Serie Científica de la Universidad de las Ciencias Informáticas
Vol. 11, No. 3, Mes Marzo, 2018
ISSN: 2306-2495 | RNPS: 2343
Pág. 16-33
http://publicaciones.uci.cu

Tipo de artículo: Artículo original
Temática: Soluciones Infrmáticas
Recibido: 12/01/18 | Aceptado: 10/03/18 | Publicado: 30/03/18

Extracción de correlaciones entre el test vocacional CHASIDE y la carrera de Ingeniería en Ciencias Informáticas

Extraction of correlations between the CHASIDE vocational test and theEngineering in Computer Science

Samuel Ojeda Pereira[1], Julio Cesar Diaz Vera[2], Guillermo Manuel Negrín Ortiz[3]

[1] Universidad de las Ciencias Informáticas, sojeda@estudiantes.uci.cu
[2] Universidad de las Ciencias Informáticas, jcdiaz@uci.cu
[3] Universidad de las Ciencias Informáticas, gmnegrin@uci.cu

[*] Autor para correspondencia: sojeda@estudiantes.uci.cu

Resumen

La sociedad cubana está comprometida con el éxito personal de cada uno de sus ciudadanos y el sistema de educación superior del país es un reflejo de ese compromiso. A pesar de ello los niveles de fracaso escolar no se corresponden con los deseados. Una causa para esta problemática puede estar asociada a que los estudiantes no matriculan carreras correlacionadas favorablemente con sus habilidades, competencias y preferencias.En este trabajo se propone relacionar estas habilidades, competencias y preferencias con la carrera de ingeniería en ciencias informáticas. El objetivo de este trabajo es obtener un modelo que permita hacer un análisis de la correlación que existe entre estos elementos. El proceso de desarrollo de software orientado al análisis de datos es el de software de predominio de cómputo que usa como parte de su metodología la extracción de conocimiento de una base de datos, representado en forma de reglas de asociación que permiten obtener un modelo de análisis en forma de correlaciones. Se usó un dataset con 150 tuplas para obtener el modelo que permite establecer las correlaciones entre las habilidades, competencias y preferencias y la carrera de ingeniería en ciencias informáticas.

Palabras clave: Habilidades; Competencias; nivel de fracaso; correlacionadas.

Abstract

Cuban society is committed to the personal success of each of its citizens and the country's higher education system puts his effort in this regard. In spite of this, the levels of school failure do not correspond to the desired ones. A cause for this problem may be associated with students not enrolling in studies correlated favorably with their skills,

Serie Científica de la Universidad de las Ciencias Informáticas
Vol. 11, No. 3, Mes Marzo, 2018
ISSN: 2306-2495 | RNPS: 2343
Pág. 16-33
http://publicaciones.uci.cu

competencies and preferences. This paper aims at correlating these skills to the Informatics Sciences studies. The target of this paper is to obtain a model that allows to analyze the correlation among these elements. The software development process oriented to data analysis is computation dominant software that uses as a part of its methodology a database extraction phase represented using association rules that allow to obtain a model to set correlations. A dataset of 150 tuples was used to obtain the model that allows to stablish the correlations between skills, competencies and preferences and the Informatics Sciences studies.

Keywords: *Skills; Competencies; level of failure, correlated.*

Introducción

La orientación vocacional comienza a considerarse un elemento importante en el desarrollo de capacidades en los trabajadores a partir del año 1908 con la creación, en Boston, Estados Unidos, del Primer Buró de Orientación Vocacional a cargo de Frank Parsons. En aquel momento se introduce el término "VocationalGuidance" para agrupar un conjunto de elementos que permitían escoger qué profesión podría resultar más adecuada para cada persona en particular.

A medida que la orientación vocacional ha evolucionado se han utilizado técnicas que permiten establecer determinadas correspondencias entre las aptitudes naturales del ser humano, las exigencias y competencias necesarias para el correcto desempeño de la profesión. Las técnicas que han acaparado la mayor atención en este tipo de tareas están asociadas a la aplicación de test que siguen las teorías factorialistas[1].

Cuba ha dedicado varios esfuerzos en el área de la orientación vocacional. Este es un proceso que, en nuestro país, se lleva a cabo en el sector de la educación, y que sirve de ayuda para que los estudiantesde enseñanza media y media superior puedan seleccionar una profesión. Sin embargo, a pesar de ellolos niveles de fracaso escolar en las universidades cubanas todavía alcanzan cuotas superiores a lasesperadas. La insuficiente orientación vocacional es uno de los factores que provoca que los estudiantesseleccionen carreras para las que no tienen las aptitudes necesarias o que coinciden con sus interesesbásicos, lo que limita su capacidad para enfrentar las tareas de esta, ya que no están motivados y lesdificulta que alcancen resultados positivos.

Se han desarrollado varios instrumentos para determinar las habilidades, competencias y preferencias de los estudiantes y establecer la relación entre estas y las áreas del conocimiento. Especialmenteinteresante en este sentido es el test CHASIDE [2]. Sin embargo, estos resultados no han sidocorrelacionados, de manera experimental, con el éxito en el estudio de la carrera de Ingenieríaen Ciencias Informáticas. Lo que constituye una limitación

importante si se pretenden desarrollarsistemas inteligentes y/o de recomendación que contribuyan a la orientación de los estudiantes con vistas a decidir la carrera más favorable para ellos.

Lo antes expresado conduce a plantear el siguiente problema: Cómo correlacionar las habilidades,competencias y preferencias determinadas mediante el test de orientación vocacional CHASIDE conla carrera de Ingeniería en Ciencias Informáticas. La investigación se centrará en el objeto de estudio del desarrollo de software y teniendo como objetivo general: Establecerun grupo de correlaciones entre las habilidades, competencias y preferencias detectadas en el test vocacional CHASIDE con la carrera de Ingeniería en Ciencias Informáticas. Teniendo como campo de acción: Proceso de desarrollo de Software de Predominio de Cómputo.

En la sección de materiales y métodos encontrará un resume del que es el proceso de desarrollo de software, y como se puede enfocar este a los softwares de predominio de cómputo siguiendo su taxonomía. Mientras que en la sección de discusión y resultado se expondrá el algoritmo de minería de datos que se escogió para realiza el proceso de descubrimiento de información en la base de datos y la descripción e implementación de la solución siguiendo la metodología Catalys.

Materiales y métodos o Metodología computacional

En la actualidad existen gran cantidad de datos almacenados de forma digital, donde hay información valiosa y que a los humanos les resulta difícil de procesar manualmente o a través de una aplicación de gestión. Este problema ha generado una nueva forma de procesar la información. Existen múltiples técnicas de extracción de conocimiento, muchas de ellas enmarcadas dentro del dominio de la minería de datos.

Unas de las técnicas son la extracción de reglas de clasificación ha ganado fuerza en los últimos años y son utilizadas en ramas de la sociedad como la salud, la educación el deporte, el comercio entre otros. Este proceso se realiza a través de los llamados algoritmos de extracción que le son aplicados a las vistas minables. Para ello la siguiente investigación se enfoca en el proceso desarrollo de software para el procesamiento de datos y en realizar un análisis detallado del software de predominio de cómputo. Para ello se descriará en la presente investigación el proceso para análisis de datos en tiempo de inactividad.

¿Qué es un software?

Un software de computadora es: el producto que construyen los programadores profesionales, o sea, instrucciones que cuando se ejecutan proporcionan las características, funciones y desempeñobuscados. Al que después le dan mantenimiento durante un largo tiempo. Incluye programas que seejecutan en una computadora de cualquier tamaño y arquitectura, contenido que se presenta a medidaque se ejecutan los programas de cómputo e información descriptiva

Serie Científica de la Universidad de las Ciencias Informáticas
Vol. 11, No. 3, Mes Marzo, 2018
ISSN: 2306-2495 | RNPS: 2343
Pág. 16-33
http://publicaciones.uci.cu

tanto en una copia dura enformato virtuales que engloban virtualmente a cualesquiera medios electrónicos.

Es importante definir las características principales de un software las cuales son: El software es unelemento de un sistema lógico o virtual y no de uno físico. Por tanto, tiene características que lodiferencia considerablemente del hardware, entre ellas se encuentran que se desarrolla o modificancon el intelecto y no se manufacturan como un producto clásico. Lo que se convierte en un productono desgastable y aunque las tendencias del mercado sean a desarrollar módulos o componentes elsoftware siempre se va a dar un uso individualizado[3].

Proceso de desarrollo de un software

A la hora de desarrollar software es vital decir que se debe hacer como un proceso ya que abarca un conjunto de actividades, acciones y tareas que se van ejecutando según se va creando el producto. La construcción de un sistema de software debe ser precedida por la construcción de un modelo, tal como se realiza en otros sistemas ingenieriles (Figura 1-1). El modelo del sistema es una conceptualización del dominio del problema y de su solución [4]. El modelo se focaliza sobre el mundo real: identificando, clasificando y abstrayendo los elementos que constituyen el problema y organizándolos en una estructura formal.

Figura 1.1: *Proseso de dasarrollo del software*

Tipologías de software por su taxonomía

Según su estructura, el software puede ser agrupado en cuatro grandes grupos, que responden a su tipología, funciones, dominio y características fundamentales. Estos a su vez se derivan en ramas o subconjuntos más específicos para poder comprender más fácil su estructura y por ende el modelo a utilizar para su elaboración. Si se agrupan todos por sus conceptos fundamentales quedarían ubicados dependiendo en los siguientes grupos según su taxonomía [5] :

1. Software de gestión de datos
2. Software de sistemas
3. Software controlador del hardware
4. Software de Predominio de Cómputo

Analizando las características de las tipologías de los softwares, se enfoca la presente investigación enlos **softwares de predominio de cómputo** los cuales se basan en como conceptualizar solucionesinformáticas que se dirigen al área de la inteligencia artificial. Dentro de ellos se encuentran los softwares de simulación, búsqueda de información y los de aprendizaje de máquina. Estos últimosse relaciona con el desarrollo de programas que mejoran el desempeño en cierta tarea, mediante laexperiencia. Los algoritmos que sutilizan para aprendizaje de máquina ha probado su valía en una

variedad de aplicaciones[6]:

- ✓ Problemas de minería de datos.
- ✓ Dominios en los que los humanos no disponen del conocimiento necesario para desarrollar algoritmos efectivos.
- ✓ Dominios en los que los programas deben adaptarse dinámicamente a condiciones cambiantes.

Normalmente estos software parten de datos almacenados, lo cuales son procesados para encontrar conocimiento útil. Es importante entender como se desarrollan estos tipos de software para comprender su proceso de desarrollo.

Desarrollo de software orientados al análisis de datos

Las primeras etapas del desarrollo de software son cruciales en la consecución de productos de calidad dentro de los límites de tiempo y coste establecidos para un proyecto. Los errores introducidos en las primeras etapas del desarrollo del software o durante su evolución son causa frecuente de dificultades en el mantenimiento, baja reutilización y comportamiento defectuoso de los programas. Estas son las principales causas por las que la medición del software en el ámbito de la especificación de requisitos (ERS) está adquiriendo cada vez mayor importancia, debido a la necesidad de obtener datos objetivos que contribuyan a mejorar la calidad desde las primeras etapas del producto [3].

Cuando se está construyendo un software de predominio de cómputo normalmente nos debemos hacer la pregunta: ¿cómo conceptualizar el problema de software práctico como el programa de Análisis de datos en tiempo de inactividad, a las áreas teóricas como inteligencia artificial? Teniendo en cuenta los objetivos a alcanzar la aplicación a desarrollar.

Serie Científica de la Universidad de las Ciencias Informáticas
Vol. 11, No. 3, Mes Marzo, 2018
ISSN: 2306-2495 | RNPS: 2343
Pág. 16-33
http://publicaciones.uci.cu

Hasta no hace mucho, el análisis de los datos de una base de datos se realizaba mediante consultas efectuadas con lenguajes, generando listas de consulta, como el SQL, y se producía sobre la base de datos operacional, es decir, junto al Procesamiento Transaccional en Línea (On-Line TransactionProcessing, OLTP por sus siglas en inglés) de las aplicaciones de gestión. El crecimiento gradual de los datos hace difícil el procesar tanta información de manera eficiente, creando la necesidad de analizar los datos de otra manera más eficiente que la tradicionales, dando paso a una nueva tendenciadel software "la minería de datos".La aplicación práctica de técnicas de minería de datos en la construcción y validación de modelos de ingeniería del software que relacionan diferentes atributos de la ERS.

La minería de datos ha dado lugar a una paulatina sustitución del análisis de datos dirigidos a la verificación por un enfoque de análisis de datos dirigido al descubrimiento del conocimiento. La principal diferencia
entre ambos se encuentra en que en el último se descubre información sin necesidad de formular previamente una hipótesis. La aplicación automatizada de algoritmos de minería de datos permite detectar fácilmente patrones en los datos, razón por la cual esta técnica es mucho más eficiente que el análisis dirigido a la verificación cuando se intenta explorar datos procedentes de repositorios de gran tamaño y complejidad elevada. Dichas técnicas emergentes se encuentran en continua evolución como resultado de la colaboración entre campos de investigación tales como bases de datos, reconocimiento de patrones, inteligencia artificial, sistemas expertos, estadística, visualización,
recuperación de información, y computación de altas prestaciones.

Los algoritmos de minería de datos se clasifican en dos grandes categorías: supervisados o predictivos y no supervisados o de descubrimiento del conocimiento [7].Los supervisados o predictivos predicen el valor de un atributo(etiqueta) de un conjunto de datos, conocidos otros atributos(atributos descriptivos). A partir de datos cuya etiqueta se conoce se induce una relación entre dicha etiqueta y otra serie de atributos. Esas relaciones sirven para realizar la predicción en datos cuya etiqueta es desconocida. Esta forma de trabajar se conoce como aprendizaje supervisado y se desarrolla en
dos fases: Entrenamiento (construcción de un modelo usando un subconjunto de datos con etiqueta conocida) y prueba (prueba del modelo sobre el resto de los datos).

Cuando una aplicación no es lo suficientemente madura no tiene el potencial necesario para una solución predictiva, en ese caso hay que recurrir a los métodos no supervisados o descubrimiento del conocimiento que descubren patrones y tendencias en los datos actuales (no utilizan datos históricos). El descubrimiento de esa información sirve para llevar a cabo acciones y obtener un beneficio (científico o de negocio) de ellas. Para comprender mejor lo planteando en la próxima sección se explica más detallado el cómo se hace y las características principales de los softwares de predominiode cómputo.

Serie Científica de la Universidad de las Ciencias Informáticas
Vol. 11, No. 3, Mes Marzo, 2018
ISSN: 2306-2495 | RNPS: 2343
Pág. 16-33
http://publicaciones.uci.cu

Tipologías de Software de Predominio de Cómputo

Partiendo del enfoque de cómo conceptual el problema de software práctico como el programa de análisis de datos en tiempo de inactividad, enfocado a las áreas teóricas de la inteligencia artificial. Los softwares de predominios de cómputo se centran en el procesamiento de la información para lograr mediante métodos de la inteligencia artificial un tratamiento automatizado de los datos.

El descubrimiento de conocimiento en bases de datos, conocido en la actualidad como "minería dedatos", es una disciplina que ha crecido enormemente en los últimos años. Esta no es más que elproceso de extraer conocimiento útil, comprensible y novedoso de grandes volúmenes de datos, siendosu principal objetivo encontrar información oculta o implícita que no es posible obtener mediantemétodos estadísticos convencionales[8]. Para ello utiliza métodos y algoritmos desarrollados enlos campos de aprendizaje automatizado (del inglés machine learning), reconocimiento de patrones,análisis estadístico de datos, visualización de datos, redes neuronales, entre otros[9]. Actualmente lasorganizaciones han comprendido que los grandes volúmenes de datos que residen en sus sistemaspueden ser analizados y explotados para obtener nuevo conocimiento a partir de los mismos.

Elementos que caracteriza la tipología de Software de predominio de cómputo

A la hora de construir un software de predomino de cómputo lo primero que debemos hacer es clasificar a qué tipo de software pertenece dentro de la taxonomía, para así saber que herramientas usar y como debe ser el resultado final. Atendiendo a la tipología del software de predominio de software se pueden clasificar en[5]:

- ✓ Software de operaciones de búsqueda.
- ✓ Administración y manipulación de información
- ✓ Software de creaciones artísticas
- ✓ Software científicos
- ✓ Software de inteligencia artificial

La investigación se centra en Análisis de datos de tiempo en inactividad, ya que se analizarán los datos recolectado en el test de orientación vocacional Chaside para poder describir la relación detallada entre competencias y aptitudes en la carrera de Ingeniería en Ciencias. Teniendo en cuenta el modelo que se desea obtener, las relaciones se van a representar mediante reglas de clasificación, la cual podrá ayudar en la perdición de las características del ingeniero en Ciencias informáticas. A la hora de construir un software con estas características se debe tener en cuenta una serie de aspectos y pasos

lógicos basados en los algoritmos de inteligencia artificial que se vaya a usar para obtener el resultado deseado.

Serie Científica de la Universidad de las Ciencias Informáticas
Vol. 11, No. 3, Mes Marzo, 2018
ISSN: 2306-2495 | RNPS: 2343
Pág. 16-33
http://publicaciones.uci.cu

Desarrollo de Software de Predominio de Cómputo.

La aplicación de los algoritmos de minería de datos requiere la realización de una serie de actividades previas encaminadas a preparar los datos de entrada debido a que, en muchas ocasiones dichos datos proceden de fuentes heterogéneas, no tienen el formato adecuado o contienen ruido. Por otra parte, es necesario interpretar y evaluar los resultados obtenidos. El proceso completo consta de las siguientes etapas [5]:

1. Determinación del objetivo.
2. Preparación de los datos:

- Selección: Identificación de las fuentes de información externas e internas y selección del subconjunto de datos necesario.
- Preprocesamiento: estudio de la calidad de los datos y determinación de las operacionesde minería que se pueden realizar.

3. Transformación de los datos: conversión de datos en un modelo analítico.
4. Minería de datos: tratamiento automatizado de los datos seleccionados con una combinaciónapropiada de algoritmos.
5. Análisis de los resultados: interpretación de los resultados obtenidos en la etapa anterior, generalmente con la ayuda de una técnica de visualización.
6. Asimilación del conocimiento: aplicación del conocimiento descubierto.

Aunque los pasos anteriores se realizan en el orden en que aparecen, el proceso es altamenteiterativo, estableciéndose retroalimentación entre los mismos. Además, no todos los pasos requierenel mismo esfuerzo, generalmente la etapa de pre-procesamiento es la más costosa ya que representaaproximadamente el 60 porciento del esfuerzo total, mientras que la etapa de minería sólo representael 10 por ciento.

A groso modo los componentes antes explicado se puede concluir que pertenece al proceso dedesarrollo de software de la Minería de Datos, específicamente a los de descubrimiento de información enbases de datos(Knowledge Discovery in Databases,KDD por sus siglas en inglés). Este es el procesocompleto de extracción de información, que se encarga además de la preparación de los datos y de lainterpretación de los resultados obtenidos. Se ha definido como “el proceso no trivial de identificaciónen los datos de patrones válidos, nuevos, potencialmente útiles, y finalmente comprensibles”. Se tratade interpretar grandes cantidades de datos y encontrar relaciones o patrones [10].

Serie Científica de la Universidad de las Ciencias Informáticas
Vol. 11, No. 3, Mes Marzo, 2018
ISSN: 2306-2495 | RNPS: 2343
Pág. 16-33
http://publicaciones.uci.cu

La minería de datos no es más que encontrar conocimiento útil a partir de datos recolectados sobre una entidad o tema en específico. Para poder entender que es minería de datos primero hay que verporque etapas pasa nuestros componentes para llegar a descubrir riquezas en nuestra información.

A continuación, se muestra una figura que describe el proceso completo de KDD:

KDD es un proceso iterativo e interactivo. Es iterativo ya que la salida de alguna de las fases puedehacer volver a pasos anteriores y porque a menudo son necesarias varias iteraciones para extraerconocimiento de alta calidad. Es interactivo porque el usuario, o más generalmente un experto enel dominio del problema, debe ayudar en la preparación de los datos, validación del conocimientoextraído, convirtiéndose en un procedimiento supervisado.

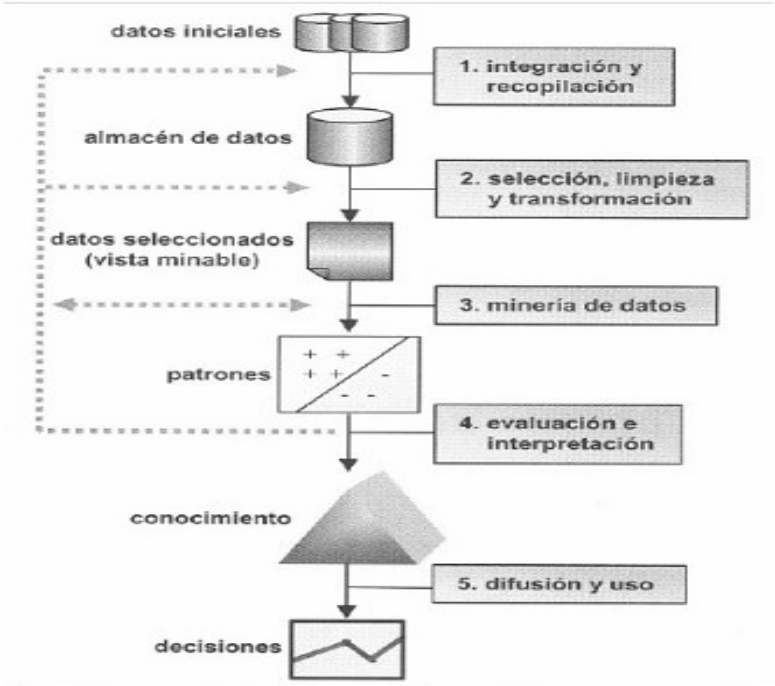

Figura 1.2: *Fases del proceso de extracción de conocimiento en bases de datos.*

Si bien el modelado en KDD puede resumirse en cinco faces principales mencionadas anteriormente, Fayyad en su libro "The KDD processforextractingusefulknowledgefromvolumes of data "define nueve etapas para llevar a cabo todo el proceso. Definidas a continuación[11],[12]:

1. Comprensión del dominio de aplicación. En esta primera etapa, se debería recolectar todo elconocimiento disponible y relevante sobre el dominio de aplicación e identificar los objetivosdel proceso KDD desde el punto de vista del usuario.
2. Creación del conjunto de datos. Esta etapa consiste en la elección de las fuentes de datos que se utilizarán, la integración de las mismas y la elección de las observaciones/atributos que conformarán la vista minable (Datos que se van a procesar). Aunque no es estrictamente necesario, en este paso podría requerirse la construcción de un almacén de datos.
3. Limpieza y pre-procesamiento de los datos. En esta fase se deberían llevar a cabo tareas comolimpieza de ruido o datos anómalos (outliers) y tratamiento de datos faltantes (missingvalues).
4. Reducción y proyección de los datos. En este paso se detectan características útiles de representación de los datos dependiendo del objetivo de la tarea de minería (descripción o predicción). Se incluye la utilización de técnicas de reducción de la dimensionalidad y métodosde transformación de los datos para reducir la cantidad de variables en discusión o para encontrar representaciones invariantes de los datos. En esta etapa es frecuente la transformación de los datos, calculando nuevos atributos o bien redefiniendo los existentes con otro formato.
5. Determinar la tarea de minería de datos. En esta fase, se deberá determinar la tarea de minería con la que se abordará el estudio (como agrupamiento, regresión, clasificación, o asociación) teniendo en cuenta los objetivos definidos en la etapa 1.
6. Determinar el algoritmo de minería. De acuerdo a la tarea de minería establecida en el punto anterior, en esta etapa se define el algoritmo (o algoritmos) que se aplicarán para la búsqueda de patrones sobre los datos. Incluye la determinación de qué modelos y parámetros son los más adecuados según la naturaleza del problema y de los datos disponibles.
7. Minería de datos(MD). Etapa en la que se aplican los algoritmos y técnicas seleccionadas al conjunto de datos en búsqueda de los patrones de interés.
8. Interpretación. Comprende la interpretación de los patrones encontrados, visualizando y traduciendo los mismos en términos comprensibles por el usuario.
9. Utilización del nuevo conocimiento. En esta fase se implementa el conocimiento descubierto, apoyando con el mismo la toma de decisiones o bien reportándolo a las partes interesadas. Incluye la verificación y resolución de potenciales conflictos con conocimiento descubierto previamente.

Luego de realizar un estudio de los conceptos básicos del software orientados a la inteligencia artificial,

específicamente al software de minería de datos, centrándonos en el proceso de KDD y las herramientas existentes para la minería de datos, se concluye parcialmente lo siguiente:

- ✓ Se define la taxonomía de nuestra solución orientada a los softwares de predominio de computo enmarcando la solución informática al área de la inteligencia artificial.
- ✓ Se describe el proceso de desarrollo de software científicos ya que la solución va orientada al análisis de datos de tiempo en inactividad, teniendo en cuentacómo hacer el procesamiento de los datos.

  El proceso de KDD es el más adecuado para nuestra solución ya que transita por 4 fases fundamentales para nuestra solución(pre-procesamiento de datos, selección de la técnica de minería de datos, post-procesamiento, visualización).

Resultados y discusión

Una vez definido y descrito el proceso de desarrollo de software se debe definir el algoritmo que se va a utilizar. Existen varios algoritmos para aplicar las técnicas de minería de datos, la diferencia entre ellos está dado a en el modo de extraer la información de los datos. Cuando se trata de reglas de asociación o clasificación uno de los algoritmos más utilizado es Apriori.

Este algoritmo está basado en la reducción de conjuntos, centrándose en un soporte mínimo(sop_min) introducido. Donde se define que, el sop(X) como la proporción de transacciones que contienen el conjunto X, donde I es un conjunto de elementos, y se utilizará |A| para denotar la cardinalidad del conjunto A(sop(X)= (|{ I| I ∈ D ∧ I ⊇ X}|) /N). De la definición de soporte tenemos que si sop(A∪C) ≤sop_min entonces sop(A → C) ≤sop_min.

Apriori genera todos los conjuntos que cumplen con la condición de tener un soporte menor o igual a sop_min. Para cada conjunto frecuente X se generan todas las reglas de asociación A → C tal que A∪C= A y A∩ C =Ø. Cualquier regla que no satisfaga las restricciones impuestas por el usuario, como por ejemplo la confianza mínima, se desechan, y las reglas que sí cumplen se conservan[13].

Como el sop(A) ≥ sop(A→C) y sop(C)≥ sop(A→C), si A S C es un conjunto frecuente entonces tanto A como C son conjuntos frecuentes. El soporte, la confianza y otras métricas por las cuales las reglas de asociación A → C son evaluadas y se puede usar el spo(A), sop(C) y sop(A∪ C) como referencias. Este algoritmo se resume en dos pasos fundamentales para obtener los resultados [14]:

Paso1:

Generación de todos los item-sets que contienen un solo elemento, utilización de estos para generar item-sets que contengan dos elementos, y así sucesivamente. Se toman todos los posibles pares de items que cumplen con las

medidas mínimas de soporte inicialmente preestablecidas; esto permite ir eliminando posibles combinaciones: aquellas que no cumplan con los requerimientos de soporte no entrarán en el análisis.

Paso2:

Generación de las reglas revisando que cumplan con el criterio mínimo de confianza. Es interesante observar que, si una conjunción de consecuentes de una regla cumple con los niveles mínimos de soporte y confianza, sus subconjuntos (consecuentes) también los cumplen; en el caso contrario, si algún ítem no los cumple no tiene caso considerar sus súper conjuntos.

Este algoritmo tiene como ventaja fundamental que para generar reglas de clasificación puede ser muy útil ya que cuando realiza el primer paso va generando los item-sets de grado 1, luego los de grado 2 y así sucesivamente. Si restringimos al algoritmo a la primera iteración podríamos obtener reglas de clasificación. Estas reglas son muy parecidas a la de asociación con la diferencia de que un solo consecuente implica a un conjunto de item-set. Ejemplo:(x→ y,z,v,n).

Una vez definido nuestro algoritmo se lleva a cabo el proceso KDD bajo la metodología t. La cual está formada por dos partes (o sub-metodologías): Metodología para el Modelado del Negocio y Metodología para la Minería de Datos. Esta metodología es muy completa y a pesar de que no es tan difundida como CRISP-DM, se puede afirmar que refleja todos los procesos de KDD, dando como resultado[12].

Modelado del Negocio:

- Escenario 1: Datos

Los datos se encuentran almacenados en una base de datos la cual contiene una tabla donde está la relación entre las preguntas marcadas y el identificador del estudiante. La otra tabla que contiene datos relevantes es la tabla estudiante la cual almacena los datos de este con su identificador de clasificación. A la hora de analizar los datos se debe unir ambas tablas para obtener por preguntas cuantos estudiantes marcaron cada pregunta y adicionar en forma de pregunta los clasificadores. Los cuales son muy importante para nuestras reglas de clasificación.

- Escenario 2: PROBLEMA/OPORTUNIDAD

En estos momentos no existe un modelo que pueda relacionar las habilidades e intereses con la calidad de los graduados en la carrera de ingeniería en ciencias informáticas. Para ellos la investigación se centra en hallar que relación existente entre los profesionales graduados de en la carrera con respecto a las habilidades e intereses que se evalúa en el test de orientación vocacional. Normalmente se hace engorroso realizar la correlación a través de consultas a la base de datos de manera tradicional, por lo que se hace necesario la implementación de un algoritmo

que permita hallar con qué frecuencia se repite cada pregunta y qué relación existe con las clasificaciones otorgada a cada profesional.

- Escenario 3: PROSPECCIÓN

Este resultado nos permitirá describir un modelo mediante reglas de clasificación las cual servirá para darle un agregado al resultado del test de orientación vocacional CHASIDE y se podrá predecir un posible comportamiento de los estudiantes dentro de la carrera. Como fundamento de la solución la investigación se basó en el concepto dado por la doctora V. G. Maura, el cual plantea que según sea la orientación vocacional del estudiante así será los resultados a obtener en la carrera.

- Escenario 4: MODELO DEFINIDO

El modelo se representará mediante reglas de clasificación, con el objetivo de usar un clasificador para poder describir cómo se puede comportar el estudiante al cursar la carrera, ya que hay otras variables que pueden incidir en los resultados. Asociado a las reglas van un soporte y una confianza la cual sirve para evaluar que tan exacto puede ser el resultado que se arroja.

Metodología para la Minería de Datos

Al aplicar la metodología, el segundo paso o parte de esta es la Metodología para la Minería de Datos en la cual se proporciona una guía de pasos para el descubrimiento de patrones/relaciones de acuerdo al problema de negocio identificado. Estos pasos se hacen llamar muy parecidos a los descrito por el modelado de KDD.

1. Preparación de los datos:

Una vez recolectada la información necesaria se pasa a clasificar los datos de los profesionales mediante un panel de experto en los cuales se recogen los datos de los profesionales y a partir de ellos se le da una clasificación a cada profesional. Creando 4 grupos, los Avanzados (títulos de oro),Buenos (destacado en eventos y con índice de ente 4.2-4.6), Satisfactorio (rendimiento adecuado con índice entre 3.9-4.1) y Malos (graduados con muchos mundiales y malo rendimiento académico). Se verificó que todos los que realizaron el test estuvieran correctamente clasificados, para que no existieran datos inconclusos.

2. Selección de herramientas y modelado inicial:

Como herramientas fundamentales se usaron el servidor de bases de datos Postgres y como lenguajes de programación se incrementarán para la selección de los datos una función en PLpgsql y para encontrar los ítem-frecuent y las reglas de clasificación se implementarán en Python el algoritmo Apriori al cual se le realizará una variación para obtener reglas de clasificación.

3. Refinar el modelo:

En el modelo se rechazan todas las reglas de clasificación que no contengan las 4 clases de clasificación. Obteniendo solo las reglas que implique uno de los clasificadores, del lado del consecuente.

4. Implementar el modelo:

Se implementó la siguiente función para seleccionar y limpiar los datos. Dando como resultado una matriz esparcida en la cual cada línea representa un estudiante con las preguntas que marco verdaderas y la clasificación al final.

```
CREATE OR REPLACE FUNCTION tipo_test()
  RETURNS SETOF text AS
$BODY$
declare
inte text;
apti text;
est integer;
res text;
z text;
clas integer;
begin
FOR est IN SELECT f_estudiantes.id_f_estudiantesFROM  public.f_estudiantes loop
res:='';

SELECT caso_aputi(est) into apti;
SELECT caso_interes(est) into inte ;
SELECT f_estudiantes.id_clasificacion into clas FROM public.f_estudiantes WHERE id_f_estudiantes=est;
case clas
when 1 then res:=res||apti||inte||'42';
when 2 then res:=res||apti||inte||'43';
when 3 then res:=res||apti||inte||'44';
when 4 then res:=res||apti||inte||'45';
else
res:='borrar';
END CASE;
z:=res;
return next z;
endloop;

end
$BODY$
  LANGUAGE plpgsql;
```

Función principal para la limpieza y transformación de los datos.

Dentro de esta función se les hace llamada a otras dos una que procesa las preguntas que se clasifican según el test de interese y la otra se encarga de las aptitudes, la cuales lo que hace es contar las cantidades de preguntas que hay por cada rama del test y evalúan que tan alta es esa puntuación.

```
CREATE OR REPLACE FUNCTION public.caso_aputi(est integer)
  RETURNS text AS
$BODY$
declare
f record;
```

```
res text;
begin
res:='';
FOR f IN (Select case
when id_f_categoria_de_la_carrera=1 and count(id_f_preguntas)<4 then '0'
when id_f_categoria_de_la_carrera=1 and count(id_f_preguntas)<=7 and count(*)>=4 then '1'
when id_f_categoria_de_la_carrera=1 and count(id_f_preguntas)>=7 then '2'
when id_f_categoria_de_la_carrera=2 and count(id_f_preguntas)<4 then '3'
when id_f_categoria_de_la_carrera=2 and count(id_f_preguntas)<=7 and count(*)>=4 then '4'
when id_f_categoria_de_la_carrera=2 and count(id_f_preguntas)>=7 then '5'
when id_f_categoria_de_la_carrera=3 and count(id_f_preguntas)<4 then '6'
when id_f_categoria_de_la_carrera=3 and count(id_f_preguntas)<=7 and count(*)>=4 then '7'
when id_f_categoria_de_la_carrera=3 and count(id_f_preguntas)>=7 then '8'
when id_f_categoria_de_la_carrera=4 and count(id_f_preguntas)<4 then '9'
when id_f_categoria_de_la_carrera=4 and count(id_f_preguntas)<=7 and count(*)>=4 then '10'
when id_f_categoria_de_la_carrera=4 and count(id_f_preguntas)>=7 then '11'
when id_f_categoria_de_la_carrera=5 and count(id_f_preguntas)<4 then '12'
when id_f_categoria_de_la_carrera=5 and count(id_f_preguntas)<=7 and count(*)>=4 then '13'
when id_f_categoria_de_la_carrera=5 and count(id_f_preguntas)>=7 then '14'
when id_f_categoria_de_la_carrera=6 and count(id_f_preguntas)<4 then '15'
when id_f_categoria_de_la_carrera=6 and count(id_f_preguntas)<=7 and count(*)>=4 then '16'
when id_f_categoria_de_la_carrera=6 and count(id_f_preguntas)>=7 then '17'
when id_f_categoria_de_la_carrera=7 and count(id_f_preguntas)<4 then '18'
when id_f_categoria_de_la_carrera=7 and count(id_f_preguntas)<=7 and count(*)>=4 then '19'
when id_f_categoria_de_la_carrera=7 and count(id_f_preguntas)>=7 then '20'     else -1
end as cas
fromf_categoria_de_la_carreraleftouterjoin (SELECT f_categoria_de_la_carreraid_f_categoria_de_la_carrera as cat,
f_preguntas.id_f_preguntas
FROM
public.f_fuente,public.f_preguntas
where  marcada=1 and f_preguntas.id_f_preguntas = f_fuente.f_preguntasid_f_preguntas and f_tipo_testid_f_tipo_test=2
and f_estudiantesid_f_estudiantes=$1)as fuentes on (fuentes.cat=id_f_categoria_de_la_carrera)
groupbyid_f_categoria_de_la_carreraorderbyid_f_categoria_de_la_carrera) loop
res=res ||f.cas||',' ;

end loop;
return res;
end
$BODY$
LANGUAGE plpgsql ;
```

Clase en Python que permite generar las reglas de clasificación

Una vez que los datos fueron agrupados, clasificados según intereses y aptitudes aplicando una función sobre las tablas donde se encuentran las preguntas marcadas o no sobre el test. Este paso arrojó 45 posibles clasificaciones sobre la respuesta del test la cual se corresponden con la salida de CHASIDE y cada categoría fue clasificada en alta, regular y baja según la cantidad de aciertos por intereses y aptitudes. Obteniéndose 15 columnas que se hace corresponder con 7 clasificaciones de aptitudes, 7 clasificaciones de intereses y la clasificación del encuestado. Al correr este algoritmo se obtuvo un modelo descriptivo el cual describes las principales características presente en lo

encuestados, con un nivel de confianza de 0.5. El algoritmo arrojó como respuesta 65 reglas que se muestran a continuación:

[35→45, 26,6→45, 35,6→45, 25,15→43, 28,40→43, 25,40→43, 40,0,12→43, 25,0,15→43, 25,12,0→43, 25,6,15→43,12,40,15→43, 25,12,40→43, 25,40,6→43, 12,28,40→43, 40,15,22→43, 12,37,40→43, 28,40,6→43, 12,40,22→43, 28,40,15→43, 25,40,15→43, 25,12,15→43, 12,40,6→43, 26,6,15→45, 25,0,12,6→43, 28,40,6,15→43, 12,37,18,15→43, 12,40,18,15→43, 12,40,37,15→43, 12,18,15,22→43, 40,18,22,15→43, 37,40,6,15→43, 9,12,18,15→44, 12,37,6,15→43,12,40,6,22→43, 28,12,6,40→43, 40,12,6,15→43, 40,6,15,22→43, 12,40,15,22→43, 40,0,12,6→43, 12,40,28,15→43, 25,0,6,15→43, 25,12,18,15→43, 25,40,6,15→43, 12,37,6,40→43, 40,0,12,15→43, 25,12,6,15→43, 25,0,12,15→43, 12,40,18,22→43, 25,12,40,15→43, 25,40,12,6→43, 12,40,6,37,15→43, 25,0,12,6,15→43, 0,40,6,12,15→43, 25,12,6,18,15→43, 28,40,6,12,15→43, 9,12,6,18,15→44, 12,6,18,22,15→43, 25,40,12,6,15→43,12,40,18,22,→43, 37,12,6,18,15→43,12,40,6,18,15→43, 40,12,6,18,22→43,40,6,18,22,15→43, 12,40,6,22,15→43, 6,22,12,15,40,18→43]

Leyenda: 0→Administrativas y Contables.baja.aptitudes, 1→Administrativas y Contables.media.aptitudes, 2→Administrativas y Contables.alta.aptitudes,3→Humanisticas y Sociales.baja.aptitudes, 4→Humanisticas y Sociales.media.aptitudes, 5→Humanisticas y Sociales.alta.aptitudes,6→Artisticas.baja.aptitudes, 7→Artisticas.media.aptitudes,8→Artisticas.alta.aptitudes, 9→Medicina y Cs. de la Salud.baja.aptitudes, 10→Medicina y Cs. de la Salud.media.aptitudes, 11→Medicina y Cs. de la Salud.alta.aptitudes, 12→Ingenieria y Computacion.baja.aptitudes, 13→Ingenieria y Computacion.media.aptitudes,14→Ingenieria y Computacion.alta.aptitudes, 15→Defensa y Seguridad.mala.aptitudes, 16→Defensa y Seguridad.regular.aptitudes,17→Defensa y Seguridad.alta.aptitudes, 18→Ciencias Exactas y Agrarias.baja.aptitudes, 19→Ciencias Exactas y Agrarias.regular.aptitudes, 20→Ciencias Exactas y Agrarias.alta.aptitudes, 21→Ciencias Exactas y Agrarias.baja.intereses, 22→Administrativas y Contables.media.intereses, 23→Administrativas y Contables.alta.intereses,24→Humanisticas y Sociales.baja.intereses, 25→Humanisticas y Sociales.media.intereses, 26→Humanisticas y Sociales.alta.intereses, 27→Artisticas.baja.intereses 28→Artisticas.media.intereses, 29→Artisticas.alta.intereses, 30→Medicina y Cs. de la Salud.baja.intereses, 31→Medicina y Cs. de la Salud.media.intereses, 32→Medicina y Cs. de la Salud.alta.intereses, 33→Ingenieria y Computacion.baja.intereses, 34→Ingenieria y Computacion.media.intereses, 35→Ingenieria y Computacion.alta.intereses, 36→Defensa y Seguridad.mala.intereses, 37→Defensa y Seguridad.regular.intereses, 38→Defensa y Seguridad.alta.intereses,

39→Ciencias Exactas y Agrarias.baja.intereses, 40→Ciencias Exactas y Agrarias.regular.intereses, 41→Ciencias Exactas y Agrarias.alta.intereses, 42→bad, 43→average, 44→good, 45→outstanding

Conclusiones

Con la realización de la herramienta informática se consigue aumentar el nivel de calidad y claridad con respecto a la orientación vocacional que recibirán los involucrados en el proceso, mejorando la informaciónreferente a las diferentes carreras que se ofertan en el país, brindando actualidad y seguridad de la misma.Sirve de soporte a las diferentes universidades del país, contribuyendo además al proceso de toma de decisionesde los estudiantes. Contribuyendo a una correcta elección y formación del futuro profesional y al aumento estadístico de promoción de la universidad comprometida debido a que el egresado está consciente de lasasignaturas que va a estudiar y las ramas de su desarrollo profesional. Logrando que luego de concluidos susestudios pueda obtener mejores resultados en su desempeño laboral. Lográndose en concreto:

- El desarrollo una herramienta informática de apoyo al proceso de orientación vocacional en los estudiantes de la enseñanza media en cuba.
- Describir características de los estudiantes en la carrera de Ciencias Informática mediante los intereses y aptitudes descrito por los profesionales graduados en la carrera.

Referencias

[1] F. De Fruyt y I. Mervielde, «The five-factor model of personality and Holland's RIASEC interest types», *Personal. Individ. Differ.*, vol. 23, n.º 1, pp. 87–103, jul. 1997.

[2] V. G. Maura, «El servicio de orientación vocacional-profesional (SOVP) de la Universidad de La Habana: una estrategia educativa para la elección y desarrollo profesional responsable del estudiante.», *Pedagog. Univ.*, vol. 6, n.º 4, 2013.

[3] N. Fenton y J. Bieman, *Software metrics: a rigorous and practical approach*. CRC Press, 2014.

[4] T. DeMarco, «Structure analysis and system specification», en*Pioneers and Their Contributions to Software Engineering*, Springer, 1979, pp. 255–288.

[5] A. Forward y T. C. Lethbridge, «A taxonomy of software types to facilitate search and evidence-based software engineering», en*Proceedings of the 2008 conference of the center for advanced studies on collaborative research: meeting of minds*, 2008, p. 14.

[6] A. E. Trujillo Arboleda, «Desarrollo de una propuesta para el uso de técnicas con base en inteligencia de negocios, para la toma de decisiones estratégicas en una empresa de viajes y courier», Quito: Universidad de las Américas, 2016., 2016.

[7] S. M. Weiss y N. Indurkhya, *Predictive data mining: a practical guide*. Morgan Kaufmann, 1998.

[8] M. Kantardzic, *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons, 2011.

[9] S. Russell, P. Norvig, y A. Intelligence, «A modern approach», *Artif. Intell. Prentice-Hall Egnlewood Cliffs*, vol. 25, p. 27, 1995.

[10] T. H. Cao, E.-P. Lim, Z.-H. Zhou, T.-B. Ho, D. Cheung, y H. Motoda, *Advances in Knowledge Discovery and Data Mining*. Springer, 2015.

[11] U. Fayyad, G. Piatetsky-Shapiro, y P. Smyth, «The KDD process for extracting useful knowledge from volumes of data», *Commun. ACM*, vol. 39, n.º 11, pp. 27–34, 1996.

[12] J. M. Moine, «Metodologías para el descubrimiento de conocimiento en bases de datos: un estudio comparativo», Facultad de Informática, 2013.

[13] J. Dongre, G. L. Prajapati, y S. V. Tokekar, «The role of Apriori algorithm for finding the association rules in Data mining», en*Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on*, 2014, pp. 657–660.

[14] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A. I. Verkamo, y others, «Fast Discovery of Association Rules.», *Adv. Knowl. Discov. Data Min.*, vol. 12, n.º 1, pp. 307–328, 1996