término k_i en el documento d_j entonces la ***frecuencia normalizada*** f_{ij} de un termino k_i en un documento d_j está dada por:

$$f_{ij} = \frac{freq_{ij}}{max_l \cdot freq_{lj}} \qquad \textbf{(5)}$$

donde el máximo es calculado sobre todos los términos indexados del documento d_j. Si el término k_i no figura en el documento d_j entonces $f_{ij} = 0$. Por otro lado, ***la inversa de la frecuencia de documentos*** de k_i se define como:

$$idf_i = \log\left(\frac{N}{n_i}\right) \qquad \textbf{(6)}$$

La medida de ponderación de términos más, conocida como ***tf-idf*** esta dada por:

$$w_{ij} = f_{ij} \times idf_i \qquad \textbf{(7)}$$

De estas métricas presentadas se puede decir que la *frecuencia normalizada* evita que documentos de mayor tamaño, con altas frecuencias en sus términos, sean favorecidos durante el cálculo de la similitud con una consulta. Por otro lado la *inversa de la frecuencia de documentos* pondera más alto a aquellos términos que no se encuentran en muchos documentos, ya que esta característica los dota de un mayor poder de discriminación. La combinación de ambas métricas es la *tf-idf* y es la mejor métrica de ponderación de términos por ser la más completa ya que combina las dos características previamente nombradas.

4.1.2 Evaluación de Recuperación de Información

Un SRI, como todo sistema de software, debe ser evaluado una vez finalizado su desarrollo. El enfoque de evaluación común utilizado para sistemas convencionales de recuperación de datos es el de la medición de tiempo y espacio. El análisis del tiempo de respuesta y de la cantidad de espacio de almacenamiento utilizado son parámetros concretos y relativamente fáciles de conseguir. El SRI no esta exento a este tipo de evaluación y debe ser sometido a la misma con el objeto de analizar tiempos de indexación y de resolución de consultas, cantidad de almacenamiento utilizado para la estructura de índice entre otros. Esta tarea se denomina *evaluación de performance*.

En los diseños y desarrollos de sistemas de Recuperación de Información otras métricas, además de las de performance, son de interés. Debido a que las consultas realizadas sobre un SRI a veces no son muy claras, es probable que los documentos recuperados no conformen respuestas exactas y necesiten ser rankeadas con respecto a relevancia, y además teniendo en cuenta que el concepto de relevancia es naturalmente subjetivo y depende esencialmente del juicio del individuo, se presenta a la evaluación de un SRI como una tarea no del todo sencilla. Es por eso que estos sistemas necesitan de la evaluación de cuan preciso es el conjunto de respuesta. Este tipo de evaluación es denominada *evaluación de recuperación de información*.

Dicha evaluación de recuperación de información está usualmente basada en una *colección de prueba* y en una *medida de evaluación*. Una colección de prueba esta compuesta por tres elementos:

- Una colección de documentos
- Un conjunto de necesidades de información, expresadas como consultas
- Un conjunto de juicios de relevancia, por lo general una asignación binaria que corresponde a *relevante* o *no relevante* para cada par consulta-documento.

La decisión de relevancia para cada par consulta-documento es determinada en consenso por un grupo de personas para evitar la subjetividad de un solo individuo. Es vital que las colecciones de pruebas no sean pequeñas, sino que se definan un número considerable de necesidades de información junto a los resultados esperados.

Según [Manning et al., 2008] , es incorrecto utilizar solo una colección de pruebas durante la evaluación, ya que el sistema podría haberse ajustado intencionalmente para presentar una buena performance sólo para esas consultas definidas en la prueba y no para una consulta realmente ocasional que puede surgir de un usuario cuando el sistema se encuentre operando. Es por eso que se recomienda utilizar más de una colección de pruebas para la evaluación. Por dicho motivo existen algunas colecciones de pruebas estándar para la tarea de evaluación de la recuperación de información: NTCIR [48], CLEF [49], TREC[50], entre otras.

4.1.2.1 Medidas de evaluación

En la evaluación del rendimiento de RI, es necesario determinar cuántos documentos relevantes se recuperaron y cómo se rankearon para la presentación al usuario. En referencia a esto último vale la pena aclarar que generalmente la respuesta que se le retorna a un usuario es un conjunto de documentos en un ranking, en el cual hay documentos que son relevantes y otros que no los son. También es razonable que el resultado sea un subconjunto de los documentos de la colección, lo que significa que va a haber documentos que no fueron retornados. Esto es porque, si tenemos en cuenta grandes colecciones, sería inaudito retornar al usuario en la respuesta todos los documentos de una colección de considerable tamaño ya que es prácticamente imposible que se encuentre dispuesto a revisar todos los resultados. Dada esta consideración se puede definir la siguiente tabla para la evaluación de resultados:

	Recuperados	**No Recuperados**	
Relevantes	Rel-Rec (**w**) Positivos verdaderos	Rel-NoRec (**x**) Falsos positivos	Total de docs Relevantes: **w + x**
No Relevantes	NoRel-Rec (**y**) Falsos negativos	NoRel-NoRec (**z**) Negativos verdaderos	Total de docs no Relevantes: **y + z**
	Total de docs Recuperados: **w + y**	Total de docs no Recuperados: **x + z**	Total docs: **w + x + y +z**

Tabla 1: Tabla de contingencia para la evaluación [Tolosa et al., 2008]

48 NII Test Collections for IR Systems (http://research.nii.ac.jp/ntcir/data/data-en.html)
49 Cross Language Evaluation Forum (http://www.clef-campaign.org)
50 Text Retrieval Conference (http://trec.nist.gov)

Existen dos medidas que son muy reconocidas en el área de RI y por ende muy utilizadas. Éstas son las medidas de Precisión (*Precision*) y Exhaustividad (*Recall*), y fueron ideadas por [Cleverdon et al., 1966].

4.1.2.1.1 Precisión y Exhaustividad

Considerando un ejemplo de una necesidad de información *I* y su conjunto de documentos relevantes *R*. Siendo |*R*| el número de documentos en este conjunto y asumiendo una estrategia de recuperación dada, se procesa la necesidad de información *I* generando un conjunto de documentos de respuesta *A*. Siendo |*A*| el número de documentos en este último conjunto y además, siendo |Ra| el número de documentos del subconjunto resultante de R∩A (corresponde al conjunto de documentos **w** en la *Tabla 1*). Las medidas de *Recall* y *Precision* se definen así [Baeza et al., 1999]:

$$Recall = \frac{|Ra|}{|R|} \qquad \textbf{(8)}$$

$$Precision = \frac{|Ra|}{|A|} \qquad \textbf{(9)}$$

Recall es la fracción de documentos relevantes que fueron traídos en la respuesta, mientras que *Precision* es la fracción de documentos obtenidos que son relevantes. Si bien se puede realizar el calculo obteniendo un único valor de cada una de las dos medidas para cada consulta realizada en la evaluación, generalmente no es lo más utilizado ya que para hacer el análisis un poco más rico se estudia la lista de documentos recuperados en cada consulta calculando precisión y Exhaustividad de forma secuencial para cada documento según el orden de aparición en la respuesta. De esta forma es posible representarlo mediante una gráfica en la cual cada una de las dos medidas corresponde a un eje como el Gráfico 12.

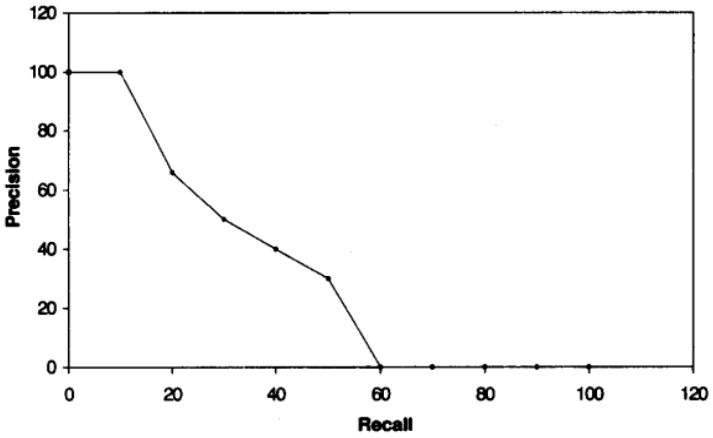


Gráfico 12: Precisión vs Exhaustividad

Un aspecto clave que cabe señalar entre las dos medidas es que existe una relación inversa entre ambas [Cleverdon, 1972], ya que se ha comprobado empíricamente que una alta exhaustividad implica una baja precisión. Es por eso que es necesario buscar que el SRI tenga una performance que equipare ambas medidas.

En la práctica se precisan evaluar una gran cantidad de consultas y comparar los resultados obtenidos. Para esto es necesario, en principio, utilizar valores de exhaustividad normalizados ya que por cada consulta pueden variar. El estándar son los once valores entre 0% y 100% con un incremento secuencial del 10%. Por otro lado también será necesaria una interpolación de la precisión, que no es más que el máximo valor de precisión entre dos valores de exhaustividad de la escala estándar.

Siendo r_j, $j \in \{0,1,2,...,10\}$, una referencia al j-ésimo nivel del estándar de exhaustividad. Entonces se define a la *precisión interpolada* como:

$$P(r_j) = max_{r_j \leq r \leq r_{j+1}} P(r) \qquad \textbf{(10)}$$

Con el objetivo de comparar la performance de distintos sistemas es posible calcular la *precisión promedio* a los efectos de utilizar la misma como objeto de comparación. La misma se define como:

$$\bar{p}(r) = \sum_{i=1}^{N_q} \frac{P_i(r)}{N_q} \qquad \textbf{(11)}$$

donde $\bar{p}(r)$ es la precisión promedio al nivel de exhaustividad *r*, N_q es el número de consultas realizadas y $P_i(r)$ es la precisión al nivel de exhaustividad *r* de la i-ésima consulta realizada.

Más allá de que no hay dudas de que estas medidas son las más utilizadas en las evaluaciones de Sistemas de Recuperación de Información existen algunas cuestiones referidas a éstas que es necesario aclarar. Primero, para poder calcular la exhaustividad de una consulta es necesario conocer la colección completa a efecto de diferenciar entre documentos relevantes y los que no lo son, siendo complicado el calculo de esta medida de forma exacta a diferencia de la precisión que no presenta esta desventaja. Segundo, como detalla [Martinez Mendez et al., 2004] no todo el público se encuentra interesado en el buena rendimiento de ambas medidas ya que los usuarios comúnmente se encuentran interesados en la precisión — resultados sin ruido — y en menos casos se valora más a la exhaustividad. Por último, una desventaja de estas medidas es que no son del todo útiles para sistemas que requieren interacciones iterativas.

4.1.2.1.2 Otras medidas

Existen otras medidas además de las de Precisión y Exhaustividad. Algunas son complementarias a éstas últimas y otras son de características distintas.

Precisión-R es una medida ideada para generar un solo valor que resuma la performance de recuperación de información. Se computa la precisión *R-ésima* posición en el ranking obtenido como resultado, donde *R* es la cantidad de documentos relevantes para la consulta. Para realizar el cálculo que defina el valor que represente la performance del sistema en general se pueden promediar los valores de esta medición para todas las consultas realizadas en la prueba. Sin embargo, el hecho de resumir el comportamiento de un SRI a través de varias consultas con un solo valor es pude ser un poco impreciso.

La ***media harmónica*** es otra medida alternativa que combina a la precisión y la exhaustividad. La misma esta definida por:

$$F(j)=\frac{2}{\frac{1}{r(j)}+\frac{1}{P(j)}} \qquad \textbf{(12)}$$

donde *r(j)* es la exhaustividad en el *j*-ésimo documento en el ranking, *P(j)* es la precisión en el *j*-ésimo documento en el ranking y *F(j)* es la media harmónica de *r(j)* y *P(j)*. Se asume que los valores de precisión y exhaustividad se encuentran entre 0 y 1. Te esta manera esta medida asume valores altos solo cuando tanto la precisión como exhaustividad son altos, implicando que la búsqueda del mejor compromiso entre ambas.

La ***medida E*** también combina la precisión y la exhaustividad pero con la posibilidad de ponderar más a uno que otro. Se define de la siguiente manera:

$$E(j)=1-\frac{1+b^2}{\frac{b^2}{r(j)}+\frac{1}{P(j)}} \qquad \textbf{(13)}$$

donde *r(j)* es la exhaustividad en el *j*-ésimo documento en el ranking, *P(j)* es la precisión en el *j*-ésimo documento en el ranking, *E* es la medida de evaluación relativa a *r(j)* y a *P(j)* y *b* es un parámetro definido por el usuario que refleja la importancia relativa de precisión y exhaustividad. Valores de *b* mayores a 1 indican que el usuario esta más interesado en la precisión mientas que valores por debajo de 1 indican que interesa más la exhaustividad.

4.1.3 Indexación

Dada una colección de documentos de texto, para que ésta pueda ser consultada a través de un SRI, en necesario obtener una representación de la misma y volcarla en una estructura de datos que pueda ser accedida y administrada rápida y eficazmente por un sistema informático. Es el proceso de indexación el que realiza esta tarea la cual es la actividad inicial de cualquier Sistema de Recuperación de información.

En esta etapa se pretende extraer de un texto los términos más significativos, las relaciones entre los mismos y toda aquella información necesaria para la correcta representación del contenido de

los documentos. La idea es utilizar solo aquellos términos más importantes ya que el uso del conjunto de todos los términos implicaría una performance de recuperación baja.

Según [Tolosa et al., 2008] existen dos enfoques de indexación: el lingüístico y el no lingüístico. En el primer caso, se utilizan técnicas estadísticas para análisis de frecuencias y cálculo de pesos de los términos, análisis de probabilidades para determinación de multipalabras y técnicas de agrupamiento (*clustering*) destinadas a la detección y extracción de relaciones. En el segundo caso, se utilizan técnicas derivadas del procesamiento del lenguaje natural[51] , las que pretenden imitar el comportamiento de los indexadores humanos. Si bien algunas de éstas aún no se encuentran completamente desarrolladas ya que no pueden generar una representación perfecta de los documentos, son utilizadas en algunos SRI.

Entre los enfoques lingüísticos más conocidos podemos nombrar:

- Procesamiento morfológico-léxico: su función principal es obtener el léxico del texto a partir de la identificación de formas sintagmáticas, siglas y locuciones. Una herramienta comúnmente utilizada son los etiquetadores de categorías gramaticales que tienen las funciones de asignar automáticamente la categoría léxica y brinda información sobre las categorías gramaticales.

- Procesamiento sintáctico: su objetivo es describir la estructura de las oraciones que componen los documentos. En el análisis sintáctico se separan las unidades lingüísticas con sentido simple o compuesto y se desambiguarlas categorías gramaticales asignadas por el analizador morfológico.

- Procesamiento semántico: la intención es obtener el significado de las palabras y de las oraciones que conforman. Generalmente esto se realiza a través de la utilización de tesauros.

Si bien las técnicas lingüísticas han asistido al proceso de indexación, estás por le momento no han presentado grandes avances siendo las técnicas no lingüísticas las que aún brindan mejores resultados y son mayormente utilizadas. A continuación se presenta una breve descripción de las mismas.

51 Abreviado PLN. El objetivo del procesamiento de lenguaje natural es el de construir sistemas y mecanismos que permitan la comunicación entre personas y máquinas mediante el uso de lenguajes naturales.

4.1.3.1 Indexación con enfoque no lingüístico

Este enfoque se basa en el estudio de las frecuencias de los términos en los documentos y su distribución dentro de los mismos. A través de este análisis se pretende determinar cuáles son los términos que tienen mayor poder de discriminación de contenido y que aportan mayor información, siendo éstos los que se van a seleccionar para conformar el índice. Para este tipo de análisis este enfoque se apoya en ciertos estudios estadísticos que pretenden describir ciertas propiedades de los textos. A continuación se mostrarán dos leyes empíricas, la *Ley de Zipf* que describe la distribución de frecuencias de las palabras en los textos y la *Ley de Heaps* que estudia el crecimiento del vocabulario con al aumento de documentos en la colección.

4.1.3.1.1 Ley de Zipf

La *Ley de Zipf* [Zipf, 1949] es una ley empírica formulada mediante el uso de estadísticas referidas al hecho que muchos tipos de datos estudiados en diversas ramas de la ciencia se ajustan a una distribución de Zipf, una de la familia de las *leyes de potencia*[52]. Mediante ésta se descubrió que la gente al escribir prefiere usar palabras más conocidas a las menos conocidas, lo que se denominó la *"ley del menor esfuerzo"*.

La ley — de Zipf — afirma que, si t_1 es el término más común en la colección, t_2 es el siguiente más común, y así sucesivamente, entonces la frecuencia de la colección cf_i del i-ésimo término más común es directamente proporcional a $1/i$:

$$cf_i \propto \frac{1}{i} \qquad \textbf{(14)}$$

Entonces el término más frecuente ocurre cf_1 veces, luego el segundo más frecuente tiene la mitad de las ocurrencias, el tercero un tercio y así sucesivamente. Se intuye que la frecuencia decrece rápidamente con el ranking, caracterizando a la distribución de tener pocos términos muy frecuentes y muchos términos poco frecuentes (sesgada).

52 La ley de potencia es un tipo especial de relación matemática entre dos cantidades. Cuando la frecuencia de un evento varía a la potencia de algún atributo de ese evento, se dice que la frecuencia sigue una ley de potencia.

Si se ordenan todos los términos de manera descendente por la frecuencia de aparición y denotando como *z(r)* a la frecuencia del término con ranking *r*, la ley de Zipf indica:

$$z(r) = z_{max} \cdot r^{-\alpha} \qquad \textbf{(15)}$$

donde z_{max} es la máxima frecuencia y α es el llamado exponente de Zipf. El exponente de Zipf indica la riqueza del texto, si las frecuencias de los términos son similares entonces α asume un valor bajo, mientras que si existen términos con frecuencias muy altas y términos con frecuencias muy bajas el valor de α pasa a ser alto.

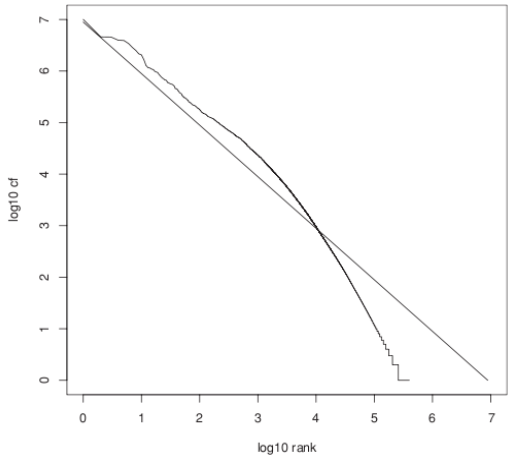


Gráfico 13: Distribución de Zipf para colección Reuters-RCV1 [Manning et al., 2008]

La colección de frecuencias de los términos en función de su ranking suele ser representada, junto a su función de Zipf, mediante una gráfica en base logarítmica en sus dos ejes. Dicha representación ayuda a verificar si los datos se corresponden a una distribución Zipf. En el caso del *Gráfico 13* los datos de las frecuencias no se ajustan particularmente bien a la distribución predicha por la ley Zipf — representada por la recta en la gráfica —, sin embargo esta última es suficiente para servir de modelo de la distribución de términos.

4.1.3.1.2 Ley de Heaps

Al igual que la ley de Zipf, la *Ley de Heaps* [Heaps, 1978] se basa en pruebas empíricas, a diferencia que ésta intenta plantear una relación entre el tamaño de la colección — cantidad de palabras — y el crecimiento del vocabulario — cantidad de palabras únicas —. De esta manera Heaps estima el tamaño del vocabulario M en función del tamaño de la colección:

$$M = kT^b \qquad \textbf{(16)}$$

donde T es el número de tokens en la colección, k y b son constantes que dependen del texto y sus valores son generalmente $30 \leq k \leq 100$ y $0.4 \leq b \leq 0.6$ (para el idioma inglés).

El parámetro k es muy variable porque el crecimiento del vocabulario depende de varias características de la colección y de como ésta es procesada. La normalización de la capitalización y el stemming (*Sección 4.1.3.2.3*) reducen la tasa de crecimiento del vocabulario, mientras que la inclusión de números y errores ortográficos la aumenta. Sin tener en cuenta valores para los parámetros en una colección en particular, la Ley de Heaps dice que el vocabulario se continuará incrementando con la incorporación de más documentos, y que el tamaño del vocabulario es un poco mayor para grandes colecciones que para las más pequeñas a comparación de su gran diferencia en el número de documentos.

4.1.3.2 Pre-procesamiento

El pre-procesamiento es una tarea que puede ser dividida en 5 etapas bien marcadas:

- Análisis léxico: en este paso se pretende quitar signos de puntuación, realizar una normalización y extraer las palabras.

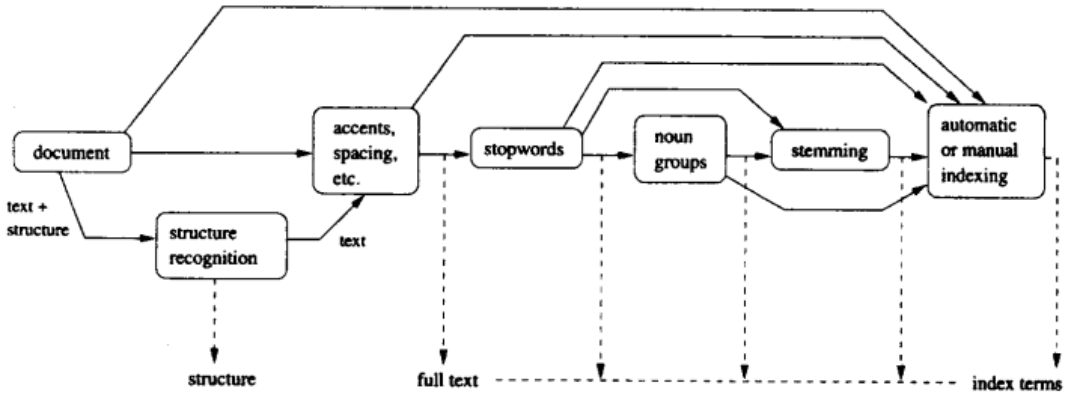
- Eliminación de palabras vacías: se quitan aquellos términos que carecen de significado y que aportan poco valor representativo al documento.

- Stemming: es el reemplazo de ciertas palabras por su raíz semántica.

- Selección de términos a indexar: se define que términos o grupos de los mismos van a

formar parte del vocabulario en el índice del SRI. También se asignan los pesos.

- Categorización de términos: en esta etapa se lleva a cabo la construcción de estructuras de apoyo como tesauros[53] para potenciar la tarea de recuperación de información.



Dibujo 5: Representación lógica de la tarea de pre-procesamiento [Baeza et al., 1999]

4.1.3.2.1 Análisis léxico

El análisis léxico — también conocido por el nombre de tokenización — lleva a cabo la conversión de un flujo de caracteres, que representan un documento de texto, en *tokens*. Dichos tokens son candidatos a convertirse en términos que formen parte del vocabulario del índice.

Entonces, explicado de otra manera, la acción que se realiza en esta etapa es la de reconocimiento de palabras. Para cumplir dicho objetivo se debe definir qué es una palabra, teniendo en consideración si términos de uno o dos caracteres son tenidos en cuenta, o si los números son descartados o no, si se considerarán palabras compuestas.

En esta etapa se realiza una normalización del texto, la cual consiste en el reemplazo de mayúsculas por minúsculas, quita de tildes y otros caracteres particulares y la eliminación de signos de puntuación, entre otras acciones. Una vez realizada la normalización se detectan los espacios y se logran extraer las palabras pero hay que tener en consideración ciertos casos particulares ya que es

53 Tesauro es un vocabulario controlado que muestra relaciones (por e.j.: semánticas) entre términos, las cuales pueden ayudar a sistemas de recuperación a expandir consultas.

probable que durante la normalización se pierdan ciertos términos que son de importancia y que aportan información al documento. Por ejemplo entidades representadas por siglas — por ejemplo "T.R.E.C." —, palabras compuestas separadas por guiones — como el caso de "pre-procesamiento" — o URLs de internet son algunos de los casos particulares de términos que quizás sean representativos pero se podrían perder en el análisis léxico. Esto quedará en consideración a la hora del diseño la forma de accionar con respecto a estos casos particulares.

4.1.3.2.2 Eliminación de palabras vacías

Naturalmente en el lenguaje existen muchos términos que carecen de significado y que en algunos casos son utilizados con altísima frecuencia careciendo de todo poder de discriminación de textos. Palabras de estas características son denominadas *palabras vacías* y es aconsejable la eliminación de las mismas por dos razones bien marcadas: no aportan ningún significado ni poder de discriminación y su eliminación reduce el tamaño del índice en un 40% aproximadamente.

Las palabras vacías son particulares del lenguaje y son presentadas en listas tal como la ofrecida por el proyecto *Snowball*[54] para la lengua castellana. La eliminación de las mismas mejora considerablemente la performance de recuperación de información evitando ruido en las respuestas y también mejora los tiempos de respuestas.

4.1.3.2.3 Stemming

Cuando se realiza una consulta sucede que quizás alguno de lo términos que se utilizan no coincidan directamente con las variantes sintácticas del mismo que están presentes en documentos que en teoría son relevantes a la consulta. Estas variaciones corresponden a pluralidad, variantes morfológicas, diferentes tiempos verbales, entre otras. De esta forma pueden ignorar ciertos documentos relevantes perjudicando la performance de recuperación.

Para atacar este tipo de situaciones existe la técnica de ***stemming*** o ***lematización***. Ésta consiste

54 http://snowball.tartarus.org/algorithms/spanish/stop.txt (lista de palabras vacías para el idioma castellano)

en el reemplazo de la palabras por su término raíz o lema. La aplicación de esta técnica trae dos ventajas: primero la reducción del tamaño del índice, y segundo, mejora en la performance de recuperación para el caso que se prevalezca más la exhaustividad que la precisión [Tolosa et al., 2008] ya que aplicando *stemming* tanto en la consulta como en el corpus se recuperan documentos que tengan todas las variantes morfológicas de los términos de la consulta.

Según [Flora et al., 2010] existen cuatro enfoques de *stemming*: quita de afijos, búsqueda en tabla, s-stemmer y n-gramas. El enfoque de *búsqueda en tabla* consiste simplemente en verificar en una tabla de palabras la raíz correspondiente. *S-stemmer* posee conocimiento de estructuras lingüísticas y se basa en la determinación de límites morfológicos. Por otro lado la aproximación de *n-gramas* corresponde más a una técnica de *clustering*[55] que de *stemming* y estaba basada en la identificación de bigramas y tri-gramas. Por último, el enfoque de *quita de afijos* es, según [Baeza et al., 1999], el mejor de los cuatro por ser intuitivo, simple y porque puede ser implementado eficientemente,

Con respecto a la perspectiva *quita de afijos*, es importante aclarar que por lo general se enfoca más en la quita de sufijos ya que en los lenguajes occidentales las variantes de palabras son generadas más comúnmente por el agregado de sufijos que de prefijos. El algoritmo más popular de quita de sufijos es el de Porter [Porter, 1980] por su simplicidad y elegancia. El mismo es dependiente del lenguaje para el cual existen varias adaptaciones del mismo para distintos de ellos.

4.1.3.2.4 Selección de términos a indexar

En esta etapa se termina definiendo aquellos términos que formarán parte del vocabulario del índice. Aquellos tokens obtenidos en el análisis léxico, luego de haber pasado por varias etapas de proceso, serán seleccionados para formar parte del índice. Se puede adoptar un enfoque para la indexación de sustantivos [Broglio et al., 1995] dado que generalmente son estos los que cargan el mayor valor semántico.

Una vez determinados los términos que se utilizaran en el índice se procede con la asignación de

55 Técnica de Recuperación de Información de agrupamiento de documentos. Se basa en la hipotesis de que documentos en el mismo cluster (o grupo) poseen similar relevancia, comportándose de las misma manera frente a necesidades de información.

pesos de los mimos dentro del documento la cual se implementa a través de alguna de las *métricas de ponderación de términos* explicadas en la *Sección 4.1.1.1.2* .

4.1.4 Estructuras de datos

Como se explicó anteriormente, un Sistema de Recuperación de Información brinda al usuario acceso a ítems de información incluidos en una colección de documentos, pero claramente, éste no realiza la búsqueda directamente sobre el texto libre ya que se trataría de una operación costosa y por ende ineficiente. Es por eso que unas de las funcionalidades básicas de un SRI es el procesamiento de la colección de texto con objetivo de la obtención de una representación lógica de la misma, la que deberá ser volcada sobre una estructura de datos que soporte consultas de manera rápida y eficaz. De esta manera se obtiene una buena performance en la recuperación sobre colecciones de un tamaño considerable dado que la búsqueda sobre texto libre para estos casos sería inviable.

Estás estructuras de datos, por lo general llamadas índices, contienen el vocabulario como la entrada del índice, el conjunto de referencias a los documentos reales y los datos que representan las relaciones entre ambos. Sin embargo, el contenido del índice varía según el modelo de RI (*Sección 4.1.1)* que se utilice ya que unos modelos precisan más información que otros.

A continuación se presentarán tres tipos de estructuras de datos comúnmente utilizadas en el área de RI.

4.1.4.1 Archivo Invertido

Un archivo invertido es un mecanismo orientado a las palabras para indexar una colección de texto con el objetivo de agilizar la tarea de búsqueda. La estructura de este índice esta conformado por dos elementos: el vocabulario y las ocurrencias del mismo en la colección. El vocabulario es el conjunto de todos los diferentes términos en el texto. Para cada término del vocabulario hay una lista en la cual se indica la información de la aparición de dicho término dentro de la colección, ésta

se denomina lista de posteo (*posting list*) y se corresponde las ocurrencias.

Este tipo de estructura se nombra invertida ya que su organización es opuesta al enfoque de un documento como un conjunto de términos. De esta manera se puede ver al índice invertido como una matriz término-documento como la que se muestra a continuación.

	d_1	d_2	d_3	...	d_m
t_1	w_{11}	w_{12}	w_{13}		w_{1m}
t_2	w_{21}	w_{22}	w_{23}		w_{2m}
t_3	w_{31}	w_{32}	w_{33}		w_{3m}
...					
t_n	w_{n1}	w_{n2}	w_{n3}		w_{nm}

Tabla 2: Matriz término-documento

En la implementación del índice invertido, cada fila correspondiente a un término corresponderá a una *posting list*. Cada elemento de la lista contiene la información necesaria para la recuperación según el modelo que se utilice. Por ejemplo, si se aplica el *Modelo Booleano* (*Sección 4.1.1*) los elementos de la lista no son más que la referencias a los documentos en los que el término aparece. Ya para el caso del *Modelo Vectorial* (*Sección 4.1.1.1*) se necesitará un poco más de información ya que además de la referencias del documento se precisará el peso del término dentro de este último. Si se desean hacer *búsquedas posicionales*[56] es necesario agregar información de las ubicaciones del término dentro del documento indicando la posiciones, relativas al inicio del documento, en donde ocurre el primera carácter de la palabra.

t1	(d5,4) (d9,2)
t2	(d1,4)
t3	(d1,2)(d8,1)(d9,3)
t4	(d2,4)
...	
tn	(d3,2)(d4,1)(d6,2)

t1	(d5:5,46,70,207)(d9:24,106)
t2	(d1:356,530,702,840)
t3	(d1:6,682)(d8:43)(d9:12,42,56)
t4	(d2:56,68,264,793)
...	
tn	(d3:10,21)(d4:19)(d6:86,350)

Tabla 3: Ejemplo de índice invertido con frecuencias (izq.) e índice invertido posicional (der.)

Las búsquedas sobre el vocabulario índice pueden realizarse de forma secuencial, mediante

56 Precisa información de localización de los términos, permitiendo búsquedas por cercanía de términos y soporta búsquedas por frases.

hashing[57] o por una *búsqueda binaria*[58]. Las dos primeras nombradas tienen un coste de $O(n)$ mientras que la búsqueda binara presenta una complejidad computacional del orden de $O(\log n)$. Mas allá de la implementación de la búsqueda, según [Baeza et al., 1999], el mayor coste en la búsqueda se lleva a cabo en la fusión de posting-lists cuando se trata de consultas que contienen varios términos.

En cuanto al espacio necesario para almacenar un índice invertido suele ser por lo general considerablemente menor que el que ocupa la colección de documentos. En principio esto de puede observar con la Ley de Heaps (Sección *4.1.3.1.2*), ya que según la misma el tamaño del vocabulario es del orden de $O(n^b)$, siendo $0.4 \leq b \leq 0.6$, aunque también hay que tener cuenta el tamaño de las posting lists correspondientes a cada término del vocabulario. Si el ahorro de espacio de almacenamiento es primordial entonces pueden utilizar técnicas de compresión del índice aunque éstas suponen un mayor costo de procesamiento por la necesidad de descompresión a la hora de ser accedido.

Un problema que tiene esta estructura de datos es el coste de mantenimiento ya que la actualización de los datos la misma implica un coste considerable, aunque se suponen que en RI este tipo de actualizaciones no son normalmente realizadas.

4.2 Recuperación de Información Distribuida

En la actualidad el fenómeno de la gran evolución de internet y el crecimiento en información textual disponible para los usuarios fue posible gracias a la aparición de diversas fuentes de información que se encuentran accesibles para los usuarios. Dado que la fuentes y la información son muy diversas en varios entornos y redes de área amplia (WANs), principalmente el de la web, se presenta como una tarea muy compleja para los usuarios en dichos contextos la búsqueda de ciertos ítems de información que satisfagan sus necesidades.

Para el ambiente de la web, una solución a esta problemática que ha resultado muy efectiva es la

57 Técnica para mapear grandes conjuntos de datos de tamaño variable (por e.j.: los términos), denominados entradas, a una conjunto de de datos más pequeños de un tamaño fijo (por e.j.: una secuencia de número que corresponde a las dimensiones de un vector).

58 Algoritmo que tiene como propósito encontrar la posición de una valor específico (la entrada) entre un arreglo de datos ordenado.

de los *motores de búsqueda*. Éstos basan su funcionamiento en una *base de datos centralizada*, es decir, que dispone de un solo modelo de base de datos para la recuperación textual. En el esquema centralizado se contiene una copia de la información existente en las numerosas fuentes. El factor de un motor de búsqueda que hace posible la extracción y copia de contenido de otras fuentes ajenas es la utilización de técnicas de *crawling*[59].

Los problemas que presentan los modelos centralizados son por un lado que requieren grandes recursos para almacenar y procesar el contenido copiado de otras fuentes si es que éstas última son de un tamaño considerable, y por otro lado muchos de los publicadores de información no están de acuerdo con que se copien sus contenidos ya por una cuestión de propiedad, monetaria o simplemente porque desean controlarlos cuidadosamente ellos mismos.

El área de Recuperación de Información Distribuida (RID) se encarga de estudiar la otra alternativa a las bases de datos centralizadas, el *enfoque de base de datos múltiples*. Según [Callan et al., 2000] para este enfoque un sitio central almacena unas breves descripciones de cada base de datos, y un servicio de selección de bases de datos utiliza estas *descripciones de recursos* para identificar aquellas que están más próximas a satisfacer una necesidad de información. El modelo de bases de datos múltiples puede ser aplicado en entornos en los cuales los contenidos de las bases de datos son propietarios, o están cuidadosamente controladas, o donde el acceso es limitado, ya que el sitio central no precisa copias de los documentos de cada base de datos.

Un sistema de Recuperación de Información Distribuida consta de una arquitectura básica. Como explica [Banchero, 2010] la misma consiste en tres componentes principales:

- *Usuario*: es aquel quien expresa su necesidad de información mediante algún lenguaje de consulta.

- *Broker*: es el servidor que aplica la mayoría de la funcionalidad. Se ocupa de descubrir nuevos repositorios, de generar y mantener una representación más acotada y eficiente de la información contenida en éstos. También procesa las consultas del usuario orientando las búsquedas a los repositorios más convenientes y retornando los resultados de una forma combinada y ordenada intentando, de esta manera, satisfacer la necesidad de información del usuario.

59 Técnicas para la aplicación en programas de computadora que navegan la World Wide Web en una manera métodica y automatizada, o mediante un estilo ordenado.

- *Repositorio*: Los repositorios es donde se almacena información de alguna temática en particular o múltiples temas. También se los conoce como fuentes de información o bases de datos textuales y se encuentran tanto en redes 30 corporativas como en redes públicas.

Un SRID opera sobre un modelo de múltiples repositorios, que a diferencia del modelo de una sola base de datos, este mucho más complejo ya que debe ocuparse de las siguientes cuestiones:

- ***Descripción del recurso***: los contenidos de cada base de datos textual deben ser descriptos.

- ***Selección de recurso***: dada una necesidad de información y un conjunto de descripciones de recursos, una decisión debe ser realizada sobre en cuál base de datos buscar.

- ***Fusión de resultados***: integración de las listas de documentos rankeadas de cada base de datos textual dentro de un único y coherente ranking general.

Éstos tres ítems presentados arriba forman parte de la problemáticas de RID, la cual varía dependiendo del ambiente en el cual se pretende llevar a cabo la tarea. De esta manera [Banchero, 2010] clasifica dos ambientes posibles para la RID y por definición plantea la mayor complejidad que presenta uno sobre el otro:

- *Ambientes cooperativos*: ambientes reducidos donde se trabaja sobre una arquitectura de red local, como podrían ser pequeñas organizaciones. Las fuentes de información generalmente soportan un único algoritmo de recuperación. En este caso la cooperación es simple y posibilita que cada una de las fuentes provea las estadísticas de su corpus. Este tipo de ambiente cooperativo provee soluciones más simples y efectivas para dar respuesta a los sub-problemas de creación de descripciones de recursos y fusión de resultados. Cuando se trabaja en ambientes donde las fuentes de información cooperan se deben definir protocolos para poder acceder a las estadísticas que se proveen.

- *Ambientes no cooperativos*: en redes WAN (Wide Area Networks) o en la Web, puede que no se conozcan los algoritmos de recuperación que utilizan las diferentes fuentes

y esto hace que la cooperación entre estas sea dificultosa. En este sentido, aunque exista voluntad para cooperar, dada la gran heterogeneidad del ambiente es difícil detectar si la información provista por cada fuente es fiable o no para el proceso de selección de recursos.

4.2.1 Descripción del Recurso

En un ambiente donde existen muchas bases de datos, la primer tarea a realizar es el descubrimiento y representación de lo que dichas bases de datos contienen. Por lo general, primero se lleva a cabo la representación, decidiendo lo que se desea representar y como se va a hacer, dejando para más adelante lo pertinente a la adquisición de dicha información.

Una forma simple y robusta de describir los recursos es representar cada base de datos mediante una descripción que consiste de palabras y sus frecuencias de ocurrencia dentro de dicho repositorio [Gravano et al., 1994] o, también, de datos estadísticos derivados de estas últimas . Este tipo de representación se denomina *modelo de lenguaje de unigramas, y* se trata de una estructura compacta y que puede ser obtenida de una forma rápida y automática revisando el índice de la base de datos o directamente procesando los documentos de texto. Este tipo de representación además puede ser extendida incluyendo el uso de frases, nombres propios y otros rasgos textuales presentes en la base de datos.

Un ejemplo simple de una representación de *modelo de lenguaje de unigramas* es la utilización de una matriz término-recurso, similar a la matriz término-documento (*Sección* 4.1.4.1). Dicha descripción de recursos contiene los valores de documento-frecuencia que se corresponden al número de documentos dentro de cada repositorio en los cuales determinado término aparece al menos una vez. El conjunto de términos en la matriz corresponde a la unión de los subconjuntos de vocabulario de los distintos repositorios del sistema distribuido.

	r_1	r_2	r_3	...	r_m
t_1	df_{11}	df_{12}	df_{13}		df_{1m}
t_2	df_{21}	df_{22}	df_{23}		df_{2m}
t_3	df_{31}	df_{32}	df_{33}		df_{3m}
...					
t_n	df_{n1}	df_{n2}	df_{n3}		df_{nm}

Tabla 4: Matriz de descripción de recursos utilizando la frecuencia de documentos

El modelo de lenguaje de unigramas es bueno para su aplicación en ambientes cooperativos, pero a la hora de enfrentarse a situaciones de escasa cooperación serán necesarias otras formas de obtención de la representación del recurso. Dado que no es posible acceder a toda la información de los repositorios, se debe realizar una estimación de aquella frecuencias o estadísticas faltantes para poder crear una representación lo más completa posible. Como el objetivo de este trabajo no incluye operar en ambientes de este tipo, se limita sólo a nombrar a continuación algunas de técnicas de obtención de descripción de los recursos ideadas para este tipo de condiciones: *Query Based Sampling (QBS) [Callan et al., 2001]* , *Capture-recapture y Sample-resample* [Si et al., 2003].

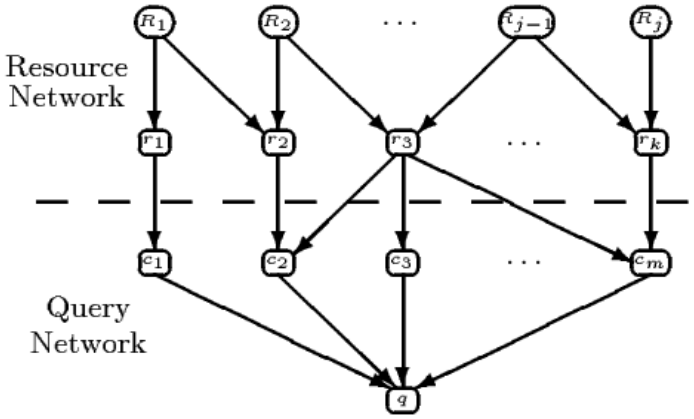
4.2.2 Algoritmos de selección

Dada una necesidad de información y un conjunto de descripciones de recursos ¿cómo hace el sistema para seleccionar sobre qué recurso buscar? El *broker* recibe una consulta y deberá escoger del subconjunto de colecciones disponibles aquellas que se estima tienen mayor probabilidad de retornar documentos relevantes a la consulta. Para realizar esta selección, es necesario el armado de un ranking de las colecciones en función a su probabilidad de satisfacer la necesidad de información.

Muchos algoritmos de selección de recursos en RID se basan en el *cálculo de similitud* (*Sección 4.1.1.1.1*) de los modelos de RI centralizados. Para poder aplicar estas técnicas al ambiente distribuido se trata a los recursos del sistema distribuidos como si fueran los documentos en un sistema centralizado. Existen dos algoritmos principales de RID para la selección de algoritmos: CORI y GlOSS [Gravano et al., 1994]. En esta sección se profundiza sólo sobre el primero nombrado anteriormente.

4.2.2.1 CORI

El *modelo de red de inferencia bayesiana*[60] de Recuperación de Información puede ser aplicado al proceso de rankear recursos. Cada recurso R_i es representado por un conjunto de nodos de representación — términos de indexación — r_j. Una necesidad de información está representada por una o más consultas q, los cuales están compuestos por conceptos de consulta c_k y operadores de consultas - no mostrados en el Dibujo 6 -.



Dibujo 6: Simple red de inferencia de selección de recursos [Callan et al., 2000]

La probabilidad $P(q|R_i)$ de que una necesidad de información representada por una consulta q sea satisfecha por la búsqueda en un recurso R_i está determinada por el nodo de instanciación R_i y la probabilidad de propagación a través de la red hacia el nodo q. La probabilidad P que el concepto de representación r_j es observado en el recurso R_i es estimada por la variación de fórmulas *tf-idf* (*Sección 4.1.1.1.2*) y es mostrada a continuación:

$$T = \frac{df}{df + 50 + 150 \cdot cw / avg_{cw}} \qquad \textbf{(17)}$$

$$I = \frac{\log\left(\frac{C + 0.5}{cf}\right)}{\log(C + 1.0)} \qquad \textbf{(18)}$$

$$p(r_k|R_i) = b + (1-b) \cdot T \cdot I \qquad \textbf{(19)}$$

60 Es un modelo probabilístico gráfico que representa un conjunto de variables aleatorias y sus dependencias condicionales a través de un grafo dirigido acíclico.

donde:

df es el número de documentos en la colección *Ri* que contienen el término *rk*,

cw es el número de términos de indexación en la colección *Ri*,

avg_{cw} es el número promedio de términos de indexación en cada colección,

C es el número de colecciones,

cf es el número de colecciones que contienen el término *rk*, y

b es el componente de probabilidad mínima (usualmente 0.4).

La ecuación (**17)** es una adopción de la fórmula de *Okapi* [Robertson et al., 1994], en donde el término-frecuencia (*tf*) es reemplazado por el documento-frecuencia (*df*), y se utilizan constantes para controlar valores de *df* muy altos. Mientras que la ecuación **(18)** es una variación de la adaptación de la fórmula *idf* presentada por [Turtle et al., 1990] en donde el número de documentos es reemplazado por el número de colecciones (*C*).

Las ecuaciones **(17)**, **(18)** y **(19)** son conocidas como el algoritmo de CORI para ranking de bases de datos [French et al., 1999], aunque también el nombre de CORI es utilizado de una manera más amplia identificando a cualquier uso de redes de inferencia para el ranking de bases de datos. Para calcular la probabilidad completa $P(q|R_i)$ correspondiente a la consulta basta con calcular el promedio de las probabilidades $P(q|R_i)$ de los términos que la componen.

El algoritmo de CORI no presenta una buena performance al operar sobre un conjunto de recursos en el cual existen diferencias significativas de tamaño entre las bases de datos [Si et al., 2003b], es decir, cuando existen grandes colecciones y otras muy pequeñas.

4.2.3 Fusión de resultados

Después de que se realiza la búsqueda sobe un conjunto de bases de datos, el ranking resultante de cada una de las mismas debe ser fusionado en una sola lista completa [Callan et al., 2000]. Esta tarea puede presentarse complicada ya que los rankings de documentos y la puntuaciones producidas por cada base de datos están basadas en diferentes estadísticas de corpus y posiblemente diferentes representaciones y/o algoritmos de recuperación. En el caso de que las bases de datos presenten una heterogeneidad por lo factores nombrados anteriormente, se hace imposible realizar una comparación directa.

Para esta tarea existen distintas soluciones que incluyen el procesamiento de *valores normalizados* manteniendo la normalización de los datos de todas las colecciones o realizando una estimación de los mismos, o por otro lado, fusionando los *valores desnormalizados* directamente.

La solución más precisa es la que utiliza *valores normalizados* de documentos de diferentes bases de datos. Esto se puede implementar tanto utilizando un corpus de estadísticas globales [Xu et al., 1998] o reprocesando los valores de ranking de los documentos en el cliente de búsqueda [Kirsch, 1997]. De todas formas esta solución solo funciona en ambientes altamente cooperativos.

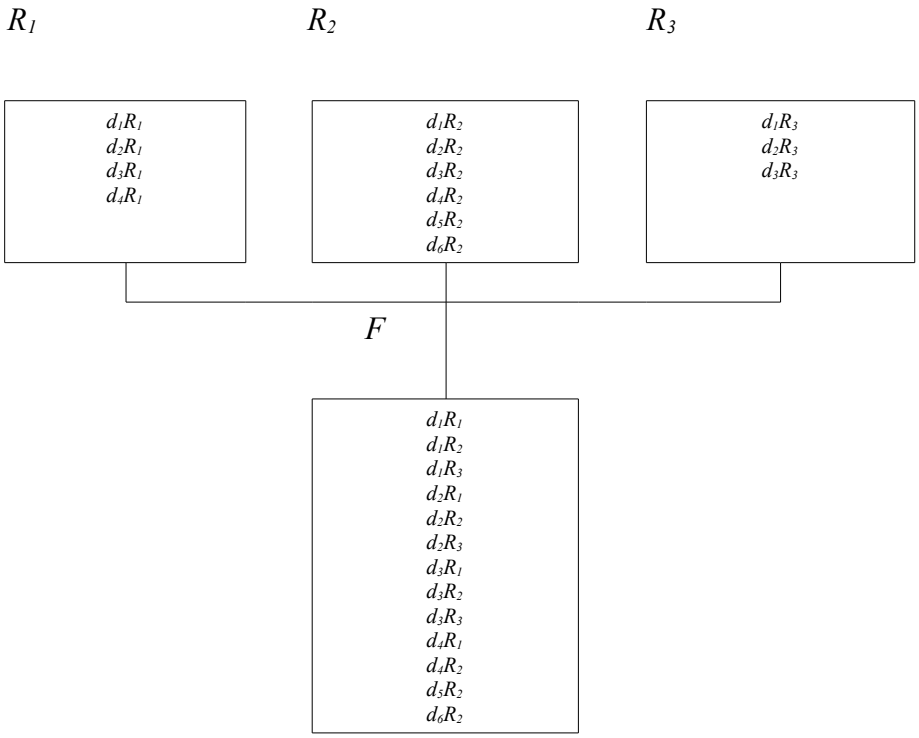


Tabla 5: Simple Ejemplo de fusión mediante Round Robin

Siguiendo otra línea, se presenta como la solución más simple la de fusión de *valores desnormalizados*. Una aproximación a este tipo de fusión de resultados es mediante el método de *round robin* [Turtle et al., 1990]. En este método se utiliza la información del ranking de documentos en una lista ordenada individualmente y la información del ranking de la fuente dada por el algoritmo de de selección de recursos. El proceso de ordenamiento consiste en tomar el primer documento relevante retornado por la primer fuente de información seleccionada como el primer elemento a fusionar en la lista, luego el primer elemento de la segunda fuente seleccionada, y así sucesivamente [Banchero, 2010]. La ventaja de esta solución es que su implementación es sencilla, pudiendo ser utilizada tanto en ambientes cooperativos como no cooperativos, y además, permitiendo el fusionado de valores que no pueden ser comparados directamente. Por otro lado presenta una desventaja ya que sus resultados por lo general no son muy precisos.

5 Diseño del sistema

5.1 Introducción

Como fue indicado al principio en los objetivos (*Sección 2*) la meta del presente trabajo es demostrar que distintos dispositivos móviles pueden interactuar conformando un sistema de Recuperación de Información Distribuida (*Sección 4.2*) utilizando, en cada uno de ellos, técnicas de Recuperación de Información clásicas (*Sección 4.1*) para administrar la colección de documentos contenida dentro de los mismos. A los efectos de validar la hipótesis planteada se diseñó un prototipo de SRID, que implementa una arquitectura propuesta y soporta una experimentación exhaustiva.

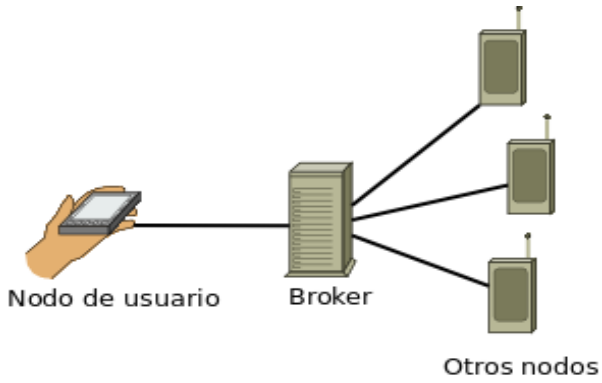
El sistema distribuido esta diseñado bajo las bases de la arquitectura de un SRID tradicional, más específicamente tomando como referencia la arquitectura del sistema presentado en [Bawa et al., 2003], pero con una gran particularidad que se debe a que se compone por dispositivos móviles que actúan tanto de usuarios — rol de cliente en el sistema — como de repositorios — rol de servidor —, interactuando todos ellos con un broker alojado en un servidor, posibilitando que los usuarios de los móviles compartan información con sus pares. Esta arquitectura define una red *Peer-to-Peer*[61] *híbrida*[Milojicic et al., 2002]. Un sistema de este tipo, a diferencia de un *P2P puro*, combina nodos, aplicaciones de usuarios comunes que actúan tanto de clientes como de servidores, con supernodos, servidores que realizan tareas de coordinación de los nodos y administración de meta-data.

El prototipo de sistema esta ideado para la interacción de diversos dispositivos móviles con un solo *broker*. Cada nodo es un repositorio, ya que administra su colección de documentos

61 Se refiere a una clase de sistema y aplicaciones que emplean recursos distribuidos para cumplir un objetivo o realizar una funcionalidad de una manera descentralizada. Los nodos efectúan tareas tanto la tarea de cliente como de servidor.

almacenada en la memoria del mismo. Por otro lado, el broker, centraliza cierta información necesaria para la coordinación de los nodos con el objetivo de que puedan compartir documentos operando todo el conjunto como un solo repositorio distribuido. Es por eso que se espera que el SRID en cuestión se comporte de una manera similar, al momento de la recuperación, a un sistema de RI centralizado en el cual su colección administrada está conformada por todos los documentos administrados por todos los nodos del sistema. También se espera que, en cuanto al rendimiento de Recuperación de Información del sistema, prevalezca la Precisión (*Sección 4.1.2.1.1*) por sobre la Exhaustividad debido a que algunos documentos no serán accesibles debido la etapa de selección de recursos (*Sección 4.2.2*).

A continuación se especifica por separado las características del nodo y el broker, y luego se explica de forma detallada la interacción entre los mismos.



Dibujo 7: Diagrama del sistema distribuido

5.2 Nodo

El nodo es un proceso de aplicación que se ejecuta en dispositivos móviles. La aplicación es compatible con sistemas Android (*Sección 3.4.1*) y puede ser utilizada en *smartphones* que funcionen con dicho sistema operativo. Esta aplicación tiene las mismas características de una Recuperación de Información clásica llevando a cabo la representación, organización, almacenamiento y acceso de los ítems de información.

Dado que el *smartphone* se está convirtiendo en la herramienta principal para el acceso a la información (*Sección 3.1*), es evidente que el usuario poseerá en el almacenamiento del dispositivo

un conjunto de documentos de texto. La aplicación entonces brindará la funcionalidad e interfaz para el acceso a dichos documentos a través de la expresión de consultas. Vale la pena remarcar que el sistema de Recuperación de Información está diseñado, en principio, para textos escritos en lenguaje castellano (*Sección 5.2.3*).

A continuación se describen las distintas características de la aplicación del dispositivo móvil según el enfoque de Recuperación de Información a través de los distintos conceptos mostrados dentro de la *Sección 4.1* .

5.2.1 Representación de los documentos

La aplicación en el dispositivo móvil utiliza como modelo de recuperación el *modelo vectorial* (*Sección 4.1.1.1*). La fácil implementación de este modelo y su buen rendimiento han sido las principales características por las que se optó el mismo.

Por otra parte, la asignación de pesos de los términos utilizada corresponde a la *frecuencia normalizada* (S*ección 4.1.1.1.2*), la cual se escogió por sobre *tf-idf* ya que su cálculo requiere un menor procesamiento. Además se desea llevar a cabo un proceso de indexación incremental para los casos en los que se requiere agregar nuevos documentos a una colección previamente indexada, y el uso de tf-idf perjudica el rendimiento de dicha tarea ya que el calculo de la inversa de la frecuencia de documentos precisa una recalculo para todos los términos incluidos previamente en el índice que tienen ocurrencias en los nuevos documentos a indexar.

A la hora de satisfacer necesidades de información expresadas en consultas, la *métrica de similitud* empleada es la del *coseno* (*Sección 4.1.1.1.1*) ya que es la más utilizada en las aplicaciones de *modelo vectorial*. Vale la pena aclarar que a efectos mejorar la eficiencia, la sumatoria del cuadrado de la frecuencias de cada término correspondiente al cálculo de la norma del vector del documento — es el $|\vec{d}_j|$ en la *Ecuación* ***(1)*** — es calculada y almacenada en el proceso de indexación. De esta forma no se precisa de su cálculo cada vez que se realizan las consultas, ya que para la realización del mismo se requiere la obtención de los pesos de todos los términos del documento, y de esta manera sólo basta con la obtención de pesos de los términos que componen la consulta.

5.2.2 Estructura de datos y almacenamiento

El índice del sistema de recuperación de información del Nodo se encuentra soportado por una *estructura de datos* del tipo correspondiente a un *archivo invertido* (*Sección 4.1.4.1*). La estructura almacena un índice invertido de frecuencias pertinente a una *matriz término-documento*.

En principio se optó por la implementación del índice invertido en un modelo relacional [Grossman et al., 1997] soportado por el motor de base de datos SQLite (*Sección 3.4.1.1*). Pero al ser sometido a las pruebas correspondientes a la experimentación (Sección *6.2.2*) se evidenció una baja performance en tiempos de respuestas.

Debido a que los resultados de la implementación en un base de datos no fueron los esperados y dado que el usuario del dispositivo móvil no almacena grandes colecciones de documentos (*Sección 6.1*), finalmente se decidió por almacenar el índice invertido en un archivo y cargar su contenido en memoria principal al momento de la inicialización de la aplicación. De esta forma la carga inicial de la aplicación precisará de mayor procesamiento y se llevará a cabo en más tiempo que de la otra forma, pero como muestra más adelante la experimentación, la performance mejora considerablemente.

5.2.3 Proceso de Indexación

El proceso de indexación es la parte más importante del funcionamiento de recuperación de información del Nodo. Se implementa un *enfoque no lingüístico* (*Sección 4.1.3.1*) ya que los *enfoques lingüísticos* (*Sección 4.1.3*) aún no están muy desarrollados y precisan mayor poder de procesamiento que los primeros.

La aplicación realiza la indexación mediante el procesamiento de documentos de texto alocados en un directorio dentro del sistema de archivos del dispositivo móvil, dicha locación puede ser indicada por el usuario mediante configuración. Mientras el sistema esté activo realizará periódicamente una comparación entre el contenido del directorio y el del índice invertido. Con el objetivo de mantener sincronizado a los mismos notificará al usuario cuando existe alguna diferencia, y éste último podrá, en el momento que desee, iniciar la actualización del índice. Dicha

actualización puede comprender tanto una indexación incremental como la eliminación de ciertos documentos que ya no están presentes en el documento.

El pre-procesamiento de los documentos de texto respeta las etapas presentadas en la *Sección 4.1.3.2* . Primero se realiza un *análisis léxico* convencional (*Sección 4.1.3.2.1*) eliminando signos de puntuación, tildes, quitando números, y palabras de menos de dos caracteres, y normalizando capitalizaciones. Luego se realiza una eliminación de *palabras vacías* (*Sección 4.1.3.2.2*) para el idioma castellano, corriendo el riesgo que en el caso que se procesan textos en otros idiomas no se obtendrá una adecuada representación de los mismos.

No se aplica la técnica de *stemming* (*Sección 4.1.3.2.3*) en el pre-procesamiento ya que, además de que la transformación y recuperación de la raíz semántica de una palabra requiere de un procesamiento adicional, la aplicación de la misma mientras mejora la exhaustividad, a su vez, disminuye la precisión, y se aclaró anteriormente (*Sección 5.1*) que se busca predominar esta última nombrada por sobre la primera.

Por último se realiza una selección de términos a indexar (*Sección 4.1.3.2.4*), el objetivo es indexar aquellos presentes en la parte central de la gráfica de frecuencias (*Sección 4.1.3.1.1*) descartando la cabeza de la misma, correspondiente a los más frecuentes, y la cola que indica aquellos términos muy poco frecuentes. La quita de aquellos términos con mayor ocurrencia se supone realizada en la eliminación de las palabras vacías mientras que lo que es necesario realizar en esta etapa es el descarte de aquellos términos poco frecuentes, es por eso que se rechazan los que poseen menos de dos ocurrencias en el documento.

5.2.4 Proceso de consulta

El usuario a través del nodo puede realizar una consulta. La misma tiene dos fases de procesamiento, uno local y otro remoto:

- El procesamiento local corresponde a la funcionalidad de consulta de un sistema de Recuperación de Información convencional, el cual compara mediante una métrica de similitud (*Sección 4.1.1.1.1*) la representación lógica de la consulta con la de los

documentos contenidos en la colección administrada por el mismo.

- El procesamiento remoto es el que involucra al broker y los otros nodos que componen el SRID. Mediante la interacción con el broker el nodo puede emitir la consulta (*Sección 5.4.1*) y esperar los resultados de otros nodos. Éstos últimos reciben la consulta desde el broker (*Sección 5.4.2*) y deben entonces realizar solamente el procesamiento local y retornar sus resultados a éste, el cual los fusionará y enviará al nodo emisor.

Vale la pena remarcar que una vez finalizado el procesamiento remoto, cuando el nodo emisor de la consulta obtiene sus resultados, puede obtener acceso tanto a documentos que estén presentes en la colección que se almacena en el mismo dispositivo como otros documentos que pueden estar almacenados en repositorios remotos, realizando la obtención de los contenidos de estos mismos mediante una interacción directa entre nodos (*Sección 5.4.3*).

5.3 Broker

Esta entidad del sistema respeta las bases indicadas en la *Sección 4.2* para la arquitectura de un *Sistema de Recuperación de Información Distribuido*. El mismo esta desarrollado bajo el lenguaje de programación Java y desplegado en un servidor de aplicación con tecnología *Apache Tomcat*[62], ofreciendo una interfaz de *Web Service*[63] para la interacción remota con los dispositivos móviles.

Administra una descripción de recursos correspondiente al contenido de los índices que manejan los nodos empleando, utilizando el formato de una *Matriz de descripción de recursos con la frecuencia de documentos* (*Sección 4.2.1*). Cuando recibe una consulta y procede con la *selección de repositorios* (*Sección 4.2.2*) emplea para dicha tarea mediante el algoritmo de CORI (*Sección 4.2.2.1*), para luego redirigir la consulta a los nodos indicados por dicho algoritmo. Éstos últimos retornan el ranking de documentos correspondiente a la consulta, y el broker pasa a fusionar dichos resultados pudiendo utilizar tanto la metodología de *Round Robin* (*Sección 4.2.3*) como la de un ordenamiento de documentos por valor de relevancia — dicho valor otorgado por el mismo nodo

62 http://tomcat.apache.org/
63 Es un método de comunicación a través de dos dispositivos electrónicos a través de la web (http://www.w3.org/TR/wsa-reqs/)

que lo suministró —.

5.4 Interacción entre las partes

Por definición “un sistema distribuido es aquel en el cual componentes ubicados en computadoras interconectadas a través de una red se comunican y coordinan sus acciones sólo mediante el intercambio de mensajes. La principal motivación de la construcción de sistemas distribuidos es la de compartir recursos.”[Colourius, 2005]. El Sistema de Recuperación de Información Distribuido en cuestión deberá, para cumplir su cometido, llevar a cabo una serie de intercambios a través de interacciones de distinto índole. Dichas interacciones se definen según qué componente toma el papel de cliente y qué otro el de servidor, existiendo tres tipos que son profundizados a continuación: Nodo-Broker, Broker-Nodo y Nodo-Nodo.

5.4.1 Nodo-Broker

Esta interacción es la principal del sistema, los distintos eventos de la misma están definidos mediante en el contrato del Web-Service presente en el broker. El intercambio de datos se realiza con el protocolo de aplicación HTTP[64] a través de mensajes definidos en el estándar SOAP[65]. Cada mensaje se corresponde con la solicitud o la respuesta de uno de los siguientes eventos definidos en el servidor:

- Alta de un nodo (mensaje sayHello): un nodo se registra al broker mediante este mensaje. En los parámetros indica datos básicos como el nombre del usuario, puerto de escucha para la recepción de conexiones y ciertas preferencias de configuración. Como respuesta el broker, que administra los nodos de sistema, le asigna un id único el cual precisará el nodo para los posteriores mensajes.

- Alta de la descripción del recurso (mensaje putIndex): luego de registrarse, el nodo

64 Hiper-text Transfer Protocol (http://www.faqs.org/rfcs/rfc2616.html)
65 Service Oriented Architecture Protocol (http://www.w3.org/TR/wsdl)

envía al broker la descripción de su colección de documentos (*Sección 4.2.1*) la cual será consultada por este último a la hora se hacer la selección de repositorios.

- Actualización de la descripción de recurso (mensaje updateIndex): cuando la colección del nodo se modifica y el usuario realiza la correspondiente actualización del índice, es necesario que se notifique al broker de tal cambio a efectos de mantener consistente la información que contiene el mismo. Mediante este evento el nodo, luego de una actualización de la colección y el índice, envía las variaciones correspondientes de su descripción del recurso para que el broker mantenga dicha información consistente.

- Consulta (mensaje query): una consulta arriba al broker pudiendo provenir tanto de un nodo como de un cliente externo. Este mensaje es el principal ya es que define e implementa el núcleo de la funcionalidad del sistema, y por ende es aquel que requiere de mayor esfuerzo de procesamiento y coordinación de las partes. A partir de la consulta se realiza la selección de repositorios a los cuales consultar (*Sección 4.2.2*). Una vez seleccionados se deriva la consulta y se esperan los resultados de cada nodo para la recolección y fusión de los mismos (*Sección 4.2.3*). La lista final fusionada se retorna en forma de ranking al emisor de la consulta.

- Baja de un nodo (mensaje sayBye): mediante este evento un nodo notifica de manera formal que se desconecta de la red, de esta manera el broker no le derivará más consultas.

5.4.2 Broker-Nodo

Esta interacción se lleva a cabo solamente a través del mensaje de consulta definido en la Sección anterior (*Sección 5.4.1*), ya que se trata del caso en el que el broker deriva la consulta a los nodos seleccionados. Aquí se invierten los papeles y es el nodo el que oficia de servidor y el broker de cliente emitiendo la consulta y esperando los resultados del primero. La interacción se realiza mediante TCP[66] y es posible ya que el nodo al registrarse indica el puerto de escucha en el cual

66 Transmission Control Protocol (http://www.rfc-es.org/rfc/rfc0793-es.txt)

espera este tipo de mensajes. La definición del mensaje en sí es muy simple ya que se trata del envío de la palabra QUERY seguido de la secuencia de términos que conforman la consulta. La respuesta del nodo es en datos carentes de formato, secuencia de bytes, que se corresponden a un *objeto Java serializado*[67] el cual representa el ranking de resultados de ese nodo y que es interpretado por el broker.

5.4.3 Nodo-Nodo

Similar a la interacción anterior (*Sección 5.4.2*), ésta también posee una definición muy simple. Se lleva a cabo cuando el usuario del nodo emisor de una consulta, que realizó un procesamiento remoto (*Sección 5.2.4*), selecciona visualizar entre los documentos que componen la respuesta obtenida un recurso que se encuentra en otro repositorio — en la colección de otro nodo — . Es entonces cuando se precisa obtener el contenido del archivo mediante una interacción directa. Cada documento contiene información referente la dirección IP y puerto a donde se puede realizar la petición y obtención de dicho archivo. El nodo cliente emite un mensaje con la clave GET y el identificador del documento en cuestión, a través de una simple conexión TCP, transmite el archivo en cuestión.

67 Conversión de un objeto alojado en memoria principal en un conjunto de bytes para la transmisión de los mismos por la red o para almacenamiento en disco, posibilitando la reconstrucción de dicho objeto a través de dichos datos (http://java.sun.com/developer/technicalArticles/Programming/serialization/).

6 Experimentos

En esta sección se presentan los distintos experimentos realizados para la evaluación de dos aspectos principales:

- Viabilidad de administración de colecciones de documentos de texto en un dispositivo móvil a través de técnicas de Recuperación de Información clásicas.

- Performance de recuperación de un Sistema de Recuperación de Información Distribuido para dispositivos móviles.

Para la experimentación y evaluación se formaron dos colecciones de prueba las cuales intentan simular los datos que contiene un usuario en su dispositivo móvil y que son caracterizadas en la siguiente sección. En las secciones subsiguientes se procede con la explicación de los experimentos, caracterización de los recursos utilizados y los resultados obtenidos.

6.1 Caracterización de los datos

Para la realización de los experimentos sobre el prototipo desarrollado se utilizaron dos colecciones de documentos de texto. Una corresponde a un conjunto de artículos de noticias de la fuente *El Universal de México*[68] — *UMEX* de aquí en adelante —, los cuales fueron capturados en el mes de marzo del 2011 a través de su sitio gracias a al servicio de suscripción RSS[69] que ofrece el mismo. La otra colección corresponde a una recopilación de correos electrónicos del año 2009 que se intercambiaron ente los miembros de la comunidad *GRULIC*[70] a mediante su lista de correo.

68 http://www.eluniversal.com.mx/

69 Really Simple Syndication. Es una familia de formatos de fuentes de contenidos web utilizados para publicar frecuentemente la actualización de información en una forma estandarizada a aquellos que se suscribieron a dicho servicio de sindicación.

70 Grupo de Usuarios de Software Libre de Córdoba (http://www.grulic.org.ar/)

Colección	Cantidad documentos	Tamaño promedio documentos
GRULIC	3240	1,78 KB
UMEX	974	2,65 KB

Tabla 6: Tamaño de colecciones de pruebas en cuanto a cantidad de documentos y tamaño promedio de los mismos

Cada colección es utilizada para la experimentación del sistema distribuido, pero para los experimentos referentes al desempeño individual del dispositivo móvil se tomo una subconjunto de documentos de cada colección para formar una versión reducida de las mismas, escogiendo 500 documentos de manera aleatoria — GRULIC' y UMEX' — ya que se considera un número suficiente de archivos de texto que puede almacenar un usuario en su dispositivo. A continuación se brinda un análisis desde varios enfoques de ambas colecciones de texto utilizadas.

6.1.1 Distribución de las palabras

Para el estudio de la distribución de las palabras de ambas colecciones se procedió con el análisis de Zipf *(Sección 4.1.3.1.1)* el cual a través de su representación en un gráfica y con la estimación del parámetro α aporta valiosa información.

Para la colección de GRULIC se obtuvo un valor de $\alpha=1,01$ mientras que para *UMEX un* $\alpha=0,96$ lo cual indica que la última tiene una distribución de frecuencias un poco más equitativa, probablemente por el uso más rico de términos en la redacción de los artículos de noticias. Los términos incluidos en la cabeza de la gráfica corresponden a las palabras vacías y los de cola son términos poco influyentes e incluso algunos escritos con faltas ortográficas. Todos estos términos presentes en estas dos secciones mencionadas son descartados en el proceso de indexación (*Sección 4.1.3*). Aquellos términos que se encuentran en la parte central de la gráfica y que se posan sobre la recta son los preferibles para formar parte del vocabulario.

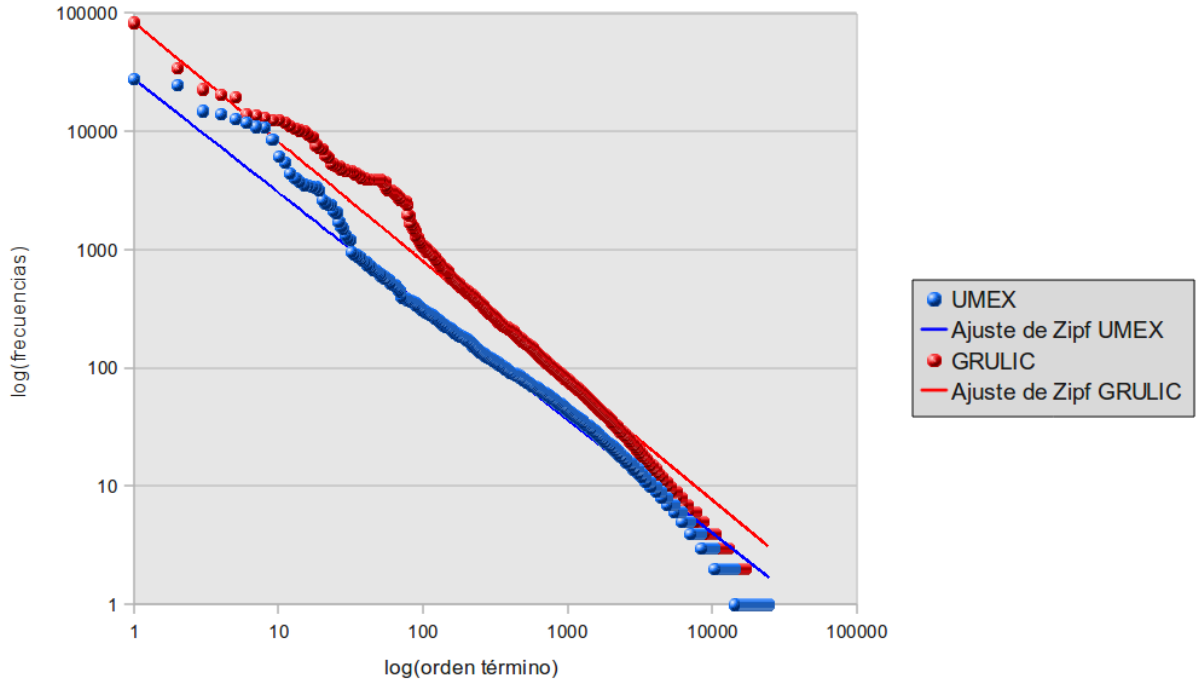


Gráfico 14: Distribución de las palabras

6.1.2 Crecimiento del vocabulario

El tamaño del vocabulario de una colección de documentos se mide por la cantidad de términos únicos existentes en la misma. Una forma de predecir dicho crecimiento es mediante la Ley de Heaps (*Sección 4.1.3.1.2*).

En los gráficos se muestra el crecimiento del vocabulario por cada documento de las colecciones GRULIC y UMEX, junto con el ajuste de la ley de Heaps. Para GRULIC los parámetros que describen el crecimiento son $k = 11{,}35$ y $b = 0{,}67$. Mientras que para UMEX son $k = 35$ y $b = 0{,}71$.

Los parámetros no coinciden completamente con el rango estándar (ver *Sección 4.1.3.1.2*) debido a que las colecciones son relativamente pequeñas y además en las mismas predominan los documentos cortos (*Sección 6.1.3*).

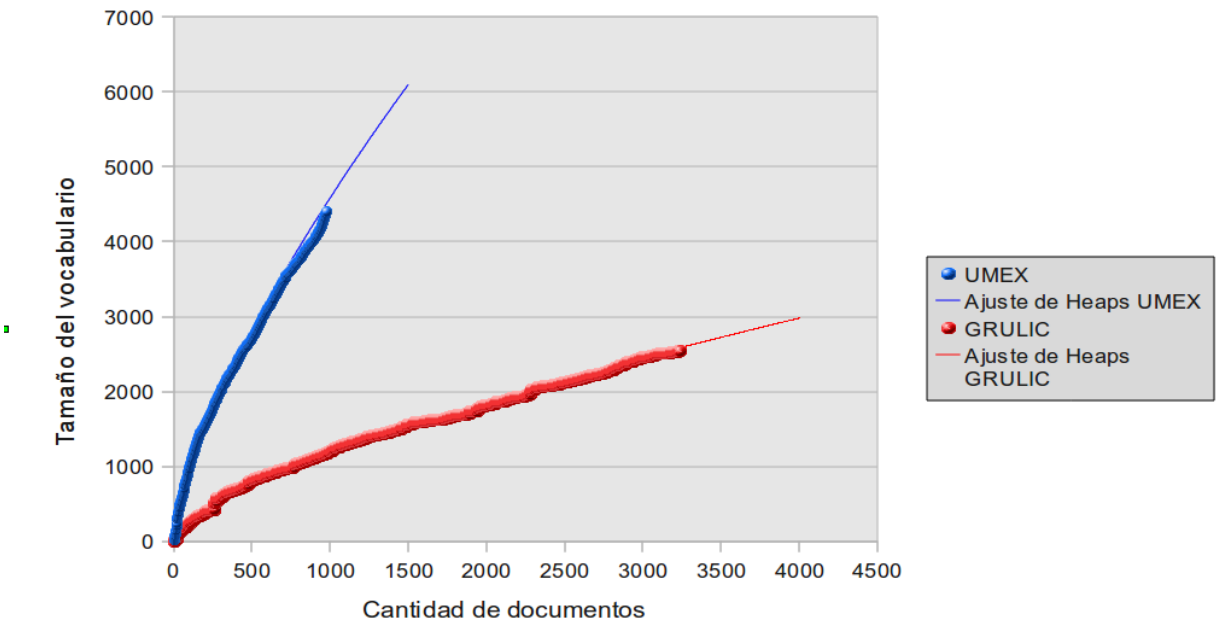


Gráfico 15: Crecimiento del vocabulario

6.1.3 Distribución del tamaño de los documentos

Este análisis es de utilidad para determinar las características de los documentos de una colección en cuanto al tamaño de los mismos. Se procedió con el estudio definiendo intervalos de 50 bytes y luego clasificando cada documento en uno de ellos. A continuación se presentan los gráficos para las colecciones en los que se puede apreciar que, en ambas, los documentos son cortos ya que predominan los valores de bajo tamaño.

Como es apreciable en el *Gráfico 16*, la distribución del tamaño de los documentos para ambas colecciones no ajusta con una distribución normal como se esperaba. En cambio presentan una considerable asimetría positiva ya que la media se ubica en un valor superior a aquel con mayor ocurrencia (Moda). Esto se corresponde únicamente al hecho de que no se trata de colecciones largas, con un gran número de documentos.

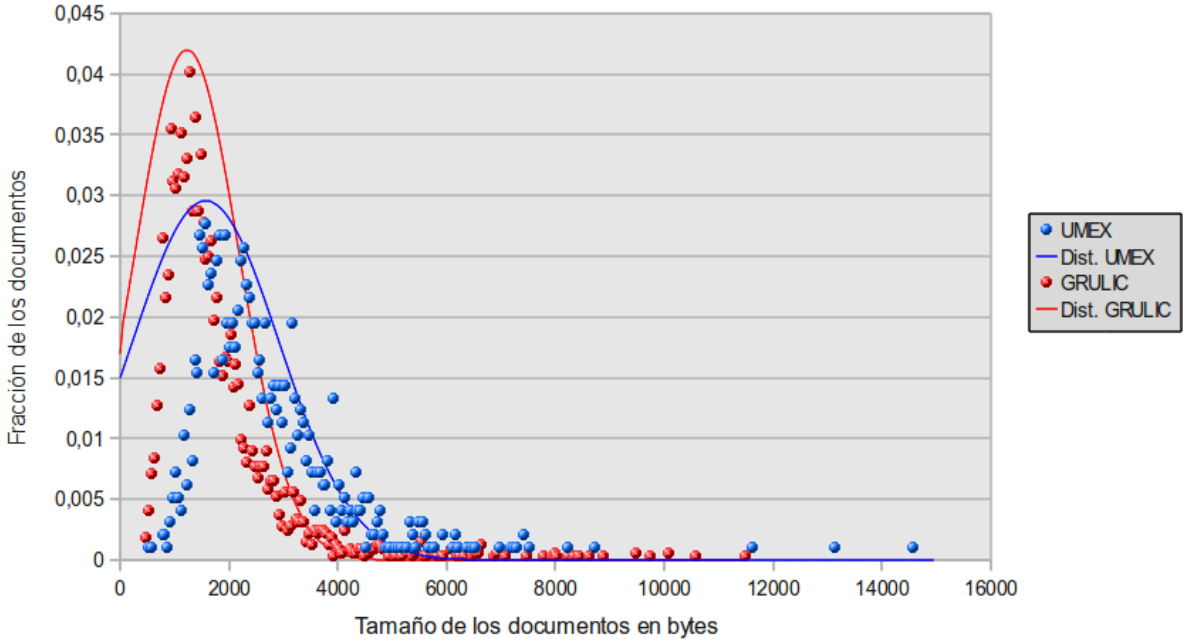


Gráfico 16: Distribución del tamaño de los documentos

6.2 Capacidad del dispositivo móvil para Recuperación de Información

El primer experimento intenta poner a prueba la capacidad de un dispositivo móvil para administrar una colección de documentos de texto a través de la utilización de técnicas de Recuperación de Información clásicas. Las colecciones de prueba utilizadas no superan los 500 documentos ya que se entiende como una cantidad razonable de archivos que puede tener almacenados ocasionalmente un usuario en su dispositivo. Dichas colecciones se tratan de subconjuntos (*Sección 6.1*) de las dos analizadas anteriormente.

El procedimiento de esta prueba tiene dos partes bien definidas en las cuales, en ambas, se mide la performance del prototipo en el dispositivo móvil en sus tareas de procesamiento de Recuperación de Información locales. Puntualmente el objetivo es evaluar tanto la performance de indexación como la de resolución de consultas.

6.2.1 Recursos

El set de pruebas se realizó sobre el prototipo de software para smartphones Android instalado en tres dispositivos de distintas características. Los recursos utilizados fueron una PC de escritorio — en la cual se ejecutó la aplicación a través de un contexto emulado que ofrece la SDK de Android (*Sección 3.4.1.2*) — y dos *smartphones* actuales de distintas características — uno de la línea *Motorola*[71] perteneciente al modelo *Milestone 2*[72] y otro de la línea LG[73] modelo *P350 Optimus*[74]— con sistema operativo Android. A continuación se presenta una tabla con las características de los dispositivos correspondientes.

Dispositivo	Sistema	Microprocesador	Memoria	Denominación
Emulador-PC	Android 2.2[75] (Emulado)	AMD Athlon 3500	128 MB (Emulados)	EMUPC
Milestone 2	Android 2.2	TI OMAP[76], 1000 MHz	512 MB	DROID2
LG P350 Optimus	Android 2.2.2	ARM[77], 600 MHz	140 MB	LGP350

Tabla 7: Dispositivos utilizados en el primer experimento

6.2.2 Indexación

El proceso de pruebas de indexación de colecciones de documentos se realizó utilizando dos corpus distintos, GRULIC' y UMEX' (*Sección 6.1*). El procedimiento de indexación de dichas colecciones se hizo de forma incremental, aumentando el subconjunto de documentos a indexar de a 20 por vez hasta abarcar la completitud de documentos, registrando en cada iteración el tiempo en el que se llevó a cabo el proceso. A su vez, dicho procedimiento fue ejecutado reiteradas veces para obtener una medida general de la performance de los dispositivos.

En un primer diseño del prototipo se ideó el archivo invertido (Sección *4.1.4.1*) volcado en un modelo relacional [Grossman et al., 1997] en el motor de base de datos SQLite (Sección *3.4.1*)

71 http://www.motorola.com/Consumers/AR-ES/Home
72 http://geekaphone.com/phones/Motorola-DROID-2-specs (especificación técnica)
73 http://www.lgmobile.com/
74 http://www.gsmarena.com/lg_optimus_me_p350-3735.php (especificación técnica)
75 http://developer.android.com/sdk/android-2.2.html
76 Texas Instruments OMAP Mobile Processors (http://www.ti.com/general/docs/gencontent.tsp?contentId=46946)
77 Advanced RISC Machine (http://www.arm.com)

ofrecido por la plataforma Android. Se llevó a cabo la implementación correspondiente y se procedió con el experimento de indexación sobre los dispositivos *EMUPC* y *DROID2*.

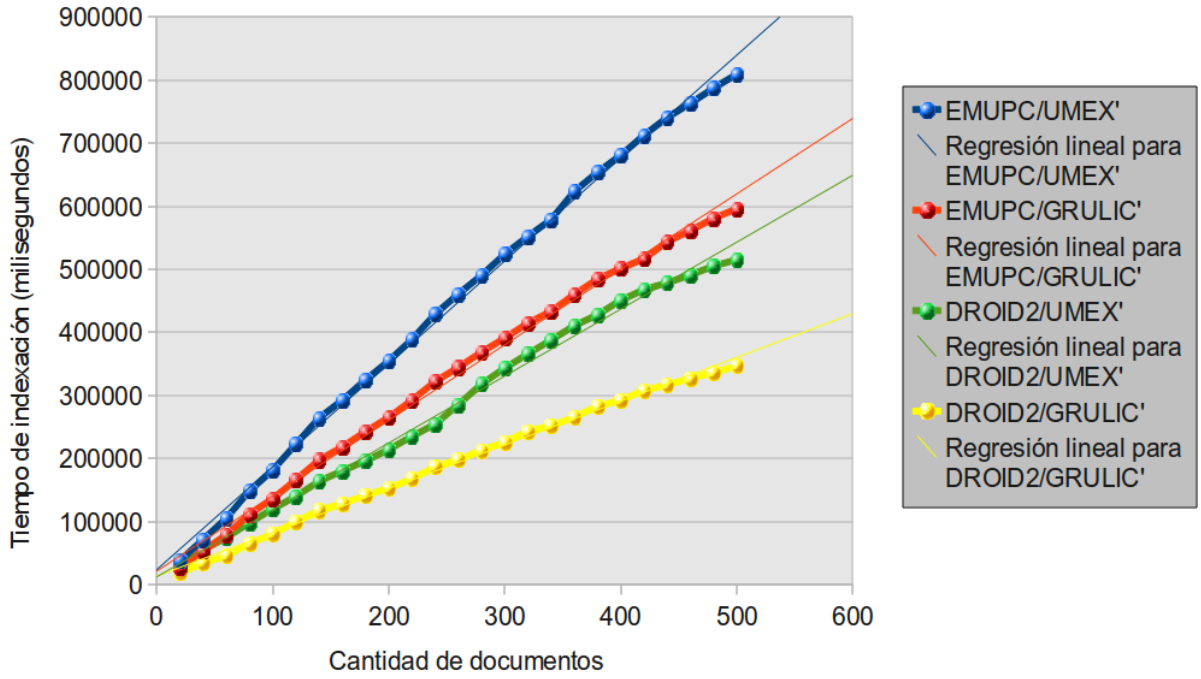


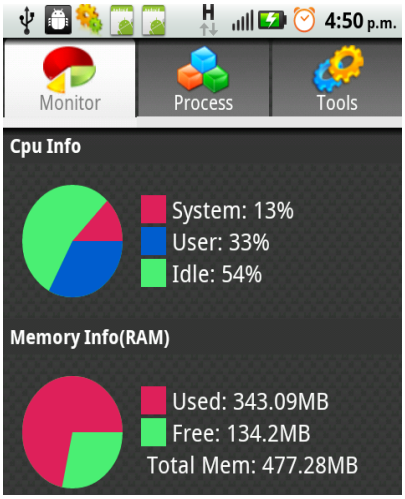
Gráfico 17: Proceso de indexación con utilización de SQLite

EMUPC/UMEX'	f(x) =1632,08x + 23446
EMUPC/GRULIC'	f(x) =1196,65x + 21084
DROID2/UMEX'	f(x) =1061,59x + 12064
DROID2/GRULIC'	f(x) =693,69x + 13245

Tabla 8: Ecuaciones de comportamiento de indexación con la utilización de SQLite

Como se muestra en *Gráfico 17* (datos disponibles en *Sección 9.1*) la performance del dispositivo DROID2 no se aleja mucho de la de EMUPC, registrando un tiempo de un poco más de 8 minutos – 515186 milisegundos – para la indexación completa de UMEX' y unos casi 6 minutos —346515 milisegundos— con respecto a GRULIC'. La gráfica se presenta con una forma lineal, por lo que a través de su tendencia se puede decir que para la colección UMEX' en el dispositivo DROID2 se estima un tiempo de indexación a razón de un segundo por documento, mientras que para GRULIC' sobre el mismo dispositivo se obtuvo un tiempo de 700 milisegundos por cada documento indexado. Éstos tiempos estimados dependen de las características del dispositivo y de la naturaleza de las colecciones de documentos.

Los resultados obtenidos evidencian una baja performance por un “cuello de botella” procedente del acceso a la base de datos SQLite, dejando ociosa la CPU por varios ciclos como muestra el *Dibujo 8*. Esto derivó en una indexación de los documentos de texto realizada en un lapso de considerables minutos.



Dibujo 8: Uso de CPU y memoria durante indexación

Por dichos motivos se modificó la implementación del índice invertido reemplazando la utilización del motor de base de datos por el simple manejo del mismo en el sistema de archivos. De esta manera se accede a la memoria no volátil durante el inicio de la aplicación para cargar las estructuras de datos en memoria RAM, y se espera de esta manera una mejora en la performance de indexación ya que se realiza el acceso al sistema de archivos sólo al momento de la finalización de este proceso. Para la experimentación sin utilización de SQLite se incorporó el *smartphone* LGP350 al set de dispositivos de pruebas.

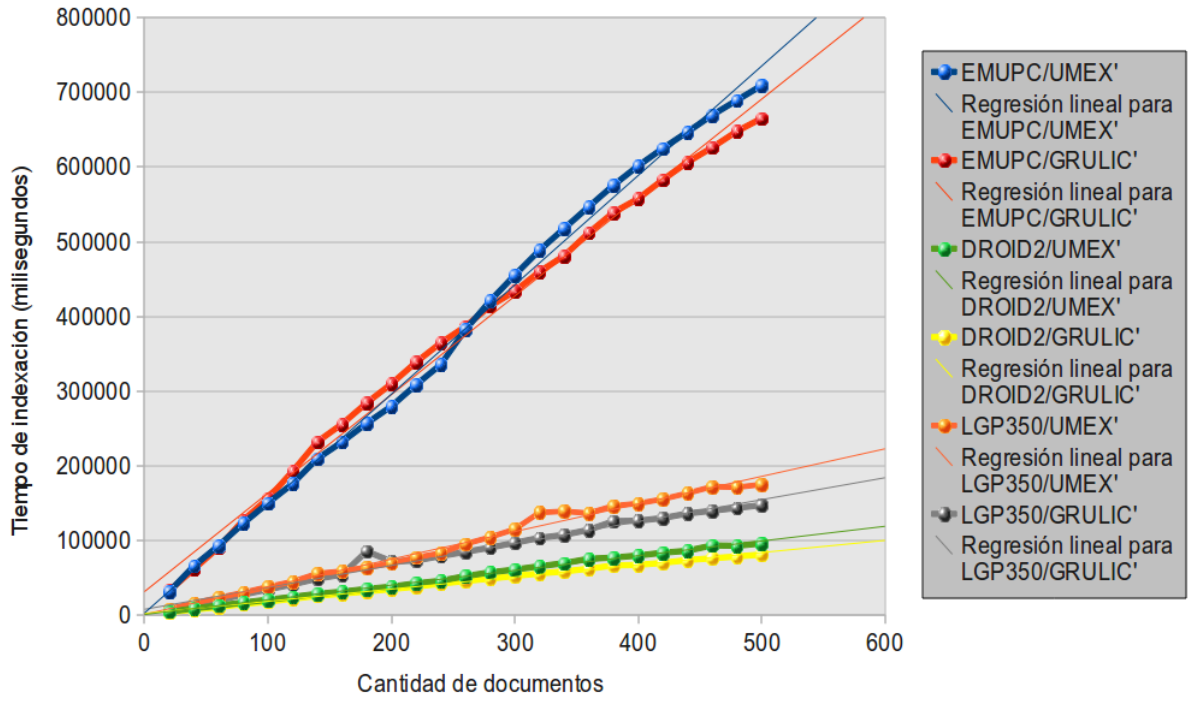


Gráfico 18: Proceso de indexación sin utilización de SQLite

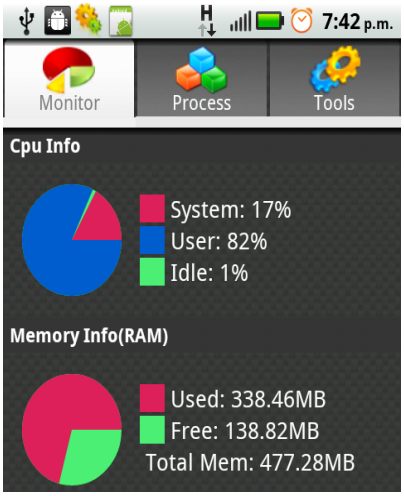
EMUPC/UMEX'	f(x) =1465,12x + 2584
EMUPC/GRULIC'	f(x) =1319,09x + 31278
DROID2/UMEX'	f(x) =196,95x + 556
DROID2/GRULIC'	f(x) =163,21x + 1959
LGP350/UMEX'	f(x) =369,44x + 886
LGP350/GRULIC'	f(x) =293,8x + 7519

Tabla 9: Ecuaciones de comportamiento de indexación sin la utilización de SQLite

Como fue previsto, y como indica el *Gráfico 18* (datos en *Sección 9.2*), efectivamente la no utilización de SQLite derivó en una notoria mejora en la performance de indexación. Sin embargo el cambio fue satisfactorio únicamente para el caso de los dispositivos móviles, ya que en el emulador los tiempos continúan siendo muy altos. Es por esto que se optó por no tener en cuenta los resultados de los experimentos realizados sobre este último ya que se consideran que no reflejan la performance de un dispositivo real debido a sus características de entorno emulado sobre un dispositivo de distintas características y su administración a través de un sistema operativo de distintos fines y naturaleza en comparación al móvil.

Retornando a la caracterización de la información presentada en el *Gráfico 18* se puede decir que, a través de la tendencia de la gráfica para DROID2 con ambas colecciones, los tiempos de indexación se han reducido en un 80% con respecto a los experimentos que implicaron el uso de

SQLite. Esto es debido a que de un proceso de indexación para la colección UMEX' que se llevaba a cabo a un ritmo de un documento por segundo, se pasó a otro muy distinto que registró un medida de 197 milisegundos por documento indexado, lo que significó un tiempo de indexación para la colección completa de 1 minuto y 20 segundos frente a los más de 8 minutos anteriores. De manera similiar sucedió para la colección GRULIC' para la cual su performance de indexación también presento una notoria mejora. Sobre el comportamiento de *LGP350* se puede decir que su performance es similar al de *DROID2* aunque menor, naturalmente debido a que se trata de un dispositivo con menor capacidad de procesamiento que el anterior.



Dibujo 9: Uso de CPU y memoria durante proceso indexación sin utilización de SQLite

En el *Dibujo 9* se puede apreciar la mejora en la utilización de ciclos de procesador durante el proceso de indexación sin el uso de SQLite para albergar el índice invertido. De aquí en adelante todos los experimentos que involucran al prototipo de software utilizan la configuración que presento mejor performance en esta sección.

6.2.3 Resolución de consultas locales

Para la parte de experimentos enfocados en la resolución de consultas se procedió con una serie de necesidades de información generadas al azar formadas con términos presentes en el vocabulario del índice. Se formaron tres conjuntos de los cuales uno corresponde a consultas de un sólo término,

otro de dos términos y el último de tres, teniendo en cuenta que los usuarios habitualmente expresan sus consultas utilizando pocas palabras [Spink et al.,2002].

Este experimento incluye el procedimiento del anterior (*Sección 6.2.2*), pero además de ejecutar la indexación en cada iteración se procede con la ejecución de las consultas. Dicha ejecución dispara varias consultas, 30 de cada una de las tres categorías según cantidad de términos, registra los tiempos de respuesta y luego se calcula la media por categoría.

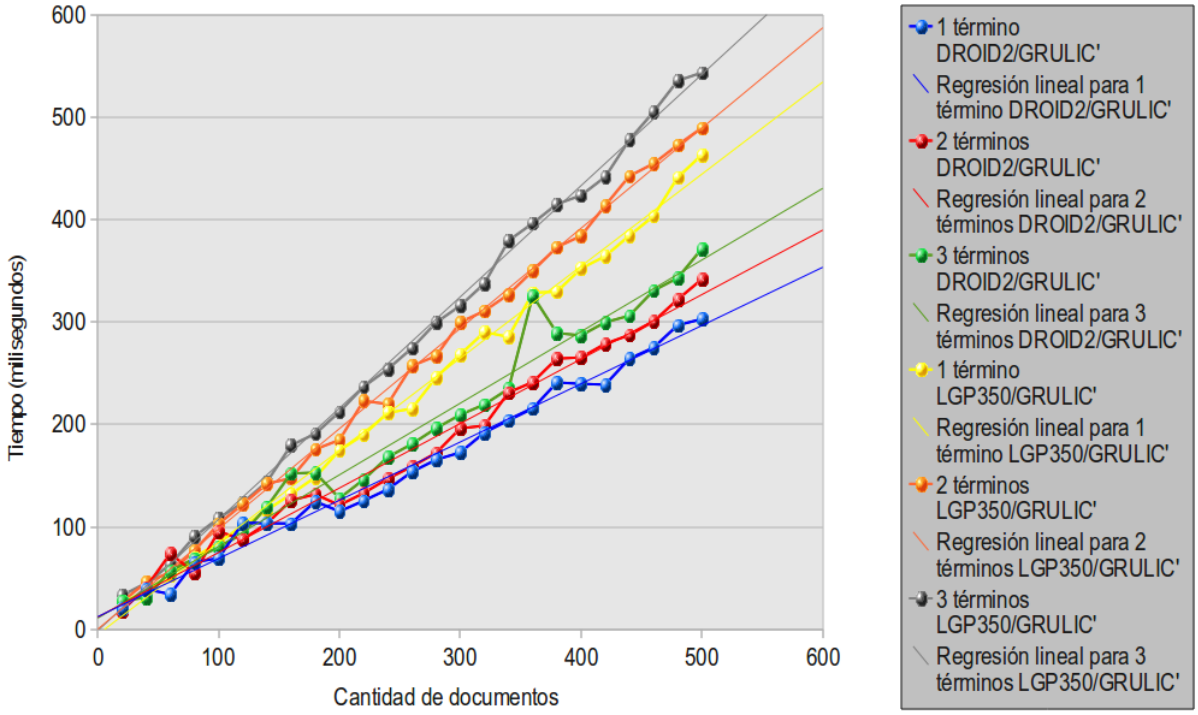


Gráfico 19: Resolución de consultas locales para la colección GRULIC'

Procediendo de esta forma se obtuvieron una serie de resultados que fueron plasmados en estas gráficas (tabla de datos en *Sección 9.3*). Otra vez, al igual que en los resultados de Indexación (*Sección 4.1.3*), es evidente la supremacía en cuanto a rendimiento del dispositivo DROID2 sobre LGP350 — la diferencia en capacidad de memoria y frecuencia del microprocesador entre ambos dispositivos (*Sección 6.2.1*) es plasmada en los resultados de rendimiento—. Por otro lado, se puede apreciar un comportamiento lineal en cuanto a tiempo de respuesta en función al tamaño de la colección. En el *Gráfico 19* correspondiente a GRULIC' se denota cierta irregularidad en el tiempo de resolución promedio de las consultas, rasgo que no se presenta en el *Gráfico 20* de UMEX' debiéndose esto principalmente a la diferencia que existe entre las colecciones en cuanto a su distribución de los términos (*Sección 6.1.1*).

Otro aspecto interesante que se aprecia en los resultados es la variación de la pendiente de la tendencia lineal en relación con la cantidad de términos empleados en la consulta. Es una característica razonable que cuanto más términos se utilicen mayor será el tiempo en el cual se realizará la consulta debido a la mayor cantidad de información que se precisa obtener de la matriz término-documento (*Sección 4.1.4.1*).

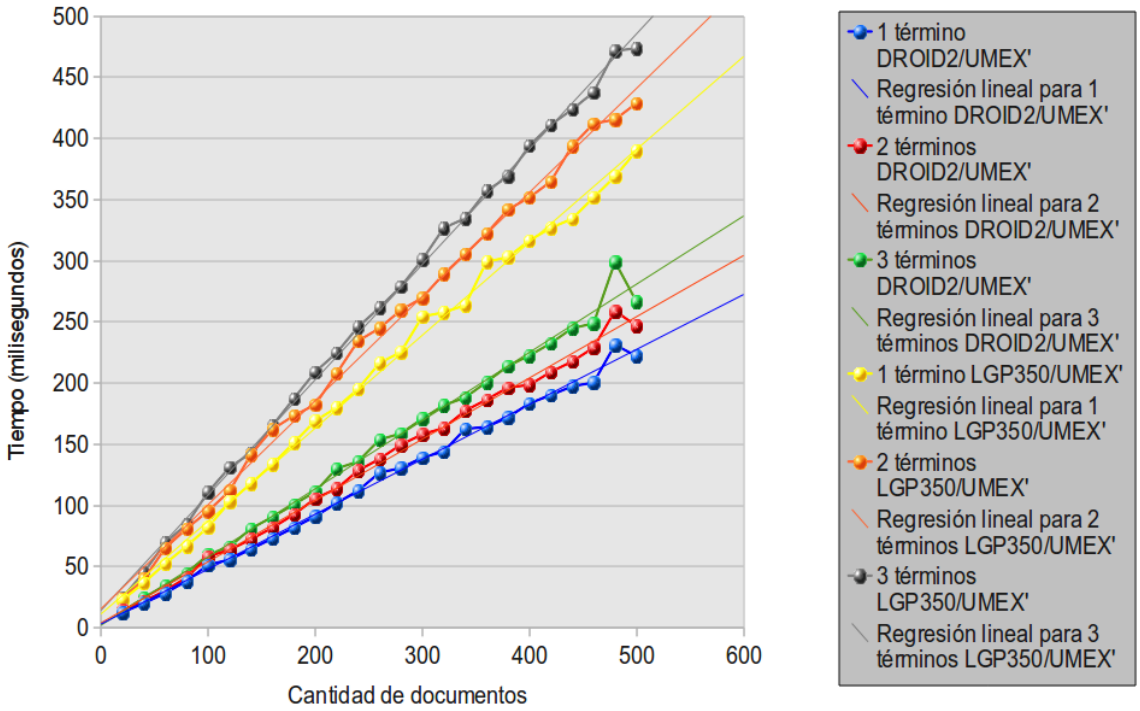


Gráfico 20: Resolución de consultas locales para la colección UMEX'

Un enfoque valioso es el análisis de las gráficas por cantidad de términos de la consulta (*Gráfico 21*, respaldado por los datos en *Sección 9.3*). De esta manera es posible analizar a mayor detalle el rendimiento del par Dispositivo/Colección para cada tipo de consulta. La particularidad que se observa aquí es que las recta correspondientes los tiempos de respuesta de las consultas realizadas en LGP350 para las dos colecciones prácticamente no difieren hasta los 300 documentos, siendo más notoria la diferencia a partir de ese punto. De otra manera, DROID2 muestra una marcada diferencia en las rectas de resolución de consultas en las dos colecciones a través de todos los valores de la abscisa. Se aprecia en las rectas de comportamiento estimadas (*Tabla 10*) que las pendientes correspondientes a cada dispositivo varían de manera similar con respecto a la colección utilizada, a diferencia de las ordenadas al origen que no presentan cierta similitud. Se atribuye esta diferencia de comportamiento a las distintas características de los dos dispositivos móviles empleados.

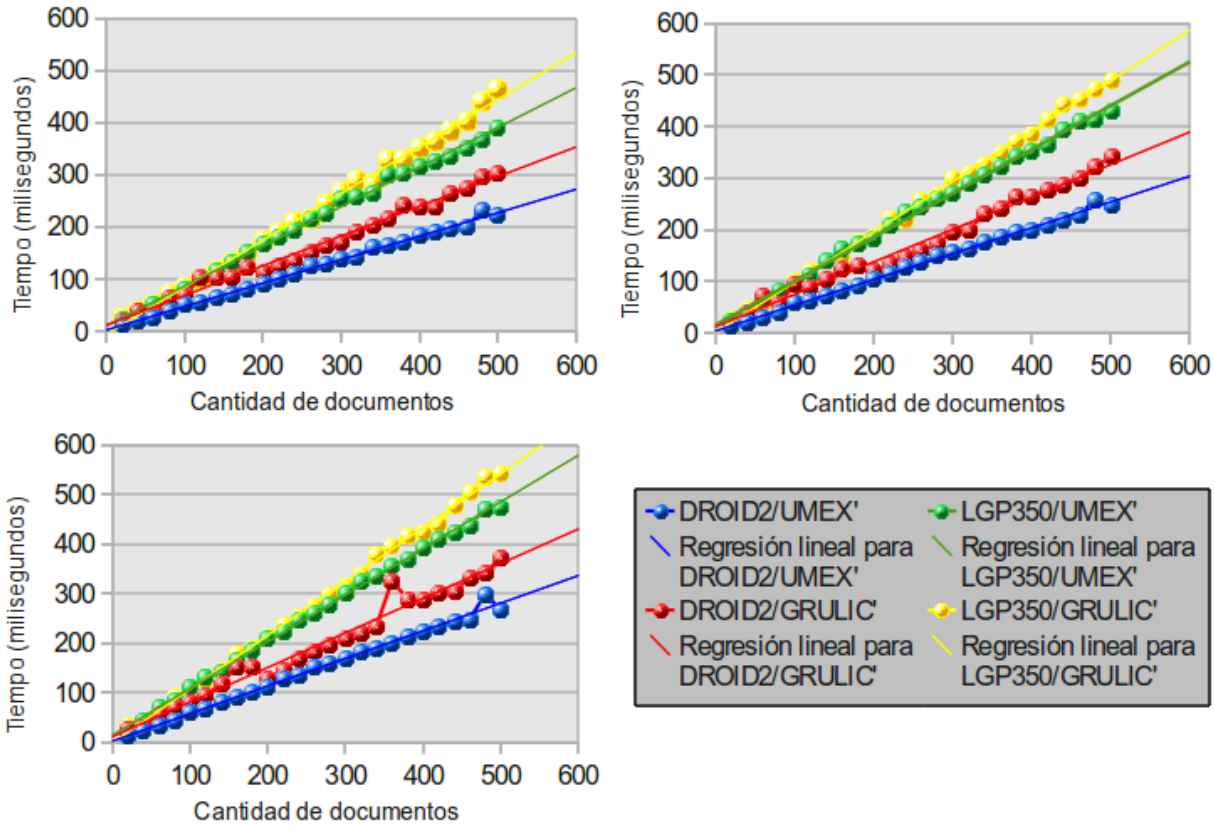


Gráfico 21: Tiempos de resolución de consultas locales agrupadas por cantidad de términos correspondientes a 1 término(esq. sup. izq.), 2 términos(esq. sup. der.) y 3 términos(esq. inf. izq.)

	1 término	2 términos	3 términos
DROID2/UMEX'	f(x) = 0,45x + 3	f(x) = 0,5x + 4	f(x) = 0,56x + 2
DROID2/GRULIC'	f(x) = 0,57x + 12	f(x) = 0,63x + 12	f(x) = 0,7x + 11
LGP350/UMEX'	f(x) = 0,76x + 11	f(x) = 0,85x + 16	f(x) = 0,94x + 14
LGP350/GRULIC'	f(x) = 0,9x – 5	f(x) = 0,98x	f(x) = 1,09x – 1

Tabla 10: Ecuaciones de comportamiento de resolución de consultas locales

Salvando las diferencias en los resultados de los experimentos sobre las dos colecciones por sus distintas características, vale la pena remarcar que la resolución de consultas locales para una colección de alrededor de 500 documentos de texto escritos en lengua castellana sobre un dispositivo móvil de las características de DROID2 (*Sección 6.2.1*), se encuentra en el orden de los 300 milisegundos. Se considera un tiempo de respuesta satisfactorio teniendo en cuenta la naturaleza del equipo (*Sección 3.1*) y se estima que el prototipo ejecutado en dispositivos de estas cualidades puede ser bien utilizado para un cooperación en distribuido con pares semejantes.

6.3 Rendimiento del Sistema de Recuperación de información Distribuido

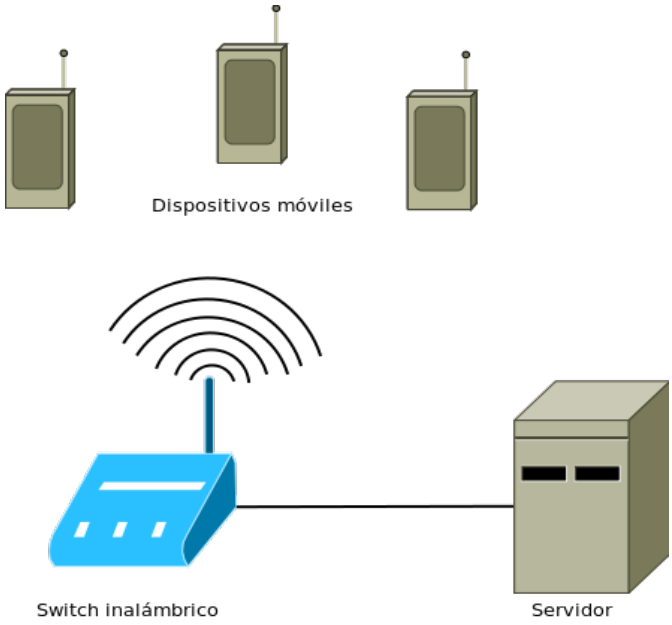
En este experimento se sometió a una serie de pruebas al Sistema de Recuperación de Información Distribuido diseñado en este trabajo en un contexto red de área local. El objetivo fue evaluar por un lado la coordinación de los dispositivos a través de los tiempos de respuesta y por otro lado le rendimiento de Recuperación de Información comparando los resultados de las consultas con los arrojados por un SRI centralizado de común utilización en el área. La arquitectura del Sistema Distribuido sometido a experimentación fue detallada en la *Sección 5* .

Se utilizaron las dos colecciones completas – GRULIC y UMEX (*Sección 6.1*) – para este experimento, distribuyendo de manera equitativa los documentos en cada uno de los dispositivos móviles utilizados (*Sección 6.3.1*) —1080 documentos por móvil para GRULIC y 325 documentos para UMEX— para que los mismos lleven a cabo la indexación de los documentos y posteriormente inicien la coordinación y comunicación con el broker (*Sección 5.4.1*) del sistema. Se realizaron una serie de consultas creadas de forma aleatoria, las cuales fueron emitidas una a la vez al broker el cual — como se explica en la *Sección 5.4.2* — procedió con la coordinación de los nodos y propagación de la consulta para una posterior recolección y fusión de los resultados (*Sección 4.2.3*). El set de consultas armado para la prueba se compone por dos conjuntos correspondientes a cada colección, los cuales están formados a su vez por 300 consultas de cada tipo según a cantidad de términos que la componen (1, 2, y 3 términos).

6.3.1 Recursos

El procedimiento se llevo a cabo sobre una red inalámbrica de de área local (WLAN), la cual interconecta tres dispositivos móviles con un servidor montado en una computadora de escritorio. Se debieron utilizar un conjunto de equipos de distintas características para esta prueba: un *switch*[78] *inalámbrico*, un equipo servidor y los dispositivos móviles. A continuación se muestra un diagrama con la disposición de los recursos y posteriormente una tabla con las características de cada uno.

78 Dispositivo digital de lógica de interconexión de redes de computadores que opera en la capa de enlace de datos del modelo OSI.



Dibujo 10: Diagrama de disposición de los recursos para el segundo experimento

Switch Inalámbrico	*TP-Link TL-WR340GD*[79], 54 Mbps, IEEE 802.11g (*Sección 3.3*)
Servidor	PC de escritorio. Procesador: AMD Athlon 3500 | Memoria: 1536 Mb | Sistema Operativo: Ubuntu 10.4
Dispositivos móviles	DROID2 (cantidad:1) | LGP350 (cantidad:2)

Tabla 11: Recursos utilizados en el segundo experimento

6.3.2 Tiempos de respuesta de consultas

A efectos de la evaluación de los tiempos de respuesta de las consultas al Sistema de Recuperación de Información se realizaron una serie de pruebas con las colecciones GRULIC y UMEX, y se registraron los tiempos en los cuales se llevaron a cabo los procesos. Se efectuaron 300 consultas de 1, 2 y 3 términos respectivamente en cada uno de los experimentos realizados con las dos colecciones de prueba, y a partir de las observaciones de los tiempos de respuesta se calculo la media y el respectivo desvío estándar que se visualizan en el *Gráfico 22*.

Cabe remarcar que en el caso del experimento con GRULIC la cantidad de documentos administrados por móvil es considerablemente mayor a la máxima aplicada en las pruebas de

79 http://www.tp-link.com/mx/products/details/?model=TL-WR340G#spec (especificación técnica)

indexación (*Sección 6.2.2*) y de resolución de consultas locales (*Sección 6.2.3*). Por eso mismo se registraron durante el procedimiento los tiempos de indexación de cada móvil para los subconjuntos de GRULIC y se compararon con los estimados según el análisis realizado en la *Sección 6.2.2* . Dicha comparación se presenta en la *Tabla 12* de la que que se puede decir que, con respecto a los tiempos de indexación para los 1080 documentos de GRULIC correspondientes a cada móvil, los datos reales se corresponden a los estimados ya que se encuentran levemente por debajo del teórico.

	Real	Estimado (x=1080)
DROID2	169541	178226
LGP350(1)	317277	324823
LGP350(2)	317232	324823

Tabla 12: Comparación de tiempos (en milisegundos) de indexación reales y estimados para el valor de 1080 documentos de la colección GRULIC

Focalizando en los tiempos de respuestas de las consultas distribuidas, se puede observar la gran diferencia que existe entre los tiempos de las pruebas con UMEX y las que emplearon la colección GRULIC. Esto es relativo a la desigualdad con respecto a la cantidad de documentos de cada una de las colecciones (*Sección 6.1*) y que se ve reflejado en el *Gráfico 22*.

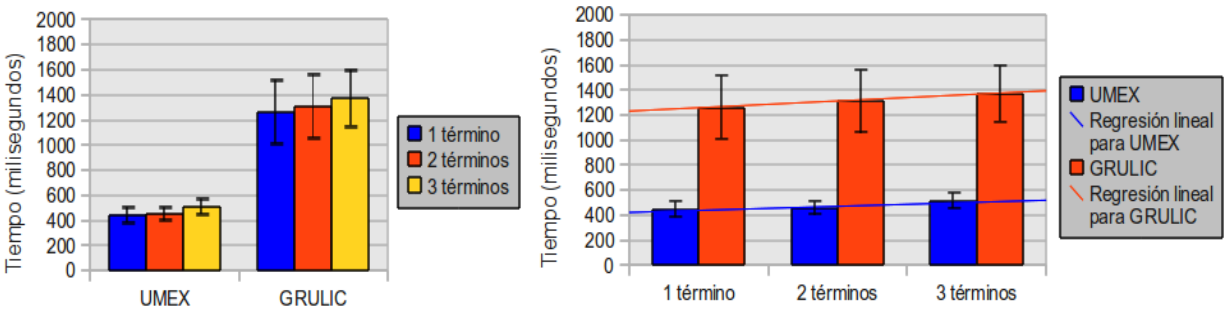


Gráfico 22: Tiempos de respuesta de consultas distribuidas agrupados por colección (der.) y por cantidad de términos de la consulta (izq.)

Los tiempos de respuesta para UMEX son buenos — oscilan entre los 380 y los 600 milisegundos como marcan las barras de error en el *Gráfico 22* —, teniendo en cuenta que en un lapso aproximado medio segundo se obtienen los resultados de una consulta sobre un sistema conformado por tres dispositivos móviles interconectados en una red de área local, en donde el móvil que procesa las consultas en mayor tiempo lo hace en unos 320 milisegundos — resultado obtenido de la ecuación de comportamiento LGP350/UMEX' de la *Tabla 10* con x = 325 —. No es

posible decir lo mismos para GRULIC, las pruebas realizadas sobre esta colección han mostrado tiempos de respuesta considerablemente mayores que los anteriores descriptos, principalmente debido a la gran cantidad de documentos ya que los 3240 de la colección se dividieron en partes iguales para los tres móviles, delegando a cada dispositivo la administración de 1080 documentos que supera por más del doble a la cantidad sugerida de 500 unidades (*Sección 6.1*). Es por esta razón que se presentan tiempos de respuesta entre los 1000 y 1600 milisegundos — ver las barras de error de GRULIC en el *Gráfico 22* —. A continuación se muestra en la *Tabla 13* la relación entre los tiempos de respuesta de las consultas distribuidas y con los tiempos estimados del procesamiento de consultas locales del dispositivo LGP350 el cual, de entre los dispositivos utilizados en la prueba, es el que presenta menor rendimiento y actúa como "cuello de botella" del sistema.

	DISTR/UMEX	LGP350/UMEX'(x=325)	DISTR/GRULIC	LGP350/GRULIC' (x=1080)
1 término	445	258	1260	967
2 términos	454	292	1309	1059
3 términos	510	320	1369	1176

Tabla 13: Comparación de tiempos (en milisegundos) de procesamiento de consultas locales del dispositivo utilizado con menor rendimiento con el procesamiento de consultas distribuido

En un análisis en mayor detalle, con respecto a los datos presentes en la *Tabla 13,* se calcularon los deltas entre los valores relacionados para deducir aproximadamente el tiempo que le lleva al broker la coordinaciónn de la consulta en conjunto con los tiempos que representan los retardos en la red debido a la transferencia de mensajes. A simple vista no se observó relación de los valores obtenidos con respecto a la cantidad de términos de la consulta, pero si se notó claramente que las consultas sobre GRULIC implicaron un leve mayor tiempo — unos 70 milisegundos más en promedio — que los que presentó UMEX. Esto se atribuye a alguna situación particular en la transferencia de datos en la red, descartando un retardo en el broker ya que el procesamiento local realizado en el mismo teóricamente no presentaría marcadas diferencias según la colección que se esté utilizando — de hecho la descripción de recursos de GRULIC, consultada en el cálculo de CORI (*Sección 4.2.2.1*), es de un tamaño menor al de UMEX debido a que posee menos términos en el vocabulario (*Sección 6.1.2*) —. Posiblemente se deba a que los rankings generados como resultado de las tareas de Recuperación de Información son de un tamaño mayor en bytes que los formados para UMEX ya que, en dichos resultados, se presenta una descripción de los documentos con los nombres de los mismos y los archivos correspondientes a GRULIC poseen nombres considerablemente más largos que los de la segunda — dichos archivos contienen como nombre al asunto completo del mail al que representan —, información que se puede avalar en la distribución

de las colecciones por longitud de nombre de los documentos en *Sección 9.6* .

Debido a que en esta prueba no se dispuso de recursos similares a los reales que se utilizan en un sistema de este tipo en su fase de producción, se puede decir que los tiempos de respuestas recolectados se consideran pesimistas. A pesar que los resultados no fueron para nada desfavorables — para el caso de UMEX —, se entiende que con el cambio de equipos por aquellos de mejor rendimiento — esto incluye servidores dedicados, switch de mayor capacidad y dispositivos móviles de gama superior — se obtendrá una sustancial mejora en los tiempos de respuesta del sistema.

6.3.3 Rendimiento de la Recuperación de Información Distribuida

En esta sección del experimento se pretende evaluar al sistema distribuido propuesto y diseñado en este trabajo desde el enfoque de Recuperación de Información, analizando si los resultados obtenidos a través de una consulta son aptos para satisfacer la necesidad de información expresada en la misma.

Para la experimentación en cuestión no se dispuso de colecciones de prueba estándar con juicios de relevancia para la evaluación de RI (*Sección 4.1.2*) — principalmente debido a que las más accesibles están compuestas por documentos escritos en inglés, los cuales no son útiles para la evaluación del sistema en cuestión —, en consecuencia se optó por un método particular de prueba, habitualmente utilizado en la comunidad de investigadores en Recuperación de Información. Dicho método se basa en la comparación de los resultados arrojados por el SRID sometido a evaluación contra un SRI centralizado bien conocido. Es pertinente aclarar que para el experimento, ambos sistemas mantienen indexada la misma colección y son sujetos al mismo conjunto de consultas. En base a esta comparación se puede inferir qué tanto se asemeja el rendimiento del prototipo de SRID con un sistema de recuperación centralizado.

Luego, a través del análisis de dicha semejanza y partiendo de la premisa que los resultados del SRI centralizado forman el conjunto de los documentos relevantes a la necesidad de información expresada en la consulta — simulando, de cierta manera, el juicio de relevancia que en colecciones de pruebas estándar es determinado por un grupo de personas — es posible realizar la evaluación de

Recuperación de Información.

El sistema de Recuperación de Información centralizado utilizado para la prueba es el motor de búsqueda *Indri* correspondiente a la herramienta del proyecto *Lemur*[80]. Dicha herramienta es altamente reconocida en la comunidad de Recuperación de Información y su buscador presenta uno de los rendimientos más alto entre sus semejantes de código libre [Middleton et al., 2007], por lo que se trata de una buena referencia en cuanto a desempeño de Recuperación de Información.

El experimento consiste en la ejecución secuencial de una serie de consultas en los dos sistemas utilizados en el experimento, y posteriormente, se lleva a cabo una comparación entre los ranking de documentos resultantes de dichas necesidades de información — ambos sistemas retornan una lista de documentos ordenada descendentemente por índice de relevancia —. La comparación esta basada en la identificación de los documentos relevantes a las consultas — corresponden a aquellos existentes en los resultados retornados por *Indri* — en los ranking respuesta a las mismas arrojados por el prototipo de SRID evaluado.

Se utilizaron dos parámetros configurables en el prototipo de SRID para poder ampliar el análisis de los resultados obtenidos y ver su influencia en el rendimiento de recuperación. Se trata de la cantidad máxima de nodos en las selección de recursos (*Sección 4.2.2*) y del método de fusión de resultados (*Sección 4.2.3*). Con respecto al primero se varió la cantidad máxima de nodos seleccionados en el rango de 1 a 3, mientras que para el segundo se utilizó por una lado la fusión de resultados mediante una ordenamiento por valor de relevancia y por el otro a través de un Round-Robin.

La cantidad máxima de documentos que *Indri* retorna como resultado fue limitada a 10, misma cantidad que se utilizó para limitar la respuesta arrojada por cada Nodo (*Sección 5.2*) en el procesamiento de consulta del SRID — siendo entonces en el mejor de los casos según los recursos utilizados (*Sección 6.2.1*) una cantidad de 30 documentos como máximo —.

En la evaluación se tuvieron en cuenta tres medidas diferentes: precisión (Sección *4.1.2.1.1*), exhaustividad (Sección *4.1.2.1.1*) y correlación. Con respecto a la última métrica, se pretendió evaluar la semejanza existente entre los rankings resultantes según las posiciones de los documentos relevantes, considerando que los más relevantes deben ir primeros en la lista. Para dicho análisis se utilizó el *Coeficiente de Correlación de Spearman [Salinas, 2007]*.

80 http://www.lemurproject.org/

En cuanto al análisis de los resultados obtenidos en este experimento (ver *Gráfico 23* y *Sección 9.4*) se puede decir, en líneas generales, que para la colección UMEX el sistema tuvo un mejor rendimiento en comparación a GRULIC en todos los parámetros evaluados. Esto se debe a que la primera posee una menor cantidad de documentos, tiene un vocabulario más rico y una mejor redacción ya que se trata de artículos de noticias. Haciendo enfoque en la Precisión se puede apreciar que en general el sistema responde de una forma aceptable, pero su rendimiento se ve en disminución en cuanto más términos se utilicen en la consulta.

Por otro lado la medida de Exhaustividad exhibe un valor mucho menor con respecto al de Precisión. Esto se debe a que, como se nombró antes, se limitó la cantidad de documentos que retornan los nodos, por lo que es probable que algunos documentos relevantes hayan quedado afuera de la respuesta. El valor de esta medida muestra un leve incremento cuanto más nodos son utilizados para la resolución de la consulta ya que se entiende que a menor cantidad de nodos, menor cantidad de documentos se encuentran accesibles. En la *Sección 5.1* se mencionó que el sistema esta diseñado para prevalecer la Precisión por sobre la Exhaustividad, en efecto se puede decir que esta medida se encuentra dentro de los parámetros razonables.

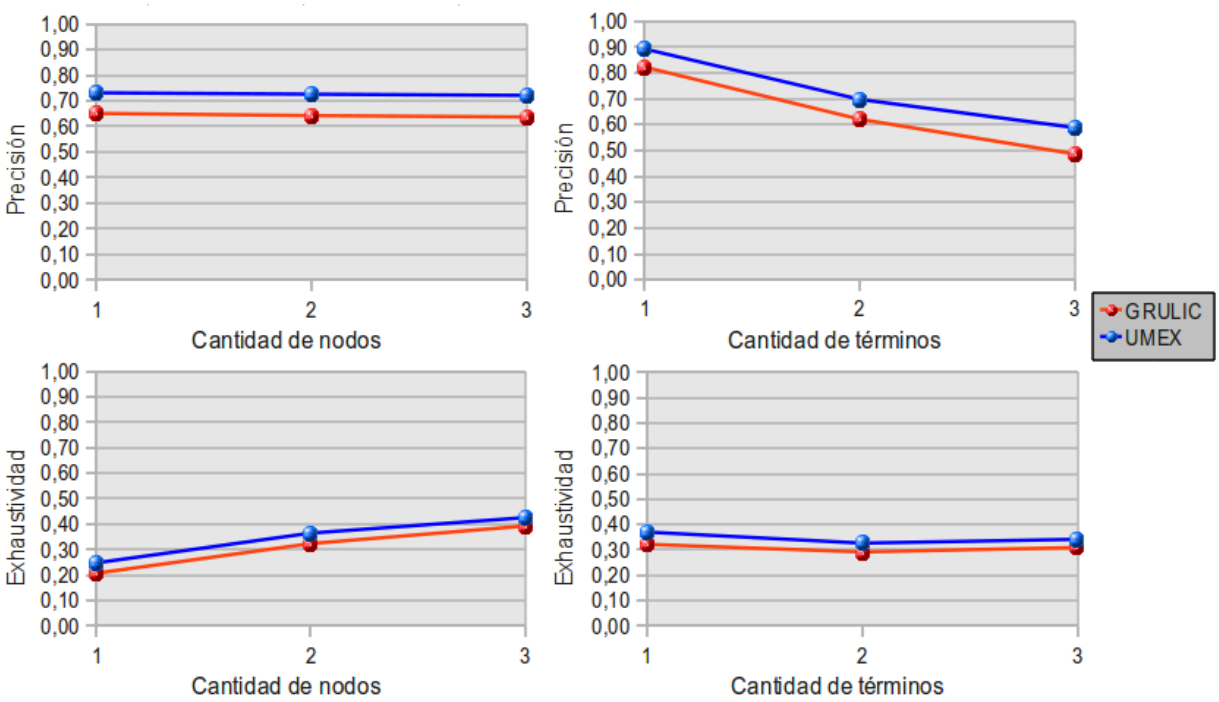


Gráfico 23: Relación de la Precisión y Exhaustividad con la máxima cantidad de nodos en la selección de recursos y la cantidad de términos de la consulta

Los datos del estudio de la correlación (*Sección 9.5*) son presentados en el *Gráfico 24*. Se puede apreciar que los resultados entre *Indri* y el prototipo SRID poseen una alta correlación en las consultas de un término, pero se da la particularidad que a mayor cantidad de términos utilizados menor es la semejanza entre los resultados de los dos sistemas. Otra aporte que arrojan las resultados de esta medida es el hecho de que la utilización de Ordenamiento por valor de relevancia (SORT), para la fusión de resultados, mejora levemente la correlación con respecto al uso de Round-Robin (RR).

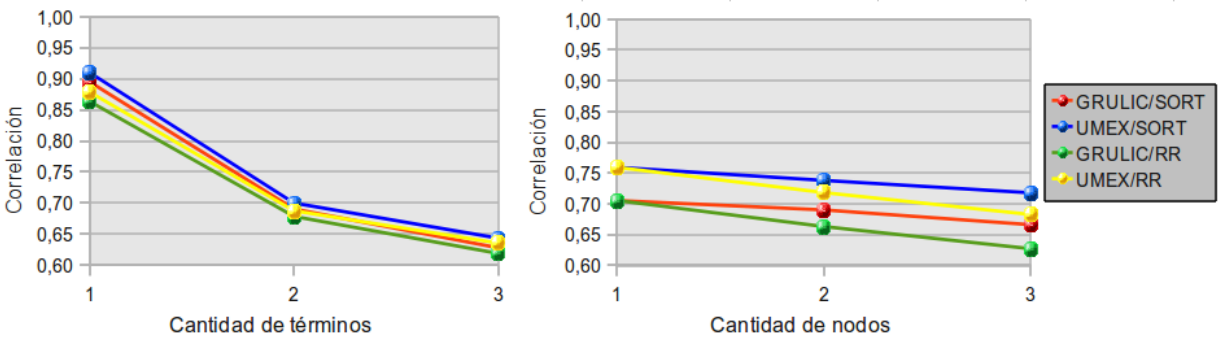


Gráfico 24: Correlación entre los rankigs del sistema Indri con los del prototipo de Sistema de Recuperación de Información Distribuido

A modo de resumen sobre los resultados de este experimento, es importante resaltar que se esperaban altos valores de Precisión ya que es la medida de mayor peso para los fines del sistema. Satisfactoriamente el prototipo presentó tal comportamiento en las pruebas, acompañado por una regular Exhaustividad, que también era supuesta ya que el sistema no fue diseñado para maximizar la misma. En los que respecta a la correlación de los rankings de resultados, hay que remarcar que el valor de relevancia, dato mediante el cual se ordena la lista de documentos, es dependiente exclusivamente del SRI debido a la métrica de similitud (*Sección 4.1.1.1.1*) y peso de términos (*Sección 4.1.1.1.2*) que aplica. Dado que los sistemas comparados utilizan peso de términos diferentes — *TF Normalizado* en los Nodos y *BM25TF* [Robertson et al., 1993] en *Indri* — se producen diferencias en el orden de relevancia de los documentos, las cuales aumentan con el incremento de términos. Sin embargo, sobres esto último se encuentra un respaldo con el hecho de que el largo de las consultas realizadas por los usuarios es de 2,3 términos en promedio [Spink et al.,2002], por lo que podemos suponer que en la práctica las medidas de Correlación y Precisión no variarán significativamente de lo valores obtenidos.

7 Conclusiones

Los dispositivos móviles ya se encuentran actualmente insertados en la vida cotidiana de las personas, con un adopción que va en crecimiento, mejorando sus capacidades de procesamiento, memoria, conexión inalámbrica y de interfaz al usuario. También existe un gran crecimiento en materia de software para móviles que ha derivado en la creación de innumerables aplicaciones y *frameworks* que posibilitan diversas funcionalidades para usuarios.

El acceso a la información es un requerimiento muy importante en la época actual. Sin embargo no existe una variedad de esfuerzos destinados a la Recuperación de Información en móviles limitando los mismos al acceso a través de la web a los grandes motores de búsqueda.

En este trabajo se ha comprobado que los dispositivos de tipo *smartphones* pueden realizar tareas de Recuperación de Información, con un enfoque clásico, sobre pequeñas colecciones de documentos almacenadas en la memoria de los mismos. En colecciones de hasta 500 documentos los dispositivos han exhibido un buen desempeño con tiempos de respuesta razonables que no superan los 500 milisegundos en promedio.

En un escenario de dispositivos móviles, se planteó y evaluó la posibilidad de coordinación entre prototipos de Sistemas de Recuperación de Información desplegados en múltiples móviles a través del enfoque de una Recuperación de Información Distribuida. Se evaluó al sistema en este contexto desde el enfoque de tiempos de respuesta y de rendimiento de Recuperación de Información. En cuanto a performance de respuesta la prueba se presentó como pesimista dado que no se tuvieron al alcance los recursos que poseen desempeño a la altura de de aquellos utilizados en sistemas en fase de producción. Mas allá de esto, los resultados fueron favorables en los casos que se respetó la cantidad de documentos sugerida que pueden ser indexada por dispositivos móviles, aunque se da por sentada la posibilidad de mejorar los tiempos de respuesta mediante la utilización de equipos de mayor capacidad para soportar la infraestructura de Sistema Distribuido.

La Recuperación de Información del sistema distribuido fue evaluada a través de la comparación

con un SRI centralizado bien conocido. El SRID propuesto presenta una alta precisión, muy cercana a la de un SRI centralizado (con un valor de 0,8). En cuanto a la exhaustividad, como se esperaba, aumenta según la cantidad de Nodos móviles que son utilizados, pero también suben los tiempos de respuesta. En un ambiente de producción hay que estudiar el trade-off que resulte óptimo a los fines del sistema.

Queda claro que un dispositivo móvil actual posee las capacidades de hardware y la plataforma de software adecuada para soportar funcionalidades de Recuperación de Información mediante técnicas clásicas. Entonces una red de dispositivos móviles, a través de conexiones inalámbricas, colaborando en el objetivo de recuperación de información es factible siempre y cuando se tomen las consideraciones adecuadas para las capacidades de los dispositivos utilizados. Dichas capacidades se ven en aumento a través de los nuevos modelos de *smartphones* y *tablets*, por lo que los resultados y consideraciones obtenidas en este trabajo se supone que serán ampliadas y mejoradas en un futuro a corto plazo. En consecuencia se espera la posible implementación de aplicaciones móviles de RI de estas características con fines más allá de lo experimental, adecuando redes de usuarios que deseen compartir colecciones de información sin dejar de tener exclusividad de administración sobre la mismas.

En resumen, los aportes de este trabajo son:

- Una primera implementación de un sistemas de RI clásico sobre dispositivos móviles, incluyendo la codificación de un prototipo y su evaluación exhaustiva en términos de performance. El código[81] del mismo se ofrece como software libre a la comunidad, bajo la licencia *GNU General Public License, version 2*[82], para la continuidad de la investigación en el tema.

- La caracterización de colecciones de noticias y correo electrónico (ambos en español) típica del uso en tales aparatos.

- Una propuesta de un sistema de RID para los móviles con búsqueda distribuida y fusión de resultados.

- La evaluación del sistema propuesto respecto de uno centralizado clásico del mundo de la

81 http://code.google.com/p/movirdroid-unlu/source
82 http://www.gnu.org/licenses/old-licenses/gpl-2.0.html

Recuperación de Información.

7.1 Trabajos Futuros

El trabajo desarrollado abre nuevas lineas de investigación a explorar en el ámbito de la RI en dispositivos móviles. En particular, se proponen la siguientes continuidad:

- ***Diseño de estructuras de datos especificas para dispositivos móviles:*** dado que en este trabajo se analizaron estructuras de datos clásicas, queda pendiente el diseño de unas nuevas adaptadas a la naturaleza de tales dispositivos esperando que mejoren la performance lograda en esta publicación.

- ***Analizar el uso de técnicas de compresión del índice y de las descripciones de recursos para RID:*** la aplicación de compresión implica una menor utilización de memoria lo que puede aumentar la escalabilidad del sistema.

- ***Modelar el problema con múltiples brokers que coordinen sus acciones en dominios de dispositivos excluyentes:*** estudiar la posibilidad de interconexión entre múltiples SRID, como el que se propuso este trabajo, es algo que precisa ser realizado si se desea llevar a cabo una Recuperación de Información con grandes número de usuarios.

- Ampliar las colecciones de prueba incorporando juicios de relevancia y analizar las performance bajo otros escenarios: le evaluación mediante la utilización de otras colecciones aportará un gran valor para la mejora del prototipo ya que la performance del mismo esta altamente relacionada con la naturaleza de las colecciones de documentos.

- ***Extender el modelo para dispositivos tipo tablets, cuyas prestaciones son diferentes:*** dichos dispositivos, en menor medida que los smartphones, son potenciales herramientas para el acceso de información. Presentan una particularidad en su interfaz que los hace más útiles para la lectura de textos.

8 Referencias

[Anderson et al., 2008] J.Q. Anderson y L. Rainie. The Future of the Internet III. Pew Internet and American Life Project. 2008.

[Android, 2011] What is Android? Online: http://developer.android.com/guide/basics/what-is-android.html

[Baeza et al., 1999] Ricardo Baeza-Yates y Berthier Ribeiro-Neto. Modern Information Retrieval 1st edition. ACM Press series. 1999.

[Banchero, 2010] Santiago Banchero, Gabriel Tolosa y Fernando Bordignon. "Selección de Recursos Distribuidos en Ambientes Dinámicos ". Universidad Nacional de Luján, Trabajo Final de la Licenciatura. 2010.

[Bawa et al., 2003] M. Bawa, R. J. Bayardo Jr, S. Rajagopalan y E. J. Shekita. "Make it Fresh, Make it Quick — Searching a Network of Personal Webservers". WWW '03 Actas de la 12° Conferencia Internacional sobre la World Wide Web. 2003.

[Broglio et al., 1995] J. Broglio, J.P. Callan, W.B. Croft y D.W. Nachbar. "Document retrieval and routing using the INQUERY system". En "Overview of the Third Retrieval Conference (TREC-3)", págs 29-38. Pulicación especial de Nist 500-225, 1995.

[Callan et al., 1995] James P. Callan, et al. "Searching distributed collections with inference networks" págs. 21-28. Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval. 1995.

[Callan et al., 2000] James P. Callan, et al. Distributed Information Retrieval (Advances in information retrieval, chapter 5, págs 127-150. Kluwer Academic Publishers. 2000.

[Callan et al., 2001] Jaime Callan y Margaret Connel. "Query-Based Sampling of Text Databases". ACM Transactions on Information Systems, Vol. 19, No. 2. Abril 2001, págs. 97–130. 2001.

[Chun et al., 2009] Byung-Gon Chun, Petros Maniatis. Augmented Smartphone Applications Through Clone Cloud Execution. HotOS. 2009.

[Claburn, 2009] T. Claburn. Google CEO Imagines Era of Mobile Supercomputers. Information Week. 2009.

[Cleverdon et al., 1966] Cleverdon, C.W., Mills, J. Y Keen, M. "Factors Determining the Performance of Indexing Systems". ASLIB Cranfield Project. Vol. 1, Design, Vol2, Test Results. 1966.

[Cleverdon, 1972] Cleverdon, C.W. “On the inverse relationship of recall and precision”. Journal of Documentation, vol. 28, págs. 195-201. 1972.

[Coffman et al., 2001] K. G. Coffman, A. M. Odlyzko . Growth of the Internet . AT&T Labs – Research . 2001.

[Colourius, 2005] G. Colourius. “Distributed Systems - Concepts and Design”. Addison-Wesley Professional, 4ta edición. 2005.

[Comscore, 2011] The comScore 2010 Mobile Year in Review . Febrero 2011. Online: http://www.comscore.com/Press_Events/Presentations_Whitepapers/2011/2010_Mobile_Year_in_Review

[Comscore, 2011b] U.S. Smartphone audience growth. Agosto 2011. Online: http://www.comscoredatamine.com/2011/08/u-s-smartphone-audience-growth/

[Ericsson, 2011] One billion mobile broadband subscriptions 2011. Ericsson. 2011. Online: http://www.ericsson.com/res/docs/2011/barca_brochure_subscriptions.pdf

[Flora et al., 2010] Flora S. Tsai, Minoru Etoh, y otros, “Introduction to Mobile Information Retrieval,” Intelligent Systems, Enero 2010. Online: http://www.computer.org/portal/web/csdl/doi/10.1109/MIS.2010.22

[Frakes et al., 1992] W.B. Frakes y R. Baeza-Yates. “Information Retrieval: Data Structures & Algorithms”. Prentice Hall, Englewood Cliffs. 1992.

[French et al., 1999] James C. French, J. Callan, A.L. Powell, C.L. Viles, T. Emmit, T. Prey y Y. Mou. “Comparing the Performance of Database Selection Algorithms”. SIGIR. 1999.

[Fitzek et al., 2009] Frank H. P. Fitzek , Hassan Charaf . MOBILE PEER TO PEER (P2P) . John Wiley & Sons Ltd . 2009.

[Fuhr et al., 1990] Fuhr, N. & Buckley, C. “Probabilistic document indexing from relevance feedback data”. ACM/SIGIR Conference. 1990.

[Gantz et al., 2007] John F. Gantz , David Reinsel , Christopher Chute, Wolfgang Schlichting, John McArthur , Stephen Minton , Irida Xheneti , Anna Toncheva , Alex Manfrediz . The Expanding Digital Universe . Marzo 2007. International Data Corporation White Paper . Online: http://www.emc.com/collateral/analyst-reports/expanding-digital-idc-white-paper.pdf

[Gantz et al., 2011] John Gantz, David Reinsel . Extracting Value from Chaos . Junio 2011. International Data Computer Iview.

[Gnutella, 2010] The Annotated Gnutella Protocol Specification v0.4. Online: http://rfc-gnutella.sourceforge.net/developer/stable/index.html

[Gravano et al., 1994] L. Gravano, H. García-Molina y A . Tomasic. “The effectiveness of GlOSS for the text database discovery problem”. Proceedings of the ACMSIGMOD International Conference on Management of Data. 1994.

[Grossman et al., 1997] D.A. Grossman, O. Frieder, D.O. Holmes, y D.C. Roberts. ”Integrating Structured Data and Text: A Relational Approach”. Journal of the American Society of Information Science, 48(2):122–132. 1997.

[GSA, 2011] The Global Mobile Suppliers Association. Estadísticas. 2011. Online: http://www.gsacom.com/news/statistics.php4

[Halepovic et al., 2009] E. Halepovic, C. Williamson, and M. Ghaderi, “Wireless data traffic: A decade of change,” IEEE Network, vol. 23, no. 2. Marzo 2009.

[Heaps, 1978] Harold Stanley Heaps. “Information Retrieval: Computational and Theoretical Aspects”,sección 7.5, págs 206–208. Academic Press, 1978.

[Horrigan, 2009] John Horrigan. Wireless Internet Use. Julio 2009. Pew Internet & American Life Project. Online: http://pewinternet.org/Reports/2009/12-Wireless-Internet-Use.aspx

[Ingwersen, 2002] Peter Ingwersen, Information Retrieval Interaction. 2002. Royal School of Library and Information Science , Denmark.

[ITU, 2011] Internation Telecommunication Union. Estadísticas. 2011. Online: http://www.itu.int/ITU-D/ict/statistics/

[IWS, 2011] Internet World Stats, Usage and Population Stastics. 2011. Online: http://internetworldstats.com/stats.htm

[Jantscher et al., 2009] Martin Jantscher, Mohammed Talhaoui, Denis De Vos, Ivan Cunha, Artur Roszcyk, Mikhail Datsyuk. Mobile Application Development 2009. Online: http://www.mad-ip.eu/files/reports/Android.pdf

[Kellogg, 2011] Don Kellogg . Average U.S. Smartphone Data Usage Up 89% as Cost per MB Goes Down 46%. Nielsen. Junio 2011. Online: http://blog.nielsen.com/nielsenwire/online_mobile/average-u-s-smartphone-data-usage-up-89-as-cost-per-mb-goes-down-46/

[Kendrick, 2011] James Kendrick. “Stats: Android Growth Continues; Passes iOS in Usage”. ZDNET. 2011. Online: http://www.zdnet.com/blog/mobile-news/stats-android-growth-continues-passes-ios-in-usage/422

[Kirsch, 1997] S.T. Kirsch. “Document retrieval over networks wherein ranking and relevance scores are computed at the client for multiple database documents”. U.S. Patent 5,659,732. 1997.

[Krester et al., 1998] Owen de Kretser, et al. Methodologies for Distributed Information Retrieval. Proceedings of the The 18th International Conference on Distributed Computing Systems. 1998.

[Lallana, 2003] Emmanuel C. Lallana. The Information Age. e-ASEAN Task Force, UNDP-APDIP. 2003.

[Laszlo, 2002] Laszlo B. Kish . End of Moore’s law: thermal (noise) death of integration in micro and nano electronics . Julio 2002. Elsevier Science.

[Li et al.,2010] Xun Li, et al. Smartphone Evolution and Reuse: Establishing a more Sustainable Model. Proceedings of ICPPW'10, 39th International Conference on Parallel Processing Workshop. IEEE Computer Society, 2010.

[Manning et al., 2008] Christopher D. Manning, et al. Introduction to Information Retrieval, chapter 6, p109-128. Cambridge University Press. 2008.

[Martinez Mendez et al., 2004] Martinez Mendez, F.J. y Rodriguez Muñoz, J.V. "Reflexiones sobre la Evaluación de clos Sistemas de Recuperación de Información: Necesidad, Utilidad y Viabilidad". Anuales de Documentación, Nro. 7, págs. 153-170. 2004.

[Middleton et al., 2007] C. Middleton y R. Baeza-Yates . "A Comparison of Open Source Search Engines ". 2007. Online: http://wrg.upf.edu/WRG/dctos/Middleton-Baeza.pdf

[Milojicic et al., 2002] Dejan S. Milojicic, et al. Peer-to-Peer Computing. HP Technical Reports. 2002.

[Moore, 1965] Cramming more components onto integrated circuits , Electronics, Volume 38, Number 8. 1965 .

[MTIS, 2011] Measuring The Information Society. International Telecommunication Union. 2011.

[Nielsen, 2010] Mobile Snapshot: Smartphones Now 28% of U.S. Cellphone Market. Online: http://blog.nielsen.com/nielsenwire/online_mobile/mobile-snapshot-smartphones-now-28-of-u-s-cellphone-market/

[Nielsen, 2011] 40 Percent of U.S. Mobile Users Own Smartphones; 40 Percent are Android. Septiembre 2011. Online: http://blog.nielsen.com/nielsenwire/online_mobile/40-percent-of-u-s-mobile-users-own-smartphones-40-percent-are-android/

[OHA, 2011] Open Handset Alliance. Android Overview. Online: http://www.openhandsetalliance.com/android_overview.html

[Porter, 1980] Porter, M. F. "An algorithm for suffix stripping". Program, 14(3), págs. 130-137. 1980.

[Reuters, 2009] Internet most popular information source. Reuters. 2009. Online: http://www.reuters.com/article/2009/06/17/us-media-internet-life-idUSTRE55G4XA20090617

[Robertson et al., 1993] S.E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu y M. Gatford. "Okapi". The Second Text REtrieval Conference (TREC-2). 1993: Online: http://trec.nist.gov/pubs/trec2/papers/txt/02.txt

[Robertson et al., 1994] S.E. Robertson y S. Walker. "Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval". Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 1994.

[Robertson, 1977] S.E. Robertson. "The probability ranking principle in IR". School of library, Archive and Information Studies, University College London. 1977.

[Salinas, 2007] M. Salinas. “Modelos de Regresión y Correlación IV. Correlación de Spearman”. Revista “Ciencia & Trabajo”. Julio / Septiembre 2007. Online: http://www.cienciaytrabajo.cl/pdfs/25/pagina%20143.pdf

[Salton et al., 1983] Salton, G.; Fox, E.A. y Wu, H. “Extended Boolean information retrieval”. Communications of the ACM, 26(11):1022-1036. Noviembre, 1983.

[Salton, 1971] Salton, G. (editor). “The SMART Retrieval System – Experiments in Automatic Document Processing”. Prentice Hall In. Englewood Cliffs, NJ. 1971.

[Si et al., 2003] Luo Si y Jaime Callan. “Relevant document distribution estimation method for resource selection”. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 2003.

[Si et al., 2003b] Luo Si y Jaime Callan. “The effect of database size distribtion on resource selection algorithms”. 2003.

[Singhal , 2001] Amit Singhal . Modern Information Retrieval: A Brief Overview . 2001. Google Inc.

[Spink et al.,2002] A. Spink, B. J. Jansen, D. Wolfram y T. Saracevic.”From e-sex to e-commerce: Web search changes”. IEEE Computer, 35(3), 107–109. 2002. Online: http://jimjansen.tripod.com/academic/pubs/ieee_computer.pdf

[Toffler, 1970] Alvin Toffler. Future Shock, capítulo 16, p350-358. Bantam Books . 1970.

[Tolosa et al., 2008] Gabriel H. Tolosa, Fernando Bordignon. Introducción a la Recuperación de Información, capítulo 1 p14, capítulo 4 págs 88-95. Laboratorio de Redes de Datos, División Estadística y Sistemas, Departamento de Ciencias Básicas, Universidad Nacional de Luján. 2008.3

[TSOTWWW, 2011] The Size Of The World Wide Web. 2011. Online: http://www.worldwidewebsize.com/

[Turtle et al., 1990] H.R. Turtle y W.B. Croft. “Efficient probabilistic inference for text retrieval”. RIAO 3 Conference Proceedings, págs. 644-661. Barcelona, España. 1991.

[Voorhess et al., 1995] E. Voorhess, N.K. Gupta y B. Johnson-laird. “Learning collection fusion strategies”. Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, New York. 1995.

[Vouk, 2008] Mladen A. Vouk . Cloud Computing – Issues, Research and Implementations. Journal of Computing and Information Technology . 2008

[Waller et al., 1979] Waller, W. G. y Kraft, D. H. “A mathematical model for a weighted Boolean retrieval system”. Information Processing and Management, Vol 15, No. 5, pp. 235-245. 1979.

[MOBMAV, 2010] Smartphones Evolving Faster Than Moore’s Law. 2010. Online: http://mobilemavens.net/?p=726

[World Bank, 2011] Indicadores del Banco Mundial. 2011. Online: http://datos.bancomundial.org/indicador

[Xu et al., 1998] J. Xu y J. Callan. “Effective retrieval of distributed collections”. Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, págs. 254-261. Berkeley. 1998.

[Zipf, 1949] Zipf, G. K. “Human Behaviour and the Principle of Least Effort” Reading, MA: Addison- Wesley Publishing Co. 1949.

9 Anexos

9.1 Anexo 1: Tabla de datos de indexación utilizando SQLite

N° Docs	EMUPC/UMEX'	EMUPC/GRULIC'	DROID2/UMEX'	DROID2/GRULIC'
20	37552	27725	26203	18447
40	71299	53314	55148	34065
60	104947	78792	74342	45372
80	148559	111382	98019	65347
100	181301	135977	119692	79915
120	222680	165977	139961	99247
140	264066	197508	163619	117387
160	292163	218089	180185	128439
180	324465	242487	196047	140923
200	354378	264876	214039	153021
220	388614	292377	234959	167353
240	428714	321761	254282	186464
260	460693	344351	285065	197905
280	490338	368477	317878	212676
300	523917	390765	343075	225494
320	552039	414552	365731	242947
340	578831	433200	387703	251407
360	623405	460089	410392	265177
380	653663	484199	427785	282857
400	680814	501392	450734	292143
420	712354	516323	467562	307803
440	740591	543724	478450	316308
460	762863	561367	490856	326956
480	787145	580579	505019	335925
500	809254	596065	515186	346515

9.2 Anexo 2: Tabla de datos de indexación

N° Docs	EMUPC/UMEX'	EMUPC/GRULIC'	DROID2/UMEX'	DROID2/GRULIC'	LGP350/UMEX'	LGP350/GRULIC'
20	31146	32447	4451	3722	7684	7730
40	65753	61326	8745	7200	15886	13814
60	92779	90504	12498	10454	23144	19562
80	122729	126893	16641	14580	29964	27301
100	149865	155880	20389	17829	37688	34207
120	175816	192485	24220	21679	44667	40885
140	209327	231928	28262	25714	55416	48602
160	231619	255625	31148	28508	58654	53947
180	256770	284069	34828	31612	63637	85251
200	279481	309558	37884	34562	69114	71717
220	308288	338496	42297	38027	76948	72805
240	335376	364370	45784	41997	82836	78886
260	382121	386440	51932	45430	95024	84452
280	420586	413789	57068	48665	104136	90188
300	455272	433609	60759	52149	114831	96575
320	488383	459837	65272	55592	137181	103184
340	517748	481059	69342	58267	138848	107251
360	546941	512007	75248	61947	136358	113949
380	575710	539238	76834	66327	146188	125571
400	601612	557614	79633	67056	149320	126214
420	625115	583014	83307	69777	155222	129754
440	647096	606457	86026	73507	163096	135893
460	669589	626234	93078	75953	171884	139579
480	689479	647899	92637	78293	170958	143314
500	709263	665229	95781	81012	174851	147077

9.3 Anexo 3: Tabla de datos de resolución de consultas

	DROID2						LGP350					
	UMEX'			GRULIC'			UMEX'			GRULIC'		
N° Docs	1 TERM	2 TERMS	3 TERMS	1 TERM	2 TERMS	3 TERMS	1 TERM	2 TERMS	3 TERMS	1 TERM	2 TERMS	3 TERMS
20	13	14	13	20	18	28	23	23	25	26	27	33
40	20	22	25	39	41	31	37	40	45	38	46	45
60	28	30	35	34	74	57	53	65	70	53	55	63
80	38	41	45	66	55	69	67	81	85	70	77	91
100	51	58	60	70	96	80	82	95	111	85	103	109
120	56	64	66	104	88	94	103	112	131	100	122	124
140	64	72	81	104	103	120	118	142	143	116	142	144
160	73	82	90	103	126	152	134	162	165	132	148	180
180	82	92	101	125	132	153	151	173	187	149	176	191
200	91	106	111	115	121	128	169	182	209	175	185	212
220	102	114	130	126	133	146	180	208	225	190	223	237
240	112	128	136	137	147	168	195	234	246	211	220	254
260	127	138	154	154	159	181	216	245	262	215	257	274
280	131	150	159	166	172	196	225	260	279	246	267	300
300	139	158	171	173	196	209	254	270	301	268	300	316
320	144	163	182	192	199	220	258	289	327	290	311	338
340	163	177	188	204	231	235	264	305	335	286	327	379
360	164	186	201	216	241	326	299	322	357	329	350	396
380	172	196	214	241	264	289	303	342	369	330	373	415
400	183	198	222	240	265	287	316	352	394	352	384	424
420	190	209	233	239	279	300	327	365	411	364	414	442
440	197	218	245	264	287	306	334	394	424	384	443	478
460	200	229	249	275	301	331	352	412	438	403	455	505
480	231	259	299	297	322	343	369	415	471	441	473	536
500	222	247	267	303	342	372	390	429	474	463	490	543

9.4 Anexo 4: Tabla de datos de Evaluación de Precisión y Exhaustividad

		GRULIC		UMEX	
N° Nodos	Long. Consulta	P	R	P	R
1 NODO	1 TERM	0,82	0,24	0,89	0,31
1 NODO	2 TERMS	0,63	0,19	0,71	0,22
1 NODO	3 TERMS	0,49	0,19	0,59	0,21
2 NODOS	1 TERM	0,82	0,29	0,89	0,35
2 NODOS	2 TERMS	0,63	0,25	0,70	0,29
2 NODOS	3 TERMS	0,49	0,26	0,59	0,29
3 NODOS	1 TERM	0,82	0,34	0,89	0,38
3 NODOS	2 TERMS	0,62	0,31	0,70	0,35
3 NODOS	3 TERMS	0,49	0,32	0,59	0,36
1 NODO	-	0,65	0,21	0,73	0,25
2 NODOS	-	0,64	0,32	0,73	0,36
3 NODOS	-	0,64	0,39	0,72	0,43
-	1 TERM	0,82	0,32	0,89	0,37
-	2 TERMS	0,62	0,29	0,70	0,33
-	3 TERMS	0,49	0,31	0,59	0,34

9.5 Anexo 4: Tabla de datos de Evaluación de Correlación

N° Nodos	Long. Consulta	GRULIC-RR	GRULIC-SORT	UMEX-RR	UMEX-SORT
1 NODO	1 TERM	0,84	0,84	0,90	0,90
1 NODO	2 TERMS	0,67	0,67	0,72	0,72
1 NODO	3 TERMS	0,61	0,61	0,66	0,66
2 NODOS	1 TERM	0,78	0,83	0,86	0,90
2 NODOS	2 TERMS	0,63	0,64	0,67	0,68
2 NODOS	3 TERMS	0,58	0,60	0,63	0,63
3 NODOS	1 TERM	0,74	0,81	0,83	0,89
3 NODOS	2 TERMS	0,60	0,62	0,64	0,66
3 NODOS	3 TERMS	0,54	0,57	0,58	0,61
1 NODO	-	0,71	0,71	0,76	0,76
2 NODOS	-	0,66	0,69	0,72	0,74
3 NODOS	-	0,63	0,67	0,68	0,72
-	1 TERM	0,86	0,89	0,88	0,91
-	2 TERMS	0,68	0,69	0,69	0,70
-	3 TERMS	0,62	0,63	0,64	0,64

9.6 Anexo 6: Gráfica de la distribución de los documentos de las colecciones según longitud de nombre