LACREST MEDELLÍN 2012
REQUIREMENTS ENGINEERING & SOFTWARE TESTING
LATIN AMERICAN CONGRESS
ISBN 978-958-46-0577-1
"POR LA PROFESIONALIZACIÓN DEL DESARROLLO DE SOFTWARE"
Medellín – Colombia / 13 y 14 de julio de 2012

Behavioral Variability of Clustering and Induction Based on Domain Features

Variabilidad del Comportamiento de Agrupamiento e Inducción Basado en las Características del Dominio

Marcelo López N.[1], Ramón Garcia-Martinez[2]

[1] Programa de Magister en Ingeniería en Sistemas de información. Escuela de Posgrado. Facultad Regional Buenos Aires. Universidad Tecnológica Nacional. Buenos Aires, Argentina. *zappapet@yahoo.com*.

[2] Grupo de Investigación en Sistemas de Información. Departamento de Desarrollo Productivo y Tecnológico. Universidad Nacional de Lanús. Buenos Aires, Argentina. *rgarcia@unla.edu.ar*.

INFORMACIÓN DEL ARTÍCULO

Tipo de artículo: Investigación

Historia del artículo:
Recibido: 23/04/2012
Correcciones: 29/05/2012
Aceptado: 31/05/2012

Palabras clave
Explotación de Información, caracterización de dominios, agrupamiento, inducción, generación de dominios.

Categories and Subject Descriptors
H.2.8 [**Database Applications**]: Data mining

General Terms
Algorithmic Analysis

Keywords
Information Mining, domains characterization, clustering, induction, domains generation procedure.

ABSTRACT
The information mining processes use different algorithms for data mining for obtaining patterns of knowledge from the examples (instances) of the problem domain. One of the hypotheses in which these algorithms are based, is that the complexity of the domain on which they are working, has no bearing on the quality of the patterns of knowledge obtained. One of the processes of information mining of interest is the rules discovery of group membership which uses clustering algorithms and induction algorithms. In this research we characterize the complexity of the domains in terms of pieces of knowledge that describe them and that processes of information mining seek to discover. We use an experiment to demonstrate that in the case of the information mining process of rules discovery of group membership, the quality of patterns of knowledge obtained differ according to the algorithms used in the process and to the complexity of the domains to which they are applied.

RESUMEN
Los procesos de explotación de información utilizan distintos algoritmos de minería de datos para obtener patrones de conocimiento a partir de los ejemplos (instancias) que se tienen sobre el dominio de problema Una de las hipótesis con las que trabajan estos algoritmos es que la complejidad del dominio sobre cuya información se trabaja, no incide sobre la calidad de los patrones obtenidos. Uno de los procesos de explotación de información de interés es el de descubrimiento de reglas de pertenencia a grupos que utiliza algoritmos de agrupamiento (*clustering*) y algoritmos de inducción. En este trabajo se caracteriza la complejidad de los dominios en términos de las piezas de conocimiento que los describen y que los procesos de explotación de información buscan descubrir. Se demuestra mediante un experimento que, en el caso del proceso de descubrimiento de reglas de pertenencia a grupos la calidad, los patrones que se obtienen difieren en función de los algoritmos que se utilizan en el proceso y de la complejidad de los dominios al cual aplican.

1. INTRODUCCIÓN

Varios autores [1-5] han señalado la necesidad de disponer de procesos de explotación de información que permitan obtener conocimiento a partir de las grandes masas de información disponible, su caracterización y tecnologías involucradas. En [6] se han definido cinco procesos de explotación de información: descubrimiento de reglas de comportamiento, descubrimiento de grupos, descubrimiento de atributos significativos, descubrimiento de reglas de pertenencia a grupos y ponderación de reglas de comportamiento o de pertenencia a grupos.

Los procesos de explotación de información utilizan distintos algoritmos de minería de datos para obtener patrones de conocimiento a partir de los ejemplos (instancias) que se tienen sobre el dominio de problema [7]. Una de las hipótesis implícitas con las que trabajan estos algoritmos es que fijados los algoritmos para el proceso de explotación de información, la complejidad del dominio sobre cuya información se trabaja, no incide sobre la calidad de los patrones obtenidos.
Sin embargo, hay indicios [8] que muestran que la complejidad de los dominios en términos de las piezas de conocimiento que los describen y que los procesos de explotación de información buscan descubrir, emerge como un componente no despreciable al momento de analizar la calidad de los resultados a obtener.

En este contexto, se demuestra mediante un experimento que para el caso del proceso de explotación de información descubrimiento de reglas de pertenencia a grupos, la calidad de los patrones que se obtiene difiere en función de la complejidad de los dominios a los cuales se aplica y de los algoritmos que se utilizan en el proceso. En suma, en este artículo se caracterizan los distintos tipos de complejidad de dominios (sección 2), las preguntas de investigación s

2. CLASIFICACION DE DOMINIOS POR COMPLEJIDAD

Para abordar el tema de la complejidad de los dominios, en [9] se propone caracterizar a los mismos en términos de piezas de conocimiento (reglas) que explican la pertenencia de una determinada instancia (ejemplo) a un determinado dominio. Es así que la complejidad del dominio queda caracterizada por la cantidad de clases que lo describe, la cantidad de reglas que definen la pertenencia a cada clase, la cantidad de atributos que

puede tener cada regla y la cantidad de valores (distintos) que puede tener cada atributo.

Con base en los atributos de clasificación enunciados y el protocolo de clasificación de dominios propuesto en [8] se pueden clasificar los dominios en función de su complejidad en los siguientes tipos:

- *Dominios de Complejidad Simple*: son aquellos dominios en que el aumento de la cantidad de ejemplos por regla, mejora el cubrimiento de reglas (independientemente de las demás dimensiones utilizadas.

- *Dominios de Complejidad Mediana:* son aquellos dominios que se explican con ejemplos con pocos atributos y pocas clases, o pocos atributos y muchas clases o pocas clases y pocas reglas por clase.

- *Dominios Oscilantes*: son aquellos dominios que se explican con ejemplos donde pueden variar el número de atributos por ejemplo, o cantidad de ejemplos soportados por una regla, o valores comunes de atributos en un conjunto de ejemplos cubiertos por la misma regla.

- *Dominios Complejos*: son aquellos dominios que se explican con ejemplos con pocos atributos y muchos valores posibles por atributo, o con muchos
- atributos y pocos valores posibles por atributo, o con muchos atributos y muchos valores posibles por atributo.

- *Dominios Hipercomplejos*: son aquellos dominios que se explican con ejemplos donde pueden variar la cantidad de posibles valores que pueden tomar los atributos, el número de atributos que cubren ejemplos, la cantidad de las reglas que cubren ejemplos, o la cantidad de clases

3. REGLAS DE PERTENENCIA A GRUPOS

Uno de los procesos de explotación de información de interés es el de descubrimiento de reglas de pertenencia a grupos que utiliza algoritmos de agrupamiento (*clustering*) y algoritmos de inducción.

El proceso de descubrimiento de reglas de pertenencia a grupos aplica cuando se requiere identificar cuáles son las condiciones de pertenencia a cada una de las clases en una partición desconocida "a priori", pero presente en la masa de información disponible sobre el dominio de problema.

Son ejemplos de problemas que requieren este proceso: tipología de perfiles de clientes y caracterización de cada tipología, distribución y estructura de los datos de un *website*, segmentación etaria de estudiantes y comportamiento de cada segmento, clases de llamadas telefónicas en una región y caracterización de cada clase, entre otros [10]. Este proceso y sus subproductos pueden ser visualizados gráficamente en la Figura 1.

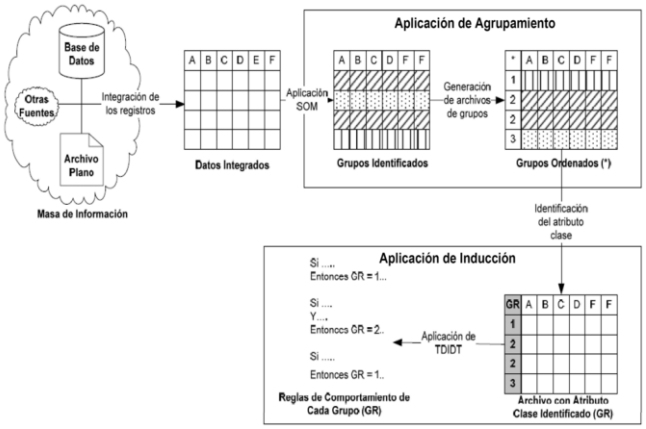

Fig. 1: Esquema y subproductos resultantes de Agrupamiento e Inducción

El proceso se puede describir de la siguiente manera [6]: en primer lugar se identifican todas las fuentes de información (bases de datos, archivos planos, entre otras), se integran entre sí formando una sola fuente de información a la que se llamará datos integrados. Con base en los datos integrados se aplica Agrupamiento. Como resultado de la aplicación de Agrupamiento se obtiene una partición del conjunto de registros en distintos grupos a los que se llama grupos identificados. Se generan los archivos asociados a cada grupo identificado. A este conjunto de archivos se lo llama grupos ordenados. El atributo "grupo" de cada grupo ordenado se identifica como el atributo clase de dicho grupo, constituyéndose este en un archivo con atributo clase identificado (GR). Se aplica Inducción sobre el atributo clase de cada grupo GR y se obtiene un conjunto de reglas que definen el comportamiento de cada grupo.

4. PREGUNTAS DE INVESTIGACIÓN

En este proyecto se han planteado las siguientes preguntas de investigación:

- ¿Es correcta la suposición que la performance de los algoritmos utilizados para agrupamiento e inducción es independiente de la complejidad del dominio?

- En caso de falseamiento de la suposición previa: ¿Cuál es el par <algoritmo de agrupamiento, algoritmo de inducción> que mejor performance proporciona dada la complejidad del dominio?

5. DISEÑO EXPERIMENTAL

Se consideraron de interés para el estudio los siguientes dominios: Dominios de Complejidad Mediana, Dominios Oscilantes, Dominios Complejos, Dominios Hiper-complejos. En orden a dar respuesta las preguntas de investigación se ha diseñado un experimento estructurado en tres pasos. El primer paso consiste en la preparación del experimento e involucra: (a) generación del dominio, basado sobre la generación de las clases y reglas que indican la pertenencia a cada clase y (b) generación de ejemplos que sean cubiertos por cada regla. La salida de este paso es un conjunto de reglas de pertenencia a cada clase y un conjunto de ejemplos del

ISBN 978-958-46-0577-1
"POR LA PROFESIONALIZACIÓN DEL DESARROLLO DE SOFTWARE"
Medellín – Colombia / 13 y 14 de julio de 2012

dominio. El segundo paso consiste en la ejecución del experimento. Este paso involucra: (a) aplicar el proceso de agrupamiento al conjunto de ejemplos del dominio para obtener el conjunto de grupos de ejemplos y (b) aplicar a cada grupo de ejemplos el proceso de inducción para obtener reglas que caractericen la pertenencia a cada grupo, obteniendo así el conjunto de reglas descubiertas. El tercer y último paso consiste en la comparación entre el conjunto de reglas de clasificación generadas en el primer paso y las reglas descubiertas en el segundo paso. El porcentaje de reglas descubiertas de forma correcta, define el éxito del experimento [11].

Las variables independientes del experimento son: (a) cantidad de clases que rigen el dominio, (b) cantidad de reglas que indican la pertenencia a cada clase, (c) cantidad de atributos en cada regla, (d) cantidad de posibles valores que puede tomar cada atributo, (e) cantidad de ejemplos de cada regla y (f) el porcentaje sobre la cantidad total de valores posibles que puede tomar cada atributo. La variable dependiente del experimento es porcentaje de reglas pertenecientes al conjunto de reglas original que se encuentra en el conjunto de reglas descubiertas. Un esquema de representación se muestra en la Figura 2.

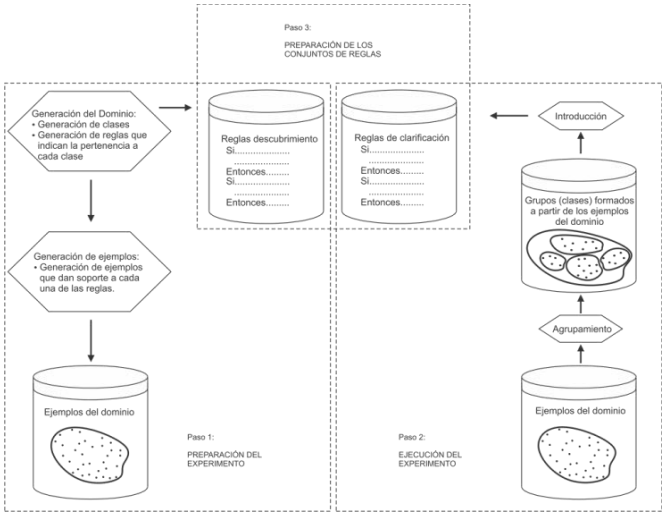

Fig. 2: Esquema de los pasos experimentales

Para llevar adelante la experimentación se utiliza entonces el banco de pruebas para la generación de dominios en condiciones de laboratorio, la generación de ejemplos en dichas condiciones, el agrupamiento por parte de diversos algoritmos de agrupamiento, la inducción de reglas sobre el conjunto de grupos de ejemplos, la obtención de reglas de pertenencia a grupos por parte de los distintos algoritmos de inducción, y finalmente la contrastación de las reglas obtenidas en el segundo paso con las reglas creadas en el primer paso, con el fin de establecer el porcentaje de reglas correctamente cubiertas. La dinámica de trabajo que tiene el banco de pruebas puede ser descrita entonces de la siguiente manera: (a) generación de clases, (b) generación de reglas de pertenencia a cada clase, (c) generación de ejemplos, (d) agrupamiento de ejemplos, (e) aplicación de inducción a los grupos obtenidos, (g) generación de reglas descubiertas y (f) comparación de ambos conjuntos de reglas. Para el experimento se utilizaron los siguientes algoritmos de agrupamiento: SOM [12], KMEANS [13], NNC [14]; y los siguientes algoritmos de inducción: M5 [15], CN2 [16] y AQ15 [17].

6. DESARROLLO DEL EXPERIMENTO

Se llevaron a cabo 210 corridas del banco de pruebas, utilizando para ello los 7 atributos antes descritos para cada uno de los escenarios que surgen de las combinaciones de atributos representativas de los cuatro dominios en estudio. Los escenarios utilizados en las experiencias, con su correspondiente grupo se presentan en la Tabla 1.

Tabla 1. Descripción de los distintos tipos de escenarios utilizados en la experimentación

NÚMERO DE SECTOR	CANTIDAD DE ATRIBUTOS DE CADA EJEMPLO	CANTIDAD DE POSIBLES VALORES QUE PUEDEN TOMAR LOS ATRIBUTOS	CANTIDAD DE CLASES QUE RIGEN LOS EJEMPLOS	CANTIDAD DE REGLAS QUE INDICAN LA PERTENENCIA A CADA CLASE	CANTIDAD DE EJEMPLOS UTILIZADOS PARA CADA REGLA	GRUPO AL QUE PERTENECE
1	3	1	1	1	1	**A**
2	3	1	1	1	10	**A**
3	3	1	1	1	20	**A**
4	3	1	1	5	1	**C**
5	3	1	1	5	10	**C**
6	3	1	1	5	20	**C**
7	3	1	1	15	1	**D**
8	3	1	1	15	10	**D**
9	3	1	1	15	20	**D**
10	3	1	5	1	1	**B**
11	3	1	5	1	10	**B**
12	3	1	5	1	20	**B**
13	3	1	5	5	1	**C**
14	3	1	5	5	10	**C**
15	3	1	5	5	20	**C**
16	3	1	5	15	1	**C**
17	3	1	5	15	10	**C**
18	3	1	5	15	20	**C**
19	3	1	10	1	1	**B**
20	3	1	10	1	10	**B**
21	3	1	10	1	20	**B**
22	3	1	10	5	1	**C**
23	3	1	10	5	10	**C**
24	3	1	10	5	20	**C**
25	3	1	10	15	1	**C**
26	3	1	10	15	10	**C**
27	3	1	10	15	20	**C**
28	3	1	1	1	1	**C**
29	3	1	1	5	10	**C**
30	3	1	1	10	20	**C**

Se aplicó en cada una de las corridas los 9 pares <algoritmo de agrupamiento, algoritmo de inducción> que ya se mencionaron en el presente trabajo. A cada dominio generado se le aplicó el proceso de descubrimiento de reglas de pertenencia a grupos utilizando cada uno de los nueve pares <algoritmo de agrupamiento, algoritmo de inducción>.

ISBN 978-958-46-0577-1
"POR LA PROFESIONALIZACIÓN DEL DESARROLLO DE SOFTWARE"
Medellín – Colombia / 13 y 14 de julio de 2012

Se analizaron los resultados obtenidos para los distintos tipos de dominios: mediano, oscilante, complejo e hipercomplejo y se verificó el porcentaje de reglas correctamente cubiertas para cada una de las combinaciones posibles de variables. Se utilizaron Diagramas de Kiviat [18] para graficar los resultados [8], lo que se muestra en la Figura 3.

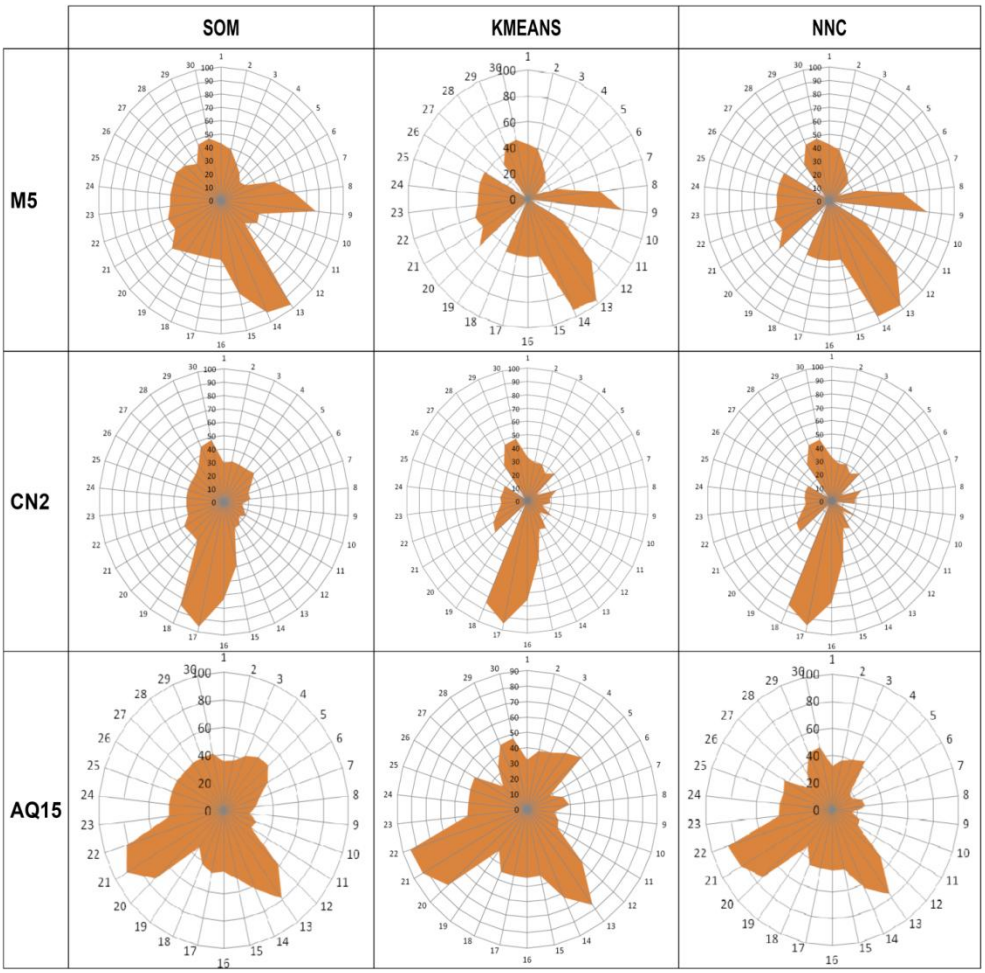

Fig. 3: Resultados descriptos en términos de Diagramas de Kiviat

Como resultado del análisis sobre los resultados de la experimentación emergieron las siguientes proposiciones experimentales:

1. Proposición Experimental 1: Para dominios de tipo mediano (pertenecientes al grupo A), se verificó experimentalmente que la combinación que ofrece mejor cubrimiento de reglas es el par SOM VS AQ15, con un valor cercano al 86 por ciento en promedio. Esto verifica la Proposición Experimental 1.
2. Proposición Experimental 2: Para dominios de tipo hipercomplejo (pertenecientes al grupo B), se verificó experimentalmente que la combinación que ofrece mejor cubrimiento de reglas es también SOM VS AQ15,, pero con un rendimiento más bajo, cercano al 65 por ciento en promedio. Esto verifica la Proposición Experimental 2.
3. Proposición Experimental 3: Para dominios de tipo complejo (pertenecientes al grupo C), se verificó experimentalmente que la combinación que ofrece mejor cubrimiento de reglas es SOM VS AQ15, con 49 por ciento como valor promedio. Esto verifica la Proposición Experimental 3.
4. Proposición Experimental 4: Para dominios de tipo oscilante (pertenecientes al grupo D), se verificó experimentalmente que la combinación que ofrece mejor cubrimiento de reglas es SOM VS M5, con 60 por ciento en promedio. Esto verifica la Proposición Experimental 4.
5. Proposición Experimental 5: Se observa y se determina experimentalmente que las características del dominio subyacente tienen influencia sobre el resultado experimental obtenido.
6. Proposición General: Se observa también que los valores obtenidos experimentalmente se ajustan sobremanera a lo que se postuló a priori en la clasificación teórica realizada a los distintos dominios según su complejidad.
7. En relación al rendimiento en términos de cubrimiento de reglas de los algoritmos de agrupamiento e inducción combinados se han obtenidos resultados que falsean la suposición que la performance de los algoritmos utilizados para agrupamiento e inducción es independiente de la complejidad del dominio.

ISBN 978-958-46-0577-1
"POR LA PROFESIONALIZACIÓN DEL DESARROLLO DE SOFTWARE"
Medellín – Colombia / 13 y 14 de julio de 2012

La Tabla 2 muestra los resultados para los Dominios Medianos, resultando la mejor combinación (con más cubrimiento) AQ15 y SOM.

Tabla 2. Resultados de mejor cubrimiento para los Dominios Medianos

DOMINIOS MEDIANOS	SOM	KMEANS	NNC
M5	37,34	37,53	37,19
CN2	30,46	30,57	30,29
AQ15	**38,26**	37,05	36,61

La Tabla 3 muestra los resultados para los Dominios Complejos, resultando la mejor combinación (con más cubrimiento) AQ15 y SOM.

Tabla 3. Resultados de mejor cubrimiento para los Dominios Complejos

DOMINIOS COMPLEJOS	SOM	KMEANS	NNC
M5	39,55	34,46	34,65
CN2	26,86	20,05	17,95
AQ15	**51,10**	47,96	47,76

La Tabla 4 muestra los resultados para los Dominios Hiper Complejos, resultando la mejor combinación (con más cubrimiento) AQ15 y SOM.

Tabla 4. Resultados de mejor cubrimiento para los Dominios Hiper Complejos

DOMINIOS HIPER COMPLEJOS	SOM	KMEANS	NNC
M5	46,74	42,46	42,58
CN2	41,12	36,23	36,22
AQ15	**49,02**	46,09	44,59

La Tabla 5 muestra los resultados para los Dominios Osciantes, resultando la mejor combinación (con más cubrimiento) M5 y SOM.

Tabla 5. Resultados de mejor cubrimiento para los Dominios Oscilantes

DOMINIOS OSCILANTES	SOM	KMEANS	NNC
M5	**60,68**	54,75	53,84
CN2	18,88	21,01	20,96
AQ15	25,44	24,65	22,04

Los resultados mostrados en las Tabla anteriores se pueden resumir en la Tabla 6, que es la combinación de los algoritmos estudiados que mejor resulta (reglas generadas con mayor cubrimiento) en función de la complejidad de cada dominio.

Tabla 6. Rendimiento de los algoritmos en términos del cubrimiento de las reglas que generan

	SOM	KMEANS	NNC
M5	Oscilantes	Oscilantes	Oscilantes
CN2	Complejos	Complejos	Complejos
AQ15	Complejos	Hiper Complejos	Hiper Complejos

7. CONCLUSIONES

Ha quedado demostrado empíricamente que la suposición sobre que la performance de los algoritmos utilizados para agrupamiento e inducción es independiente de la complejidad del dominio es falsa.

Se ha encontrado que: [a] para Dominios Medianos, Complejos e Hipercomplejos la combinación de algoritmos con más cubrimiento es la de AQ15 y SOM; y [b] para Dominios Oscilantes la combinación de algoritmos con más cubrimiento es la de M5 y SOM. Con lo expuesto se logra identificar el par <algoritmo de agrupamiento, algoritmo de inducción> que mejor performance proporciona dada la complejidad del dominio.

Se observa que con independencia del algoritmo de agrupamiento, el algoritmo de inducción M5 es el que mejor funciona para Dominios Oscilantes y el algoritmo CN2 es el que mejor funciona para Dominios Complejos.

Como futuras líneas de investigación se prevé extender en una primera etapa estos estudios a la combinación de otros algoritmos de agrupamiento e inducción; y en una segunda etapa a otros procesos de explotación de información.

8. AGRADECIMIENTOS

Las investigaciones que se reportan en este artículo han sido financiadas parcialmente por los proyectos UNLa 33A105 y UNLa 33B102 de la Universidad Nacional de Lanús.

9. REFERENCIAS

[1] Chen, M.; Han, J. & Yu, P. 1996. Data Mining: An Overview from a Database Perspective. IEEE Transactions on Knowledge and Data Engineering, 8(6), 866-883.

[2] Chung, W.; Chen, H. & Nunamaker, J. 2005. A Visual Framework for Knowledge Discovery on the Web: An Empirical Study of Business Intelligence Exploration. Journal of Management Information Systems, 21(4), 57-84.

[3] Chau, M. et al. 2007. Redips: Backlink Search and Analysis on the Web for Business Intelligence Analysis. Journal of the American Society for Information Science and Technology, 58(3), 351-365.

[4] Golfarelli, M.; Rizzi, S. & Cella, L. 2004. Beyond data warehousing: what's next in business intelligence? Proceedings 7th ACM international workshop on Data warehousing and OLAP, 1-6.

[5] Koubarakis, M. & Plexousakis, D. 2000. A Formal Model for Business Process Modeling and Design. Lecture Notes in Computer Science, 1789, 142-156.

[6] Britos, P. 2008. Procesos de explotación de información basados sobre sistemas inteligentes. Tesis presentada para obtener el grado de Doctor en Ciencias Informáticas, Facultad de Informática, Universidad Nacional de La Plata, Argentina.

[7] Britos, P. & García-Martínez, R. 2009. Propuesta de Procesos de Explotación de Información. Proceedings XV Congreso Argentino de Ciencias de la Computación. Workshop de Base de Datos y Minería de Datos, 1041-1050.

ISBN 978-958-46-0577-1
"POR LA PROFESIONALIZACIÓN DEL DESARROLLO DE SOFTWARE"
Medellín – Colombia / 13 y 14 de julio de 2012

[8] Lopez-Nocera, M. et al. 2011. Un Protocolo de Caracterización Empírica de Dominios para Uso en Explotación de Información. Proceedings XVII Congreso Argentino de Ciencias de la Computación, pp. 1047-1055.

[9] Rancan, C.; Pesado, P. & García-Martínez, R. 2010. Issues in Rule Based Knowledge Discovering Process. Advances and Applications in Statistical Sciences Journal, 2(2), 303-314.

[10] Britos, P. et al. 2005. Minería de Datos Basada en Sistemas Inteligentes. Editorial Nueva Librería.

[11] Kogan, A. et al. 2007. Algunos resultados experimentales de la integración de agrupamiento e inducción como método de descubrimiento de conocimiento. Proceedings IX Workshop de Investigadores en Ciencias de la Computación, 11-15. Universidad Nacional de la Patagonia San Juan Bosco, Trelew, Argentina.

[12] Kohonen, T. (2001). Self-Organizing Maps. Springer series in information sciences. Ed Springer, Helsinki University of Technology Neural Networks Research Centre P.O. Box 5400 02, 3rd Edition.

[13] McQueen, J. 1967. Some Methods for Classification and Analysis of Multivariate Observation. 5th Berkeley Proceedings of Simposium on Mathematics, Statistics and Probability, 1, 281-297.

[14] Yang, Y. 1999. An evaluation of statistical approaches to text categorization. Information Retrieval, 1(1-2), 69–90.

[15] Witten, I. H. & Frank, E. 2005. Data Mining Practical Machine Learning Tools and Techniques. Ed. Morgan Kaufmann.

[16] Clark, P. & Niblett, T. The CN2 Induction Algorithm. Machine Learning, 3(4), 261-283.

[17] Michalski, R. et al. 1986. The Multi-Purpose Incremental Learning System AQ15 and Its Testing Application to Three Medical Domains. Proceedings of AAAI-86, 1041-1045.

[18] Soman, K. P.; Diwakar, S. & Ajay, V. 2006. Insight into Data Mining: Theory and Practice. Prentice-Hall.