Bibliografía Anotada

Andrés Romero Rodríguez

May 15, 2006

References

[1] Kjersti Aas and Line Eikvil. Text categorisation: A survey., 1999.

Este artículo hace una revisión de las técnicas utilizadas en el campo de la categorización de texto, se enfoca principalmente en tres componentes:

- Extracción de Características: Se mencionan y explican brevemente las principales técnicas de representación de documentos tales como:
  - Boolean Weighting
  - Word Frequency Weighting
  - tf*idf Weighting
  - tfc Weighting
  - ltc Weighting
  - Entropy Weighting

  Además, se abordan consideraciones a tener en cuenta para reducir la dimensionalidad del conjunto de datos y se explican algunas técnicas.
- Métodos de Clasificación: Se explican métodos para clasificación tales como:
  - Rocchio
  - Bayes
  - knn
  - Árboles de decisión

– SVM

Adicionalmente, se explican algoritmos de *bagging y boosting*

- Medidas de Depsempeño: Se abordan consideraciones a tener en cuenta para la evaluación del desempeño de cualquier técnica de clasificación que se utilice.

Se mencionan además algunos trabajos previos en cuanto a clasificación de documentos usando el conjunto de datos *Reuters-21578* y se comparan los resultados obtenidos hasta el momento.

[2] Akiko N. Aizawa. Linguistic techniques to improve the performance of automatic text categorization. In *NLPRS*, pages 307–314, 2001.

El artículo presenta una metodología para adicionar componentes de procesamiento de lenguaje natural en tareas de clasificación de texto. Dado que las tcnicas utilizadas tradicionalmente para llevar a cabo el procesamiento del texto con miras a la clasificación (Extracción de términos, asignación de pesos y reducción de la dimensionalidad), se llevan a cabo de una forma muy simple, es necesario contar con mecanismos que permitan adicionar este tipo de técnicas de lenguaje natural para mejorar el desempeño de los métodos de categorización. El autor propone una metodología que adiciona tantas palabras como sea posible usar en la fase de clasificación, incluyendo términos que tienen baja frecuencia. Se introduce una extensión a la noción de td-idf que tiene en cuenta estás palabras poco frecuentes. Además de tener en cuenta palabras simples, se tienen en cuenta términos compuestos por varias palabras, aunque no se entra en detalles de cómo se obtienen dichos términos compuestos, los experimentos demuestran que son mas efectivos en el proceso de clasificación.

[3] Rie Kubota Ando. Latent semantic-space: iterative scaling improves precision of inter-document similarity measurement. In *SIGIR*, pages 216–223, 2000.

En este artículo se presenta una variación al algoritmo SVD (Singular Value Decomposition) utilizado en LSI (Latent Se-

mantic Indexing), el algoritmo propuesto difiere del SVD en cuanto a que éste tiene en cuenta los documentos que son *anormales*, los cuales son clasificados como ruido por SVD y por lo tanto no son tenidos en cuenta, este algoritmo los tiene en cuenta aplicando un escalamiento a dichos datos de modo que al momento de hacer la descomposición, tengan un efecto significativo sobre los demás datos. Esto se hace con el fin de mejorar la medida de similitud entre documentos. El artículo plantea además que se puede llegar a encontrar esos factores de escalamiento de los vectores anormales de manera dinámica, lo cual mejoraría el desempeño del algoritmo.

[4] Maria-Luiza Antonie and Osmar R. Zaïane. Text document categorization by term association. In *ICDM*, pages 19–26, 2002.

El artículo presenta una técnica para clasificación de documentos de texto basada en reglas de asociación. La idea general es aplicar algoritmos conocidos como *association rule mining* para encontrar reglas que asocien conceptos dados en los documentos con las clases a las que pertenece dicho documento. Una vez se han obtenido tales reglas a partir de los datos de entrenamiento, estas se usan para construir un clasificador. Una de las ventajas que se plantean en el artículo, es que este clasificador no asume que los términos encontrados en un documento son independientes entre si, como lo hacen la mayoría de técnicas de clasificación de texto. Otra ventaja que se muestra, es que las reglas generadas mediante el proceso de minería son fácilmente entendibles por los humanos.

[5] Pierre Baldi, Paolo Frasconi, and Padhraic Smyth. *Modeling the Internet and the Web: Probabilistic Method and Algorithms*. John Wiley, 2003.

[6] A. Basu, Carolyn R. Watters, and Michael A. Shepherd. Support vector machines for text categorization. In *HICSS*, page 103, 2003.

Este artículo presenta un algoritmo de clasificación de texto usando SVM (Support Vector Machines), se plantea el algoritmo general de clasificación y se compara con una red neuronal artificial, el cual muestra que el algoritmo con SVM

funciona mucho mejor dando buenos resultados en la fase de clasficiación. Posteriormente, se emplea un algoritmo para reducir la dimensionalidad del conjunto de datos. Este algoritmo llamado KSS, el cual utiliza un umbral para reducir el número de términos que se van a considerrar en la generación del diccionario. Al utilizar esta redución en el tamaño del diccionario, también se obtienen mejores resultados con el algoritmo basado en SVM.

[7] Ron Bekkerman, Ran El-Yaniv, Naftali Tishby, and Yoad Winter. Distributional word clusters vs. words for text categorization. *Journal of Machine Learning Research*, 3:1183–1208, 2003.

Este artículo presenta una comparación entre dos técnicas para representar los documentos en un entorno de clasificación, estas técnicas son:

- Bag of Words: en donde cada palabra simplemente corresponde a una característica dentro del vector que representa a cada documento; este es el enfoque tradicional
- Word Clusters: en donde lo que se busca es generar clusters de palabras relacionadas en cuanto a las categorías que representan, aquí se usó un algoritmo de clustering llamdo *Information Bottleneck*, el cual genera representaciones muy compactas que permiten mejorar el desempeño de los clasificadores

Entonces el algoritmo queda de la siguiente manera: en la primera etapa se obtiene un esquema de representación de los documentos como clusters de distribución de palabras (distributional clusters); y en la segunda etapa se aplica un algoritmo basado en SVM para realizar la clasificación. Los resultados que se obtuvieron al aplicar estas técnicas en diversos conjuntos de datos no son muy contundentes, puesto que para un conjunto de datos particular, el esquema de clusters funcionó mucho mejor que el esquema BOW, mientras que para los otros dos conjuntos de datos que probaron, funcionó mejor el esquema BOW.

[8] V. Richard Benjamins, Dieter Fensel, and Asunción Gómez-Pérez. Knowledge management through ontologies. In *PAKM*, 1998.

Este artículo presenta metodologías encaminadas a la administración del conocimiento al interior de las organizaciones, de modo que dicho conocimiento sea fácilmente accesible a las personas al interior de la organización, una ventaja importante de este enfoque es que toma en cuenta aspectos sociales. Las fases propuestas para llegar a un manejo apropiado del conocimiento son las siguientes:

- Obtención del conocimiento
- Organización y estructuración del conocimiento
- Refinamiento del conocimiento
- Distribución del conocimiento

La idea final de este proceso es obtener soluciones inteligentes a problemas difíciles mediante la utilización adecuada de este conocimiento.

[9] Caterina Caracciolo, Willem Robert van Hage, and Maarten de Rijke. Towards topic driven access to full text documents. In *ECDL*, pages 495–500, 2004.

Se presenta un enfoque para el acceso a la información contenida en documentos de texto a partir de los tópicos encontrados en dichos documentos. La idea básica, es combinar técnicas de segmentación de tópicos y técnicas de extracción de información. La extracción se realiza a partir de una ontología construida previamente, a partir de la cual se utilizan técnicas de extracción de información para conectar segmentos de texto obtenidos con los conceptos dados por loa ontología. Finalmente lo que se obtiene es un conjunto de subdocumentos mas pequeños para cada documento que tienen tópicos homogeneos.

[10] Soumen Chakrabarti. *Mining the Web. Discovering Knowledge from Hypertext Data*. Morgan Kaufmann Publishers, 2002.

[11] Wai chiu Wong and Ada Wai chee Fu. Incremental document clustering for web page classification, August 31 2000.

[12] Stephen D'Alessio, Keitha Murray, Robert Schiaffino, and Aaron Kershenbaum. The effect of using hierarchical classifiers in text cate-

gorization. In *Proceeding of RIAO-00, 6th International Conference "Recherche d'Information Assistee par Ordinateur"*, pages 302–313, Paris, FR, 2000.

En este artćulo se presenta una técnica para realizar clasificación de texto jerárqica, esto es, dada una jerarquía de categorias representada como un árbol, un documento puede ser asignado a una o mas de las categorías representadas por los nodos hoja de dicho árbol. El artćulo presenta el esquema de construcción de una jerarquía de clasificadores que corresponde a la jerarquía de categorías que se quiere clasificar; en esta técnica, el vocabulario y los pesos de los términos están asociados con las categorías, lo que reduce el número de érminos que se deben utilizar. Para construir los clasificadores, se usa un algoritmo voraz que empieza con todas las categorías en el mismo nivel y posteriormente va adicionando categorías intermedias durante la fase de entrenamiento. Experimentalmente los autores muestran que se obtienen buenos resultados en cuanto a eficiencia del algoritmo.

[13] Franca Debole and Fabrizio Sebastiani. Supervised term weighting for automated text categorization. In *SAC*, pages 784–788, 2003.

El artículo se enfoca en el ajuste de pesos de los términos de un documento para su clasificación. La primera etapa consiste en la selección de las características mas relevantes del documento, la segunda es el ajuste de pesos de dichos conceptos (o términos); así, por cada término de cada documento, se genera un peso que representa la importancia que tiene dicho término para la discriminación semántica del documento. Se muestran experimentos usando técnicas de ajuste de pesos en los principales algoritmos de clasificación, tales como:

- Rocchio
- Knn
- SVM

[14] Offer Drori. Identifying the subject of documents in digital libraries automatically using frequently-occurring words - study and findings, May 23 2003.

En este artículo se muestra un estudio realizado sobre un conjunto de documentos en los que se busca identificar el tópico tratado a partir de la frecuencia de ocurrencia de ciertas palabras, para ello utilizan una herramienta llamada TextAnalysis, la cual obtiene automáticamente una clasificación a partir del contenido del documento. El estudio que hacen los autores se basa en identificar qué tantas de esas palabras que utiliza esta herramienta hacen parte del título del documento y un conjunto de keywords definido por el autor del documento. Muestran también que la clasificación obtenida por esa herramienta puede ser generada a partir del título y los keywords únicamente, lo que mejoraría el rendimiento del clasificador, dado que se observa que entre mas largos son los documentos, menor es la tasa de clasificaicón efectiva dada por TextAnalysis.

[15] Dave Elliman. Automatic derivation of on-line document ontologies, July 26 2001.

En este artículo se muestra una metodología para construir ontologías automáticamente a partir de la información contenida en los documentos que están siendo evaluados y en documentos adicionales. El enfoque que le dan a la construcción de dicha ontología es dirigido por los intereses de cada usuario, así, cada uno podrá tener una ontología diferente de acuerdo a lo que esté buscando. Lo que se hace es tomar un conjunto de keyword generados por el usuario junto con la información contenida en el documento para ver si esta clasificación representa una abstracción mas general que el propio contenido del documento. La ontología se construye en principio realizando un conteo de las palabras que generalizan los keywords dados por el usuario y sus sinónimos. Posteriormente se aplica PCA para encontrar las palabras mas importantes. Aunque la técnica parece interesante, no se muestran resultados obtenidos siguiendo esta metodología y únicamente se hace una discusión acerca del uso que se le pueden dar a esas ontologías generadas.

[16] Dave Elliman and J. R. G. Pulido. Visualizing ontology components through self-organizing maps. In *IV*, page 434, 2002.

Este artículo describe una metodología para construir ontologías a partir de componentes visualizados utilizando un mapa auo-organizativo (SOM), el esquema general del agoritmo es el siguiente:

1. Se obtiene un conjunto de documentos relacionados
2. Se realiza un proprocesamiento de dichos documentos eliminando palabras que aportan poca información.
3. Se obtienen las palabras base (por ejemplo, play para plays, playing, played); además se asignan pesos a cada uno de los términos obtenidos usando tf*idf y se crea un vector para cada documento
4. Con los vectores individuales de cada documento, se construye lo que llaman el espacio del documento
5. Se constryue una red SOM utilizando ese espacio del documento generado
6. Se crea una ontología a partir de los componentes visualizados por el mapa

No se dan detalles acerca de la forma de obtener la ontología a partir de los resultados del SOM, y se muestran algunos resultados preliminares.

[17] Radu Florian and David Yarowsky. Dynamic nonlocal language modeling via hierarchical topic-based adaptation. In *37th Annual Meeting of the Association for Computational Linguistics*, pages 167–174, 1999.

En este artículo se presenta un modelo para generar auntomáticamente árboles de clasificación que corresponden a un modelo jerárquico de lenguaje. Se muestra que es importante tener en cuenta que la frecuencia de las palabras en un documento está fuertemnte relacionada con el contexto del documento. Para atacar este problema, se utilizan técnicas de estimación de probabilidad de un tópico. El algoritmo general consta de 2 fases:

1. Agrupamiento de los documentos: se utiliza una técnica de clustering no supervisado para agrupar documentos que tienen tópicos relacionados
2. Generación del árbol: en esta etapa se utiliza un algoritmo de clustering jerárquico aglomerativo, el árbol queda construido de modo que los documentos similares van a quedar juntos.

Una vez generado el árbol, se discute acerca de su aplicación en el contexto de clasificación.

[18] Hichem Frigui and Olfa Nasraoui. Simultaneous categorization of text documents and identification of cluster-dependent keywords, April 07 2002.

En este artículo se presenta una técnica para la categorización de documentos y generación simultanea de keywords para cada categoría. Se presenta un algoritmo no supervisado basado en K-means en donde cada cluster está representado como un grupo de keywords. Se plantean 2 principales ventajas que puede llegar a tener este enfoque:

- Los clusters generados van a tener un significado semántico mas apropiado
- Es posible generar, a partir de dichos keywords, una descripción de cada cluster automáticamente

En este algoritmo, cada cluster tiene un grupo de características y un peso asociado, a partir de los cuales se obtendrán posteriormente los keywords de cada categoría. En cada iteración del algoritmo se van ajustando los pesos lo que hace que al final se obtengan aquellos que son mas relevantes.

[19] Evgeniy Gabrilovich and Shaul Markovitch. Feature generation for text categorization using world knowledge. In *IJCAI*, pages 1048–1053, 2005.

Los autores proponen una técnica para la generación de características adicionales a las contenidas en los documentos. estas características adicionales van a mejorar el desempeño de los algoritmos de clasificación de texto, ya que estos se basan únicamente en las características contenidas en el documento.

Para la generación de dichas características adicionales se debe contar con un conocimiento previo acerca del dominio en el cual se encuentra el documento que se quiere clasificar. Este conocimiento es extraido de ontologías públicas y de libre acceso, de modo que se logra una generalización de los conceptos a partir de las ontologías. Finalmente se aumenta el conjunto de palabras que hacen parte de la representación de los documentos con los conceptos generados automáticamente a partir de la base de conocimiento.

[20] Gongde Guo, Hui Wang, David A. Bell, Yaxin Bi, and Kieran Greer. An kNN model-based approach and its application in text categorization. In Alexander F. Gelbukh, editor, *Proceedings of CICLING-04, 5th International Conference on Computational Linguistics and Intelligent Text Processing*, pages 559–570, Seoul, KO, 2004. Springer Verlag, Heidelberg, DE.

En este artículo los autores hacen un análisis de los algoritmos Knn y Rocchio usados para clasificación de texto, muestran tanto sus fortalezas como sus debilidades y proponen un nuevo algoritmo, que combina las fortalezas de ambos, llamado Knn-Model para el cual se presentan los conceptos básicos como representación de los documentos y se definen las medidas de similaridad que se van a usar. Además se describe de manera general (sin entrar en muchos detalles) el esquema que sigue el algoritmo tanto en su fase de entrenamiento como en la de clasificación. Finalmente, presentan algunos resultados obtenidos utilizando 2 conjuntos de datos conocidos (Reuters-21578 y 20 NewsGroups), en los cuales se muestra que este nuevo algoritmo propuesto en la mayoría de los casos es un poco mejor que Knn y Rocchio, aunque la mejora no es significativa; dada la complejidad del algoritmo se esperaría que los resultados fueran mucho mejores comparados con 2 técnicas que son conceptualmente muy sencillas.

[21] Eui-Hong Han and George Karypis. Fast supervised dimensionality reduction algorithm with applications to document categorization & retrieval. In *CIKM*, pages 12–19, 2000.

El artículo presenta una técnica para reducir la dimensionalidad de documentos de texto en una fase previa a su clasificación. Este algoritmo de reducción de la dimensionalidad tiene en cuenta las dependencias que existen entre los diferentes conceptos contenidos en un documento, la idea básica es generar un vector de conceptos por cada documento, luego por cada categoría se genera un vector centroide a partir de los vectores de cada documento en la categoría. A partir de estos vectores centroides se presenta un algoritmo supervisado y uno no supervisado para reducir la dimensionalidad y se muestra que el algoritmo no supervisado tiene mejores resultados en una fase posterior de clasificación.

[22] Eui-Hong Han, George Karypis, and Vipin Kumar. Text categorization using weight adjusted k-nearest neighbor classification. In *PAKDD*, pages 53–65, 2001.

El artículo presenta una variación al tradicional k nearest neighbor, en el cual se utiliza una medida de distancia basada en el ajuste de pesos del vector de clasificación, dicho ajuste se realiza basánndose en algoritmos de hill climbing. Además se tratan aspectos en cuanto al incremento en el desempeño de está técnica y se muestran algunas variantes que pueden mejorarlo. Además, se realizan experimentos con varios conjuntos de datos y se compara su rendimiento, el cual en la mayoría de los casos es superior a las otras técnicas.

[23] Thomas Hofmann. Probabilistic latent semantic indexing. pages 50–57.

Se presenta una extensión al indexamineto semántico (LSI) puesto que tiene deficiencias en cuanto a sus fundamentos estadísticos, la idea es mapear tanto los documentos como los términos a una representación en el espacio semántico, reduciendo así la dimensionalidad y facilitando la extracción de información.

[24] Andreas Hotho, Steffen Staab, and Gerd Stumme. Explaining text clustering results using semantic structures. In *PKDD*, pages 217–228, 2003.

El artículo aborda técnicas para dar explicaciones en el contexto de la clasificación de texto. Los principales resultados que se quieren obtener despues de un proceso de clasificación son los siguientes:

- Por qué se generan los clusters
- Como se relacionan varios clusters
- Explicación a través de jerarquias semánticas.

Además se explican técnicas para representación de texto, Clustering, extracción de características; siempre teniendo en mente la meta de explicar los resultados del algoritmo.

[25] Andreas Hotho and Gerd Stumme. Conceptual clustering of text clusters, May 23 2002.

Se presenta una técnica de dos etapas para el agrupamiento de documentos y además para obtener una descripción de cada uno de los clusters obtenidos. Las fases del algoritmo son las siguientes:

1. Clustering Tradicional: Utilizan una variante de K-means llamada Bisecting K-means, con esta obtienen un conjunto de clusters que se pasarán a la siguiente etapa.
2. Clustering Conceptual: Se utilizan los clusters obtenidos en la fase anterior junto con un conjunto de tesauros para obtener descripciones de los clusters usando un algoritmo llamado *Formal Concept Analysis*

Se realizaron pruebas usanbdo el conjunto de datos Reuters-21578, se muestran algunas gráficas de los clusters obtenidos en donde se observan las distancias o similitudes entre los diversos clusters. Finalmente, se plantea el uso de técnicas de cluster no disyuntivas para explorar si se obtienen mejores resultados.

[26] M. Jarrar and R. Meersman. Scalability and knowledge reusability in ontology modeling. In Veljko Milutinovic, editor, *Proceedings of the International conference on Infrastructure for e-Business, e-Education, e-Science, and e-Medicine*, volume SSGRR2002s, Rome, Italy, 2002. SSGRR education center.

Se presenta un framework para el manejo de Ontologas llamado DOGMA. Sobre este framework, se discuten aspectos relacionados con la escalabilidad y reusabilidad de las ontologás, en cuanto a que estas se puedan extender y unificar fácilmente a partir de ontologías existentes previamente. Aunque el artículo habla acerca de este framework, no se hace una presentación formal del mismo; simplemente se presentan algunas consideraciones técnicas relacionadas con los aspectos de reusabilidad y escalabilidad; mencionan que detalles adicionales acerca de la implementación de DOGMA se dejan para otros artículos. Adicionalmente, se presentan algunos ejemplos del uso e integración de diversas ontologías relacionadas, tomando como base un conexto de alquiler de vehículos.

[27] George Karypis and Eui-Hong Han. Concept indexing: A fast dimensionality reduction algorithm with applications to document retrieval and categorization. Computer science department TR-00-0016, University of Minnesota, 2000.

Los autores presentan una técnica para reducir la dimensionalidad de un conjunto de documentos. Este algoritmo debe funcionar adecuadamente para un ambiente tanto supervisado como no supervisado; para ello se establecen dos fases:

1. Se agrupan los documentos en k categorías.
   - En el caso de reducción de la dimensionalidad supervisada, estas categorías se obtienen a partir de las etiquetas que tengan los documentos.
   - Para el caso de reducción no supervisada, se utiliza un algoritmo de clustering (no se especifica cual, en principio, cualquier algoritmo que utilice un tiempo lineal es apropiado para este procedimiento)
2. Cada categoría será usada para calcular los ejes del espacio reducido.

Se muestran los resultados de algunos experimentos y enfatizan en que esta técnica reduce el tiempo considerablemente comparada con otras como LSI.

[28] Taku Kudo and Yuji Matsumoto. A boosting algorithm for classification of semi-structured text. In *Proceedings of EMNLP-04, 9th Conference on Empirical Methods in Natural Language Processing*, Barcelon, ES, 2004.

Este artículo propone una técnica para clasificación de documentos basada en las opiniones representadas en el documento más que en el contenido sintáctico de este.El algoritmo se basa en una representación de los documentos en forma de árbol,los cuales son clasificados y luego se aplica una técnica de boosting. Posteriormente se obtiene un conjunto de árboles de todos los generados que van a representar las categorías; este conjunto es reducido y con el tamaño limitado (solo son escogidos los árboles más relevantes) Adicionalmente, se compara el desempeño de esta técnica contra SVM basándose en dos conjuntos de datos de prueba.

[29] Alberto Lavelli, Bernardo Magnini, and Fabrizio Sebastiani. Building thematic lexical resources by term categorization. In *SIGIR*, pages 415–416, 2002.

Se introduce una técnica para la generación semi-automática de diccionarios temáticos mediante algoritmos de clasificación de texto; para estos algoritmos, se emplean técnicas de extracción de información. Se tiene en cuenta las asociaciones entre los términos y los temas, dichos términos son representados por un vector, pero además cada término tiene un tema asociado, en lugar de asociar este a un documento completo. Para el proceso de aprendizaje se utilizan técnicas de boosting.

[30] Dawn Lawrie and W. Bruce Croft. Discovering and comparing topic hierarchies, October 13 2000.

En este artículo se hace una introducción a la creación automática de jerarquías en el contexto de la clasificación de documentos. Primero se hace una revisión general de los enfoques actuales en esta área y posteriormente se seleccionan dos enfoques que se analizan con más detalle y se realiza una coparación entre ellos. Estos enfoques son:

- Subsumption Hierarchies
- Lexical Hierarchies

Estos enfoques se comparan realizando una búsqueda mostrando la jerarquá generada por cada uno de los enfoques. Adicionalmente se hace una evaluación de las jerarquías asignándoles puntajes y midiendo la similaridad dentro de las jerarquías.

[31] Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S. Yu. Building text classifiers using positive and unlabeled examples. In *ICDM*, pages 179–188, 2003.

El artículo trata el problema de la construcción de un clasificador de texto a partir únicamente de ejemplos positivos, se cuenta además con ejemplos no etiquetados. Se muestra como este problema tradicionalmente ha sido atacado con un enfoque de dos pasos:

- Paso 1: Identificar ejemplos negativos a partir de los datos no etiquetados
- Paso 2: Construir clasificadores a parir de ejemplos positivos y negativos.

Se introducen algunos nuevos métodos para ambos pasos y se evaluan las posibles combinaciones. Finalmente, se propone una técnica basada en SVM para la cual los experimentos muestran que se comporta mejor que las técnicas anteriores.

[32] Tao Liu, Zheng Chen, Benyu Zhang, Wei-Ying Ma, and Gongyi Wu. Improving text classification using local latent semantic indexing. In *ICDM*, pages 162–169, 2004.

Se establece que el indexamineto semántico (LSI) no es óptimo dado que cuando se aplica de manera global no tiene en cuenta la discriminación entre las diferentes clases y solo se concentra en la representación. Por lo tanto se propone un nuevo método para indexamiento semántico basado en relevancia, teniendo en cuenta una discriminación entre clases. Este método debe mejorar el desempeño en la fase de clasificación. Además se hace una comparación entre LSI global y varias técnicas de

LSI local, y se llega a la conclusión de que es más apropiada utilizar un esquema local.

[33] R. E. Madsen, J. Larsen, and L. K. Hansen. Part-of-speech enhanced context recognition. In S. Douglas A.K. Barros, J. Principe, J. Larsen, T. Adali, editor, *Proceedings of IEEE Workshop on Machine Learning for Signal Processing XIV*, pages 635–644, Piscataway, New Jersey, September 2004. IEEE Press.

Este artículo se enfoca en el uso de técnicas tomadas de metodologías de procesamiento de lenguaje natural para tratar de mejorar los resultados de los clasificadores de documentos. En particular, se estudia el efecto que tiene la identificación de tipos de palabras al interior de los documentos en su representación, además se hace una evaluación del comportamiento de un enfoque híbrido: Bag of words y Part-of-speech y se analizan los efectos que tiene esta unión. Se realizan algunos experimentos al respecto, pero los resultados y conclusiones obtenidas a partir de estos no son muy claros, aunque se establece que hay una mejora respecto a otras técnicas.

[34] Alessandro Moschitti and Roberto Basili. Complex linguistic features for text classification: A comprehensive study. In *ECIR*, pages 181–196, 2004.

Este artículo hace un estudio acerca de las características lungüisticas complejas que se deben tener en cuenta para mejorar el desempeño de los clasificadores de textoG generalmente, las representaciones utilizadas no tienen en cuenta dichas características; por lo que se plantean nuevas formas de representación, además, se hace una experimentación utilizando estás nuevas representaciones.

[35] Terri Oda and Tony White. Developing an immunity to spam. In *GECCO*, pages 231–242, 2003.

Se plantea un Sistema Inmune Artificial para detectar Spam, el cual es dividido en 2 capas, una social y una tecnológica, en la capa social no se profundiza mucho, en la tecnológica se establece el concepto de anticuerpo cono una expresión regular

que es capaz de identificar varias formas de spam. Adem"as, se establece un sistema de memoria implementando pesos para los anticuerpos que han sido activados previamente.

[36] Seong-Bae Park and Byoung-Tak Zhang. Co-trained support vector machines for large scale unstructured document classification using unlabeled data and syntactic information. *Information Processing and Management*, 40(3):421–439, 2004.

En este artículo primero se hace una introducción al problema de clasificación de texto y luego se analizan las formas de mejorarlo, en partícular los problemas que se atacan son los siguientes:

1. Falta de ejemplos etiquetados: para esto se emplea un algoritmo llamado co-training, el cual, a partir de un pequeño conjunto de datos etiquetados y un grupo de datos no etiquetados, se aumentan los datos etiquetados basándose en una medida de confianza que indica si un documento no etiquetado se puede asignar a una categoría. Este enfoque se usa conjuntamente con una máquina de soporte vectorial para construir el clasificador (co-trained SVM)
2. Se ignora la información lingüistica: se plantean dos enfoques para tener en cuenta información adicional contenida en los documentos (no únicamente usar la representación Bag-of-words):
   - Información Léxica.
   - información Sintáctica.

Se hace un estudio de estos enfoques planteados y se realizan algunos experimentos usando los conjuntos de datos Reuters 21578 y TREC 7. Finalmente se definen unas medidas de desempeño y se comparan los enfoques, el que usa solo información léxica, el que usa solo información sintáctica y el que usa ambos tipos de información.

[37] William M. Pottenger and Ph. D. Detecting patterns in the LSI term-term matrix. Technical report, September 25 2002.

Este artículo básicamente hace un estudio de los valores obtenidos por el algoritmo LSI para tratar de encontrar patrones de co-ocurrencias de términos y ver que tan importantes son estos. Se propone un framework teórico enfocado a detectar correlaciones entre los diferentes términos, lo cual, según los autores, ayudaría a entender la semántica aportada por el algoritmo LSI; dichas correlaciones se obtienen a partir de co-ocurrencias de términos similares. Se presenta además una prueba matemática acerca del uso de las co-ocurrencias en los algoritmos SVD y LSI, y se realizan experimentos con algunos conjuntos de datos tratando de descubrir dichas correlaciones.

[38] David Ramamonjisoa. Towards automated research topics discovery on scientific domain by agents system, January 02 2003.

Aqu se presenta un sistema de agentes llamado Karoka; la idea de este sistema es obtener información automáticamente de un tópico en particular a partir de información encontrada en la web y presentara al usuario en forma adecuada. Para hacer esto, el sistema trata de extraer tópicos, reglas de asociación e información útil a partir de técnicas tales como categorización de texto, aprendizaje de máquina, detección de tópicos y clustering. El artículo muestra un caso de estudio en donde se busca determinar algunos tópicos prometedores en la investigación en el área de redes computacionales. Para mostrar este caso de estudio, se presenta desde la arquitectura del sistema, la forma de obtener reglas de asociación, y algunos experimentos desarrollados; aunque el resultado obtenido de estos experimentos no es muy bien explicado.

[39] Anthony Scime. *Web Mining: applications and techniques*. Idea Group, 2005.

[40] Sam Scott and Stan Matwin. Feature engineering for text classification. In *ICML*, pages 379–388, 1999.

Se examinan formas alternativas para la representación de texto en el contexto de la clasificación de texto, mas allá del tradicional *bag of words* (en donde cada palabra corresponde

a una característica); dicha representación nueva se basa en relaciones sintácticas y semánticas entre las palabras (frases, sinónimos, etc). El artículo plantea además la hipótesis de que estas nuevas técnicas deben mejorar el desempeño de los clasificadores, aunque la nueva representación no da muy buenos resultados experimentales.

[41] Mark Sinka and David Corne. Evolving document features for web document clustering: A feasability study. In *Proceedings of the 2004 IEEE Congress on Evolutionary Computation*, pages 891–897, Portland, Oregon, 20-23 June 2004. IEEE Press.

[42] Noam Slonim and Naftali Tishby. The power of word clusters for text classification. In *Proceedings of ECIR-01, 23rd European Colloquium on Information Retrieval Research*, Darmstadt, DE, 2001.

[43] Alexander Strehl, Joydeep Ghosh, and Raymond Mooney. Impact of similarity measures on web-page clustering. In *Proceedings of the 17th National Conference on Artificial Intelligence: Workshop of Artificial Intelligence for Web Search (AAAI 2000), 30–31 July 2000, Austin, Texas, USA*, pages 58–64. AAAI, July 2000.

[44] Gerd Stumme. Formal concept analysis on its way from mathematics to computer science. In *ICCS*, pages 2–19, 2002.

El artículo hace una revisión de un enfoque para la representación de conocimiento llamado FCA (Formal Concept Analysis), este enfoque tiene sus orígenes en las matemáticas como la formalización matemática del concepto C̈oncepto; se muestra además la relación entre este concepto mateático y su apolicación en las ciencias de la computación. FCA se aborda como una técnica de representación del conocimiento. Se utilizan técnicas de extracción y procesamiento de conocimiento conceptual, las cuales se dividen en:

- Adquisición
- Representación
- Inferencia
- Comunicación

[45] Aixin Sun and Ee-Peng Lim. Hierarchical text classification and evaluation. In *ICDM*, pages 521–528, 2001.

Se desarrolló un algoritmo de clasificación de texto jerárquico, en donde la novedad es que un documento puede ser asignado a cualquier nodo del árbol de jerarquias, no solo a los nodos hoja como lo hacen los algoritmos tradicionales. Se propone además, una medida de similaridad entre las categorías para tener en cuenta el grado de error al clasificar mal un documento. El artículo presenta entonces dos principales enfoques:

- Se definen un conjunto de medidas de desempeño que consideran la relación semántica entre categorías de la jerarquía. Cuando un documento es clasificado incorrectamente, se debe medir qué tan diferente es la clasificación de la categoría correcta
- Se desarrollo un método de categorización usando SVM a partir de las nuevas medidas de desempeño definidas.

[46] Domonkos Tikk, Jae Dong Yang, and Sun Lee Bang. Hierarchical text categorization using fuzzy relational thesaurus, April 22 0.

[47] J. J. Verbeek. Supervised feature extraction for text categorization, February 14 2002.

[48] Ian H. Witten, Gordon W. Paynter, Eibe Frank, Carl Gutwin, and Craig G. Nevill-Manning. KEA: Practical automatic keyphrase extraction. *CoRR*, cs.DL/9902007, 1999.

[49] Osmar R. Zaïane and Maria-Luiza Antonie. Classifying text documents by associating terms with text categories. In *Australasian Database Conference*, 2002.