



## Artículos

**Reconocimiento y síntesis de voz (I)**

Evaluación del modelado acústico y prosódico del sistema de conversión texto-voz Cotovía <i>Francisco Campillo, Eduardo Rodríguez</i> .....	5
Reconocimiento automático de emociones utilizando parámetros prosódicos <i>Iker Luengo, Eva Navas, Inmaculada Hernández, Jon Sánchez</i> .....	13
New Advances in Cross-Task and Speaker Adaptation for Air Traffic Control Tasks <i>Ricardo Córdoba, Javier Macías, Valentín Sama, Roberto Bara, José Manuel Pardo</i> .....	21
Main Issues in Grapheme-to-Phoneme Conversion for TTS <i>Tatyana Polyakova, Antonio Bonafonte</i> .....	29

**Análisis automático del contenido textual**

NERUA: sistema de detección y clasificación de entidades utilizando aprendizaje automático <i>Oscar Fernández, Zornitsa Kozareva, Andrés Montoyo, Rafael Muñoz</i> .....	37
Análisis de los fenómenos lingüísticos de los mensajes de correo electrónico en catalán desde la perspectiva de la traducción automática <i>Joaquim Moré, Salvador Climent, Antoni Oliver, Mariona Taulé</i> .....	45
Construcción y minimización eficiente de transductores de letras a partir de diccionarios con paradigmas <i>Sergio Ortiz, Mikel L. Forcada, Gema Ramírez</i> .....	51
Evaluación de resúmenes automáticos mediante QARLA <i>Enrique Amigó, Anselmo Peñas, Julio Gonzalo, Felisa Verdejo</i> .....	59

**Traducción automática (I)**

Modelo estocástico de traducción basado en N-gramas de tuplas bilingües y combinación log-lineal de características <i>José B. Mariño, Rafael Banchs, Josep M<sup>o</sup> Crego, Adrià de Gispert, Patrik Lambert, José A. R. Fonollosa</i> .....	69
Traducción automática estadística basada en n-gramas <i>Antoni Oliver, Gemma Boleda, Maite Melero, Toni Badia</i> .....	77
Algoritmo de Decodificación de Traducción Automática Estocástica basado en N-gramas <i>José M<sup>o</sup> Crego, José B. Mariño, Adrià de Gispert</i> .....	85
Bilingual phrases for statistical machine translation <i>Ismael Garcia-Varea, Francisco Nevado, Daniel Ortiz, Jesús Tomás, Francisco Casacuberta</i> .....	93

**Extracción y recuperación de información monolingüe y multilingüe**

El tratamiento de la polisemia en la extracción de léxicos bilingües a partir de corpora paralelos <i>Pablo Gamallo, Susana Sotelo</i> .....	103
Un entorno para el desarrollo y la evaluación de un sistema de búsqueda de respuestas en euskara <i>Olatz Ansa, Xabier Arregi, Itsaso Esparza, Andoni Valverde</i> .....	111
Text Categorization using bibliographic records: beyond document content <i>Arturo Montejo, Luis Alfonso Ureña, Ralf Steinberger</i> .....	119
Detección automática de spam utilizando regresión logística bayesiana <i>M<sup>o</sup> Teresa Martín, Antonio J. Ortiz, Luis Alfonso Ureña, Miguel Angel García</i> .....	127

**Lexicografía computacional**

Explotación computacional del metalenguaje en corpus especializados para la generación de lexicones no convencionales <i>Carlos Rodríguez</i> .....	137
Transforming a Constituency Treebank into a Dependency Treebank <i>Alexander Gelbukh, Hiram Calvo, Sulema Torres</i> .....	145
Algoritmo de stemming para el gallego <i>Miguel Rodríguez, Marisa Moreda, Angeles S. Places, Eloy Vázquez</i> .....	153
A Proposal for a Shallow Ontologization of Wordnet <i>Salvador Climent, Jordi Aterias, Joaquim Moré, German Rigau</i> .....	161

**Resolución de la ambigüedad léxica**

Exploiting Rules for Word Sense Disambiguation in Machine Translation <i>Lucia Specia, Maria das Graças Volpe, Mark Stevenson</i> .....	171
Un Enfoque Integrado para la Desambiguación <i>Jordi Aterias</i> .....	179
Uso flexible de soluciones evolutivas para tareas de Generación de Lenguaje Natural <i>Raquel Hervás, Pablo Gervás</i> .....	187
Exploring the construction of semantic class classifiers for WSD <i>Luis Villarejo, Luíís Márquez, German Rigau</i> .....	195

**Semántica, pragmática y discurso**

Nueva Técnica de Generación Automática de Gramáticas Para Sistemas de Diálogo <i>Zoraida Callejas, Ramón López-Cózar</i> .....	205
Dos aproximaciones basadas en reglas para la gestión del diálogo <i>David Griol, Lluís F. Hurtado, Emilio Sanchis, Encarna Segarra</i> .....	213
Verificación de tema en sistemas de diálogo mediante la aplicación de un test de hipótesis bayesiano <i>David Pérez-Piñar, Carmen García</i> .....	221
Utilización de medidas de confianza en sistemas de comprensión del habla <i>Valentín Sama, Javier Ferreiros, Fernando Fernández, Rubén San, José Manuel Pardo</i> .....	229

**Modelos lingüísticos, matemáticos y psicológicos del lenguaje**

Evaluating the LHLA lexical aligner on Spanish, Brazilian Portuguese and Basque parallel texts <i>Helena de Medeiros, Maria das Graças Volpe, Mikel L. Forcada</i> .....	237
Identificación de formas lógicas en el caso del español: propuesta de un modelo basado en reglas y aprendizaje automático. <i>Fernando Martínez, Miguel Angel García</i> .....	245

Syntax-driven bindings of Spanish clitic pronoun <i>Ivan Vladimir Meza, Luis Alberto Pineda</i> .....	253
Diccionarios basados en taxonomías con estructura de grafo orientado acíclico <i>Antonio Ramón Vaquero, Francisco José Alvarez, Fernando Sáenz</i> .....	259
<b>Reconocimiento y síntesis de voz (II)</b>	
Comparación de modelos de lenguaje en tareas de transcripción automática de noticieros televisivos <i>Francisco Javier Diéguez, Carmen García, Antonio Cardenal</i> .....	269
Analysis of prosodic features towards modelling of emotional and pragmatic attributes of speech <i>Jordi Adell, Antonio Bonafonte, David Escudero</i> .....	277
Un Sistema de Diálogo Multicanal para Acceder a la Información y Servicios de las Administraciones Públicas <i>Meritxell González, Marta Gatiús</i> .....	285
Analisis y síntesis de expresión emocional en cuentos leídos en voz alta <i>Virginia Francisco, Pablo Gervás, Raquel Hervás</i> .....	293
<b>Aplicaciones Industriales del PLN</b>	
Una propuesta de infraestructura para el Procesamiento del Lenguaje Natural <i>Lorenza Moreno, Armando Suárez</i> .....	303
Hacia una arquitectura flexible para sistemas de predicción de palabras: propuesta de diseño y evaluación <i>Sira Elena Palazuelos, José Luis Martín, Lisset Hierrezuelo, Javier Macías</i> .....	311
A Named Entity Recognition System based on a Finite Automata <i>Muntsa Padró, Lluís Padró</i> .....	319
Topic Identification based on Bayesian Belief Networks in the context of an Air Traffic Control Task <i>Fernando Fernández, Luis Fernando D'Haro, Javier Ferreiros, Juan Manuel Montero, Rubén San</i> .....	327
<b>Traducción automática (II)</b>	
Clasificación y generalización de formas verbales en sistemas de traducción estocástica <i>Adrià de Gispert, José B. Mariño, Josep M<sup>a</sup> Crego</i> .....	335
Sistema de Traducción Oral para el Castellano, Catalán e Inglés <i>Elisabet Comelles, Victoria Arranz, David Farwell</i> .....	343
Técnicas mejoradas para la traducción basada en frases <i>Marta Ruiz, José A. R.</i> .....	351
Computer-Assisted Translation using Finite-State Transducers <i>Jorge Civera, Elsa Cubel, Antonio Luis Lagarda, Francisco Casacuberta, Enrique Vidal, Juan Miguel Vilar</i> .....	357
<b>Lingüística de corpus</b>	
3LB-LEX: léxico verbal con frames sintáctico-semánticos <i>Montserrat Civit, Izacum Aldezabal, Eli Pociello, Mariona Taulé, Joan Aparicio, Lluís Màrquez, Borja Navarro, Joan Castellví, María Antònia Martí</i> .....	367
Designing an active learning based system for corpus annotation <i>Bertjan Busser, Roser Morante</i> .....	375
Uso del punto de transición en la selección de términos índice para agrupamiento de textos cortos <i>Héctor Jiménez, David Pinto, Paolo Rosso</i> .....	383
Una aproximación multilingüe a la clasificación de preguntas basada en aprendizaje automático <i>David Tomás, Empar Bisbal, José Luis Vicedo, Armando Suárez, Lidia Moreno</i> .....	391
<b>Gramáticas y formalismos para el análisis morfológico y sintáctico</b>	
Generación automática de analizadores sintácticos a partir de esquemas de análisis <i>Carlos Gómez, Jesús Vilares, Miguel A. Alonso</i> .....	401
Tipología de errores gramaticales para un corrector automático <i>Ana M<sup>a</sup> Díaz</i> .....	409
Evaluación del clustering de páginas web mediante funciones de peso y combinación heurística de criterios <i>Raquel Martínez, Víctor Fresno, Arantza Casillas, Soto Montalvo</i> .....	417
Identifying Jargon in Texts <i>Stephen Helmreich, Jesús Llevadías, David Farwell</i> .....	425
<b>Proyectos</b>	
VILE: Estudio acústico de la variación inter e intralocutor en español <i>Joaquim Llisteri</i> .....	435
The project HOPS: Enabling an Intelligent Natural Language Based Hub for the Deployment of Advanced Semantically Enriched Multi-channel Mass-scale Online Public Services <i>Marta Gatiús, Meritxell González</i> .....	437
OAC-onto: Open Archive Cataloger, ontologías y metadatos <i>Inés Jacob, Joseba Abaitua, JosuKa Díaz, Fernando Quintana, Jon Fernández, Txus Sánchez</i> .....	439
Presentación del proyecto MeLLANGE (Multilingual eLearning in LANGuage Engineering) <i>Carme Colominas, Toni Badia</i> .....	441
El proyecto METIS-II <i>Toni Badia, Gemma Boleda, Maitte Melero, Antoni Oliver</i> .....	443
Implementación de Sistemas de Diálogo en Dial-XML <i>Ramón López-Cózar, Zoraida Callejas, Miguel Gea, Nuria Medina, Domingo Martín</i> .....	445
<b>Demostraciones</b>	
VENSES - A Linguistically-Based System for Semantic Evaluation <i>Delmonte Rodolfo</i> .....	449
Demostración de una interfaz vocal para el control de un sistema de alta fidelidad <i>Fernando Fernández, Javier Ferreiros, Valentín Sama, Juan Manuel Montero, Rafael García</i> .....	451
Sistema de diálogo para el Proyecto DIHANA <i>Lluís F. Hurtado, Fernando Blat, Sergio Grau, David Griol, Emilio Sanchis, Encarna Segarra</i> .....	453
TXALA un analizador libre de dependencias para el castellano <i>Jordi Atserias, Elisabet Comelles, Aingeru Mayor</i> .....	455
GAG: Generación Automática de Gramáticas en un Sistema Conversacional de Interacción Oral <i>Zoraida Callejas, Ramón López-Cózar</i> .....	457
PHILIPS: Intelligent Speech Interpretation - la tecnología inteligente de reconocimiento de voz <i>Javier Viver</i> .....	459

# XXI Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural



Universidad de Granada, 14-16 de Septiembre de 2005



I Congreso Español de Informática (CEDI 2005)



Universidad de Granada

## **EDITADO POR:**

Ramón López-Cózar Delgado (Universidad de Granada)

L. Alfonso Ureña López (Universidad de Jaén)

## **COMITÉ CIENTÍFICO:**

### **Presidentes:**

Ramón López-Cózar Delgado (Universidad de Granada)

L. Alfonso Ureña López (Universidad de Jaén)

### **Miembros:**

Prof. José Gabriel Amores Carredano (Universidad de Sevilla)

Prof. Toni Badia i Cardús (Universitat Pompeu Fabra)

Prof. Manuel de Buenaga Rodríguez (Universidad Europea de Madrid)

Prof.<sup>a</sup> Irene Castellón Masalles (Universitat de Barcelona)

Prof.<sup>a</sup> Arantza Díaz de Ilarraza (Euskal Herriko Unibertsitatea)

Prof. Antonio Ferrández Rodríguez (Universitat d'Alacant)

Prof. Mikel Forcada Zubizarreta (Universitat d'Alacant)

Prof.<sup>a</sup> Ana María García Serrano (Universidad Politécnica de Madrid)

Prof. Koldo Gojenola Gallettebeitia (Euskal Herriko Unibertsitatea)

Prof. Xavier Gómez Guinovart (Universidade de Vigo)

Prof. Julio Gonzalo Arroyo (Universidad Nacional de Educación a Distancia)

Prof. José Miguel Goñi Menoyo (Universidad Politécnica de Madrid)

Prof. Joaquim Llisterri (Universitat Autònoma de Barcelona)

Prof. Ramón López-Cózar Delgado (Universidad de Granada)

Prof. Javier Macías Guarasa (Universidad Politécnica de Madrid)

Prof. José B. Mariño Acebal (Universitat Politècnica de Catalunya)

Prof.<sup>a</sup> M. Antonia Martí Antonín (Universitat de Barcelona)

Prof. Lluís Padró (Universitat Politècnica de Catalunya)

Prof. Manuel Palomar Sanz (Universitat d'Alacant)

Prof. José Manuel Pardo Muñoz (Universidad Politécnica de Madrid)

Prof. Germán Rigau (Euskal Herriko Unibertsitatea)

Prof. Horacio Rodríguez Hontoria (Universitat Politècnica de Catalunya)

Prof. Kepa Sarasola Gabiola (Euskal Herriko Unibertsitatea)

Prof. L. Alfonso Ureña López (Universidad de Jaén)

Prof.<sup>a</sup> M<sup>a</sup> Felisa Verdejo Mailló (Universidad Nacional de Educación a Distancia)

Prof. Manuel Vilares Ferro (Universidade de Vigo)

## **REVISORES EXTERNOS:**

Alicia Ageno, Pablo Daniel Agüero, Iñaki Alegria, Francesc Alías, Laura Alonso, Javier Artilles, Xabier Artola, Zoraida Callejas, Rafael C. Carrasco, Montserrat Civit, Adrià de Gispert, Helenca Duxans, Nerea Ezeiza, David Farwell, Izaskun Fernández, Miguel Ángel García, Carmen García, Manuel García, José María Gómez, José Carlos González, Luis Hernández, Mikel Lersundi, Fernando Llopis, Fernando López, Pilar Manchón, Manuel J. Maña, Montserrat Marimon, María Teresa Martín, Patricio Martínez-Barco, José Luis Martínez, David Martínez, Fernando Martínez, Juan Manuel

Montero, Andrés Montoyo, Roser Morante, Asunción Moreno, Borja Navarro, Albino Nogueiras, Antoni Oliver, Víctor Peinado, Anselmo Peñas, Jesús Peral, Guillermo Pérez, Juan Antonio Pérez, Enrique Puertas, Eduardo Rodríguez, José Adrián Rodríguez, Maximiliano Saiz, Rubén San-Segundo, Mariona Taulé.

## **COMITÉ LOCAL**

### **Presidente:**

Ramón López-Cózar Delgado (Universidad de Granada)

### **Secretaria:**

Zoraida Callejas Carrión (Universidad de Granada)

### **Miembros:**

Prof.<sup>a</sup> María V. Hurtado Torres (Universidad de Granada)

Prof. José María Guirao Miras (Universidad de Granada)

Prof.<sup>a</sup> Nuria Medina Medina (Universidad de Granada)

Prof. Marcelino J. Cabrera Cuevas (Universidad de Granada)

Prof. Miguel Gea Megías (Universidad de Granada)

## **COLABORADORES**

### **Maquetación de la revista:**

Antonio F. Díaz García (Universidad de Granada)

ISSN: 1135-5948

Depósito Legal: B:3941-91

## Detección automática de Spam utilizando Regresión Logística Bayesiana

Antonio Jesús Ortiz Martos

María Teresa Martín Valdivia

L. Alfonso Ureña López

Miguel Ángel García Cumbreiras

Grupo Sistemas Inteligentes de Acceso a la Información  
Departamento de Informática  
Universidad de Jaén

<http://sinai.ujaen.es>

e-mail: [ajortiz@amsystem.es](mailto:ajortiz@amsystem.es), {maite,laurena,magc}@ujaen.es

**Resumen:** Este artículo presenta un sistema de detección automática de Spam, o correo no deseado, aplicando Regresión Logística Bayesiana (BBR) como técnica de aprendizaje automático, sobre la colección de correos electrónicos SPAMBASE. A modo de comparativa se han aplicado otros dos algoritmos de aprendizaje: el algoritmo SVM (Support Vector Machine), y el algoritmo PLAUM (Perceptron Algorithm with Uneven Margins). La finalidad de este estudio es comprobar la eficiencia y efectividad del algoritmo BBR en la tarea concreta de filtrado de Spam. Como muestran los experimentos, el algoritmo BBR no solo obtiene unos resultados satisfactorios en cuanto a precisión y recall, sino que además es el algoritmo más rápido de los estudiados.

**Palabras clave:** filtrado de correo, Spam, SPAMBASE, BBR.

**Abstract:** This paper presents an Spam automatic detection system using Bayesian Logistic Regression (BBR) as machine learning algorithm, over the SPAMBASE collection. We have also used two machine learning algorithms: SVM and PLAUM, in order to compare the results. Our aim is to check the efficiency and effectiveness of the BBR method. The obtained results show good results in terms of precision and recall. We have also noticed that BBR is the faster algorithm.

**Keywords:** e-mail filter, Spam, SPAMBASE, BBR.

### 1 Introducción

El uso del servicio de correo electrónico ha sufrido en los últimos años un gran crecimiento. Actualmente existe un serio problema que afecta tanto al destinatario de los correos electrónicos como a las comunicaciones a través de Internet: el envío masivo de correo no deseado o Spam. Esta práctica fraudulenta proporciona a determinados publicistas un medio de llegar a miles de posibles clientes con un coste muy reducido.

Estudios recientes ponen en evidencia la importancia de los sistemas de detección y filtrado de dichos correos. Robert Horton, de la Australian Communication Commission, señala

que el costo generado por toda esta basura electrónica a individuos y empresas se calcula en unos 10.000 millones de dólares anuales sólo en Europa y podría alcanzar en todo el mundo alrededor de los 25.000 millones de dólares.

Los términos asociados habitualmente en Internet a estos tipos de abuso son *spamming*, *mail bombing*, *unsolicited bulk email* (UBE), *unsolicited commercial email* (UCE) o *junk mail*.

Para solventar este problema se plantean diversas alternativas, unas preventivas, otras disuasorias, y, como medida final, alternativas denunciadoras.

Las alternativas preventivas van orientadas a evitar, en la medida de lo posible, la recepción de Spam.

Las alternativas denunciadoras pretenden identificar al agresor informático y denunciarlo ante la autoridad competente.

Por último, las alternativas disuasorias, más interesantes desde el punto de vista de la investigación, están enfocadas a la detección y rechazo del mayor número de mensajes utilizando diversas técnicas. Entre esas técnicas caben destacar: detección de palabras clave en el campo asunto del correo electrónico, limitación del tamaño del correo, selección de remitentes autorizados y, sobre todo, técnicas de filtrado o clasificación de correos.

El filtrado de Spam utilizando técnicas de categorización de texto permite introducir técnicas de Machine Learning [6] que, tras el adecuado entrenamiento, arrojan unos resultados muy interesantes.

Estas técnicas exigen la construcción de un sistema parcialmente automático de aprendizaje a partir de datos de entrenamiento (ejemplos etiquetados manualmente por un experto humano). Entre las colecciones de correos electrónicos disponibles en Internet destacan SPAMBASE (UCI Machine Learning Repository) [10], LingSpam [9] o PUI [1].

En este trabajo se estudia el comportamiento del algoritmo BBR [4] en la tarea de filtrado de Spam sobre la colección experimental SPAMBASE. Con el fin de comparar los resultados obtenidos, se han aplicado otros dos algoritmos de aprendizaje sobre dicha colección, concretamente el algoritmo SVM y el algoritmo PLAUM.

El resto del trabajo está organizado de la siguiente forma: en la sección 2 se presentan los algoritmos utilizados en esta investigación. La sección 3 describe la colección SPAMBASE, su contenido, organización, y los resultados obtenidos. La sección 4 muestra las conclusiones y los trabajos futuros.

## 2 Algoritmos de aprendizaje automático

Los sistemas comerciales disponibles actualmente utilizan algoritmos de aprendizaje bayesianos simples. Los sistemas experimentales suelen utilizar otros algoritmos más complejos basados tanto en aprendizaje bayesiano como en otros sistemas automáticos de clasificación. Trabajos recientes en el filtrado de spam, hacen uso del algoritmo de aprendizaje SVM con unos resultados satisfactorios.

Nuestra investigación se ha centrado principalmente en la utilización del algoritmo

de aprendizaje basado en regresión logística bayesiana BBR. Por otra parte, a modo de comparativa, se ha aplicado el algoritmo SVM y el algoritmo PLAUM. A continuación, se describen brevemente dichos algoritmos.

### 2.1 BBR (Bayesian Binary Regression)

Se trata de una implementación de la regresión logística bayesiana, aplicada a la clasificación binaria. La clave de este algoritmo es la utilización de una distribución de probabilidad previa (ver ecuación 1) y algoritmos de optimización sucesiva de los ejemplos de entrenamiento suministrados.

$$p(y = 1 | \beta, x) = \psi(\beta^T x)$$

Ecuación 1

Este algoritmo inicialmente realiza una regresión logística de los datos de entrenamiento a partir de la distribución de probabilidad elegida (Gausiana o Laplace), por medio de una función de enlace  $\psi$  (ver ecuación 2).

$$\psi(z) = \exp(z) / (1 + \exp(z))$$

Ecuación 2

Una vez obtenido el modelo de regresión, se va optimizando sucesivamente a través de la aplicación de un algoritmo de regresión logística en cadena. Se trata de un algoritmo de optimización de coordenada cíclica descendente. Se comienza poniendo todas las variables a algún valor inicial, y se busca qué valor de la primera variable minimiza la función objetivo, asumiendo que todas las otras variables mantienen constantes sus valores iniciales. Este es un problema de optimización unidimensional. El mismo método se lleva a cabo con la segunda variable, y así sucesivamente hasta que se han cruzado todas las variables. Este proceso se repite varias pasadas hasta encontrar un criterio de convergencia.

De forma resumida, el algoritmo funciona como sigue [4]:

1. Inicializar las variables  $\beta_j=0$  (vector de parámetros),  $\Delta_j=0$  (conjunto de datos, o región explorada), para  $j=1$  hasta  $d$  (número de parámetros del modelo);  $\tau_i=0$  (estimador previo de confianza), para  $i=1$  hasta  $n$  (número de ejemplos).

2. Desde  $k=1, 2, \dots$  hasta que se produzca la convergencia, hacer

2.1. Para  $j=1$  hasta  $d$ , hacer:

2.1.1. Calcular

$$\Delta v_j = - \frac{\left( \sum_{i=1}^n x_{ij} y_i \frac{1}{1 + \exp(\tau_i)} \right) + 2\beta_j / \tau_j^2}{\left( \sum_{i=1}^n x_{ij}^2 F(\tau_j; \Delta_j | x_{ij}) \right) + 2 / \tau_j}$$

2.1.2. Calcular

$\Delta \beta_j = \min(\max(\Delta v_j - \Delta_j), \Delta_j)$ , en los datos spam del subconjunto tratado.

2.1.3. Calcular

$$\Delta \tau_i = \Delta \beta_j x_{ij} y_i, \quad \text{con} \\ \tau_i = \tau_i + \Delta \tau_i, \quad \text{para } i=1, \dots, n.$$

2.1.4. Calcular  $\beta_j = \beta_j + \Delta \beta_j$

2.1.5. Calcular

$$\Delta_i = \max(2|\Delta \beta_j|, \Delta_i / 2),$$

ampliando el tamaño del subconjunto de spam tratado.

entrenamiento y maximizar el margen.

2. Seleccionar la función de kernel y cualquier parámetro del mismo.

3. Solucionar el problema cuadrático dual resultante de la ecuación 3 o una formulación alternativa que use la programación cuadrática de forma apropiada o un algoritmo de programación lineal.

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m y_i y_j \alpha_i \alpha_j K(x_i, x_j) - \sum_{i=1}^m \alpha_i$$

s.t.  $\sum_{i=1}^l y_i \alpha_i = 0$

$C \geq \alpha_i \geq 0 \quad i=1, \dots, m$

Ecuación 3

4. Extraer la variable de umbral  $b$ , utilizando los vectores de apoyo.

5. Clasificar un punto nuevo  $x$ , siguiendo la ecuación 4:

$$f(x) = \text{sign}\left(\sum_i y_i \alpha_i K(x, x_i) - b\right)$$

Ecuación 4

## 2.2 SVM

El algoritmo SVM (Support Vector Machine) fue utilizado por primera vez en la Clasificación de Texto en 1998 por T. Joachims [7]. En términos geométricos, se puede ver como el intento de encontrar un espacio  $n$ -dimensional, que permita separar los ejemplos positivos de entrenamiento de los negativos, permitiendo especificar el margen más amplio posible.

El objetivo perseguido por este algoritmo es encontrar el hiperplano óptimo que maximice la distancia entre los casos positivos y los casos negativos.

Como argumenta Joachims [7], las máquinas de vectores de soporte ofrecen dos grandes ventajas para la categorización de texto:

- Evita los problemas de sobrecarga de pruebas en espacios de grandes dimensiones.
- Realiza una optimización global, sin óptimos locales.

Estos son los pasos principales del algoritmo SVM [2]:

1. Seleccionar el parámetro  $C$  como representante de la compensación entre la reducción al mínimo del error de clasificación del conjunto de

donde,  $x_i$  ( $i=1, \dots, m$ ) son los miembros del conjunto de puntos de entrenamiento;  $y_i = \pm 1$  son las etiquetas de la clase; y  $\alpha_i$  son los vectores de soporte.

En la figura 1 se pueden apreciar ejemplos de entrenamiento positivos y negativos. Las líneas representan los hiperplanos de separación. La línea más gruesa es la que minimiza la distancia para cualquier ejemplo de entrenamiento.

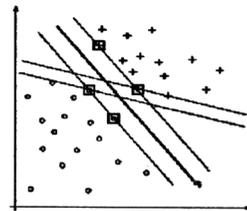


Figura 1: Representación de puntos en SVM

## 2.3 PLAUM (Perceptron Algorithm with Uneven Margins)

Se trata de algoritmo rápido y eficaz para realizar clasificaciones lineales. El algoritmo PLAUM es una extensión del algoritmo del

Perceptron, adaptada para tratar problemas de separación lineal de datos a través de un hiperplano [14]. Tal como SVM se basa en la idea de encontrar un margen entre hiperplanos, y sus autores aseguran que funciona mejor que SVM para tareas de clasificación de texto. Este algoritmo requiere:

- Un conjunto de entrenamiento linealmente separable de la forma:  

$$z = (x, y) \in (X \times \{-1, +1\})^m$$
- Un índice de aprendizaje  $\eta \in \mathcal{R}^+$ .
- Un número máximo de iteraciones T.
- Dos parámetros que limitan los ejemplos negativos y positivos:  
 $\tau_{-1}, \tau_{+1} \in \mathcal{R}^+$ .

El algoritmo funciona de acuerdo a los siguientes pasos [11]:

1. Inicialización de variables:

iteración = 0; i = 1; paso = m;  $\bar{w} = \vec{0}$ ; b = 0;

$$R = \max_{z_i \in X} \|\bar{x}_i\|$$

2. Repetir.

2.1. Si  $y_i((\bar{w}, \bar{x}_i) + b) \leq \tau_i$  entonces

2.1.1.  $\bar{w} = \bar{w} + \eta y_i \bar{x}_i$ .

2.1.2.  $b = b + \eta y_i R^2$ .

2.1.3. *paso* = i.

2.2.  $i = i + 1$ .

2.3. Si ( $i > m$ ) entonces

2.3.1.  $i = 1$ .

2.3.2. *iteración* = *iteración* + 1.

Hasta ( $i = \textit{paso}$ ) o ( $\textit{iteración} \geq T$ ).

3. Devolver (w,b).

### 3 Experimentos

#### 3.1 Metodología de Experimentación

Los experimentos se han realizado utilizando como colección de entrenamiento y prueba de correos electrónicos SPAMBASE<sup>1</sup>. Esta colección fue creada en Julio de 1999 por Mark Hopkins, Erik Reeber, George Forman, y Jaap Suermondt, a partir de una recopilación de mensajes donada por George Forman. Está

<sup>1</sup>Disponible en el ftp:

<ftp://ftp.ics.uci.edu/pub/machine-learning-databases/spambase>

compuesta por 4601 casos, de los cuales 1813 (39.4%) son clasificados como "Spam".

Cada ejemplo está representado por un vector de 58 atributos, donde el último contiene la clase a la que pertenece el documento ("Spam" o "NO Spam"). El resto de atributos representan la aparición de determinados símbolos o palabras.

La colección SPAMBASE no es linealmente separable, y la selección de atributos claves, permite la separación, casi perfecta de las dos clases.

La tabla 1 muestra los tiempos medios expresados en segundos, empleados en el entrenamiento con las particiones tratadas. Los resultados de prueba son inmediatos una vez entrenado el modelo.

	ENTRENAMIENTO
BBR	5,1 seg.
SVM	13396,37 seg.
PLAUM	10,61 seg.

Tabla 1: Tiempos de entrenamiento sobre la colección SPAMBASE

Cabe destacar que el algoritmo BBR es el más rápido de los utilizados para entrenamiento, con un margen muy amplio respecto del algoritmo SVM.

#### 3.2 Resultados obtenidos

Como método de evaluación de los experimentos realizados hemos utilizado Validación Cruzada o "10-fold Cross Validation" [13], consistente en hacer diez particiones iguales de la colección y utilizar de forma alternativa nueve partes para entrenamiento y la parte restante para prueba. Este proceso se repite 10 veces variando la partición utilizada en la evaluación. El resultado final obtenido es una media de los diez resultados parciales.

Para cada una de las ejecuciones, hemos obtenido una tabla de contingencia, del estilo mostrado en la tabla 2, donde:

- A es el número de documentos Spam clasificados como Spam;
- B es el número de documentos No Spam, clasificados como Spam;
- C es el número de documentos Spam, clasificados como No Spam;
- D es el número de documentos No Spam, clasificados como No Spam.

Partición $P_i$	SPAM	NO SPAM
Asigna SPAM	A	B
Asigna NO SPAM	C	D

Tabla 2: Tabla de contingencia.

A partir de estos datos se obtienen las métricas de precisión y recall y F medida, expresadas en las ecuaciones 5, 6 y 7 [12]:

$$P = \text{Precisión} = \frac{A}{A+B}$$

Ecuación 5: Precisión

$$R = \text{Recall} = \frac{A}{A+C}$$

Ecuación 6: Recall

$$F1 = \frac{2 * P * R}{P + R}$$

Ecuación 7: Métrica F1

La tabla 3 muestra los resultados obtenidos en términos de precisión y recall. Se ha incluido también la métrica F1 (ver ecuación 7) con el fin de tener una valoración global del comportamiento de los algoritmos

	Precisión	Recall	F1
BBR	87,10%	88,93%	87,99%
SVM	80,52%	40,68%	54,05%
PLAUM	58,10%	67,52%	62,46%

Tabla 3: Resultados sobre la colección SPAMBASE

Como se puede observar el algoritmo BBR supera ampliamente a los otros dos algoritmos presentados. La mejora de precisión obtenida con el algoritmo BBR sobre los algoritmos SVM y PLAUM es de un 7,55% y 33,30%, respectivamente. En cuanto al recall, la mejora es aún mayor (54,26% sobre SVM y 24,08% sobre PLAUM).

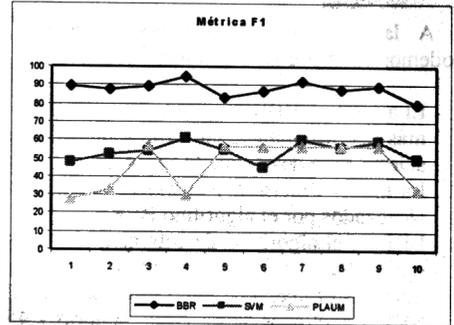


Figura 2. Comportamiento global de los algoritmos utilizando la métrica F1.

Desde un punto de vista global (es decir, utilizando la métrica F1), el algoritmo que presenta el mejor comportamiento es con diferencia el algoritmo BBR. La figura 2 muestra los valores obtenidos en cada uno de los 3 algoritmos para la métrica F1 en cada una de las 10 particiones utilizadas durante la evaluación. Se puede observar como en todos los casos (en cada una de las 10 particiones) la regresión logística bayesiana supera en más de 20 puntos a los otros dos algoritmos. Concretamente, promediando los resultados, el algoritmo BBR mejora el algoritmo SMV en un 38,57% y el algoritmo PLAUM en un 29,02%.

Por último, cabe destacar además que el algoritmo BBR es el más rápido de los algoritmos presentados tanto en el entrenamiento como en la evaluación.

#### 4 Conclusiones y trabajo futuro

La tarea de filtrado de correo electrónico no deseado o Spam, aplicando métodos de categorización de texto y usando métodos de aprendizaje automático es una tarea muy interesante e importante actualmente, tanto desde el punto de vista de la investigación como desde una perspectiva comercial o empresarial.

Una cuestión importante a la hora de elegir el algoritmo más interesante para tratar el filtrado de Spam es determinar la estrategia más adecuada: filtrar el mayor número posible de correos válidos, o relajar las condiciones para dejar pasar más correos y evitar el rechazo de correos correctos.

También es importante el tiempo en el que se procesa y filtra toda la información y el sistema toma la decisión de clasificarlo como Spam o no.

A la vista de los resultados obtenidos, podemos adelantar las siguientes conclusiones:

1. El algoritmo BBR consigue unos resultados más que aceptables, en torno a un 90% de acierto en la detección de Spam.
2. La mayor Precisión y el mayor Recall son alcanzados por el algoritmo BBR.
3. BBR es el algoritmo de entrenamiento más rápido (en el peor de los casos, 102 seg.), mientras el más lento, con diferencia, es el SVM (en el peor de los casos, 2 horas y 45 minutos).

Podemos concluir que el método de Regresión Logística Bayesiana es muy adecuado para esta tarea de filtrado de Spam, ya que se obtienen unos resultados muy satisfactorios y en unos tiempos realmente interesantes.

De cara a futuros trabajos, planteamos la posibilidad de tratar estos algoritmos con otras colecciones disponibles, y la incorporación de otras técnicas de clasificación de texto, estableciendo un sistema de voto para determinar si un correo es o no spam, en función de la fiabilidad de cada uno de los algoritmos utilizados.

### **Agradecimientos**

Este trabajo ha sido financiado con el proyecto (MCYT) TIC-2003-07158-C04-04.

### **Bibliografía**

- [1]. I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, and C.D. Spyropoulos, "An Experimental Comparison of Naive Bayesian and Keyword-Based Anti-Spam Filtering with Personal E-mail Messages". In Belkin, N.J., Ingwersen, P. and Leong, M.-K. (Eds.), Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2000), Athens, Greece, pp. 160-167, 2000.
- [2]. Kristin P. Bennet, Colin Campbell. Support Vector Machines: Hype or Hallelujah?. SIGKDD Explorations. ACM SIGKDD 2000.
- [3]. Pai-Hsuen Chen, Chih-Jen Lin, and Bernard Schölkopf. A Tutorial on  $v$ -Support Vector Machines. 2001.
- [4]. Alexander Genkin, David D. Lewis, and David Madigan. Large-Scale Bayesian Logistic Regression for Text Categorization. Dimacs. 2004.
- [5]. José María Gómez Hidalgo. Evaluating cost-sensitive unsolicited bulk email categorization. Proceedings of the ACM Symposium on Applied Computing. 2002.
- [6]. José María Gómez Hidalgo, Enrique Puertas Sanz, Francisco Carrero García, Manuel de Buenaga Rodríguez. Categorización sensible al coste para el filtrado de contenidos inapropiados en Internet. Universidad Europea de Madrid. 2003.
- [7]. T. Joachims, Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.
- [8]. Bryan Klimt, Yiming Yang. The Enron Corpus: A New Dataset for Email Classification Research. Proceedings of the 2004 European Conference on Machine Learning (ECML).
- [9]. I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, George Paliouras, and C.D. Spyropoulos, "An Evaluation of Naive Bayesian Anti-Spam Filtering". In Potamias, G., Moustakis, V. and van Someren, M. (Eds.), Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning (ECML 2000), Barcelona, Spain, pp. 9-17, 2000.
- [10]. SPAMBASE, creada por Mark Hopkins, Erik Reeber, George Forman, y Jaap Suermondt Hewlett-Packard Labs, 1501 Page Mill Rd., Palo Alto, CA 94304. Donada por Donor: George Forman (gforman at nospam hpl.hp.com). Generado: Junio-Julio 1999.
- [11]. S. E. Robertson, S. Walker, H. Zaragoza, R. Herbrich. Filtering track. Microsoft Cambridge at TREC 2002.
- [12]. G. Salton, M.J. McGill. Introduction to modern information retrieval. McGraw-Hill 1983.
- [13]. Stone, M.: Cross-validators choice and assessment of statistical predictions (with discussion). Journal of the Royal Statistical Society B 36 1974.
- [14]. Miguel Ángel García Cumberas, L. Alfonso Ureña López, Fernando Martínez

Santiago y Arturo Montejo Ruez: Búsqueda de Respuestas Multilingüe: Clasificación de preguntas en español basada en aprendizaje. Revista SEPLN n° 34. 2005.