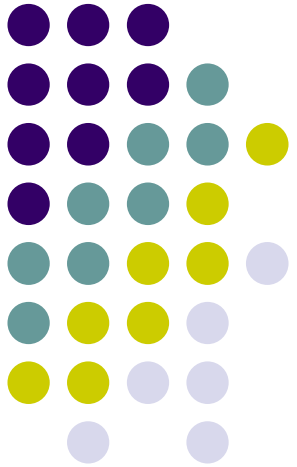
Clasificación de texto con adversario

Técnicas de clasificación y filtrado aplicadas a la detección de spam en la Web

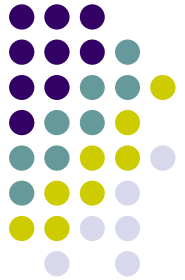


José María Gómez Hidalgo

Optenet I+D

http://www.esp.uem.es/jmgomez

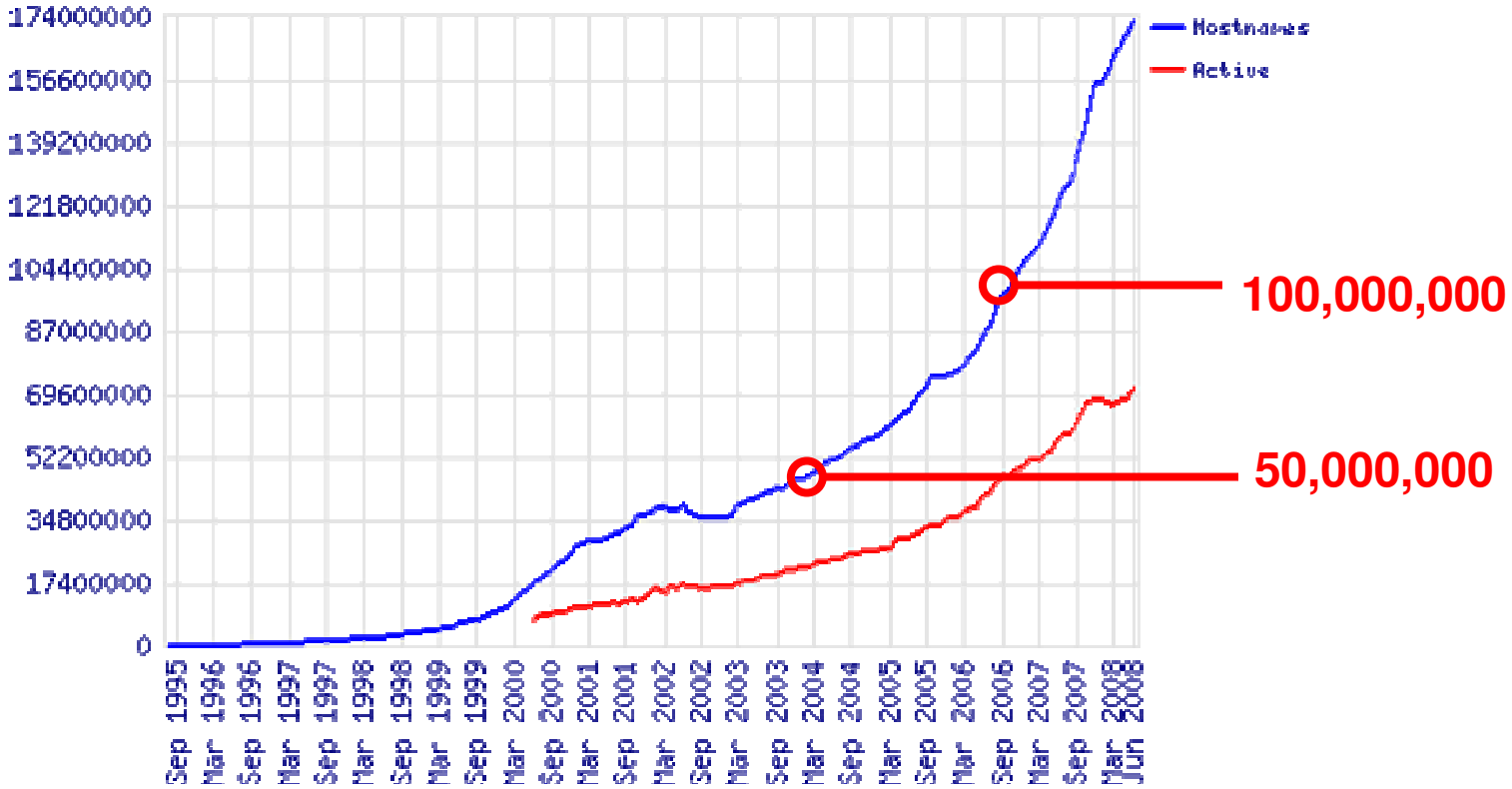
Índice



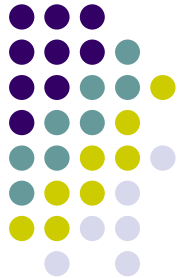
1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Introducción

- Crecimiento exponencial de la Web/Internet



Introducción



- INTECO: El 84,6% de los correos electrónicos que circulan en España son spam (92M, Marzo/08)
- MAAWG: El 85,2% de los correos mundiales son spam (0.1B, Diciembre/07)

Introducción

- ¿Qué buscamos en la Web?

Rank	1997 Excite data set (2,414 queries)	1999 Excite data set (2,539 queries)	2001 Excite data set (2,453 queries)
1	19.9% Entertainment or recreation	24.5% Commerce, travel, employment, or economy	24.7% Commerce, travel, employment, or economy
2	16.8% Sex and pornography	20.3% People, places, or things	19.7% People, places, or things
3	13.3% Commerce, travel, employment, or economy	10.9% Computers or Internet	11.3% Non-English or unknown
4	12.5% Computers or Internet	7.8% Health or sciences	9.6% Computers or Internet
5	9.5% Health or sciences	7.5% Sex and pornography	8.5% Sex and pornography
6	6.7% People, places, or things	7.5% Entertainment or recreation	7.5% Health or sciences
7	5.7% Society, culture, ethnicity, or religion	6.8% Non-English or unknown	6.6% Entertainment or recreation
8	5.6% Education or humanities	5.3% Education or humanities	4.5% Education or humanities
9	5.4% Performing or fine arts	4.2% Society, culture, ethnicity, or religion	3.9% Society, culture, ethnicity, or religion
10	4.1% Non-English or unknown	1.6% Government	2.0% Government
11	3.4% Government	1.1% Performing or fine arts	1.1% Performing or fine arts

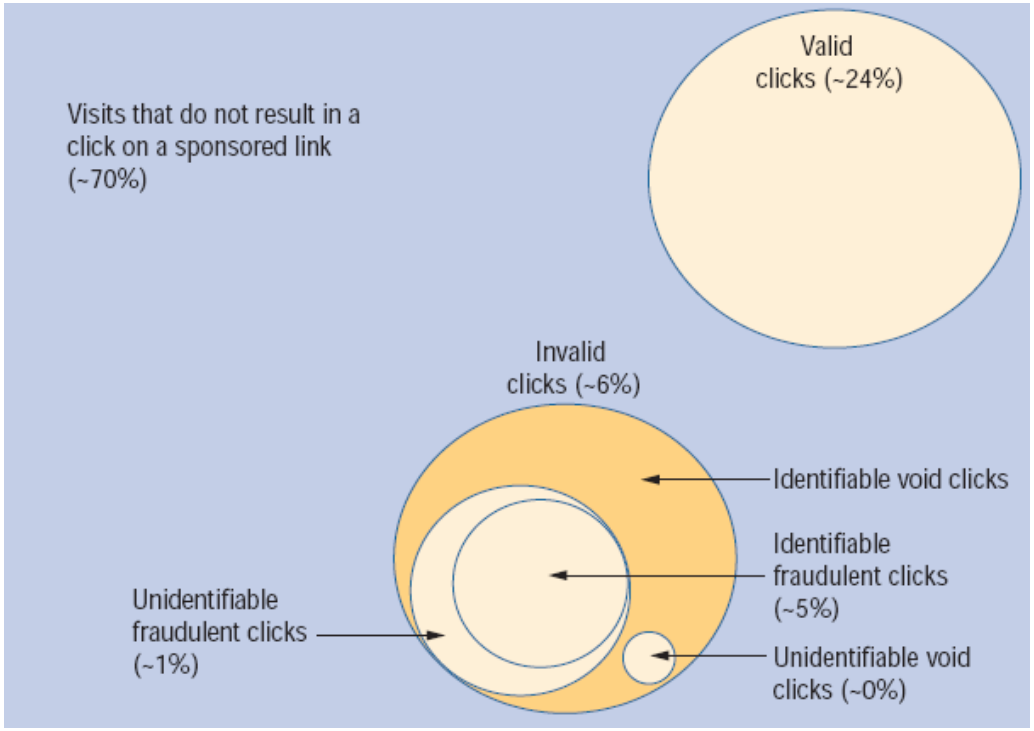
Introducción

- Google, publicidad, fraude

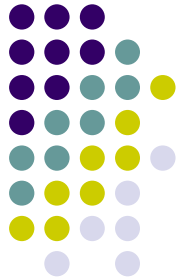
Year Ended December 31,				
2003	2004	2005	2006	2007
(in thousands, except per share amounts)				
$ 1,465,934	$ 3,189,223	$ 6,138,560	$ 10,604,917	$ 16,593,986

Google ingresa $16,000M
Por publicidad

¿Es extraño que haya
Web Spam y fraude de
clicks?

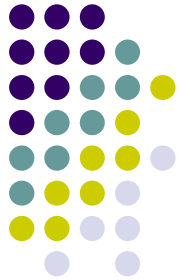


Introducción



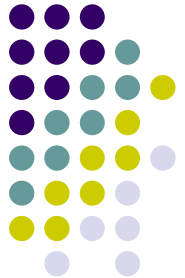
- Objetivo
  - Abordar distintas tareas de clasificación de contenidos ilícitos/inapropiados de manera unificada
  - Clasificación de texto con adversario
    - El adversario pretende burlar al sistema
  - Un problema general de **categorización de texto**

Índice



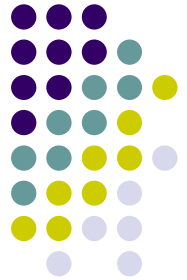
1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Categorización de texto



- Categorización (automática) de texto
  - Clasificación (automática) de documentos en categorías predefinidas
    - *Automated Text Categorization* (ATC)
  - RI = categorías NO predefinidas = consultas
  - Agrupamiento = categorías NO predefinidas
  - Importante = número y tipo de categorías
    - Solapamientos, jerarquías, dimensiones temáticas, etc.

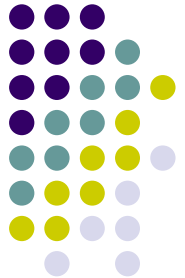
Categorización de texto



- Ejemplos de aplicaciones CAT

Documentos		***Categorías***
libros, noticias	→	descriptores temáticos
informes técnicos	→	palabras clave
páginas Web	→	directorios Web
	→	contenidos inapropiados
mensajes de correo electrónico	→	carpetas personales
	→	correo basura

Categorización de texto

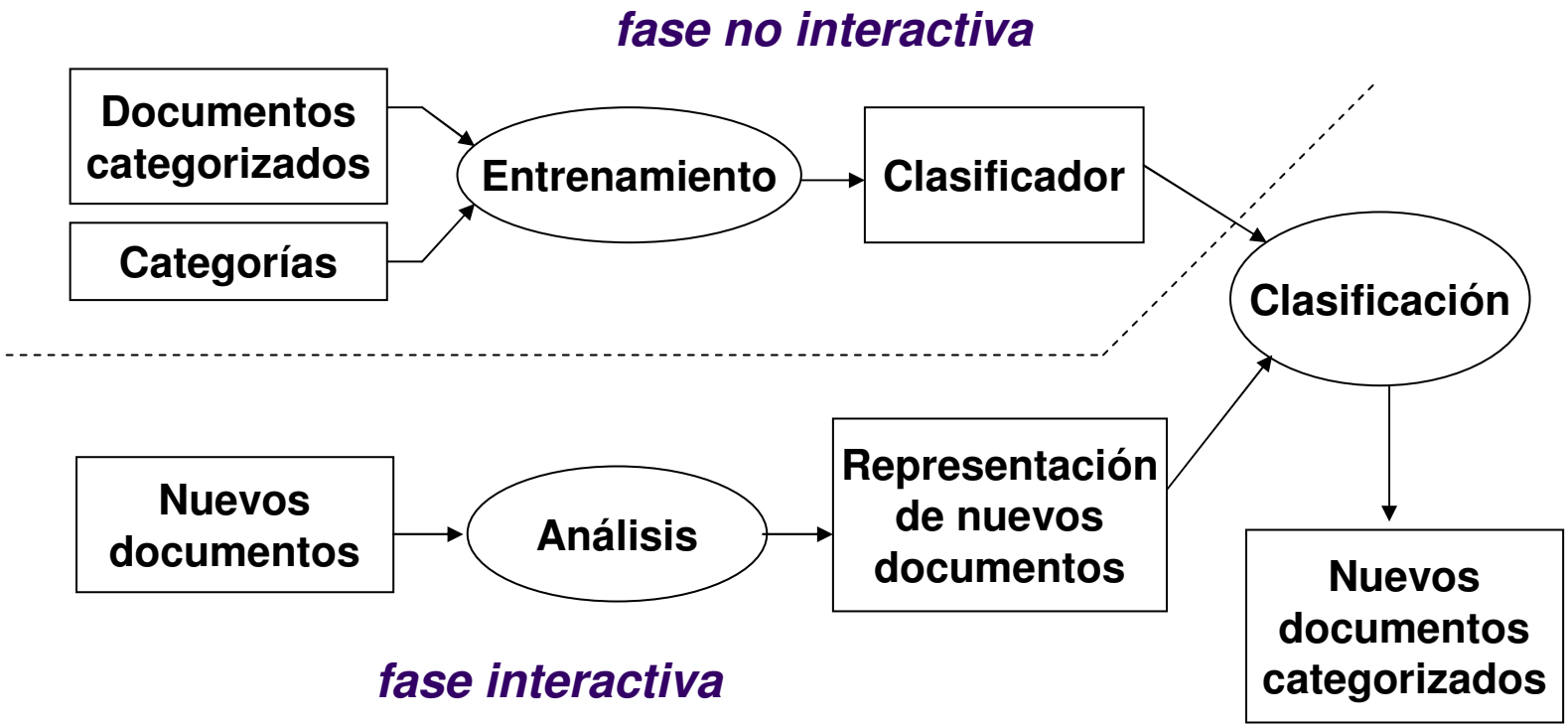


- Técnicas de automatización de CT
  - Construidos manualmente = sistemas expertos
    - Conjuntos de reglas derivadas por expertos anotadores humanos (catalogadores)

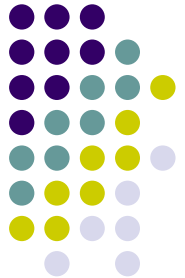
      IF (“sex” IN D) THEN (D IN porn)
  - Construidos automáticamente (CAT basada en aprendizaje)
    - Aplicación de técnicas de recuperación de Información y Aprendizaje Automático (AA)
    - Reducción del cuello de botella de adquisición del conocimiento

Categorización de texto

- CAT basada en aprendizaje

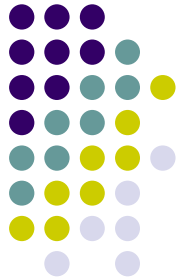


Categorización de texto



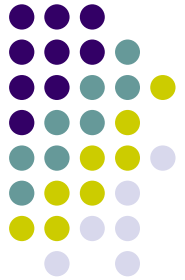
- Técnicas de CAT basada en aprendizaje
  - Representación de documentos
    - Vectores de pares atributo-valor
    - Usualmente, Modelo del Espacio Vectorial
    - Vectores de pesos de términos
      - Términos = raíces de palabras y filtrado con lista de parada
      - Pesos = binarios, TF, TF.IDF

Categorización de texto



- Técnicas de CAT basada en aprendizaje
  - Reducción de la dimensionalidad
    - Selección de términos
      - Métricas de selección (Ganancia de Información, Frecuencia en documentos, etc.)
        - E.g. 10% de términos más frecuentes en documentos
      - Agresividad
    - Extracción de términos
      - Agrupamiento de términos
      - Indexación Semántica Latente

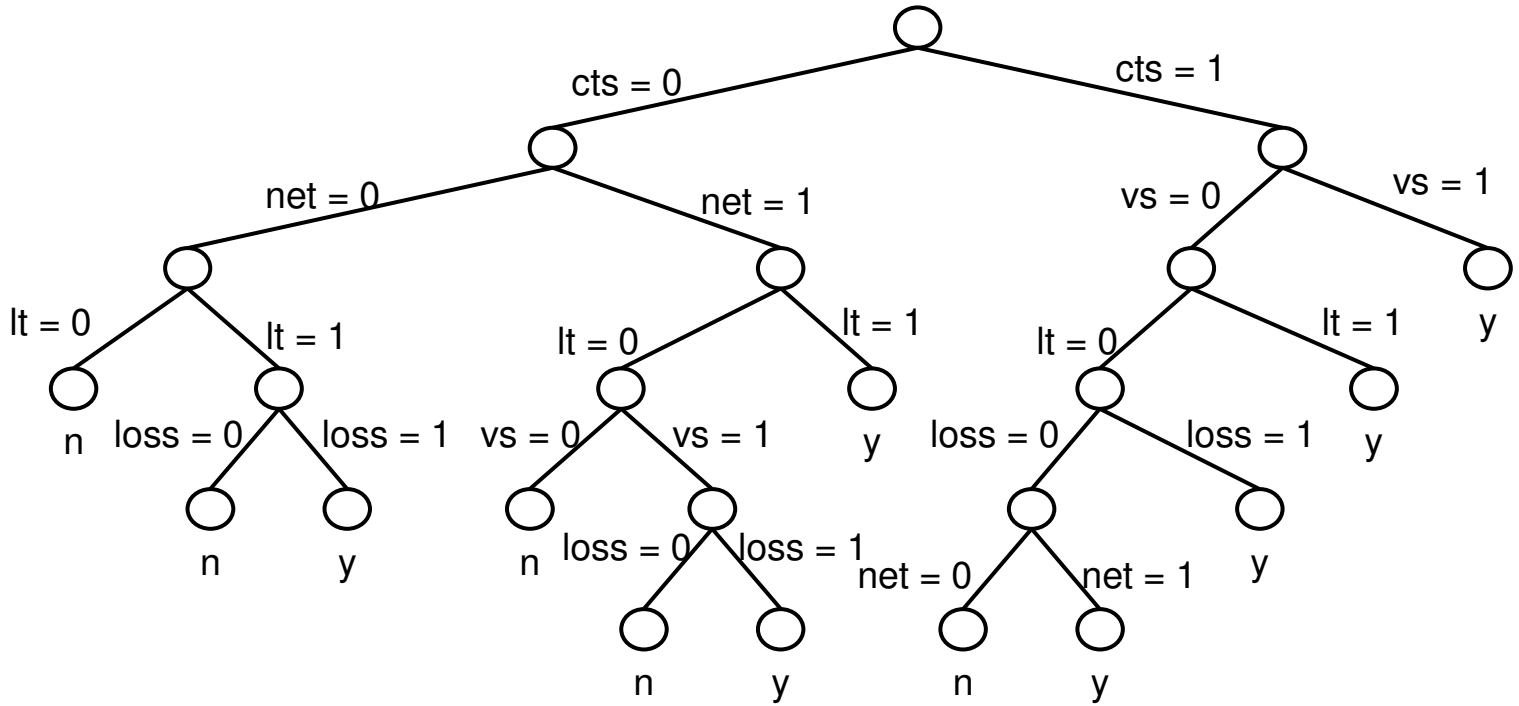
Categorización de texto



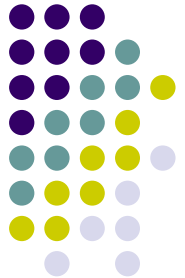
- Técnicas de CAT basada en aprendizaje
  - Aprendizaje
    - Clasificadores lineales (Rocchio, Winnow)
    - Clasificadores probabilísticos (Bayes Ingenuo)
    - Inducción de árboles de decisión (C4.5)
    - Generadores de reglas (Ripper)
    - Support Vector Machines
    - Redes Neuronales (perceptrón)
    - Aprendizaje perezoso (K-Nearest Neighbors)
    - Comités (Boosting)

Categorización de texto

- Ejemplo de aprendizaje en CAT (C4.5)

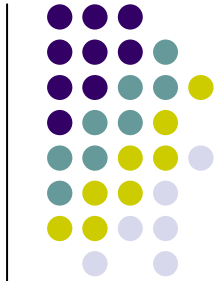


Categorización de texto



- Evaluación
  - Eficiencia
  - Efectividad
    - Estimación de la bondad de la aproximación
    - Colección de prueba
      - Reuters, OHSUMED, RCV1
    - Métricas de IR y ML
      - Cobertura (recall), precisión, F_1
      - Exactitud (accuracy), error

Categorización de texto



- Efectividad: métricas

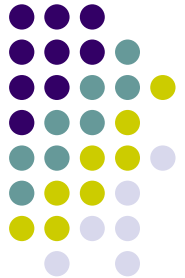
Sistema =>	Puestos en C	No puestos en C
En C	TP	FN
No en C	FP	TN

$$\textit{Cobertura } (R) = \frac{TP}{TP + FN}$$

$$\textit{Precisión } (P) = \frac{TP}{TP + FP}$$

$$F_1 = \frac{2RP}{R + P}$$

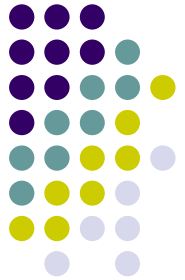
Categorización de texto



- Efectividad de la CAT
  - Promedio
    - Micro-media vs. macro-media
  - Tratamiento de la colección de evaluación
    - Partición
    - Validación cruzada
  - [Sebastiani02]

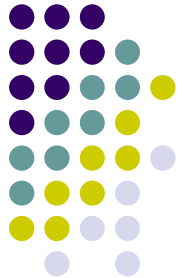
    “Automatic TC (...) has reached effectiveness levels comparable to those of trained professionals.”

Índice



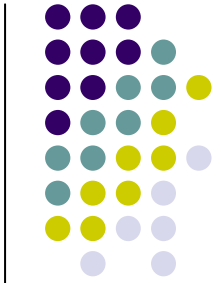
1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Clasificación de Texto con Adversario



- Tarea de clasificación de texto en la que existe un adversario
  - Su objetivo es que el clasificador deje de ser útil
  - Demasiados errores por exceso (falsos positivos)
    - Correos legítimos clasificados como spam
    - Paginas Web legítimas bloqueadas
    - Contenidos legítimos eliminados del ranking
  - Demasiados errores por defecto (falsos negativos)
    - Correos spam pasan el filtro
    - Demasiadas páginas perniciosas accedidas
    - Demasiadas páginas de spam en el ranking

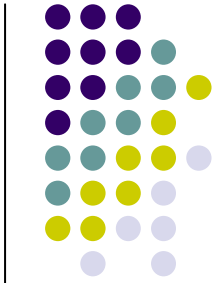
Clasificación de Texto con Adversario



- Coste de errores

Tarea	Error	Coste	Ejemplo y resultado
Filtrado de correo basura	FN	Bajo	El usuario final recibe un correo basura **Resultado**: El usuario borra el correo basura
	FP	Alto	El usuario no recibe un correo legítimo, que queda en cuarentena o es eliminado **Resultado**: El usuario debe revisar la cuarentena periódicamente, y siempre existe el riesgo de que pierda mensajes críticos (e.g. el pedido de un cliente)

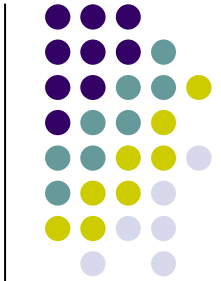
Clasificación de Texto con Adversario



- Coste de errores

Tarea	Error	Coste	Ejemplo y resultado
Filtrado de contenido Web	FN	Alto	El usuario accede a un contenido inapropiado (pornografía, juego de casino, racismo, etc.) **Resultado**: El usuario puede sentirse disgustado. Alternativamente, el usuario puede no informar del error. El supervisor debe inspeccionar la navegación periódicamente.
	FP	Medio	El usuario ve bloqueado un contenido legítimo. **Resultado**: El supervisor debe inspeccionar y reclasificar el contenido.

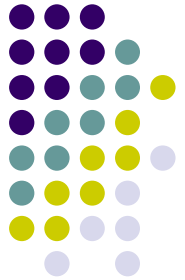
Clasificación de Texto con Adversario



- Coste de errores

Tarea	Error	Coste	Ejemplo y resultado
Detección de spam Web	FN	Alto	Una página logra un resultado muy alto en una búsqueda popular. **Resultado**: El buscador pierde credibilidad y sus supervisores deben inspeccionar periódicamente los resultados más altos en las búsquedas más populares.
	FP	Medio	Una página es eliminada en una búsqueda afín y procedente. **Resultado**: Su autor se queja, el buscador pierde credibilidad, y sus supervisores deben desbloquearla.

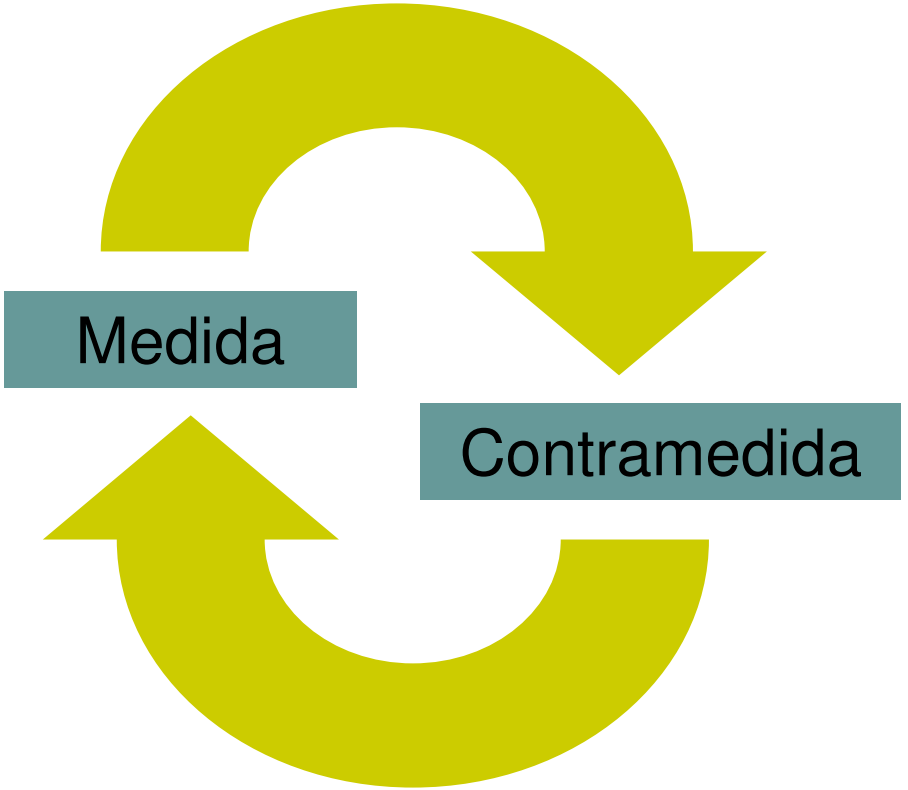
Clasificación de Texto con Adversario



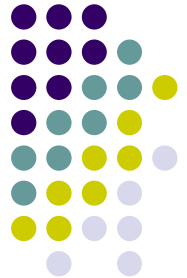
- Análisis comparativo de costes
  - Los costes dependen de la tarea
  - Los costes son asimétricos en todas las tareas
  - Los costes pueden ser variables y dependientes del entorno operativo
- La distribución también es variable
- Impacto ***crítico*** en el método de evaluación

Clasificación de Texto con Adversario

- Problema de la espada y el escudo o *carrera armamentística*

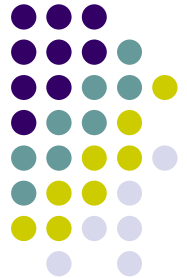


Clasificación de Texto con Adversario



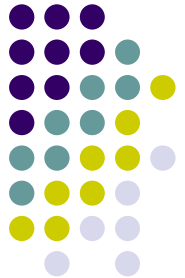
- Formalización en teoría de juegos [Dalvi04]
  - Interacción filtro – adversario = juego de la clasificación con adversario
  - En ciertos casos, se puede alcanzar un equilibrio de Nash
    - Problema no tratable en general
    - Versión reducida
      - Un movimiento por oponente = un cambio en el filtro/mensaje
      - Cada adversario conoce los parámetros del oponente
      - Cada adversario realiza su movimiento óptimo

Clasificación de Texto con Adversario



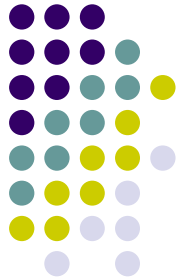
- Formalización en teoría de juegos [Dalvi04]
  - Clasificador y adversario sensibles al coste
    - Costes conocidos (escenarios 10/100/1000)
  - Clasificador bayesiano ingenuo
  - Estrategias para el clasificador y el adversario
  - Aplicación a filtrado de correo basura
    - Ataques: agregar palabras/longitud, sinónimos
    - Medida de utilidad
    - Se aumenta la utilidad (menos errores más costosos)
    - No se llega a equilibrio en juego repetido

Clasificación de Texto con Adversario



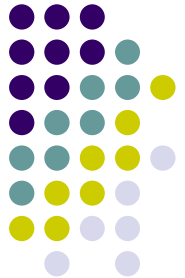
- Ingeniería inversa de clasificadores de adversario [Lowd05]
  - Condiciones más realistas, e.g.
    - Desconocimiento de parámetros del clasificador
    - Prueba iterativa con mensajes legítimos y spam
  - Método de aprendizaje ***para el adversario***
    - Evaluación con correo basura
    - Ataques de léxico (agregar/eliminar palabras)
    - Reducción en el número de pruebas a realizar para violar el filtro

Clasificación de Texto con Adversario



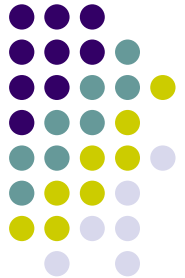
- Conclusiones
  - No se han testado otros problemas aparte del correo basura
  - Existe un marco teórico muy preliminar
    - No ha sido desarrollado en lo sucesivo
    - No existen evidencias de su utilización práctica
    - Los costes y distribuciones siguen siendo variables y desconocidos

Índice



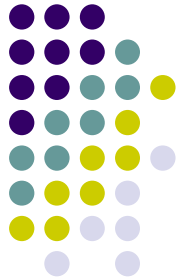
1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Filtrado de correo basura



- Motivación
  - Más del 85% del correo es basura
  - Problemas y pérdidas para
    - Usuarios (tiempo, coste de conexión, desconfianza)
    - Empresas (productividad, inversiones extraordinarias, falsos positivos de clientes, etc.)
    - Proveedores (infraestructura extra, pérdida de imagen)

Filtrado de correo basura

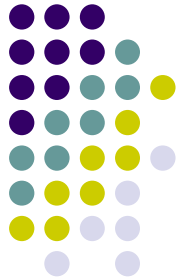


- Medidas de control
  - Legales => internacionalidad
  - Económicas => difícil implementación
  - Tecnológicas
    - Múltiples métodos de detección y filtrado

Listas negras y blancas	Franqueo computacional
Franqueo de Turing	CAPTCHAs
Direcciones prescindibles	Filtrado colaborativo
Tarros de miel	Reputación
Firma de correo/servidor	Análisis heurístico

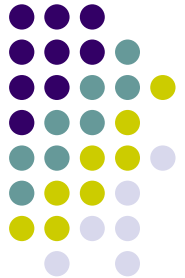
LA CLAVE ES LA INTEGRACIÓN DE TÉCNICAS

Filtrado de correo basura



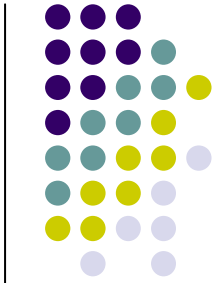
- Filtros bayesianos
  - Análisis del contenido basado en aprendizaje
    - Aprendizaje = adaptación a cambios del adversario
    - Efectividad = se ha alcanzado el 99%
    - Simplicidad de implementación
  - Resistencia “criptográfica”
  - Tecnología de Categorización de Texto

Filtrado de correo basura



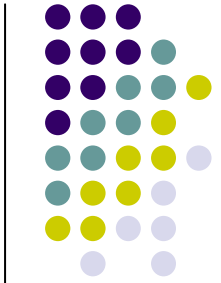
- Representación de los mensajes
  - Múltiples experimentos con
    - Tokenización = separación en cadenas/atributos
    - Reducción = paso a minúsculas, stemming, stoplist
    - Valores/Pesos = binarios, TF, TF.IDF
  - Hasta las decisiones más simples (e.g. separadores) pueden tener impacto dramático

Filtrado de correo basura



- Tokenización de Graham [Graham02,03]
  - Separador /= caracteres alfanuméricos, guiones, apóstrofes, símbolos de exclamación y símbolos de dólar
  - Secuencias de números y comentarios HTML ignorados
  - Mayúsculas y minúsculas, no stemming/stoplist
  - Puntos y comas entre dígitos = parte de tokens (IPs, precios)
  - Rangos de precios ($20-25) se separan ($20 y $25)
  - Se marcan términos en algunos campos especiales (From, To, Subject, Return-Path) y en URLs (Subject*gratis)

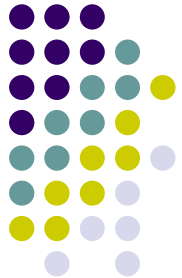
Filtrado de correo basura



- Tokens encadenados [Zdziarski04]
  - Bigramas tradicionales
  - Adicionales a los tokens individuales
  - Mejora de estadísticas

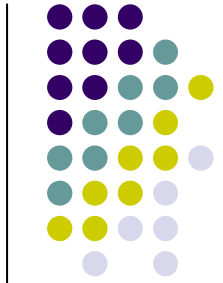
Términos	$P(spam\|w_1)$	$P(spam\|w_2)$	$P(spam\|w_1 * w_2)$
w_1=FONT, w_2= face	0,457338	0,550659	0,208403
w_1=color, w_2=#000000	0,328253	0,579449	0,968415
w_1=that, w_2= sent	0,423327	0,404286	0,010099

Filtrado de correo basura



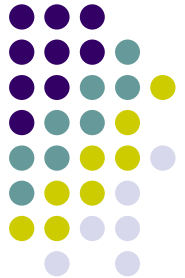
- Sparse Binary Polynomial Hash (SBPH), [Yerazunis03/04]
  - Sólo bigramas separados por otros tokens
    - “Tu puedes conseguir porno gratis”
      - “Tú gratis”
      - “puedes gratis”
      - “conseguir gratis”
      - “porno gratis”
  - Junto con Regla en Cadena Bayesiana => 99,9%

Filtrado de correo basura



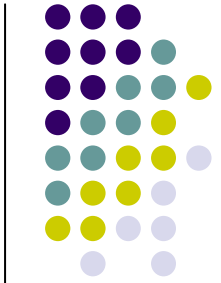
- Reducción de la dimensionalidad
  - Lo habitual es no hacerla
  - Selección de atributos
    - Ganancia de Información
  - Extracción de atributos
    - Indexación Semántica Latente

Filtrado de correo basura



- Algoritmos de aprendizaje
  - Clasificadores lineales (Rocchio, Winnow)
  - Clasificadores probabilísticos (Bayes Ingenuo)
  - Inducción de árboles de decisión (C4.5)
  - Generadores de reglas (Ripper)
  - Support Vector Machines
  - Redes Neuronales (perceptrón)
  - Aprendizaje perezoso (K-Nearest Neighbors)
  - Comités (Boosting)

Filtrado de correo basura



- Filtro bayesiano de Graham

PROBABILIDADES DE TOKENS

ST = veces que T aparece en spam
S = # mensajes spam
LT = # veces que T aparece en legítimo
L = # mensajes legítimos

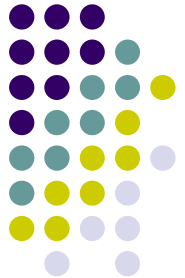
$$P(T) = \frac{\frac{ST}{S}}{\frac{2.LT}{L} + \frac{ST}{S}}$$

PROBABILIDAD MENSAJE

$$P(S) = \frac{\prod_{T \in TM} P(T)}{\prod_{T \in TM} P(T) + \prod_{T \in TM} (1 - P(T))}$$

TM = 15 tokens más extremos (lejanos de 0,5)

Filtrado de correo basura



- Compresión [Bratko06]
  - Regla de asignación

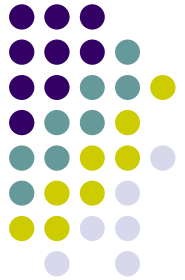
$$\text{class}(m) = \arg\min_{c=S,L}\left\{ \left|C(c \mid m)\right| \right\}$$

  - Aproximación

$$\left|C(c \mid m)\right| \approx \left|C(c.m)\right| - \left|C(c)\right|$$

  - Algoritmos = Dinamic Markov Compression, Prediction by Partial Matching

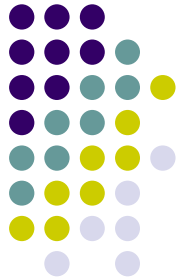
Filtrado de correo basura



- Resultados de compresión
  - Extraordinariamente efectiva
  - Superior a bayesianos y SVMs
  - Extremadamente resistente a la ingeniería inversa
  - Trivial de implementar

At TREC 2005, arguably the best-performing system was based on adaptive data compression methods

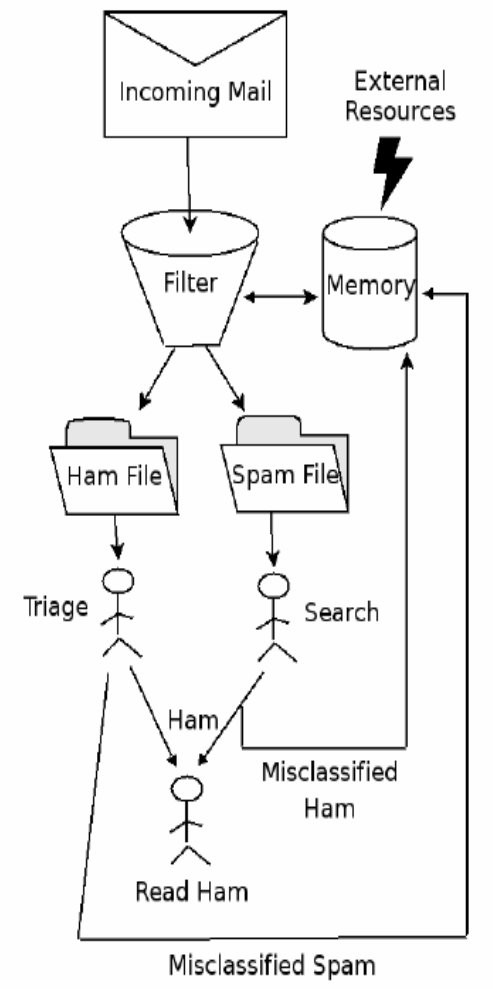
Filtrado de correo basura



- Evaluación (TREC Spam Track)
  - Aproximación a condiciones más realistas
    - Protocolo interactivo
    - Métricas independientes del coste y distribución
    - Colecciones públicas
  - Resultados más sólidos que los previos

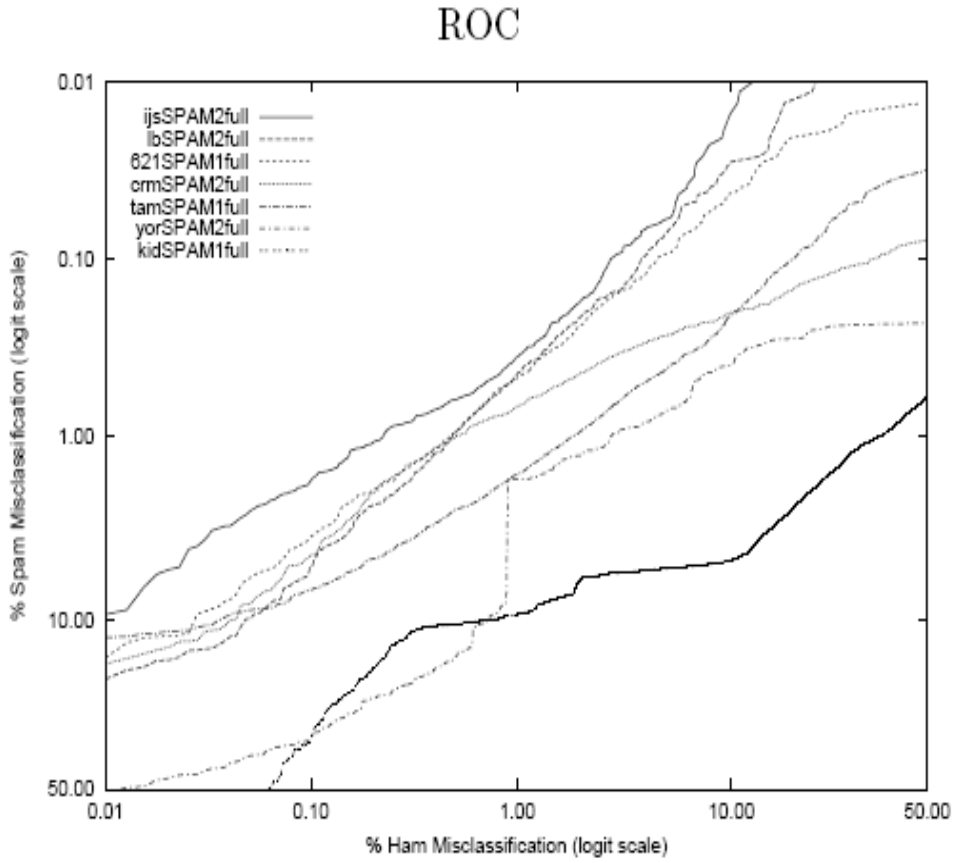
Filtrado de correo basura

- Operativa TREC
  - *TREC Spam Filter Evaluation Toolkit*
    - initialize
    - classify *message*
    - train ham *message*
    - train spam *message*
    - finalize



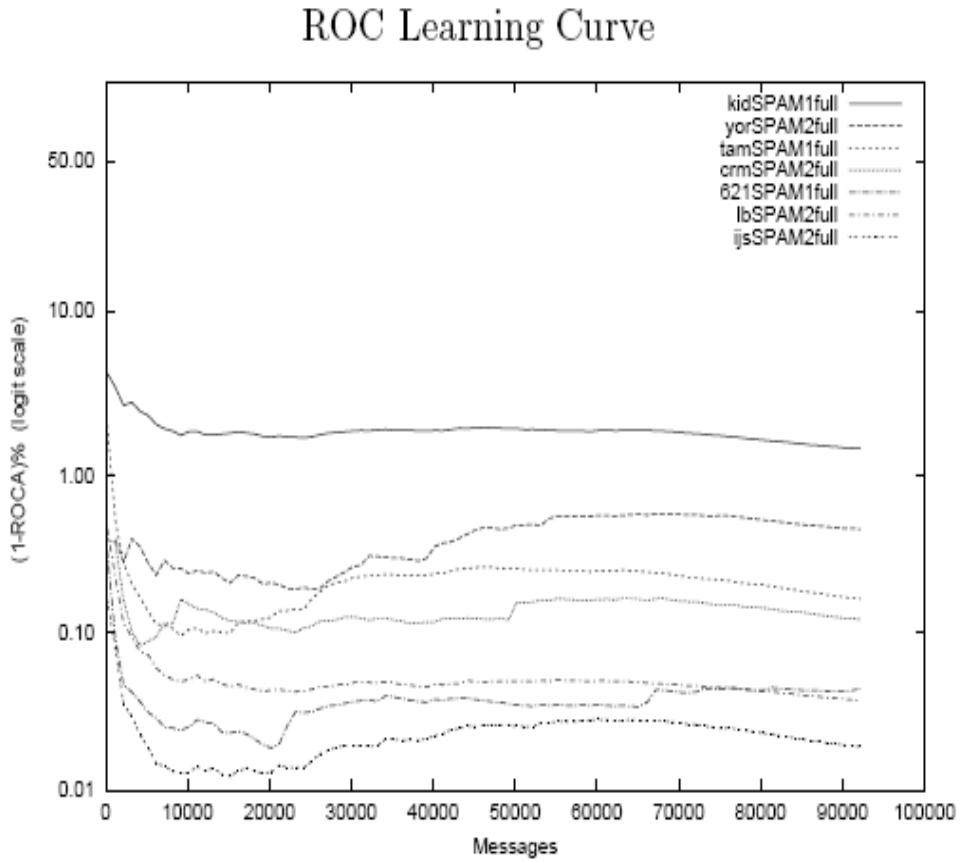
Filtrado de correo basura

- Análisis ROC/AUC (TREC05)

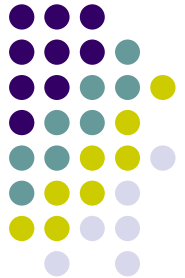


Filtrado de correo basura

- Curva de aprendizaje (TREC05)

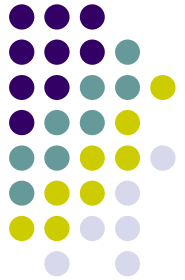


Filtrado de correo basura



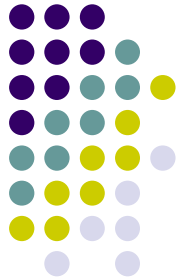
- Ataques
  - *Ataques de tokenización*
    - “viagra” => “v i a g r a” “viagra” => “via<!-->gra”
  - *Ataques de ofuscación*
    - “viagra” => “\ / 1 /\ G 4 / \”
  - *Ataques estadísticos*
    - + tokens positivos, - tokens negativos “viagra”
  - *Ataques de ocultación*
    - imágenes, PDF, URLs

Índice



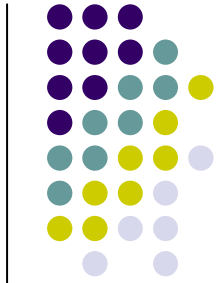
1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Filtrado de contenidos Web



- Motivación
  - Contenidos inapropiados según público/lugar
    - Público infantil => pornografía, racismo, anorexia
    - Puesto de trabajo => deportes, casinos, empleo
  - Considerables riesgos para los menores
  - Importantes pérdidas de productividad
  - Adicciones y dependencias

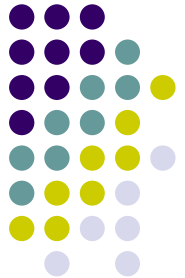
Filtrado de contenidos Web



- Medidas de control
  - Legales => limitadas por internacionalidad
  - Tecnológicas
    - Listas blancas y negras
    - Palabras clave
    - Análisis inteligente (basado en contenido con aprendizaje)
      - Análisis textual (Categorización de Texto)
      - Análisis de imagen (pornografía, violencia/sectas)

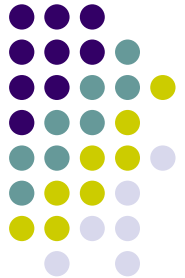
LA CLAVE ES LA INTEGRACIÓN

Filtrado de contenidos Web



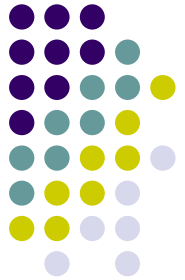
- Análisis del contenido textual con aprendizaje
  - Categorización de Texto
    - Representación
    - Reducción de la dimensionalidad
    - Aprendizaje
    - Evaluación

Filtrado de contenidos Web



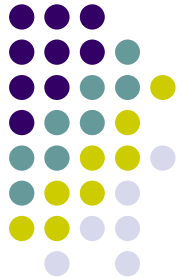
- Representación y reducción dimensionalidad
  - HTML => estructura = secciones
  - Tokenización => palabras = secuencias de caracteres alfanuméricos
  - Stemming, stoplist, Part Of Speech
  - Ngramas
  - Pesos binarios, TF, TF.IDF
  - Filtrado por Ganancia de Información

Filtrado de contenidos Web



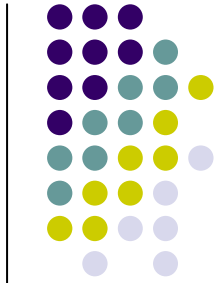
- Aprendizaje
  - Clasificador bayesiano ingenuo
  - Arboles de decisión como C4.5
  - Algoritmos basados en reglas como PART
  - Aprendizaje perezoso como kNN
  - Redes neuronales
  - Support Vector Machines
  - Aprendizaje sensible al coste

Filtrado de contenidos Web



- Evaluación
  - Carencia de colecciones estándar / competiciones
  - Evaluación por lotes
  - Medidas de Recuperación de Información / Aprendizaje Automático
    - Excepción => análisis ROC en [Gomez03]

Índice

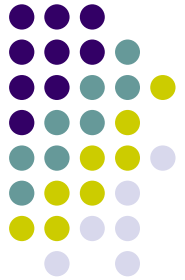


1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Detección de Spam Web

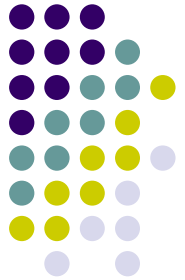
- Motivación
  - Motores de búsqueda = modelo de negocio basado en publicidad
  - Impacto Web = rentabilidad
    - Search Engine Optimization
  - Fraude en buscadores (*spam Web*)
    - Ranking inmerecido => pérdida de calidad = usuarios
  - Fraude de clicks (*click fraud*)
    - Clicks falsos en anuncios => impacto económico

Detección de Spam Web



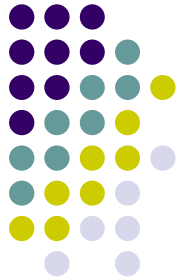
- Tipos de spam Web [Gyongyi05],
  - Spam de términos = palabras clave
  - Spam de enlaces = estructuras de enlaces contra métricas de calidad (e.g. PageRank)
  - Ocultación = colores, estructuras, redirecciones
- Existen métodos específicos para cada tipo

Detección de Spam Web



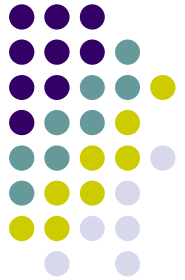
- Análisis del contenido
  - Genéricamente = CT orientada al género
    - Atributos estilísiticos, no temáticos
  - Trabajos adaptados al tipo de spam
    - Spam de comentarios de blog
    - Blogs spam (splogs)
    - Spam de etiquetas (tags)

Detección de Spam Web



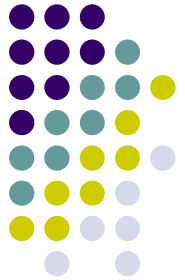
- Distintos ejemplos de atributos
  - Tomados del correo basura
  - Modelos del lenguaje = palabras
  - Consultas populares (extrínsecos)
  - Longitud de oraciones, aparición de palabras frecuentes (stoplist), etiquetas sintácticas
  - Número de palabras en la página y en el título, longitud media de las palabras, cantidad de texto presente en los hiperenlaces, fracción de contenido visible, etc.

Detección de Spam Web



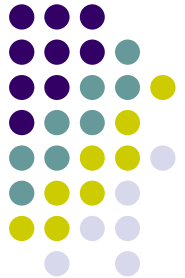
- Clasificadores
  - Bayesianos
  - Compresión
  - Árboles de decisión
  - Support Vector Machines
  - Regresión Logística

Detección de Spam Web



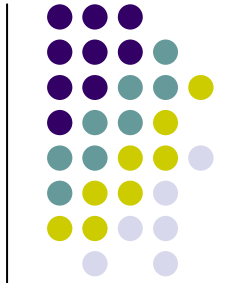
- Evaluación
  - Competiciones
    - Web Spam Challenge
    - ECML/PKDD 2008 Discovery Challenge
  - Colecciones públicas
  - Evaluación por lotes
  - Métricas similares a correo spam
    - ROC, AUC, curva de aprendizaje

Detección de Spam Web



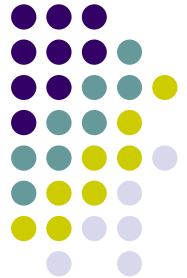
- Web Spam Challenge 2007/I => triunfó compresión sobre atributos de contenido
- Web Spam Challenge 2007/II => triunfó combinación de contenido + enlaces
- Más cerca de la estandarización que el filtrado Web
  - Más interés económico

Índice



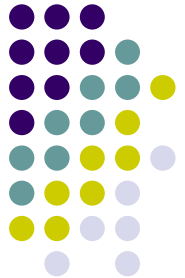
1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Algunas conclusiones



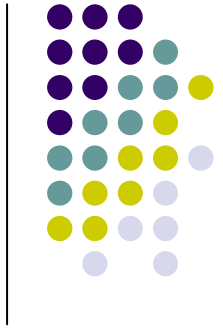
- Clasificación de texto con adversario = CT con oponente
  - No existe un análisis teórico general multitarea
  - Cada tarea puede tener sus propios costes
- Filtrado de correo basura
  - Tarea de referencia
  - Resultados excelentes (compresión)
  - ***Y sin embargo, el correo basura sigue creciendo …***

Algunas conclusiones



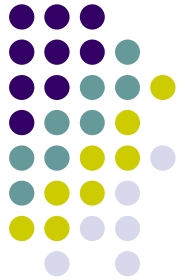
- Filtrado de contenidos Web
  - Poco estandarizada
  - Buenos resultados pero sin buenas prácticas
- Detección de spam Web
  - Reciente pero muy atractiva
  - Resultados intermedios por su complejidad
    - El análisis del contenido es importante pero funciona mejor combinar con enlaces

Índice



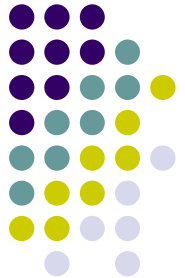
1. Introducción
2. Categorización de Texto
3. Clasificación de Texto con Adversario
4. Filtrado de correo basura
5. Filtrado de contenidos Web
6. Detección de Spam Web
7. Algunas conclusiones
8. Demostraciones

Demostraciones



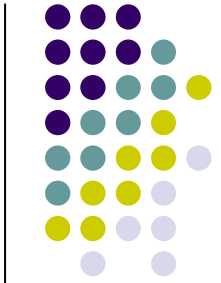
- Herramienta
  - WEKA (*Waikato Environment for Knowledge Analysis*)
  - Herramienta de minería de datos de propósito general
    - Capacidad para análisis de texto
  - Empleada en múltiples experimentos
  - http://www.cs.waikato.ac.nz/ml/weka/

Categorización de texto



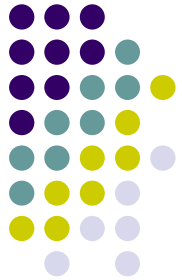
- Colección Reuters-21578
  - Noticias periodísticas económicas en inglés
  - Categorías económicas en inglés
  - Multiclase con solapamiento
  - 1504 (13 clases – 11 a 608 ejemplares)
- Representación
  - Minúsculas, stemming, stoplist, pesos binarios

Filtrado de spam SMS



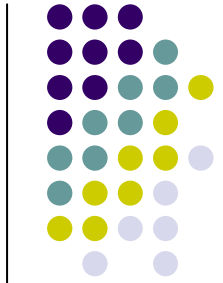
- Proyecto FISME
  - 1324 (322 spam)
- Representación simplista
  - WEKA – StringToWordVector
- Representación avanzada
  - Palabras, minúsuculas, bigramas, trigramas, bigramas de palabras

Filtrado de pornografía



- Proyecto TEFILA
  - 5992 (996 porno)
  - Español, extraídas del ODP
- Representación
  - Stemming, stoplist, pesos binarios, ganancia de información

Web Spam



- Web Spam Challenge 2007/I
- http://webspam.lip6.fr/
- 77,9M páginas, 11400 hosts
- 6552 hosts analizados

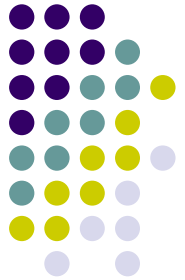
  Normal - 4,046 - 61.75%

  Spam - 1,447 - 22.08%

  Borderline - 709 - 10.82%

  Could not be classified - 350 - 5.34%

Web Spam



- Atributos basados en contenido
  - Número de palabras en la página, título, longitud media de las palabras
  - Fracción palabras en *anchor text* sobre total
  - Tasa de compresión bzip
  - Precisión y cobertura sobre el corpus
  - Precisión y cobertura sobre consultas
  - Probabilidad de independencia de trigramas
  - Entropía de trigramas