Construcción de una Vista Minable para aplicar Mineria de Datos Secuenciales Temporales

Oscar Eduardo Quinteros[1], Ana Funes[2], Hernán César Ahumada[1]

[1] Facultad de Tecnologías y Ciencias Aplicadas - Universidad Nacional de Catamarca, Catamarca, Argentina,
{oequinteros,hcahumada}@tecno.unca.edu.ar
[2] Universidad Nacional de San Luis
San Luis, Argentina
afunes@unsl.edu.ar

Resumen Sobre datos de exámenes rendidos y aprobados de las materias del Ciclo Común de Articulación (CCA) de las carreras de Ingeniería de la Facultad de Tecnología y Ciencias Aplicadas, se propone conformar la vista minable apropiada para la aplicación de métodos de minería de secuencias temporales, como parte de en un proyecto de extracción de conocimiento. El proceso de elaboración de esta vista minable se lleva a cabo siguiendo las actividades de la fase de selección y preparación de datos, según la metodología CRISP-DM. Una vez definidos los elementos de la secuencia, Identificador, Tiempo y Evento, se genera una vista minable y se realiza un estudio de frecuencias en las secuencias de aprobación de materias.

Keywords: Minería de Datos, Minería de Secuencia Temporal, Vista Minable, Metodología CRISP-DM, Frecuencias de secuencias

1. Introducción

La Minería de Secuencias [1] consiste en la búsqueda de patrones secuenciales frecuentes en una base de datos de eventos con fecha y hora [9] [4]. Existen además, diversos métodos para minar secuencias. Ahola [2] explicita varios, tales como SPADE [12], o PrefixSpan [8], entre otros. Como antecedente en la aplicación de minar secuencias sobre datos provenientes de entornos educativos se tiene el trabajo de Guerra [5], donde usan los métodos SPAM [3] y PexSPAM [6] para descubrir patrones comunes en secuencias de ejercicios parametrizados usados en herramientas de aprendizaje on-line.

Los algoritmos de minería de secuencias, requieren hacer una transformación de los datos para llevarlos al formato adecuado. Una *Vista Minable* es la consolidación en una única tabla de todas las observaciones y los atributos sobre los que se aplicarán los algoritmos de minería de datos.

La metodología CRISP-DM (CRoss Industry Standard Process for Data Mining) [11], respecto a otras metodogías, posee ventajas comparativas según Moine [7]. CRISP-DM es una metodología para el descubrimiento de conocimiento

en bases de datos, estructurada en un proceso jerárquico, compuesto por tareas descriptas en cuatro niveles diferentes de abstracción, que van desde lo general a lo específico. CRISP-DM propone, en el nivel más alto, seis fases para el proceso de minería de datos: *1.Entendimiento del negocio*, *2.Entendimiento de los datos*, *3.Selección y Preparación de los datos*, *4.Modelado*, *5.Evaluación* y *6.Implementación*. Cada fase plantea tareas generales que se proyectan a tareas específicas, que describen las acciones que deben ser desarrolladas para situaciones específicas en procesos de extracción de conocimiento a partir de datos.

Las fases 1 y 2 fueron planteadas en el trabajo "Extracción de Conocimiento en el Cursado del Ciclo Común de Articulación de Carreras de Ingeniería"[10] con el objetivo de caracterizar y analizar el recorrido académico de exámenes rendidos por alumnos del Ciclo Común de Articulación (CCA) de las Carreras de Ingeniería en la Facultad de Tecnologías y Ciencias Aplicadas (FTyCA) de la Universidad Nacional de Catamarca (U.N.CA.)

En el presente trabajo se lleva a cabo la fase *3.Selección y Preparación de Datos*, que consta de dos sub-fases: *Comprensión de los Datos* (Apartado 3.1) y *Preparación de los Datos* (Apartado 3.2). En la primera de ellas se realizan las actividades de *Recolección de datos iniciales*, *Descripción de los datos* y *Exploración inicial de los datos*. En la segunda, se ejecutan las actividades de *Seleccionar los datos*, *Limpieza de los datos*, *Construcción de los datos*. Se busca de este modo, a partir de registros de actas de exámenes, construir la vista minable que sea apta para ser procesada mediante técnicas de minería de secuencias.

En la Sección 4 se explica el proceso de transformación de los datos para generar la vista minable. Luego, en la Sección 5, se muestran resultados del análisis frecuencial sobre la vista minable. Finalmente, la Sección 6 cierra el trabajo con las conclusiones y trabajo futuro.

2. Minería de Secuencias Temporales

La Minería de Secuencias Temporales consiste en encontrar patrones de secuencias, generalmente bajo la forma de asociaciones del tipo: *cuando ocurre A, entonces ocurre B dentro de algún lapso de tiempo*[2]. El proceso de descubrir patrones secuenciales involucra dos etapas: representar las secuencias y la aplicación del algoritmo que encontrará patrones frecuentes en las secuencias.

La formulación del problema de minería de secuencias frecuentes involucra los siguientes elementos básicos [12]:

- Alfabeto: conjunto de ítems. $I = \{i_1, i_2; ...; i_m\}$
- Evento: n-upla no ordenada de ítems. $\alpha_i = (i_1; i_2; ...; i_n)$
- Secuencia: lista ordenada de eventos $\alpha = (\alpha_1 \rightarrow \alpha_2 \rightarrow ... \rightarrow \alpha_k)$

Una secuencia α está compuesta de uno o más eventos α_i que a su vez incluyen n ítems del alfabeto I.

El tamaño de un evento está dado por la cantidad n de ítems que lo integran.

El ancho de la secuencia lo determina el máximo tamaño de los eventos α_i que componen dicha secuencia. El largo de una secuencia está determinado por la cantidad k de eventos que componen la misma.

El primer paso para aplicar un método de minería de secuencias es representar los datos a procesar bajo el formato de *transacciones*. Una transacción o secuencia se identifica mediante un número (Id de transacción) y está compuesta por una serie de eventos.

Las transacciones o secuencias temporales incorporan además información sobre el momento de ocurrencia de cada evento (variable tiempo). Por ello en una secuencia temporal, la serie de eventos se ordena según la variable tiempo y se consigna el tiempo de ocurrencia del evento.

El total de transacciones constituye la base de datos D para realizar la minería de secuencias. Por lo tanto D es una colección de secuencias de entrada. Cada secuencia de entrada tiene un identificador único (S_{id}) y, a su vez, cada evento de una secuencia tiene también un identificador único (E_{id}). Además, cada evento de una secuencia tiene asociado un valor de la variable tiempo ($t(E_{id})$), donde no puede haber 2 eventos con el mismo valor de tiempo. Por lo tanto, $t(E_{id})$ se puede usar como identificador del evento. Dentro de una transacción, los eventos se ordenan de manera ascendente según la variable tiempo. Es decir, si el evento α_i ocurre antes que el evento α_j, entonces $t(\alpha_i) < t(\alpha_j)$.

En el presente trabajo, se obtiene el conjunto de datos inicial y se lo procesa para conformar un conjunto D de secuencias de exámenes aprobados que tenga las variables (S_{id}), (E_{id}) y $t(E_{id})$, que requiere una vista minable de minería de secuencia temporal. Luego, como primer estudio de las secuencias, se realizan diversos análisis frecuenciales sobre el conjunto de secuencias D.

3. Selección y Preparación de Datos

3.1. Comprensión de los Datos

Esta fase de comprensión de los datos, comprende la recolección inicial de datos, con el objetivo de establecer un primer contacto con el problema, familiarizándose con ellos, identificar su calidad y establecer las relaciones más evidentes que permitan definir futuras acciones.

3.1.1. Recolección de datos iniciales. Se definió como unidad observacional los exámenes rendidos por alumnos de las carreras de Ingeniería de la Facultad de Tecnología y Ciencias Aplicadas, particularmente los exámenes correspondientes a las materias del CCA, acotando la consulta de información a los exámenes hasta el 02/06/2016 de alumnos de las cohortes entre los años 2004 y 2013 inclusive.

3.1.2. Descripción de los datos La información extraída del sistema SIU-GUARNI fue presentada en un archivo en formato Microsoft Excel con la totalidad de exámenes rendidos por los alumnos de las carreras de ingeniería.La consulta consta de 14166 registros (exámenes), con los siguientes campos:

- Carrera: el nombre de la carrera de Ingeniería que se dicta en la FTyCA.

- IdAlumno: el número de legajo que identifica unívocamente a cada alumno.
- Cohorte: el año de ingreso a la Carrera del alumno.
- Materia: Código alfanumérico que identifica a cada materia del CCA.
- Fecha: fecha en la cual el alumno ha rendido el examen de la materia.
- Resultado: resultado evaluatorio que ha obtenido el alumno en un examen.
- Forma: forma en la que se ha tomado el examen ya sea rindiendo el mismo o solicitando equivalencia.

En la tabla 1 se puede apreciar la descripción de las variables extraídas.

Tabla 1. Descripción de las variables extraídas

Atributo	**Tipo**	**Valores Posibles**
Carrera	Categórica	[MINAS,ELECTRONICA,AGRIMENSURA,INFORMATICA]
IdAlumno	Entero	
Cohorte	Categórica	2004-2005-2006-2007-2008-2009-2010-2011-2012-2013
Materia	Categórica	M1-M2-M3-M4-M5-M6-M7-M8-M9-M10-M11
Fecha	Fecha	[01/01/2004 al 31/12/2013]
Resultado	Categórica	[APROBADO - AUSENTE - REPROBADO]
Forma	Categórica	[EXAMEN - EQUIVALENCIA]

3.1.3. Exploración inicial de los datos. Los datos recolectados se han explorado con el software WEKA para observar las distribuciones de las variables categóricas. No se contabilizaron frecuencias de las variables *"Fecha"* y *"IdAlumno"*. La variable *"Fecha"* no se evaluará en esta etapa porque más adelante será transformada. La variable *"IdAlumno"* no se considera porque no aporta información al problema de estudio, por ser sólo un identificador.

De los 14436 registros de exámenes, se tiene que en 5660 de ellos el **Resultado** es *APROBADO*, en 3991 es *REPROBADO* y 4785 con *AUSENTE*.

En la Figura 1 se observa la distribución de las variables categóricas del conjunto de datos considerando como variable de clase a la variable *"Resultado"*. El bloque inferior representa los exámenes con resultado *Aprobado*, el bloque medio aquellos con resultado *Reprobado* y el bloque superior a los exámenes con resultado *Ausente*.

En la tabla 2 se detalla la cantidad de exámenes que existen por materia.

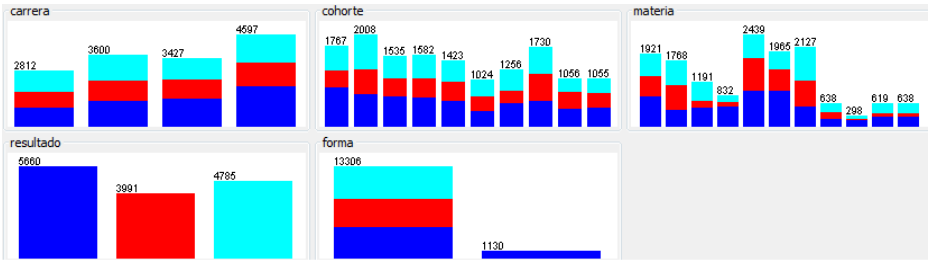

Figura 1. Distribución de las variables categóricas del conjunto de datos con variable de clase *"Resultado"*

Tabla 2. Cantidades de Exámenes por Materia

Materia	Codigo de Materia	Cantidad
FUNDAMENTOS DE INFORMÁTICA	M1	1921
QUÍMICA	M2	1768
FÍSICA I	M3	1191
SISTEMA DE REPRESENTACIÓN	M4	832
ÁLGEBRA	M5	2439
GEOMETRÍA ANALÍTICA	M6	1965
ANÁLISIS MATEMÁTICO I	M7	2127
ANÁLISIS MATEMÁTICO II	M8	638
CÁLCULO AVANZADO	M9	298
PROBABILIDAD Y ESTADÍSTICA	M10	619
FÍSICA II	M11	638

3.2. Preparación de los datos

En esta fase y una vez efectuada la recolección inicial de datos, se procede a su preparación para adaptarlos a las técnicas de Minería de Datos que se utilicen posteriormente.

3.2.1. Seleccionar los datos. Se descarta para este estudio el atributo *"Carrera"*, porque se considera que no aporta información en el análisis de secuencia deseado ya que el conjunto de materias del CCA es común en todas las carreras.

3.2.2. Limpieza de los datos. Se eliminaron todas las observaciones de aquellos alumnos que hubieran aprobado al menos una materia bajo la forma de aprobación *EQUIVALENCIA*. Los exámenes rendidos de esta forma no permiten determinar la fecha original de su aprobación. También se eliminaron las observaciones donde el resultado del examen fuera *Reprobado* o *Ausente*, debido a que el objeto de estudio es la secuencia de exámenes aprobados.

3.2.3. Construcción de los datos. Para poder aplicar los algoritmos de minería de secuencia temporal se necesita que la vista minable tenga un atributo que exprese el tiempo del evento. El atributo *"Fecha"* podría ser candidato para marcar el tiempo del evento, pero tiene la dificultad de que estamos considerando en el conjunto de datos a alumnos de diferentes cohortes con los cual sus exámenes no se han rendido en el mismo intervalo de tiempo. Resulta necesario normalizar los valores del atributo *"Fecha"* en una escala común a todos los alumnos, y para ello se decidió normalizar el valor de la variable *"Fecha"* sobre una variable que represente cuántos semestres académicos trancurrieron desde el ingreso a la carrera hasta la fecha del aprobación del examen. Como el año académico va de Abril a Marzo, se considera como un semestre académico al período Abril-Septiembre y el otro semestre comprende los meses Octubre-Marzo. Para determinar en qué semestre aprobó las diferentes materias cada alumno, se inicia en 1 el conteo a partir del año de ingreso a la carrera indicado el valor de la variable *"Cohorte"*. Luego, se incrementa en 1 por cada semestre que avanza en el tiempo de la vida académica del alumno. En la tabla 3 se muestra un ejemplo de la normalización de la fecha. En el caso particular del alumno 1 cuya cohorte es 2006 y fecha de examen 01/12/2006, el valor del semestre se calcula sumando la cantidad de semestres transcurridos desde 01/04/Cohorte hasta el semestre en el que corresponde la fecha de examen, en este caso es [Abril-2006 a Septiempre-2006] + [Octubre-2006 a Marzo-2007] = 2 semestres.

Tabla 3. Ejemplo de normalización de la variable *"Fecha"* en la variable *"Semestre"*

IdAlumno	**Cohorte**	**fecha del examen**	**Semestre**
1	2006	04/07/2006	1
1	2006	01/12/2006	2
2	2004	02/07/2004	1
2	2004	23/11/2005	4
2	2004	20/02/2006	4
3	2005	02/12/2005	2
3	2005	30/11/2006	4

4. Modelado de la Vista Minable

Luego de las etapas de limpieza y construcción de datos se ha obtenido un conjunto de datos a analizar de 4198 registros de exámenes aprobados. En la Tabla 4 se muestra un ejemplo del conjunto de datos así obtenido, donde las filas se ordenan según la fecha de examen.

Tabla 4. Ejemplo de representación de datos de actas de examen ordenadas por fecha de examen.

fecha del examen	semestre	IdAlumno	Materia
02/07/2004	1	2	M4, M5
23/11/2005	4	2	M1, M3
02/12/2005	2	3	M1
20/02/2006	4	2	M6
04/07/2006	1	1	M6
30/11/2006	4	3	M2, M7
01/12/2006	2	1	M3

Para generar la vista minable apta para la aplicación de minería de secuencias temporales, el siguiente paso consiste en agrupar para cada alumno, las actas de exámenes de aquellas materias que haya aprobado. Para ello, se reordenan las filas y columnas de la Tabla 4, quedando conformada la estructura que se muestra en Tabla 5.

Tabla 5. Ejemplo de detalle cronológico de materias aprobadas por cada alumno.

IdAlumno	Fecha del examen	Semestre	Materia
1	04/07/2006	1	M6
1	01/12/2006	2	M3
2	02/07/2004	1	M4, M5
2	23/11/2005	4	M1, M3
2	20/02/2006	4	M6
3	02/12/2005	2	M1
3	30/11/2006	4	M2, M7

Un conjunto de datos secuenciales se caracteriza por contar con tres columnas: *Objeto* (S_{id}), *Tiempo* $t(E_{id})$, *Ítems*. Se considera como variable *Tiempo del Evento* $t(E_{id})$ a la variable *Semestre* de la Tabla 5.

En la Tabla 6 se muestran las secuencias temporales construidas a partir de la Tabla 5, cuyo formato es el apropiado para ser procesado con algoritmos de *Minería de Secuencias*. Dicho formato no permite que una secuencia tenga eventos en un mismo tiempo. Por ello, para el alumno 2 que en la Tabla 5 tiene 2 eventos en el semestre 4, en la Tabla 6 ambos eventos se unifican.

Así la Tabla 6 tiene los elementos básicos descriptos en la Sección 2:

- **Identificador único de secuencia:** S_{id} establecido según el atributo *"Legajo"*.
- **Tiempo del evento:** $t(E_{id})$ es el atributo *"Semestre"* definido en el Apartado 3.2.3.
- **Ítems del evento:** representa la n-upla de ítems, atributo *"Materia"* aprobadas por un alumno en un semestre académico en particular.

Tabla 6. Ejemplo de secuencia temporal de materias aprobadas por cada alumno.

IdAlumno (S_{id})	**Tiempo del evento** $t(E_{id})$	**Ítems del evento**
1	1	$(M6)$
1	2	$(M3)$
2	1	$(M4, M5)$
2	4	$(M1, M3, M6)$
3	2	$(M1)$
3	3	$(M2, M7)$

5. Análisis Frecuencial de Secuencias Temporales

Para realizar un estudio de la frecuencias de eventos en la secuencia temporal, se utilizó el software *R*,ejecutado funciones del paquete *arulesSequences* sobre la vista minable construida previamente.

Aplicando la función *summary* a las transacciones generadas mediante la función *read_basket*, se obtiene la cantidad eventos según su tamaño, es decir la cantidad de ítems que componen cada evento. En la Tabla 7 se detalla la cantidad de veces que se aprobaron n materias en un semestre académico. La primera columna informa que 2078 veces se aprobó solo una materia en un semestre, mientras que la última columna expresa que 15 veces se aprobaron 6 materias en un semestre académico.

Tabla 7. Cantidad de eventos según la cantidad de ítems

Cant. Ítems	**1**	**2**	**3**	**4**	**5**	**6**
Cant. Eventos	2078	642	143	48	25	15

La Tabla 8, obtenida aplicando la función *timeFrequency*, muestra la cantidad de eventos ocurridos para cada valor de la variable tiempo. Representa para cada semestre académico en particular (valor de tiempo) cuántas veces se aprobó al menos una materia del CCA, por ejemplo durante el semestre 7 han ocurrido 144 eventos de aprobación de exámenes (cada evento incluye una o mas materias).

Tabla 8. Cantidad de eventos según valor de tiempo $t(E_{id})$

1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
402	684	283	410	174	262	144	177	81	99	44	48	32	36	16	23	11	10	7	2	4	1	1

La función *timeFrequency(vista_minable,"gaps")* calcula los valores de la Tabla 9, con la cantidad de veces que eventos consecutivos de las secuencias ocurren con una cierta brecha (*gap*) de tiempo (semestres). La brecha (gap) entre 2

eventos consecutivos es la diferencia en tiempo de los mismos [2]. En símbolos: $t(\alpha_j) - t(\alpha_{j-1})$. Por ejemplo, en 89 oportunidades la distancia o brecha entre exámenes aprobados es de 4 semestres.

Tabla 9. Cantidad de eventos cada cierto intervalo de tiempo

Brecha	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**13**	**14**	**15**	**16**
Eventos	1123	579	168	89	35	29	11	10	10	3	3	1	1	2	1

La función *timeFrequency(vista_minable, "span")* calcula los valores de la Tabla 10, que muestra la cantidad de secuencias (alumnos) según el lapso *(span)* de tiempo (semestres). El lapso es la duración neta (en semestres) del recorrido académico de aprobación de materias. El lapso (span) de una secuencia se determina restando el tiempo del último evento menos el tiempo del primer evento de la misma [2]. En símbolos: $span(S_k) = max(t(\alpha_j(S_k))) - min(t(\alpha_i(S_k)))$. Por ejemplo, son 66 las secuencias (alumnos) que entre la primera y la última materia aprobada tienen un lapso de 4 semestres. Con $lapso = 0$ hay 242 secuencias (alumnos), es decir que esas secuencias tienen un único evento.

Tabla 10. Cantidad de secuencias por lapso de tiempo

Lapso	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**22**
Secuencias	242	101	68	57	66	57	41	57	38	35	17	19	18	19	14	13	6	8	4	4	1	1

La Tabla 11 muestra la cantidad de secuencias (alumnos) según la cantidad de ítems de cada secuencia (exámenes). Por ejemplo con 11 exámenes aprobados existen 101 alumnos.

Tabla 11. Cantidad de secuencias por cantidad de ítems

Cant. Ítems	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Cant. Secuencias	166	134	129	77	66	49	56	49	40	19	101

6. Conclusiones y Trabajo Futuro

Siguiendo la metodología CRISP-DM se pudo construir una vista minable apta para algoritmos de minería de secuencias, en datos del contexto educativo.

El análisis frecuencial realizado a las secuencias permite conocer diferentes características del fenómeno de aprobación de materias del CCA. Lo más común es que se apruebe una sola materia en un semestre académico (Tabla 7). Los alumnos aprueban mayor cantidad de materias del CCA durante el segundo semestre de su historia académica (Tabla 8). Lo usual es que se aprueben materias

con un semestre de diferencia (Tabla 9). La mayoría de los alumnos registra un sólo evento de aprobación de materias (Tabla10). De los 886 alumnos considerados, sólo 101 (el 11 %) de ellos han aprobado las 11 materias del CCA (Tabla 11).

El resultado de este trabajo posibilita que a futuro se cumpla con las fases 4 y 5 de CRISP-DM, utilizando algoritmos específicos para encontrar patrones frecuentes de aprobación de materias y relacionarlos con el grado de avance en la carrera y los plazos que demanda aprobar las materias del CCA. De esta manera se espera contar con información estratégica y novedosa, la cual no es brindada por las herramientas habituales de informes y que se encuentra oculta en los datos y solo puede ser obtenida por técnicas de minería de datos.

Referencias

1. Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In *Data Engineering, 1995. Proceedings of the Eleventh International Conference on*, pages 3–14. IEEE, 1995.
2. Jussi Ahola. Mining sequential patterns. 2001.
3. Jay Ayres, Jason Flannick, Johannes Gehrke, and Tomi Yiu. Sequential pattern mining using a bitmap representation. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 429–435. ACM, 2002.
4. Dante Conti and Fco Javier Martínez de Pisón Ascacíbar. Reglas de asociación en series temporales: panorama referencial y tendencias.
5. Julio Guerra, Shaghayegh Sahebi, Yu-Ru Lin, and Peter Brusilovsky. The problem solving genome: Analyzing sequential patterns of student work with parameterized exercises. In *Educational Data Mining 2014*, 2014.
6. Joshua Ho, Lior Lukov, and Sanjay Chawla. Sequential pattern mining with constraints on large protein databases. In *Proceedings of the 12th International Conference on Management of Data (COMAD)*, pages 89–100, 2005.
7. Juan Miguel Moine. *Metodologías para el descubrimiento de conocimiento en bases de datos: un estudio comparativo*. PhD thesis, Facultad de Informática, 2013.
8. Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Mei-Chun Hsu. Mining sequential patterns by pattern-growth: The prefixspan approach. *Knowledge and Data Engineering, IEEE Transactions on*, 16(11):1424–1440, 2004.
9. Prof Pinkal Shah and AK Dua. Algorithm for sequence mining using gap constraints. *International Journal of Engineering Research and Development*, pages 37–49, 2014.
10. Oscar Eduardo Quinteros, Ana Funes, and Hernán César Ahumada. Extracción de conocimiento en el cursado del ciclo común de articulación de carreras de ingeniería. In *XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina)*, 2016.
11. Rüdiger Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, pages 29–39. Citeseer, 2000.
12. Mohammed J. Zaki. Spade: An efficient algorithm for mining frequent sequences. *Mach. Learn.*, 42(1-2):31–60, January 2001.