Método híbrido para categorización de texto basado en aprendizaje y reglas*

Hybrid approach for text categorization based on machine learning and rules

Julio Villena-Román
Universidad Carlos III de Madrid
Av. de la Universidad 30 – E-28911 Leganés
jvillena@it.uc3m.es

Sonia Collada-Pérez
DAEDALUS, S.A.
Av. de la Albufera 321 – 28031 E-Madrid
scollada@daedalus.es

Sara Lana-Serrano
Universidad Politécnica de Madrid
E.U.I.T. Telecomunicación
Crta Valencia km 7 – E-28031 Madrid
slana@diatel.upm.es

José Carlos González-Cristóbal
Universidad Politécnica de Madrid
E.T.S.I. Telecomunicación
Ciudad Universitaria s/n – 28040 E-Madrid
josecarlos.gonzalez@upm.es

Resumen: En este artículo se presenta un nuevo método híbrido de categorización automática de texto, que combina un algoritmo de aprendizaje computacional, que permite construir un modelo base de clasificación sin mucho esfuerzo a partir de un corpus etiquetado, con un sistema basado en reglas en cascada que se emplea para filtrar y reordenar los resultados de dicho modelo base. El modelo puede afinarse añadiendo reglas específicas para aquellas categorías difíciles que no se han entrenado de forma satisfactoria. Se describe una implementación realizada mediante el algoritmo kNN y un lenguaje básico de reglas basado en listas de términos que aparecen en el texto a clasificar. El sistema se ha evaluado en diferentes escenarios incluyendo el corpus de noticias Reuters-21578 para comparación con otros enfoques, y los modelos IPTC y EUROVOC. Los resultados demuestran que el sistema obtiene una precisión y cobertura comparables con las de los mejores métodos del estado del arte.
Palabras clave: Clasificación automática de texto, aprendizaje computacional, sistema basado en reglas, kNN, Reuters-21578, IPTC, EUROVOC, evaluación.

Abstract: This paper discusses a novel method for text categorization that combines a machine learning algorithm able to build a base model with low effort by using a labeled available corpus, along with a rule-based expert system in cascade used to filter and rerank the output of the previous classifier. The model can be fine-tuned by adding specific rules for those difficult classes that have not been successfully trained. We describe an implementation based on kNN algorithm and a basic rule language that expresses lists of terms appearing in the text. The system is trained and evaluated in different scenarios, including the popular Reuters-21578 news corpus for comparison to other approaches, and the IPTC and EUROVOC models. Results show that this approach achieves a precision that is comparable to other top state-of-the-art methods.
Keywords: Text categorization, machine learning, rule-based system, kNN, Reuters-21578, IPTC, EUROVOC, evaluation.

1 Introducción

La cantidad de información disponible en Internet en cualquier área de conocimiento ha aumentado de forma exponencial en los últimos años. Una de las claves del éxito en esta sociedad del conocimiento es la capacidad para presentar los contenidos de forma atractiva, bien ordenados, con posibilidad de búsquedas, e incluyendo cualquier valor añadido como vínculos a contenidos relacionados, información sobre entidades o eventos involucrados, opinión en blogs o redes sociales, etc. Así, los sistemas de recuperación y clasificación au-

* Esta investigación ha sido parcialmente financiada por los proyectos de I+D BUSCAMEDIA (CEN-20091026), MULTIMEDICA (TIN2010-20644-C03-01) y BRAVO (TIN2007-67407-C03-01).

tomática de información surgen para almacenar, procesar, filtrar y organizar ese enorme volumen de datos, a fin de convertirlo en información útil y, potencialmente, en conocimiento.

Este artículo propone un método híbrido para la clasificación automática de textos, combinando un algoritmo de clasificación basado en aprendizaje automático con un sistema basado en reglas, conectados en cascada. El objetivo es construir buenos modelos en precisión y cobertura (*recall*), con un esfuerzo reducido. En los siguientes apartados se describe en detalle sus fundamentos y arquitectura lógica y la implementación que se ha realizado. En el apartado de evaluación se describen diferentes escenarios en los que se ha aplicado el sistema con éxito.

2 Fundamentos

La categorización (o clasificación) automática de textos consiste en asignar automáticamente una o varias categorías (o clases) predefinidas a un determinado texto en lenguaje natural, según su similitud con respecto a otros textos etiquetados previamente, empleados como referencia.

Típicamente existen dos enfoques para la clasificación de textos (Sebastiani 2002). Por un lado está el enfoque *basado en conocimiento*, común en los años 80, que consiste en la creación de un sistema experto con reglas de clasificación definidas de forma manual, típicamente una por categoría, como una expresión lógica que combina términos del texto con los operadores booleanos AND, OR y NOT:

if (expresión lógica) **then** (categoría)

Se asume comúnmente que se pueden producir reglas tan precisas como sea necesario. Por ejemplo, uno de los sistemas más conocidos de este tipo, CONSTRUE (Hayes et al. 1990), desarrollado por el Grupo Carnegie para la agencia de noticias Reuters, alcanzaba un *breakeven* de 0,90. Sin embargo, un inconveniente es que requiere, por un lado, un conocimiento experto sobre el dominio de clasificación, y por otro, un conocimiento específico sobre el lenguaje de reglas, todo ello sin mencionar la dificultad intrínseca de modelar una categoría con una lista de operaciones lógicas sobre términos. En cualquier caso, el principal problema es que la construcción del conjunto de reglas cuando se trata con muchos cientos o incluso miles de categorías es una ardua tarea que hace que esta aproximación sea inabordable en la mayoría de los escenarios del mundo real.

Por otro lado, el enfoque de *aprendizaje automático* se ha convertido en el más popular desde los 90. En este caso, se proporciona al sistema un conjunto de textos pre-clasificados (etiquetados) para cada categoría, que se usa como conjunto de entrenamiento para construir un clasificador. La ventaja es que sólo se necesita un mínimo conocimiento del dominio para asignar una categoría a cada texto existente en el conjunto de entrenamiento, lo que implica una carga de trabajo mucho menor que la escritura de las reglas. Se han propuesto numerosos algoritmos y técnicas de aprendizaje supervisado para construir los clasificadores.

Aunque se ha demostrado que el enfoque de aprendizaje automático puede generar clasificadores igual de buenos que los sistemas basados en reglas, pero con un menor esfuerzo, también tienen sus inconvenientes, fundamentalmente relacionados con el hecho de que (en la mayoría de los algoritmos de aprendizaje empleados) el modelo no es comprensible por el ser humano, con lo que es difícil diagnosticar la razón de los falsos positivos/negativos para poder ajustar el sistema. En la práctica, la única manera de mejorar el clasificador es invertir un mayor esfuerzo en la construcción del conjunto de entrenamiento y evaluar distintas alternativas para construir los vectores de rasgos.

3 Enfoque Híbrido

Durante varios años investigamos diferentes estrategias estadísticas y/o semánticas de expansión automática de las consultas para mejorar la precisión y la cobertura en diferentes tareas de recuperación de información y clasificación automática (Villena-Román et al., 2009a). Sin embargo, al añadir términos extra a la información original en la etapa de preprocesamiento (como sinónimos o términos que coaparecen con un cierto valor de confianza), aunque mejorábamos la cobertura, fuimos incapaces de encontrar alguna estrategia para aumentar (o incluso mantener) los valores base de precisión. Por ello decidimos adoptar una estrategia diferente y actuar en la etapa de postprocesamiento de la salida del clasificador automático, llegando finalmente al método híbrido mostrado en la Figura 1.

Hay otros trabajos que proponen métodos híbridos para diferentes tareas de clasificación, pero fundamentalmente todos combinan dos o más clasificadores de aprendizaje automático como (Kim y Myoung 2003).

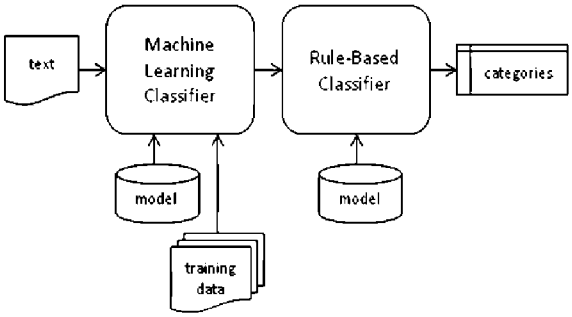

Figura 1: Arquitectura lógica del sistema

3.1 Clasificador basado en Aprendizaje Automático

El primer paso es entrenar un modelo base desde cero usando un corpus de entrenamiento disponible. No hay requisitos: es válido cualquier algoritmo de aprendizaje que pueda dar respuesta a los requisitos de cada escenario en concreto. Sin embargo, en general, el clasificador utilizado debería ser capaz de proporcionar una clasificación multietiqueta manejando un elevado número de categorías, probablemente no balanceadas (algunas categorías con muy pocos textos y otras con demasiados). Además, desde una perspectiva pragmática (comercial), pueden ser interesantes otros requisitos no funcionales, como una clasificación en tiempo real (por ejemplo, tiempo de respuesta menor de 50 ms) o la edición en tiempo real del modelo.

En nuestro caso, elegimos el algoritmo de los k vecinos más cercanos (*k-Nearest Neighbour* o kNN) basado en la distancia euclídea, por su simplicidad y buen rendimiento demostrado en otros trabajos (Joachims, 1998). Nuestra implementación se basa en Apache Lucene[1], un motor de recuperación de información de código abierto y altas prestaciones.

Independientemente del algoritmo de aprendizaje, todos los clasificadores se basan en el hecho de que cuantas más veces aparezca un término en el texto, más relevante es ese término. Según el Modelo de Espacio de Vectores (Salton, et al., 1975), cada texto del corpus se representa como un vector multidimensional $(w_{i1}\, w_{i2}\, \ldots w_{iN})$, donde cada valor w_{ij} del vector representa el grado en el que el termino está presente (o ausente) en el texto.

En nuestro sistema, utilizamos TF*IDF[2] para construir los vectores, y reducimos la dimensionalidad por selección de características escogiendo mediante Bag-Of-Words los N términos con más peso (200 términos).

3.2 Clasificador basado en Reglas

A continuación, se conecta en cascada un sistema basado en reglas para post-procesar la salida del clasificador anterior. Este sistema experto utiliza reglas basadas en expresiones lógicas de términos en lenguaje natural, tan complejas como sea necesario. En general, cada categoría puede tener ninguna, una o varias reglas asociadas. Cada regla se evalúa sobre el texto de entrada q para aceptar (validar), rechazar (invalidar) o proponer (incluir) dicha categoría en la lista de resultados, según si el texto satisface o no las condiciones expresadas en la regla. El rechazo de una categoría elimina los falsos positivos devueltos por el clasificador basado en aprendizaje automático, por lo que mejora la precisión. La inclusión de una nueva categoría resuelve los falsos negativos, con lo que mejora la cobertura.

Además, las reglas se utilizan para reordenar la lista de resultados: las reglas incrementan (refuerzan) la relevancia de una categoría dada, por ejemplo, dependiendo del número de términos que satisfacen la expresión lógica. Esto también logra mejorar la precisión.

En nuestra implementación, diseñamos un lenguaje de reglas básico que simplifica la creación de reglas. Para cada i-ésima categoría, la regla tiene cuatro componentes:

- Términos *positivos* $P_i = \{p_{i1},\ p_{i2} \ldots p_{ip}\}$: al menos uno de estos p términos debe aparecer obligatoriamente en el texto, es decir:
  if $(p_{i1}\ OR\ p_{i2}\ OR \ldots OR\ p_{ip})$ **then**
  (categoría aceptada) (1)
  else
  (categoría rechazada)
- Términos *negativos* $N_i = \{n_{i1},\ n_{i2} \ldots n_{in}\}$: ninguno de estos n términos debe aparecer en el texto, es decir:
  if $(n_{i1}\ OR\ n_{i2}\ OR \ldots OR\ n_{in})$ **then**
  (categoría rechazada) (2)
  else
  (categoría aceptada)
- Términos *relevantes* $R_i = \{r_{i1},\ r_{i2} \ldots r_{ir}\}$: se usan para incrementar la relevancia de la categoría, como se describe más adelante.

[1] http://lucene.apache.org

[2] Frecuencia de Término (nº de veces que el término aparece en el texto) y Frecuencia Inversa de Documento (logaritmo del nº total de documentos dividido por el nº que contienen a dicho término).

- Términos *irrelevantes* $I_i = \{i_{i1}, i_{i2} \dots i_{ir}\}$: similar al caso anterior, pero reduciéndola.

El factor de *boosting* de una determinada categoría se muestra en la Ecuación 3. Los términos negativos se utilizan para rechazar la categoría, así que su relevancia final es cero.

El algoritmo de aprendizaje proporciona una lista de categorías ($c_i \in C$), ordenadas según su *categorization status value* (CSV)[3] con respecto al texto de entrada *q*:

$$D_q = \begin{pmatrix} csv_{q,1} \\ \dots \\ csv_{q,K} \end{pmatrix} \quad (3)$$

El resultado de este segundo bloque, y por tanto, del sistema global, es una lista de categorías reordenada con su nuevo CSV:

$$D'_q = D_q * B_q = \begin{pmatrix} csv'_{q,1} \\ \dots \\ csv'_{q,K} \end{pmatrix} \quad (4)$$

$$B'_{q,i} = \begin{cases} 0 & si \exists t_i \in N \\ 1 + cuenta(t_i \in P) + cuenta(t_i \in R) & \text{si no} \\ \quad - cuenta(t_i \in I) & \end{cases}$$

$$B_{q,i} = \begin{cases} B'_{q,i} & if\ B'_{q,i} \geq 0 \\ 1/B'_{q,i} & \text{si no} \end{cases} \quad (5)$$

Los términos en las reglas pueden ser palabras individuales (por ejemplo *gasolina*) o unidades multipalabra (como *producto interior bruto*). En este caso, la condición booleana es *true* cuando todas las palabras están en el texto:

$$t_i \equiv t_{i1}\ AND\ t_{i2}\ AND \dots AND\ t_{ik} \quad (6)$$

Por último, se definen dos reglas adicionales para el caso en que no se haya definido ninguna regla para una categoría dada. La regla ACCEPT valida la categoría, sin importar los términos del texto ($B_{q,i} = 1$), y es la regla por defecto del sistema. La regla REJECT invalida la categoría ($B_{q,i} = 0$).

4 *Evaluación*

Este método ha sido evaluado en diferentes escenarios. El primero de ellos como comparación con otros algoritmos, utilizando el famoso corpus de noticias de prensa en inglés Reuters-21578. El segundo escenario trata también sobre clasificación de noticias pero empleando el modelo IPTC. Otros escenarios incluyen el tesauro EUROVOC y la clasificación de texto médico y transcripciones de vídeo.

4.1 Modelo Reuters-21578

El objetivo de este modelo es establecer un escenario base de validación y comparación del sistema con otros métodos, para probar que el método propuesto consigue mejores resultados que otros algoritmos, o, al menos los mismos, pero con un menor esfuerzo de desarrollo.

La colección Reuters-21578 (Sebastiani 2002) es un conjunto de 21.578 artículos de noticias en inglés publicadas por la agencia Reuters en 1987, ampliamente adoptado en las últimas décadas por la comunidad de investigadores como un marco común de referencia para la evaluación de tareas de clasificación de textos. Aunque la colección contiene 115 categorías (el denominado corpus R115), muchos investigadores se han centrado en el corpus R90, que contiene las 90 categorías con al menos un ejemplo de entrenamiento y otro de test.

Método	**Propuesto en**	**BEP**
Bayes	(Joachims 1998)	,720
Bayes	(Li y Yamanishi 1999)	,773
C4.5	(Joachims 1998)	,794
RIPPER/reglas	(Cohen y Singer 1999)	,820
DL-ESC/reglas	(Li y Yamanishi 1999)	,820
Rregresión	(Yang y Liu 1999)	,849
Widrow-Hoff	(Lam y Ho 1998)	,822
Rocchio	(Cohen y Singer 1999)	,776
Rocchio	(Joachims 1998)	,799
Red neuronal	(Yang y Liu 1999)	,838
GisW	(Lam y Ho 1998)	,860
kNN	(Joachims 1998)	,823
kNN	(Yang y Liu 1999)	,856
SVMLight	(Joachims 1998)	,864
SVM	(Dumais et al. 1998)	,870
AdaBoost	(Weiss et at. 1999)	,878
Red bayesiana	(Dumais et al. 1998)	,800
Alg. genéticos	(Hirsch y Saeedi 2007)	,800
Media		**,824**

Tabla 1: Resultados de otros sistemas

La Tabla 1 muestra los resultados obtenidos por diferentes enfoques, tomados de (Sebastiani 2002) y (Debole y Sebastiani 2004), empleando el corpus R90. Aunque hoy ha quedado algo en desuso, en la tabla se muestra el valor del punto de equilibrio micro-promediado (*Microaveraged Breakeven Point*, el punto donde la precisión y la cobertura se igualan).

La Tabla 2 muestra los resultados de nuestro sistema en sucesivas iteraciones. El experimen-

[3] Cuanto mayor es el CSV de una categoría para un documento, más pertenece ese documento a dicha categoría (Sebastiani 2002).

to I emplea únicamente el modelo kNN sin ninguna regla (es decir, todas las categorías usan la regla por defecto ACCEPT), como experimento base. Como se observa en la tabla, su rendimiento es similar y consistente con los valores mostrados en la Tabla 1. El experimento II incluye un conjunto de reglas básicas para las 10 categorías principales (aquellas con un mayor número de noticias) y el experimento III utiliza reglas básicas para todas las 90 categorías. Finalmente, el experimento IV hace uso de reglas avanzadas, expresamente para cada categoría, descritas más adelante.

Exp.	Descripción	BEP
I	Sólo kNN (sin reglas)	,817
II	Reglas básicas para 10 categorías	,846 (+3,5%)
III	Reglas básicas para todas las categorías	,858 (+5,0%)
IV	Reglas avanzada para todas las categorías	,877 (+7,3%)

Tabla 2: Resultados del sistema propuesto

La mejora conseguida mediante el *boosting* de las reglas puede observarse claramente en la tabla anterior. El resultado final supera a todos los métodos listados en la Tabla 1, excepto AdaBoost, con la ventaja clara de una menor complejidad y esfuerzo de implementación.

La Tabla 3 muestra el conjunto de reglas escritas para las 10 categorías principales en los experimentos II y III. Como se ve, estas reglas son realmente sencillas de escribir, ya que sólo incluyen términos relevantes que contribuyen a reforzar el modelo de clasificación base de cada categoría. Estos términos son los títulos de las categorías más una selección trivial de forma manual entre los términos más frecuentes del conjunto de noticias de entrenamiento de cada una de las categorías. No se emplean términos positivos porque la mayoría de esas 10 categorías tratan sobre conceptos generales como adquisiciones de empresas o comercio, que no se representan de forma clara con una lista cerrada de términos específicos.

En el experimento IV se reescriben las reglas para aquellas categorías que pueden expresarse por medio de palabras específicas, utilizando términos positivos que fuerzan la presencia de dichas palabras, por ejemplo: *wheat*, *corn OR maize*, *aluminum OR aluminium*, *sorghum*, *bop OR balance_of_payments*, etc.

Categoría	Términos relevantes
acq (*fusiones*)	mergers|merge|acquisition| acquisition|share|shares|company| companies
corn (*maíz*)	corn|maize
crude (*crudo*)	crude|oil|barrel|barrels|petroleum
earn (*beneficios*)	earnings|dividend|dividends| benefit|benefits|loss|losses|growth| income|incomes|net|company| companies|deficit|deficits|debt| debts|reduce|increase
grain (*cereales*)	grain|grains|crop
interest (*intereses*)	interest|interests|rate|rates|prime| discount
money-fx (*tipos de cambio*)	money_exchange|exchange| exchanges|change|changes| money|value|monetary|currency| currencies|money_market
ship (*transporte marítimo*)	shipping|shippings|ship|waterway
trade (*comercio*)	trade|commerce|deficit|import| imports|export|exports|trade_deficit
wheat (*trigo*)	wheat

Tabla 3: Reglas para las 10 categorías principales en los experimentos II y III

Categoría	BEP (Debole y Sebastiani 2004)	F1
acq	,853	,891
corn	,877	,911
crude	,755	,924
earn	,961	,969
grain	,891	,879
interest	,491	,790
money-fx	,694	,740
ship	,809	,875
trade	,592	,879
wheat	,855	,805

Tabla 4: Resultados de las 10 primeras categorías del experimento IV

La Tabla 4 muestra el valor de la medida-F1 para las 10 categorías principales alcanzada en el último experimento. Los resultados son consistentes con los obtenidos en otros experimentos (Debole y Sebastiani 2004) que obtienen un peor rendimiento para las categorías de *money* y *interest*, que parecen ser difíciles de modelar, y un mejor rendimiento para la categoría *earn*.

4.2 Modelo de clasificación IPTC

Este sistema ha sido entrenado también con la jerarquía de clasificación del *International Press Telecommunication Council*[4] (IPTC), para textos en castellano. IPTC es un consorcio internacional de los principales editores, agencias de noticias y empresas de comunicación, que desarrolla y mantiene estándares técnicos para el intercambio de noticias que son empleados por virtualmente todas las agencias de noticias importantes del mundo. Entre otros, IPTC ha propuesto conjuntos de metadatos, llamados *newscodes*, para estandarizar la codificación de los atributos de los objetos de publicación, aplicados a textos, fotografías, clips de audio o vídeo, etc., y que representan diferentes características como el género, tema, formato, escena, técnica, etc.

Utilizando el sistema, se ha construido un modelo para la clasificación automática de los *newscodes* descriptivos, que tienen tres niveles jerárquicos (tema, subtema y detalle - *Subject*, *Matter* y *Detail*), codificados con 8 dígitos de la forma SSMMMDDD. Se ha empleado la edición a febrero de 2010, con 1.349 categorías.

Para entrenar el clasificador kNN, utilizamos un corpus de 108.838 artículos en castellano, publicados por El País[5] durante 2006 y 2007, etiquetados manualmente con los códigos IPTC. En este corpus, de media, cada artículo pertenece a 2,4 categorías.

Las reglas se crearon mediante un proceso iterativo de desarrollo, validación y refinamiento, llevado a cabo por un equipo de lingüistas durante 3 semanas, ayudados por las evaluaciones periódicas de rendimiento realizadas por un equipo externo empleando noticias reales. La versión final del sistema incluye reglas para todas las categorías, con casi 8.000 términos, y tras una evaluación informal alcanza los resultados mostrados en la Tabla 5.

Algunas categorías son relativamente fáciles de modelar simplemente con una lista de términos positivos: por ejemplo, los artículos de la clase *15073046 (deportes – eventos deportivos – Super Bowl)* deben incluir obligatoriamente los términos *superbowl* o *super bowl*.

Las categorías más generales se modelan mejor con una lista de términos relevantes, puesto que es difícil listar un conjunto de términos obligatorios. Por ejemplo, un artículo que contenga términos como *fotografía, fotografías, foto, fotos, fotógrafo, fotógrafa, fotógrafos, fotógrafas* y *fotografiado*, será susceptible de ser clasificado en la categoría *01013000 (arte, cultura y espectáculos – fotografía)*.

Parámetro	Valor
Nº de artículos evaluados	756
Nº medio de categorías por artículo	5,16
Artículos con todas las categorías bien	75,4%
Artículos con todas las categorías mal	0%
Artículos con algunas categorías bien	100%
Artículos con algunas categorías mal	25%
Precisión media de categorías	0,948

Tabla 5: Resultados del clasificador IPTC

En otros muchos casos, ha sido necesario incluir términos negativos para poder establecer diferencias entre categorías similares. Por ejemplo, la regla para la categoría *15039000 (deportes – automovilismo)* incluye como términos relevantes *automovilismo, coche, circuito* y *equipo*, pero las noticias aquí clasificadas no deben incluir los términos *trucki, camiones, nascar* ni *fórmula 1*. Obviamente, las noticias de la categoría *15039007 (deportes – automovilismo – NASCAR)* deben a su vez incluir *nascar* y excluir *fórmula 1, trucki*, etc.

Actualmente este sistema de clasificación IPTC está siendo empleado por varias empresas de comunicación punteras en España, como núcleo de su sistema de producción y organización de contenidos[6], unas utilizando una configuración supervisada (los editores humanos seleccionan las categorías de anotación de la noticia a partir de la lista propuesta por el sistema) y otras no supervisada (el sistema anota las noticias de forma automática, sin intervención humana).

4.3 Otros modelos

Tesauro EUROVOC. Otro escenario en el que se ha aplicado el enfoque híbrido descrito es la clasificación de textos legales publicados en diarios o boletines oficiales de diferentes administraciones públicas, según el tesauro multilingüe EUROVOC que se emplea extensamente en la Unión Europea[7].

Por una parte, el alto número de categorías (actualmente 6.797 categorías) hace que la construcción y mantenimiento de un modelo de clasificación con suficiente precisión sea prácti-

[4] http://www.iptc.org/

[5] El País (http://www.elpais.com) es el periódico no deportivo de mayor difusión en España.

[6] http://www.daedalus.es/showroom/newscl/

[7] http://www.daedalus.es/showroom/eurovoc/

camente imposible con modelos exclusivamente de aprendizaje computacional. Por otra parte, la constante actualización del tesauro conlleva que la tarea sea ardua y costosa empleando modelos puramente basados en reglas. El enfoque planteado solventa ambos problemas y consigue un buen compromiso entre precisión y esfuerzo.

Específicamente se ha desarrollado un sistema (para uso comercial) que incorpora reglas para clasificación de textos en castellano y catalán. Inicialmente se emplearon los nombres y alias de los descriptores de las categorías como términos positivos (45.217 en total), y posteriormente se refinaron las 100 categorías más frecuentes (más del 80% del corpus de evaluación) con términos negativos (123 términos). De forma similar se podrían desarrollar reglas para los restantes 22 idiomas del tesauro.

Tras una revisión informal, el nivel de precisión alcanzado para ambos idiomas está sobre un 78%, considerando sólo el primer resultado devuelto por el sistema, y un 84% considerando los 5 primeros resultados. Estos valores de precisión son más que suficientes para un proceso supervisado como el escenario que se plantea.

Clasificación de texto médico. Una versión preliminar del método propuesto se aplicó en MIDAS (Medical Diagnosis Assistant) (Sotelsek-Margalef 2008), un sistema experto avanzado capaz de proporcionar un diagnóstico médico (automatizando la asignación de códigos ICD-9-CM) a partir de los registros radiológicos y clínicos del historial de los pacientes, basándose en extracción de información y aprendizaje automático sobre los historiales de pacientes anteriormente diagnosticados. Este sistema se diseñó e implementó expresamente para participar en la *Medical Natural Language Processing Challenge* en 2007, logrando un buen nivel de precisión tal y como se muestra en la Tabla 6 (los resultados para la medida F1 del mejor y el peor sistema propuestos fueron 0.8908 y 0.1541 respectivamente).

Algoritmo	F1
kNN (k=1)	,767
kNN (k=2)	,762

Tabla 6: Clasificación de textos médicos

Clasificación de vídeo. Otra aplicación de este mismo método (Villena-Román 2009) fue la clasificación temática de vídeos de programas de televisión bilingües (en inglés y holandés), utilizando las transcripciones obtenidas mediante reconocimiento automática del habla y, opcionalmente, metadatos del programa (título y descripción). La Tabla 7 muestra resultados prometedores, sobre todo considerando que se trata de una investigación en marcha.

Categoría	Precisión	Cobertura
Cine	,25	,33
Historia	,26	,50
Música	,65	1,00
Pintura	,00	,00
Artes visuales	,20	,40

Tabla 7: Clasificación de vídeos

5 *Conclusiones y Trabajos Futuros*

En este artículo se ha presentado un método híbrido para clasificación de texto que combina un algoritmo de aprendizaje, que proporciona un modelo de clasificación base relativamente poco costoso de entrenar, con un sistema experto basado en reglas, que post-procesa los resultados del primer clasificador, mejorando su precisión y cobertura filtrando los falsos positivos y resolviendo los falsos negativos. Además se ha descrito una implementación viable basada en kNN y en un lenguaje básico de reglas que permite expresar listas de términos positivos, negativos, relevantes e irrelevantes. Por último, se han presentado los resultados de la evaluación realizada en diferentes escenarios.

La conclusión principal que se puede obtener de la evaluación usando el corpus de noticias Reuters-21578 es que el método propuesto es capaz de obtener una precisión como mínimo similar a la de los métodos avanzados tradicionales, y en algunos casos superior, con el importante valor añadido de que este modelo se construye con un esfuerzo manual mucho más reducido. Si la salida del clasificador base es satisfactoria, no es necesario escribir ninguna regla para esa(s) categoría(s). Sin embargo, si alguna categoría resulta ser demasiado ruidosa y obtiene un valor excesivamente bajo de precisión o cobertura, el sistema propuesto se puede afinar cuanto sea necesario añadiendo reglas específicas para dicha(s) categoría(s).

Actualmente estamos trabajando en la aplicación de este sistema híbrido a otros escenarios como la moderación automática de foros, el análisis de opinión y la creación de marcadores sociales y bibliotecas digitales (Heymann 2010) en los que este método puede aportar grandes beneficios.

Bibliografía

Cohen, W. W. y Singer, Y. 1999. Context sensitive learning methods for text categorization. *ACM Transactions On Information Systems*. 17, 2, 141–173.

Debole, F., y Sebastiani, F. 2004. An Analysis of the Relative Hardness of Reuters-21578 Subsets. *Journal of the American Society for Information Science and Technology*, vol 56.

Dumais, S. T., Platt, J., Heckerman, D., y Sahami, M. 1998. Inductive learning algorithms and representations for text categorization. In *Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management* (Bethesda, MD, 1998), 148–155.

Hayes, P. J., Andersen, P. M., Nirenburg, I. B., y Schmandt, L. M. 1990. Tcs: a shell for content-based text categorization. In *Proceedings of CAIA-90, 6th* IEEE *Conference on Artificial Intelligence Applications* (Santa Barbara, CA, 1990), 320–326.

Heymann, P., Paepcke, A., Garcia-Molina, H. 2010. Tagging human knowledge. In 3rd ACM International Conference on Web Search and Data Mining (WSDM), 51–60.

Hirsch, L., Hirsch, R., y Saeedi, M. 2007. Evolving Lucene search queries for text classification. In *Proceedings of 9th annual conference on Genetic and Evolutionary Computation* (GECCO '07), pp 1604–1611.

Joachims, T. 1998. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of ECML-98, 10th European Conference on Machine Learning* (Chemnitz, Germany, 1998), 137–142.

Kim, I.C., y Myoung, S. 2003. Text Categorization Using Hybrid Multiple Model Schemes. *Advances in Intelligent Data Analysis V.* Lecture Notes in Computer Science, 2003, Volume 2811/2003, 88–99.

Lam, W. y Ho, C. Y. 1998. Using a generalized instance set for automatic text categorization. In *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval* (Melbourne, Australia, 1998), 81–89.

Li, H., y *Yamanishi*, K. 1999. Text classification using ESC-based stochastic decision lists. In *Proceedings of CIKM-99, 8th ACM International Conference on Information and Knowledge Management* (Kansas City, MO, 1999), 122–130.

Salton, G., Wong, A., y Yang, C.S. 1975. A vector space model for automatic indexing. *Communications of the ACM*, volume 18 num 11, pp 613–620.

Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1), pp 1–47.

Sotelsek-Margalef, A., y Villena-Román, Julio. MIDAS: An Information-Extraction Approach to Medical Text Classification. *XXIV Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural*, SEPLN08, Leganés, Spain, Septiembre 2008.

Villena-Román, J., Lana-Serrano, S., y González-Cristóbal, J.C. MIRACLE-GSI at ImageCLEFphoto 2008: Different Strategies for Automatic Topic Expansion. *Evaluating Systems for Multilingual and Multimodal Information Access.* 9th Workshop of the Cross-Language Evaluation Forum, CLEF 2008, Aarhus, Denmark, September 17-19, 2008, Revised Selected Papers. Carol Peters et al (Eds.). Lecture Notes in Computer Science, Vol. 5706, 2009.

Villena-Román, J., y Lana-Serrano, S. MIRACLE at VideoCLEF 2008: Topic Identification and Keyframe Extraction in Dual Language Videos. *Evaluating* Systems *for Multilingual and Multimodal Information Access.* 9th Workshop of the Cross-Language Evaluation Forum, CLEF 2008, Aarhus, Denmark, 2008, Revised Selected Papers. Carol Peters et al (Eds.). Lecture Notes in Computer Science, Vol. 5706, 2009.

Weiss, S. M., Apté, C., Damerau, F. J., Johnson, D. E., Oles, F. J., Goetz, T., y Hampp, T. 1999. Maximizing text-mining performance. *IEEE Intelligent Systems*. 14, 63–69.

Yang, Y. y Liu, X. 1999. A re-examination of text categorization methods. In *Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval* (Berkeley, CA, 1999), 42–49.