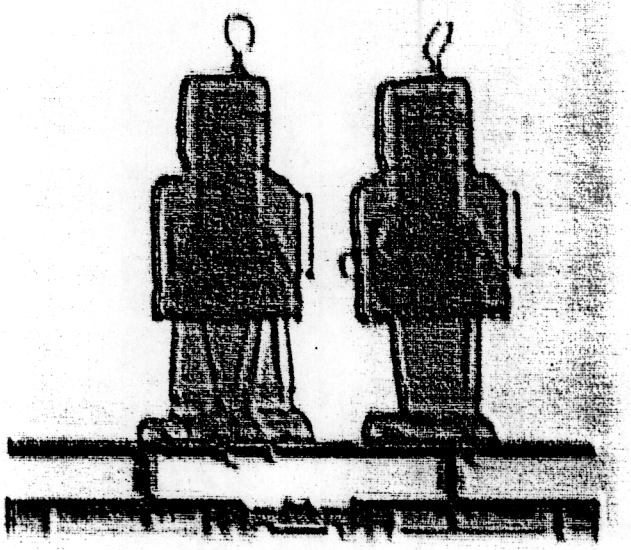
2004 Iberamia

IX IBERO-AMERICAN WORKSHOPS ON ARTIFICIAL INTELLIGENCE

National Institute of Astrophysics, Optics and Electronics



November 22-23, 2004

Editors: Guillermo De Ita Luna
Olac Fuentes Chávez
Mauricio Osorio Galindo

WORKSHOPS ON ARTIFICIAL INTELLIGENCE

- Deduction and Reasoning Techniques
- Lexical Resources and the Web for Word Sense Disambiguation
- Herramientas y Recursos Linguísticos para el Español y el Portugués
- Machine Learning for Scientific Data Analysis
- Virtual Laboratories

Proceedings

Editors: Guillermo De Ita Luna
Olac Fuentes Chávez
Mauricio Osorio Galindo

ISBN 968-863-786-6

This work is subject to copyright. All rights are reserved. Reproduction of this publication in any form by any electronic or mechanical means is permitted only under the provisions of the Mexican Federal Copyright Law and the prior permission in writing of Autonomous University of Puebla.

L. Alfonso Ureña López

Printed in Mexico

Puebla, 2004 November

TABLE OF CONTENT

	Page
Workshop on Deduction and Reasoning Techniques	1
Image Based Reasoning Applied to the Comparison of the Popocatépetl Volcano's Fumaroles *Antonio Aguilera, S. Gerardo Lázzeri, Ricardo Pérez-Aguila*	3
Quantum Search in Databases with Group Action Structure *César Bautista Ramos*	9
An Exact Algorithm for the #2-SAT Problem *Guillermo De Ita, Pedro Bello, Erica Vera*	16
A Linear-Time Algorithm for the #2μ,2-SAT Problem Using Connected Components *Guillermo De Ita, Carlos Guillén, David Pinto*	24
Encoding Conformant Planning in A-Prolog *Michael Gelfond, A. Ricardo Morales*	32
An Action Semantics of the PCF Language *I. Everardo Bárcenas Patiño, José de Jesús Lavalle M.*	42
A Principle-Based Chess Playing Program *Santos Gerardo Lázzeri Menéndez*	50
Using CR-Rules for Evacuation Planning *Mauricio Osorio, Magdalena Ortiz, Claudia Zepeda*	56
A CBR Model for Diagnosis of Math Trouble in Children *Ernesto Brau, Alberto Ochoa, Ramiro Avila*	64
Irrelevant Actions in Plan Generation *Vladimir Lifschitz, Wanwan Ren*	71
An Extension of the A-Pol System *Yana Maximova Todorova*	79

Workshop on
Lexical Resources and the Web for Word Sense Disambiguation 87

Web-based WSD using Adjective-Noun pairs 89
Davide Buscaldi, Manuel Montes y Gómez, Paolo Rosso

Automatic Acquisition of Sense Examples using ExRetriever 97
Montse Cuadros, Jordi Atserias, Mauro Castillo, German Rigau

Una Metodología para la Creación de Thesauri 105
Cupertino Lucero, David Pinto, Héctor Jiménez-Salazar

Invited Talk : Making Sense Out of the Web 112
Rada Mihalcea

Desambiguación Léxica de Sustantivos usando la Web 118
Aarón Pancardo-Rodríguez, Manuél Montes-y-Gómez, Paolo Rosso, Davide Bucaldi, Luis Villaseñor-Pineda

Improving Word Sense Discrimination with Gloss Augmented Feature Vectors 123
Amruta Purandare, Ted Pedersen

Using SENSEVAL System Scores to Optimize Lexical Resources for WSD and Knowledge Acquisition 131
Harri M.T. Saarikoski

Multilingual Extended WordNet 137
Sonia Vazquez, Andrés Montoyo

Workshop on
Herramientas y Recursos Lingüísticos para el Español y el Portugués 145

Linguateca: um Centro de Recursos Distribuído para o Processamento Computacional da Língua Portuguesa 147
D. Santos, A. Simões, A. Frankenberg, A. Pinto, A. Barreiro, etAl

SENSEM: Base de Datos Verbal del Español 155
Ana Fernández, Gloria Vázquez, Irene Castellón

87

89

97

105

112

118

ctors 123

D 131

137

145

147

155

Inclusão de Informação Semântica dos Adjetivos na Base da Rede Wordnet para o Português do Brasil 164
Ariani Di Felippo, Bento Carlos Dias-da-Silva

Refinamiento Semántico para Recuperación de Información desde la Web 172
R. Motz, C. Deco, C. Bender, J. Saer, M. Chiari

Avaliação de Centering em Resolução Pronominal da Língua Portuguesa 180
A. M. Aires, J.C. B. Coelho, S. Collovini, P. Quaresma, R. Vieira

Búsqueda de Respuestas Basada en Redundancia: un estudio para el Español y el Portugués 188
Luis Villaseñor, Manuel Montes-y-Gómez, Alejandro del Castillo

Banco Terminológico, una Base de Conocimientos Léxica Multipropósito 196
Gerardo Sierra, Gabriel Garduño, Marlene Moreno, Gabriel Castillo

CLI: An Open Linguistic Corpus for Engineering 203
A. Medina, G. Sierra, G. Garduño, C. Méndez, R. Saldaña

Analizador Morfológico Disponible:un recurso importante para PLN en Español 209
Alexander Gelbukh, Grigori Sidorov

Estándares de Anotación Morfosintáctica para el Español 217
Montserrat Civil, M. Antònia Martí

Modelo de Base de Datos Léxica para el Desarrollo de Materiales de EVAO 225
Roser Morante, M. Antònia Martí

Anotación de Roles Semánticos en el Corpus 3LB 233
B. Navarro, P. Moreda, B. Fernández, R. Marcos, M. Palomar

3LB-RAT: una Herramienta para la Anotación Referencial 241
Maximiliano Saiz-Noeda, Rubén Izquierdo

Safe Resource Sharing in an Application Building Environment 249
D. M. de Matos, Ângelo Reis, Marco Costa, Nuno J. Mamede

A Run-Time Shared Repository for NLP Tools 257
João Graça, Nuno J. Mamede, João D. Pereira

Towards a Flexible Syntax/Semantics Interface 265
Luísa Coheur, Fernando Batista, Nuno J. Mamede

A Tool for Automatic Detection of Antonymy Relations 273
Cupertino Lucero, David Pinto, Héctor Jiménez-Salazar

Índices homogéneos versus heterogéneos en la fusión de colecciones en entornos de recuperación de información distribuida 282
F. Martínez-Santiago, M.A. García-Cumbreras, L. A. Ureña-López

Adapting a Named Entity Recognition System for Spanish to Portuguese 292
Thamar Solorio, Aurelio López-López

Term Spoting with Linguistics and Statistics 298
Joana L. Paulo, Nuno J. Mamede

Workshop on
Machine Learning for Scientific Data Analysis 305

Noise Elimination with a Re-Sampling Algorithm 307
H. Jair Escalante, Olac Fuentes

Concept Lattice Learning in Analytic Induction 317
M. Martínez-Morales, A. Alatriste-Trujillo, A. Guerra-Hernández

Feature Selection in an Electric Billing Data Base using Concave Programming 326
M. Mejía-Lavalle, J. Frausto, G. Rodríguez

NSC: A New Progressive Sampling Algorithm 335
Alfonso Estrada, Eduardo F. Morales

Web Site Usage Patterns Discovery Using a Graph-Based Representation 345
Ivan Olmos, Jesús A. Gonzalez

Subgraph Isomorphism Detection with a Representation based on Lists of codes 355
Ivan Olmos, Jesús A. Gonzalez, Mauricio Osorio

Using Simulations to Check a Scientific Model´s Metaphysical Underpinnings 363
Joseph Phillips, Raghuveer Kumarakrishnan

Identification of Stellar Population in Galactic Spectra using the Hierarchial Decision Ensemble 371
Trilce Estrada, Olac Fuentes

Índices homogéneos versus heterogéneos en la fusión de colecciones en entornos de recuperación de información distribuida

Fernando Martínez-Santiago[1], Miguel A. García-Cumbreras[1], and L. Alfonso Ureña-López[1]

Grupo Sistemas Inteligentes de Acceso a la Información, Departamento de Informática, Universidad de Jaén. Spain
{dofer, magc, laurena}@ujaen.es
http://wwwdi.ujaen.es/~gsi

Resumen Este artículo presenta la aplicación del algoritmo de fusión de colecciones 2-step RSV (retrieval status value) aplicado a entornos de recuperación de información distribuída (DIR). Los experimentos realizados muestran que el algoritmo propuesto es estable ante distintas variantes de las funciones de pesado utilizadas en los índices locales. Además, se obtienen unos valores de precisión más altos que los obtenidos con el algoritmo de fusión de colecciones CORI (COllection Retrieval Inference network).

1. Introducción

Un sistema DIR clasifica normalmente las colecciones de documentos por relevancia en base a una consulta, seleccionando el mejor conjunto de colecciones de un ranking, y fusionando la clasificación de documentos devuelta para cada una de las colecciones seleccionadas. Este último aspecto se conoce como el problema de la fusión de colecciones [8]. El principal objetivo de este trabajo es un exhaustivo análisis de la utilización de índices homogéneos y heterogéneos en la fusión de documentos en recuperación de información distribuida.

El cálculo de la relevancia documental en dos pasos es un algoritmo que obtiene buenos resultados en ambientes distribuidos.

Nuestro objetivo es el estudio de la estabilidad de 2-step RSV cuando se utilizan distintas funciones de pesado para los índices locales, y establecer una comparación entre CORI y 2-step RSV como algoritmos de fusión de colecciones. La selección de colecciones se ha realizado con CORI en ambos casos. Los resultados obtenidos muestran que, mientras que CORI requiere unos índices hasta cierto punto comparables, 2-step RSV es más estable en ambientes muy heterogéneos.

Los siguientes capítulos están organizados de la siguiente forma. Primero presentamos una breve revisión y descripción del método que proponemos, integrado dentro del modelo CORI [2]. En la sección 3 detallamos los experimentos llevados a cabo así como los resultados obtenidos. Finalmente presentamos nuestras conclusiones y líneas de trabajo futuro.

F. Martínez, M. A. García, L. A. Ureña, Taller de Herramientas y Recursos Linguísticos para el Español y el Portugués, pp.282-291, Copyright Iberamia 2004.

Workshop on
Virtual Laboratories 379

Relational Probabilistic Model for Virtual Laboratories for Teaching and Learning 381
Julieta Noguez, L. Enrique Sucar

Using Spoken Language for Human-Robot Communication and Telecontrol 390
E. Uraga-Serratos, C. Miranda-Palma, O. Mayora-Ibarra, E. Sucar

Modelo Genérico y Modular para Desarrollar Laboratorios Virtuales en Telerobótica 399
Fernando D. Von Borstel, José L. Gordillo

Extending OODVR, a Collaborative Virtual Robotics Enviroment 409
L. Muñoz-Gómez, M. Alencastre-Miranda, I. Rudomín, R. Swain-Oropeza, G. Arechavaleta, J. Ramirez-Uresti

Laboratorio Virtual para el diseño, ajuste y análisis de reguladores analógicos convencionales 419
J.L. Díaz Rodríguez, Aldo Pardo García, Antonio Gan Acosta

Un Sistema de Diálogo para Agentes Conversacionales en Laboratorios Virtuales 426
Ramón Ramirez Guzmán, Ana Luisa Solis González Cosío

Using Cognitive Maps to Develop a Student Model 435
Alejandro Peña, Humberto Sossa

Laboratorio Virtual Simulado y Remoto para el Control de Motores a Través de Internet 446
A. Pardo García, F. Rozo García, D. E. Murillo Márquez

Plant Lab: an AR Plant Laboratory for the Typical Home 453
Isaac Rudomín, Marissa Díaz, Daniel Rivera

2. Cálculo de la relevancia documental en dos pasos

El cálculo de la relevancia documental en dos pasos o 2-step RSV consiste en agrupar las frecuencias documentales de un término dado de una consulta [4]. El método requiere calcular la puntuación obtenida por cada documento cambiando la frecuencia documental de cada término que aparece en la consulta: dado un término de la consulta, su nueva frecuencia documental será el resultado de sumar a su frecuencia documental original la frecuencia documental alcanzada por tal término en el resto de las subcolecciones seleccionadas. Por ejemplo, si las colecciones I_1 e I_2 son seleccionadas, y la consulta contiene el término "goverment", entonces la nueva frecuencia global será $df_{I_1}(government) + df_{I_2}(government)$.

Dada una consulta los dos pasos son:

1. La fase de preselección de documentos se corresponde con el lanzamiento de la consulta sobre cada subcolección seleccionada I_j. Como resultado de unir los documentos más relevantes recuperados para cada colección obtenemos una única colección (lista) de documentos preseleccionados I'.
2. La fase de reordenamiento consiste en reindexar la colección (ahora ya será un ranking) I', pero considerando tan sólo el vocabulario de la consulta. Finalmente, se elabora un nueva consulta formada por los términos anotados y se lanza tal consulta sobre el nuevo índice.

La hipótesis de este método es como sigue: Las puntuaciones logradas por documentos en sistemas de IR autónomos no son comparables aun utilizando el mismo modelo IR, debido a que la puntuación alcanzada por el documento es relativa a la colección a la que pertenece. Es más, la frecuencia documental alcanzada por cada término (depende de la colección) es determinante en la puntuación alcanzada por cada documento. Es posible recalcular el peso de cada documento como si todas las colecciones independientes formaran parte de una única colección local, considerando las siguientes simplificaciones: a) sólo es necesario reindexar los términos aparecidos en la consulta inicial y b) sólo es necesario reindexar las listas de documentos devueltos por cada sistema IR independientemente. Para ello, sólo es necesario almacenar localmente una lista de términos índice junto con la frecuencia documental de tal término en cada colección considerada. Así, el sistema DIR debe enviar la consulta a cada colección seleccionada, descargar los documentos más relevantes, crear un nuevo índice con tales documentos por medio de una fórmula de pesado y, finalmente la consulta se formula contra el nuevo índice. Los documentos I' se ordenan utilizando un nuevo índice. Nótese que la creación de este índice en tiempo de consulta es posible porque el vocabulario del nuevo índice está compuesto sólamente por términos de la consulta, y porque el índice puede crearse sobre unos pocos documentos, conforme el usuario solicite más resultados, tal como se expone en la sección 2.2.

2.1. Elementos para el cálculo del índice global

En este trabajo, hemos adoptado la función OKAPI-BM25 [5], como esquema de pesado durante la segunda fase, el cual conjuga tanto el peso dado a cada término de la consulta, como la expansión de ésta. OKAPI-BM-25 se formula como sigue:

$$\sum_{T \in Q} W^{(1)} \frac{(k_1 + 1)tf}{K + tf} qtf \tag{1}$$

donde

- Q es una consulta conformada por los términos T.
- K se calcula como $K = k_1((1 - b) + b * dl/avgdl)$.
- k_1 y b son constantes que se fijan en $k1 = 1,25$ y $b = 0,75$.
- tf es la frecuencia del término en un documento dado.
- qtf es la frecuencia del término en la consulta Q.
- dl y $avdl$ son respectivamente la longitud del documento y la longitud media del documento en la colección.
- $W^{(1)}$ es el peso del término T en Q propuesto por Robertson y Spark-Jones [6].

$$W^{(1)} = \log \frac{(r + 0{,}5)/(R - r + 0{,}5)}{(n - r + 0{,}5)/(N - n - R - r + 0{,}5)} \tag{2}$$

  - N es el número de documentos en la colección.
  - n es la frecuencia documental (número de documentos que contienen el término).
  - R es el número de documentos conocidos que son relevantes para la consulta dada.
  - r es el número de documentos relevantes que contienen el término.

La cuestión es cómo aplicar la fórmula globalmente (sobre la colección I') mejor que localmente (para cada colección I_i). Se conocen varios elementos y otros se aprenden o aproximan:

- La frecuencia del término se conoce porque los documentos deben descargarse por el sistema DIR antes del segundo paso del *2-step RSV*.
- La frecuencia del documento n es la suma de las frecuencias locales de los documentos n_i. El número de documentos devueltos es df del término. Además, existen varios enfoques para aprender este factor, tales como el conocido algoritmo *Query-based sample* [1].
- El número de documentos en la colección N se obtiene de la misma manera que la frecuencia documental global: agrupando cada una por tamaño colección local.; si el tamaño de cada colección no está disponible, se aproxima usando los algoritmos *Capture-recapture* [3] o *sample-resample* [7].
- Finalmente, la longitud del documento medio se aprende por medio del algoritmo *Query-based*.

2.2. Un posible sistema DIR basado en el cálculo 2-step RSV

De todos los factores involucrados en la fórmula 1, la frecuencia documental es el más costoso, ya que requiere descargarse cada documento como paso previo a evaluarlo.

Por ejemplo, si deseamos crear un sistema que muestre los documentos ordenados de 10 en 10 el procedimiento que minimiza la descarga, queda como sigue:

1. Sea la colección distribuida formada por L subcolecciones, $I = I_1, ...I_L$. Dada una consulta, con L' ($L' \leq L$) el número de subcolecciones seleccionados por el sistema DIR (por ejemplo utilizando el algoritmo CORI).
2. El sistema descarga m documentos para cada subcolección $I_i, 1 \leq i \leq L'$. De esta manera, el sistema DIR descarga un conjunto de documentos $I' = \{I'_1, \cdots, I'_{L'}\}$, $|I'| = L' * m \geq 10$, y $I'_i = \{d_1^i \cdots d_n^i\}$, $|I'_i| = m$ es el conjunto de documentos descargados de I_i.
3. Los documentos I' son repesados y alineados de acuerdo al 2-step RSV.
4. Los 10 primeros documentos dentro las lista puntuada I' se eliminan de cada conjunto de documentos I'_i de donde han sido extraidos. Si hay algún subconjunto vacío I'_i (alguna lista local vacía) antes de llegar al número de resultados globales deseado, se descargan los m documentos de I_i y repesa el conjunto de nuevos documentos I'_i. Se repite el paso previo hasta que cada subconjunto I'_i contenga al menos un documento.
5. Mostrar los 10 primeros documentos al usuario.
6. Si el usuario requiere más documentos, volver al paso 2.

3. Experimentos y resultados

3.1. Metodología de Experimentación

Los pasos seguidos para realizar cada experimento son los siguientes: El escenario tiene un diseño off-line como se describe a continuación:

1. Funciones de pesado usadas en los índices. Decidimos medir la estabilidad del modelo propuesto, y para ello hemos trabajado con dos conjuntos de índices, *OKAPI* y uno *aleatorio*. Trabajando con el primero todas las colecciones han sido indexadas con el método de pesado *OKAPI*. En el segundo caso hemos tomado de forma aleatoria una función de pesado de entre las que tiene implementadas el sistema ZPRISE. La colección TREC-1 ha sido indexada utilizando sólo OKAPI.
2. Se aplican algoritmos de muestreo basados en la consulta para obtener el modelo del lenguaje: vocabulario, tamaño medio del documento y tamaño de la colecciones. La frecuencia de los términos del documento, del vocabulario aprendido, se devuelve, para cada colección local, utilizando consultas con un único término. La suma de los documentos devueltos para cada colección es justo la frecuencia de documento para cada consulta.

Una vez que los índices están creados y el modelo de lenguaje aprendido, el sistema se evalúa. Para evaluar el método propuesto hemos usado los conjuntos de consultas del TREC-1 y TREC-2, un total de 100 consultas. Para cada una:

1. Utilizamos el algoritmo CORI para puntuar cada colección para esa consulta.
2. Elegimos las más prometedoras mediantes un algoritmo de *clustering*, como el descrito en [2].
3. Cada colección seleccionada obtiene una lista local de documentos lanzando de nuevo cada consulta sobre el índice local (CORI o un índice aleatorio).
4. En ciertos experimentos, aplicar realimentación por pseudo-relevancia (PRF) en cada colección, a nivel local. Se ha utilizado OKAPI-BM25 (eq. 1 y 2). Se tienen en cuenta los 10 primeros términos de los 10 primeros documentos.
5. La fusión de documentos se ha realizado utilizando los algoritmos de fusión CORI y 2-step RSV.

Finalmente, la evaluación se ha medido en términos de precisión y cobertura. La medida de precisión es la precisión media obtenida en los 5,10,20 y 100 primeros documentos, así como la precisión media en 11 puntos de cobertura.

3.2. Descripción de las colecciones de prueba

Los experimentos se han realizado utilizando tres particiones del TREC1 y TREC2. Las colecciones del TREC1 y TREC2 corresponden con textos publicados entre 1987 y 1990 en diversos diarios, agencias de noticias y editoriales. En total, son más de dos gigabytes de datos repartido entre 740.000 documentos.

Sobre estas trece colecciones se han realizado tres juegos de prueba, descritos en la tabla 1:

- **TREC-1**. Las 13 colecciones indexadas bajo un único índice. Representa el mejor caso posible.
- **TREC-13**. Cada una de las 13 colecciones originales han sido indexadas separadamente.
- **TREC-80**. Las 13 colecciones originales se ha particionado hasta crear 80 subcolecciones, cada una de las ellas indexadas separadamente. Estas ochenta colecciones se han creado por origen y mediante una distribución aleatoria.

Cuadro 1. Descripción de las colecciones de prueba.

	# of docs.			Size (in MB)		
	Min.	Med.	Max.	Min.	Med.	Max.
TREC-1	741.991	741.991	741.991	2168	2168	2168
TREC-13	10.163	57.066	226.087	33	159,23	260
TREC-80	2473	9273	32.401	23	25,81	30

Las consultas utilizadas se corresponden con las dos primeras ediciones de las conferencias TREC, sumando un total de 100 consultas. En todos los experimentos se han utilizado los campos título y descripción exclusivamente.

3.3. Experimentos con índices homogéneos

En esta sección se comparan CORI y 2-step RSV . Todas las colecciones han sido indexadas utilizando la misma función de pesado, OKAPI.

Cuadro 2. Experimentos DIR con índices homogéneos (colección TREC-13, consultas 101-150)

Fusión	5-prec	10-prec	20-prec	100-prec	Avg-prec
clustering					
CORI	0,468	0,416	0.363	0,265	0,130
2-step RSV	0,480	0,458	0,426	0,329	0,181
Centralizado	0,492	0,492	0,444	0,346	0,194

Cuadro 3. Experimentos DIR con índices homogéneos (colección TREC-80, consultas 51-100)

Fusión	5-prec	10-prec	20-prec	100-prec	Avg-prec
clusterig					
CORI	0,368	0.366	0,362	0,254	0,086
2-step RSV	0,416	0,424	0,396	0,263	0,097
Centralizado	*0,556*	*0,514*	*0,492*	*0,371*	*0,210*

La tabla 2 muestra los resultados obtenidos. Los mejores resultados son para 2-step RSV , cuyo incremento es mejor, en términos de precisión media, entre un 19,4 %, si tomamos los cinco primeros documentos sobre TREC-13, y 107 % si tomamos los veinte primeros documentos sobre TREC-80. En este último caso, en términos absolutos, 2-step RSV mejora a CORI en 7,6 puntos, pero CORI comienza desde una precisión muy baja (0,07), luego el incremento parece mayor de lo que realmente es. CORI consigue sus mejores resultados cuando se seleccionan las primeras diez colecciones. Sin embargo, su rendimiento cae de una forma importante si se seleccionan las veinte primeras colecciones. En este punto 2-step RSV es más estable, ya que seleccionar más colecciones implica una precisión mejor. Este aspecto se muestra en la figura 1, donde se muestra el rendimiento de 2-step RSV en relación con CORI.

Finalmente, comparando CORI y 2-step RSV con un modelo centralizado, un resultado especialmente llamativo es la precisión alcanzada por 2-step RSV sobre 13 subcolecciones, tomando $Top-10$. En este caso, 2-step RSV supera al modelo centralizado, lo cual demuestra que no contemplar todas las subcolecciones no conlleva necesariamente una pérdida de precisión. El buen nivel mostrado por 2-step RSV sobre TREC-13 no se mantiene sobre TREC-80, bajando sensiblemente la precisión obtenida, si bien el motivo más probable de esto sea debido a la baja proporción de colecciones consideradas en relación con el total disponible.

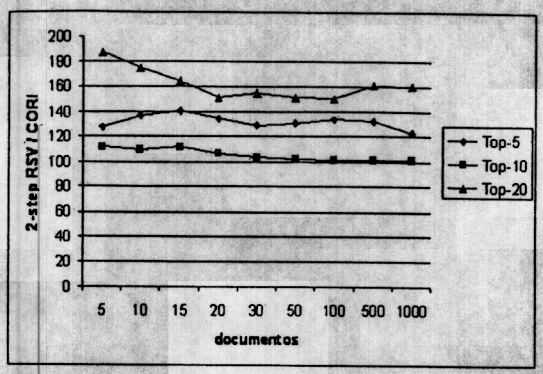


Figura 1. Comparación entre *2-step RSV* y CORI (col. TREC-80, preguntas 51-100). El caso base (100 %) muestra la precisión de CORI.

De hecho, si se toman las primeras 20 colecciones, que representan el 25 % del total, se alcanza una precisión media ligeramente por encima del 70 % de la precisión alcanzada por el modelo centralizado.

3.4. Experimentos con índices heterogéneos

En este apartado se estudia cómo afectan las variaciones en la función de pesado de cada subcolección. Mientras en el anterior apartado todas las subcolecciones están indexadas con el esquema OKAPI, en este apartado cada subcolección ha sido aleatoriamente indexada con una de las funciones disponibles en el sistema IR ZPrise.

Las tablas 4 y 5 demuestran que los resultados obtenidos siguen la misma tendencia de la sección anterior:

- Para el juego de prueba TREC-13, 2-step RSV anda cerca de duplicar la precisióon media registrada por CORI en los procedimientos de selección probados (Top-5, Top-10 y *clustering*).
- En el caso de TREC-80, esta diferencia no es tan pronunciada excepto si tomamos las veinte primeras colecciones, en cuyo caso, de nuevo, CORI se desploma no alcanzando ni la tercera parte del rendimiento registrado por 2-step RSV . En el resto de los casos la mejora de precisión queda alrededor del 40 %.

Tanto 2-step RSV como CORI pierden precisión cuando se introduce cierta aleatoriedad en la selección de funciones de pesado. Esto hasta cierto punto es inevitable pues los resultados locales alcanzados por cada subcolección son en muchos casos peores que con OKAPI. Sin embargo, la pérdida de precisión no es la misma para CORI que para 2-step RSV . Centrándonos en los resultados registrados sobre TREC-80, mientras que 2-step RSV pierde en general entre

Cuadro 4. Experimentos DIR con índices heterogéneos (colección TREC-13, consultas 101-150)

Fusión	5-prec	10-prec	20-prec	100-prec	Avg-prec
clustering					
CORI	0,376	0,312	0,281	0,208	0,091
2-step RSV	0,480	0,464	0,423	0,327	0,179
Centralizado	*0,492*	*0,492*	*0,444*	*0,346*	*0,194*

Cuadro 5. Experimentos DIR con índices heterogéneos (colección TREC-80, consultas 51-100)

Fusión	5-prec	10-prec	20-prec	100-prec	Avg-prec
clusterig					
CORI	0,232	0,258	0,245	0,180	0,044
2-step RSV	0,408	0,392	0,355	0,234	0,074
Centralizado	*0,556*	*0,514*	*0,492*	*0,371*	*0,210*

un 10 % y un 20 %, la pérdida de precisión media de CORI con la inclusión de diversos esquemas de pesado se cifra, en general, por encima del 40 %. Esta situación queda resumida en las figuras 2 y 3, en las cuales se muestra el cociente entre la precisión alcanzada con OKAPI y con los índices aleatorios utilizados en este apartado. Tanto utilizando *clustering* como Top-10, se aprecia que, mientras 2-step RSV se mantiene en torno al 80 % del rendimiento registrado con índices homogéneos, CORI pasa a penas del 60 %. Por lo tanto, es posible afirmar que 2-step RSV es un algoritmo estable frente a variaciones en las funciones de pesado utilizadas en cada subcolección.

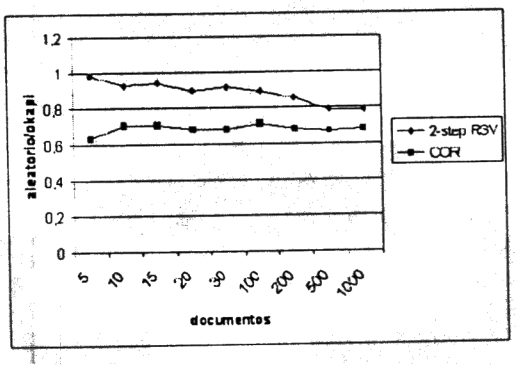


Figura 2. Impacto del uso de índices heterogéneos en el rendimiento de CORI y 2-step RSV (col. TREC-80, consultas 51-100, clustering). El caso base (100 %) representa la precisión alcanzada por el algoritmo usando índices homogéneos (sólo OKAPI).

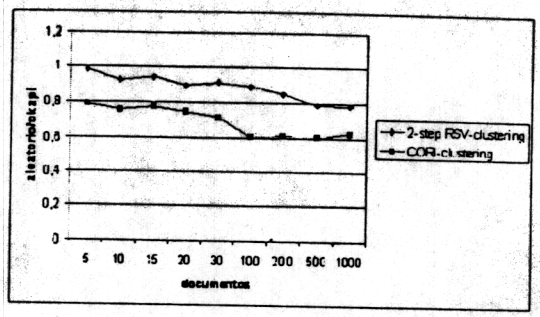


Figura 3. Impacto del uso de índices heterogéneos en el rendimiento de CORI y 2-step RSV (col. TREC-80, consultas 51-100, Top-10). El caso base (100 %) representa la precisión alcanzada por el algoritmo usando índices homogéneos (sólo OKAPI).

4. Conclusiones y Trabajos Futuros

Los sistemas DIR necesitan fusionar listas de documentos de varias colecciones. Normalmente, distintos enfoques de fusión de documentos tienen en cuenta sólo los términos originales de la consulta para re-evaluar cada documento. En este trabajo nos hemos centrado en el uso de diferentes funciones de pesado a la hora de indexar las colecciones. Los experimentos muestran que 2-step RSV es robusto cuando se aplican variantes de las funciones de pesado. Tanto 2-step RSV como CORI pierden precisión cuando se introduce un método aleatorio en la selección de la función de pesado. Esta pérdida de precisión en CORI es superior al 40 %, mientras que en 2-step RSV se sitúa entre un 10 % y un 20 %.

Un trabajo futuro es el estudio de PRF enviando la consulta expandida a nivel global sobre cada colección. Así la consulta original del usuario es enviada a cada colección seleccionada. Entonces, el 2-step RSV es aplicado y la consulta original del usuario es expandida. Esta consulta expandida es enviada a cada colección. De esta forma esperamos mejorar los resultados locales.

Finalmente, también estamos interesados en las capacidades del algoritmo 2-step RSV sobre otros escenarios DIR: con más colecciones, con mayor número de modelos de pesado distintos y con diferentes tamaños.

5. Agradecimientos

Esta investigación ha sido financiada por el Ministerio de Ciencia y Tecnología (MCYT) con el proyecto TIC2003-07158-C04-04.

Referencias

1. J. P. Callan, M. Connell, and A Du. Automatic discovery of language models for text databases. In *ACM-SIGMOD International Conference on Management of Data*, pages 470–490, 1999.
2. J. P. Callan, Z. Lu, and W. B. Croft. Searching distributed collections with inference networks. In *Proceedings of the 18th International Conference of the ACM SIGIR'95*, pages 21–28, New York, 1995. The ACM Press.
3. K. Liu, C. Yu, W. Meng, A. Santos, and C. Zhang. Discovering the representative of a search engine. In *Proceedings of 10th ACM International Conference on Information and Knowledge Managemenet (CIKM)*, 2001.
4. F. Martínez-Santiago, A. Montejo-Ráez, L.A. Ureña, and M.C. Diaz. SINAI at CLEF 2003: Merginig and decompounding. In Carol Peters, editor, *Proceedings of the CLEF 2003 Cross-Language Text Retrieval System Evaluation Campaign. Lecture Notes in Computer Science*, pages 99–109, 2003.
5. S. E Robertson, S. Walker., and M. Beaulieu. Experimentation as a way of life:Okapi at TREC. *Information Processing and Management*, 1(36):95–108, 2000.
6. S.E. Robertson and K. S. Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–146, 1976.
7. L. Si and J. Callan. Relevant document distribution estimation method for resource selection. In The ACM Press., editor, *Proc. of the 26 Annual Int'l ACM SIGIR Conference on Research and Development in Information Retrieval*, Toronto, 2003.
8. E. Voorhees, N. K. Gupta, and B. Jhonson-Laird. The collection fusion problem. In D. Harman, editor, *Proceedings of the 3th Text Retrieval Conference TREC-3*, volume 500-225, pages 95–104, Gaithersburg, 1995. National Institute of Standards and Technology, Special Publication.

Machine Learning for Scientific Data Analysis

Virtual Laboratories

Deduction and Reasoning Techniques

Herramientas y Recursos Linguísticos para el Español y el Portugués

Lexical Resources and the Web for Word Sense Disambiguation