Desarrollo de una plataforma astroinformática para la administración y análisis inteligente de datos a gran escala

Reglas de Asociación para Líneas Moleculares

Nicolás Miranda,
Mauricio Solar, Marcelo Mendoza, Jonathan Antognini, Walter Fariña,
Jorge Ibsen, Lars Nyman, Eduardo Vera, Diego Mardones, Guillermo Cabrera,
Paola Arellano, Karim Pichara, Nelson Padilla, Ricardo Contreras,
Neil Nagar, Victor Parada.

Universidad de Chile, Santiago, 30 de agosto de 2014

Resumen

Dentro del ámbito de la espectroscopía astronómica, los cubos de datos tipo ALMA ofrecen nuevas oportunidades para realizar detección de lineas espectrales de manera automática. Estos poseen grandes cantidades de pixeles presentes en cada observación, en diferentes rangos de frecuencia dentro de las bandas milimétricas/sub-milimétricas. Esto último supone nuevos desafíos para las herramientas existentes de minería de datos. El presente trabajo apunta a estudiar y aplicar una de estas herramientas, el aprendizaje mediante reglas de asociación, a los cubos de datos tipo ALMA con el fin de encontrar nuevas relaciones entre líneas desconocidas y transiciones moleculares. En este documento, en particular, se presenta el estado del arte en aprendizaje mediante reglas de asociación y los requerimientos del software a desarrollar en el proyecto.

Palabras Claves: Data Mining, Association Analysis, Association Rules, Unsupervised Learning.

1. Resumen Ejecutivo

En este proyecto se busca implementar algoritmos de *Association Rule Learning (ARL)* con el fin de realizar asociaciones entre transiciones moleculares presentes en espectros de frecuencia obtenidos a partir de cubos de datos tipo *ALMA*. De esta forma, se podrá, entre otras cosas, asociar transiciones previamente no identificadas a moleculas conocidas, agrupar moleculas relacionadas en lugares del espacio en particular, y facilitar la identificación de líneas moleculares desconocidas.

Esta aplicación se encuentra diseñada para ser ejecutada mediante una interfaz de línea de comandos. Mediante esta el usuario puede ingresar una lista de conjuntos de líneas espectrales, y además especificar parámetros opcionales con el fin de obtener información más detallada. La aplicación además incluye módulos para realizar evaluaciones del desempeño de los algoritmos de ARL y para la visualización de la información relacionada con este.

Índice

Índice de figuras

Índice de cuadros

2. Metodología de Trabajo

Durante el transcurso del trabajo se han estudiado distintos algoritmos de ARL, y se está llevando a cabo un proceso de evaluación de los mismos. Estos se han implementado y posteriormente se experimentará, en particular, con aquellos que mejor se acomoden a los casos de uso en el contexto de una plataforma en la cual se opera sobre datos (observaciones) provenientes de distintas fuentes (i.e. distintas regiones del espacio, distintas bandas de frecuencia, etc.). De estos, se seleccionarán aquellos que muestren un mejor desempeño y eficiencia en grandes volúmenes de datos.

En primera instancia, para probar los algoritmos de ARL subyacentes se han utilizado conjuntos de líneas correspondientes a espectros de frecuencia simulados; con datos generados de manera pseudo-aleatoria y otros generados bajo condiciones específicas que por sus características (como el tipo de moléculas presentes) aseguren similitud con datos reales obtenidos a partir de la observación de alguna región del espacio.

Una vez que se comparen los resultados utilizando distintas métricas definidas con anterioridad, los algoritmos pasarán a ser probados sobre datos provenientes de cubos de datos tipo ALMA reales, publicados por ALMA. Cabe, además, la posibilidad de que previamente a esto se prueben los algoritmos sobre espectros extraídos desde bandas de frecuencias y observaciones distintas de las de ALMA; como, por ejemplo, las del *Sloan Digital Sky Survey (SDSS)*[YAAJ$^+$00]. Si bien estos datos son obtenidos a partir de observaciones en el espectro óptico visible de frecuencias, poseen la ventaja de ser muy masivos, estar muy bien documentados, estudiados, y de estar desde ya disponibles para el público en general. Además, los algoritmos de ARL deberían ser, en principio, lo suficientemente independientes de las características de los datos como para funcionar de forma similar a como lo harían con los espectros de ALMA.

A continuación se presenta el plan de trabajo general de este proyecto:

1. **Datos de prueba**
   - *a*) Generar datos de prueba simulados
   - *b*) Investigar sobre posibles fuentes de datos reales (ALMA y otras fuentes)
   - *c*) Seleccionar y pre-procesar los datos reales según corresponda
2. **Algoritmos de ARL**
   - *a*) Estudiar algoritmos de ARL
   - *b*) Seleccionar aquellos que se ajusten más a los requerimientos teóricos y la ontología del problema
   - *c*) Implementar algoritmos de ARL seleccionados
   - *d*) Evaluar eficiencia de algoritmos sobre datos de prueba utilizando métricas predefinidas
   - *e*) Seleccionar algoritmos con mejor desempeño e incluirlos en una versión preliminar del sistema
3. **Testing**
   - *a*) Probar algoritmos sobre datos simulados
   - *b*) Probar algoritmos sobre datos reales
   - *c*) Evaluar la calidad de las reglas obtenidas y generar un reporte.

2.1. Trabajo realizado

Hasta el momento se ha logrado asimilar nociones básicas dentro del dominio de la espectroscopía astronómica, que sin lugar a dudas son de suma importancia a la hora de generar una solución que logre generar valor a los usuarios finales del sistema.

Junto con esto, se ha llevado a cabo una investigación detallada de los distintos algoritmos de Aprendizaje por Reglas de Asociación, la teoría matemática y estadística subyacente, y de múltiples aplicaciones en los más diversos ámbitos; con el fin de encontrar soluciones, y metodologías de evaluación de las mismas, que se adapten al problema de este trabajo.

Actualmente se encuentra implementado el algoritmo *Apriori* de análisis de reglas de asociación en el lenguaje de programación *Python*[pyt14]. Deben aún realizarse optimizaciones que permitan ejecutarlo en tiempos razonables sobre 100 o más espectros, dado que actualmente toma más de 20 minutos. Entre ellas se encuentran el utilizar el framework *NumPy*[num14] y hacer uso de matrices binarias, junto con otras posibles optimizaciones.

Aun así, se ha logrado efectuar pruebas preliminares de este algoritmo sobre una cantidad reducida de datos generados artificialmente, de la siguiente forma:

1. Se selecciona una rango de frecuencias y se obtiene a partir del catálogo SPLATALOGUE todas las líneas que se encuentren dentro de este.
2. Se selecciona al azar una molécula o átomo en particular dentro de esta banda, y se extraen todas las lineas que correspondan a este.
3. Se define un soporte a priori y un número de líneas por espectro.
4. Se generan espectros seleccionando al azar líneas de transición dentro del rango de frecuencias. Se asegura, también que cierto porcentaje de los espectros contengan todas las líneas del átomo o molécula seleccionado con el fin de cumplir con el soporte definido a priori.
5. Se ejecuta el algoritmo Apriori sobre los espectros generados y se evalúan los resultados.

Los resultados obtenidos han sido analizados, mediante medidas estadísticas sencillas, y graficados con el fin de tener una noción, en primera instancia, del desempeño del algoritmo. Por supuesto, tanto la forma de generar datos como las herramientas de análisis serán refinadas a futuro, a lo largo del presente trabajo.

También, se han estudiado posibles fuentes de datos que, posteriormente, podrían ser un contraparte valioso a los datos que ALMA haga disponible al público, y que desde ya son súmamente útiles a la hora de efectuar pruebas con los algoritmos de ARL.

2.1.1. Especificación de Requerimientos

El propósito general del sistema a desarrollar en el presente proyecto es realizar la asociación de líneas de transición de distintas moléculas, tomando como fuente de datos los conjuntos de líneas de transición

ya identificadas, presentes en espectros de frecuencia de diversas bandas (aunque más comunmente en frecuencias milimétricas y sub-milimétricas).

El sistema deberá ser capaz, junto con esto, de generar conjuntos de transiciones moleculares, tanto de manera aleatoria como a partir de descriptores que impongan ciertas restricciones al tipo de moléculas presentes en un cierto espectro de frecuencias.

Estos conjuntos de frecuencias presentes en un espectro serán almacenados en estructuras de datos de tipo vector. Estos vectores serán entregados al sistema, el cual ejecutará los algoritmos de ARL sobre estos; identificando cada vector como una transacción y cada transición molecular como un ítem.

En caso de llegar a desarrollar interfaces de usuario, estas deben ser lo suficientemente sencillas como para realizar de forma directa las operaciones más básicas por parte del usuario final; que son, especificar una o varias fuentes de datos de espectros de frecuencia, y seleccionar diversos parámetros que especifiquen el tipo de reglas de asociacion que sean de mayor interés. Es, además, deseable que los resultados sean mostrados de forma clara e intuitiva una vez obtenidos, de tal manera que el usuario pueda refinar los parámetros y realizar una nueva busqueda de manera simple, en caso de no ser los resultados esperados.

En la Figura 1 se muestran los casos de uso preliminares del sistema a desarrollar.

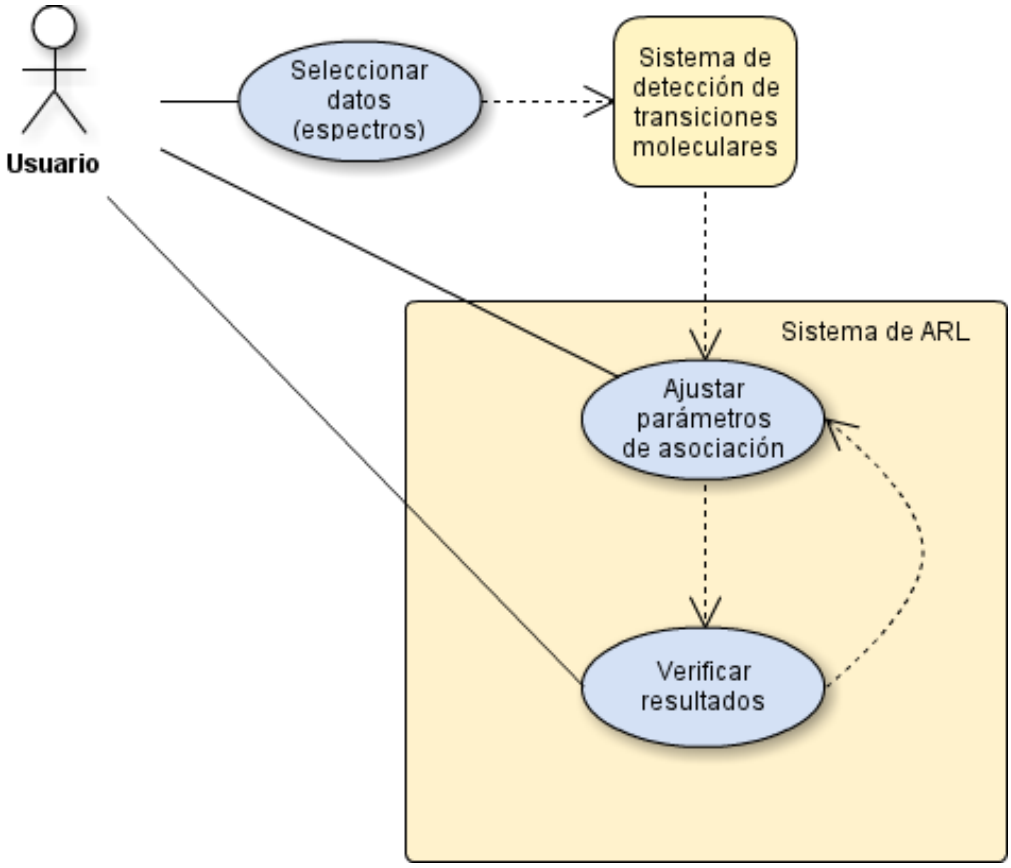

Figura 1: Casos de uso del sistema.

En cuanto a eficiencia, se espera que los algoritmos subyacentes sean, en lo posible, lineales en el número

de transacciones, y que escalen de manera acotada con grandes cantidades de datos. Debe minimizarse, también, el número de lecturas y peticiones a la base de datos. Es muy deseable que los algoritmos sean paralelizables con el fin de efectuarse en arquitecturas distribuidas o concurrentes.

Específicamente, el sistema en general será desarrollado y los algoritmos subyacentes implementados en el lenguaje de programación *Python*, debido a que esto facilirará la integración en los servicios del futuro *Chilean Virtual Observatory, ChiVO*. Se utilizará, además una base de datos MySQL para almacenar los datos del catálogo de líneas moleculares obtenido a partir de SPLATALOGUE[RMK08].

3. Prototipo de solución

3.1. Solución Propuesta

La solución consiste de una aplicación con interfaz por línea de comandos. En su versión preliminar, esta debe recibir como entrada del usuario un archivo de tipo *CSV* (*Comma Separated Values*) que contenga los espectros correspondientes a una observación en particular. Es deseable, además que el usuario pueda ingresar múltiples archivos con el fin de extraer relaciones entre líneas espectrales a partir de diversas observaciones a la vez.

Una vez ingresados los datos del usuario, la aplicación procederá a ejecutar el algoritmo de ARL sobre los espectros y a entregar al usuario mediante la misma interfaz los resultados obtenidos. Los resultados entregados por el algoritmo pueden ser vistos en detalle por el usuario si este así lo desea, entregando los parámetros correspondientes mediante línea de comando. De ser así, una vez obtenidos los resultados del algoritmo, la aplicación procederá a recopilar la información de las líneas obtenidas a partir de la base de datos.

3.2. Diseño del Prototipo

La interfaz de usuario de la aplicación es mediante línea de comando. En esta, el usuario especifica el archivo que contiene los espectros correspondientes en formato *CSV*. Cada línea del archivo debe corresponder a un espectro de frecuencias, y cada una de estas debe contener las frecuencias absoluta, en gigahertz (*GHz*), de cada una de las líneas de emisión separadas por comas; como se muestra en el siguiente ejemplo:

```
84.00110,84.01081,84.02501,84.05221
84.00986,84.02088
84.02853,84.04412,84.05202
```

Junto con esto, el usuario además debe especificar mediante flags (parámetros opcionales) si desea que las reglas de asociación resultantes se entreguen con información detallada obtenida a partir de la base de datos de líneas espectrales.

Todos los datos y parámetros entregados por el usuario mediante línea de comando son recibidos, parseados y procesados por el módulo controlador. Posteriormente, estos datos son entregados en sus estructuras adecuadas al módulo principal encargado de ejecutar el algoritmo de *ARL*. Una vez obtenidos los resultados, estos son entregados al módulo de acceso a la base de datos con el fin de extraer cualquier información adicional requerida por el usuario. Finalmente, las reglas de asociación resultantes y la información adicional recopilada sobre las líneas espectrales que las componen son entregadas al módulo de interfaz gráfica, el cual los entrega al usuario mediante imprimirlos en pantalla o a archivo.

Junto con esto, se dispone de un sistema interno de testeo; que contiene módulos de generación de datos de prueba y de gráficos de análisis de desempeño de los algoritmos utilizados.

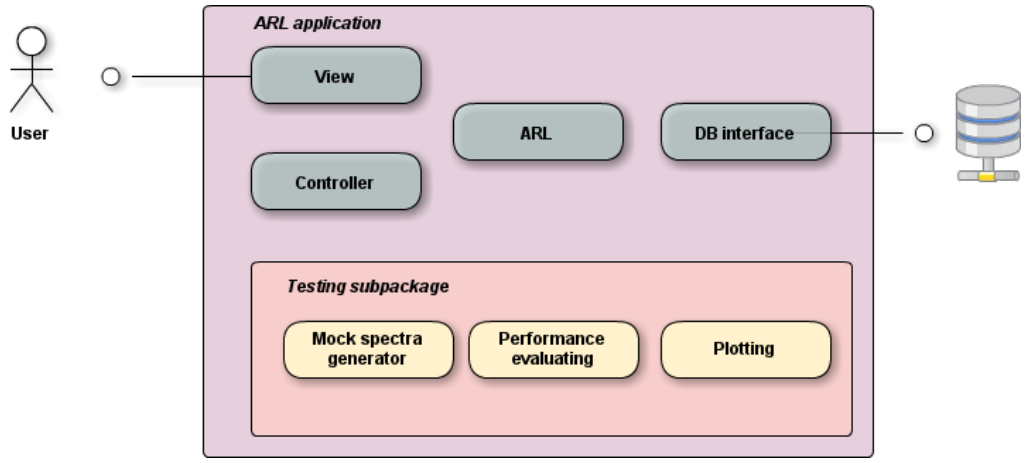

Figura 2: Diagrama de diseño del sistema. Se pueden apreciar en particular los módulos principales de la aplicación y los del sistema de testeo

3.3. Implementación

La implementación de la aplicación, sus módulos y sus algoritmos subyacentes se llevan a cabo en el lenguaje de programación *Python*, haciendo uso de librerías tales como *NumPy* y *SciPy* (que contienen estructuras y algoritmos óptimos para datos de tipo numérico), junto con *argparse* (para la interfaz de usuario) y *PyMySQL* (para acceso a la base de datos), entre otros.

4. Anexos

Referencias

[num14] Numpy – numpy. `http://www.sdss.org`, 2014. online, accesed July 2014.

[pyt14] Welcome to python.org. `http://www.python.org`, 2014. online, accesed July 2014.

[RMK08] Anthony J Remijan and Andrew J Markwick-Kemper. Splatalogue: Database for astronomical spectroscopy. 2008.

[YAAJ+00] Donald G York, J Adelman, John E Anderson Jr, Scott F Anderson, James Annis, Neta A Bahcall, JA Bakken, Robert Barkhouser, Steven Bastian, Eileen Berman, et al. The sloan digital sky survey: Technical summary. *The Astronomical Journal*, 120(3):1579, 2000.