Descubriendo Variables Relevantes y Reglas de Decisión: Una Aplicación al Conocimiento Botánico

Sonia I. Mariño,
Departamento de Informática, Facultad de Ciencias Exactas y Naturales y Agrimensura, Corrientes, Argentina, Universidad Nacional del Nordeste

Doi: 10.19044/esj.2019.v15n15p425 URL:http://dx.doi.org/10.19044/esj.2019.v15n15p425

Resumen

La habilidad de aprender en un ambiente desconocido o incierto es un componente esencial de un sistema inteligente y es crucial para su funcionamiento. El objetivo del estudio es proponer alternativas en la automatización del proceso de descubrimiento de conocimientos en la fase de Modelado de la metodología CRISP-DM en la cual se incorporan dos pasos: uno para reducir el número de variables relevantes y el segundo para descubrir automáticamente las reglas, en este caso usando Árboles de Decisión. Se adapta CRISP-DM para guiar metodológicamente el proceso inferencial. La propuesta se valida en el proceso de identificación botánica y los resultados obtenidos en la simulación se justifican corroborando la bondad predictiva de estos métodos tanto en el proceso de reducción de variables como al simular al experto en la identificación taxonómica. Finalmente se exponen las conclusiones y algunas relaciones entre los contextos de descubrimiento y validación del proceso científico de identificación.

Palabras Clave: Inteligencia, Árboles de decisión, reducción de variables, Simulación, dominios botánicos

Discovering Relevant Variables and Rules of Decision: An Application to Botanical Knowledge

Sonia I. Mariño,
Departamento de Informática, Facultad de Ciencias Exactas y Naturales y Agrimensura, Corrientes, Argentina, Universidad Nacional del Nordeste

Abstract

The ability to learn in an uncertain or unknown environment is an essential component of intelligent and artificial systems and is crucial to its functioning. This paper focuses on proposing alternatives in the automation of the knowledge discovery process. The Modeling phase of CRISP-DM methodology is incorporated into two steps: one to reduce the number of relevant variables and the second to discover rules automatically, in this case using decision trees. So, CRISP-DM is adapted to methodologically guide the inferencial process. The proposal is validated in the process of botanical identification, and the results obtained in the simulation are justified by corroborating the predictive goodness of these methods, both in the process of reducing variables and by simulating the expert in taxonomic identification. Finally, some conclusions and relationships with the contexts of discovery and validation of the scientific research process are exposed.

Keywords: Artificial Intelligence, Decision trees, reduction of variables, Simulation, Botanical Domains

Introducción

La habilidad de aprender en un ambiente incierto o desconocido es un componente esencial de los sistemas inteligentes naturales y artificiales, y es crucial para su funcionamiento. La complejidad de los entornos cambiantes demanda el diseño de sistemas rebatibles y que simulen la capacidad de aprendizaje de los expertos.

En lo que respecta a la Inteligencia Artificial (IA), los métodos de aprendizaje automático desempeñan un papel central desde sus comienzos. Esto puede deberse a que la habilidad para aprender, adaptarse y modificar el comportamiento, es un componente fundamental de la inteligencia humana (Rudin & Wagstaff, 2014; Somvanshi & Chavan, 2016).

Este artículo se organiza iniciando con una síntesis de las generalidades y los fundamentos en torno al aprendizaje automático y en

particular a los Árboles de Decisión (AD). Además, con la finalidad de vincular este proceso de generación automática de conocimientos con los procesos de la actividad tecno-científica se presentan los contextos asociados a la misma. Se incluye una adaptación de la metodología CRISP-DM para reducir el número de variables como estrategia previa a la construcción automática de Árboles de Decisión. La propuesta se valida en un contexto botánico, exponiendo los resultados obtenidos con una herramienta de software libre para simular el proceder de un experto del dominio. Finalmente, se vierten las discusiones más significativas y posibles líneas de trabajo futuro.

Fundamentos

La presente propuesta de aprendizaje automático se sustenta en cuestiones metodológicas, con la finalidad de relacionarla con uno de los contextos científicos-tecnológicos: el contexto de descubrimiento o innovación, que involucra implícita o explícitamente al contexto de evaluación.

1.1 Los contextos científicos-tecnológicos

Echeverría (1995) amplía la reflexión epistemológica tradicional-que diferencia entre contexto de descubrimiento y contexto de validación-fundamentando su propuesta al considerar a la praxis científica como transformadora del mundo. Distingue cuatro contextos en la actividad tecno-científica: i) el contexto de educación, que implica la enseñanza de conceptos lingüísticos e imágenes científicas, técnicas operatorias y manejo de equipos, y que involucra la difusión y la divulgación, reflejándose en un número mayor de destinatarios que los implicados profesionalmente con el tema; ii) el contexto de innovación; es decir, todo descubrimiento o invento que renueva la realidad; iii) el contexto de evaluación, que amplía la justificación con la valoración del descubrimiento o invento; iv) el contexto de aplicación, que muestra cómo la notabilidad de la ciencia aplicada, la técnica y la tecnología transformaron al mundo actual.

Díaz y Rivera (2013, p.6) mencionan la relevancia de la tecnología como parte del proceso tecno-científico. Es importante, dado que se requiere para la formación de científicos, para la justificación de hipótesis y para la evaluación de las teorías. Y además porque la investigación "básica", en general, está condicionada por la viabilidad de su transferencia hacia la sociedad.

Se propone entonces en este estudio relacionar el contexto de innovación del proceso de la actividad tecno-científica con la construcción de conocimientos, utilizando algoritmos de aprendizaje automático para modelar y simular las inferencias implícitas de los expertos en la toma de decisiones. En este caso se valida con la identificación taxonómica botánica. Lo expuesto

se debe a que estos algoritmos de la IA, descubren conocimiento o patrones proponiendo alternativas para entender la problemática planteada renovando la realidad abstraída. Igualmente, se aborda la vinculación de los contextos de descubrimiento y validación con los procesamientos realizados en un dominio particular de conocimiento.

1.2 Aprendizaje automático

El Aprendizaje Automático o *Machine Learning* aborda el estudio del aprendizaje y las funciones de decisión a partir de ejemplos etiquetados; es decir, requiere una representación que codifica la información sobre el dominio de la función de decisión que se debe aprender (Rudin & Wagstaff, 2014; Somvanshi & Chavan, 2016).

Rudin y Wagstaff (2014:1) mencionan que en "la era de los 'grandes datos', hay una necesidad de aprendizaje automático para hacer frente a importantes problemas de aplicación a gran escala". Implica el uso de algoritmos de aprendizaje que, a partir de los datos, infieren la estructura de relaciones más adecuada, los parámetros del modelo, o ambos simultáneamente, dando lugar a dos posibles soluciones, las estructuras adaptadas a los datos y las estructuras fijas en donde cambian los parámetros del modelo según el conjunto de datos.

Los métodos comprendidos en la denominada Minería de Datos, utilizan tecnologías simbólicas, subsimbólicas y estadísticas.

En la construcción de estos modelos cognitivos intervienen sujetos identificados como el experto en el dominio de conocimiento (EDC), el ingeniero de conocimiento (IC), los analistas y otros perfiles de las Ciencias de la Computación, originando abordajes interdisciplinarios.

En este artículo se incorpora a la fase Modelado de la metodología adaptada de CRISP-DM, el desarrollo de dos pasos para el descubrimiento de conocimiento:

- El primero orientado a descubrir conocimiento utilizando algoritmos para seleccionar variables relevantes, y
- El segundo para descubrir conocimiento utilizando técnicas de Minería de datos. Esta propuesta en particular se centra en la construcción automática de reglas de decisión utilizando Árboles de Decisión (AD). Esta técnica de la IA genera automáticamente reglas de inferencia que modelan las inferencias de los expertos del dominio.

Con la finalidad de validar la propuesta, se modela y simula la identificación de especies vegetales pertenecientes a la familia *Myrtaceae* del NE Argentino. Cabe aclarar que en botánica, el proceso de clasificación computacional se asocia al proceso de identificación de taxones.

1.3 Descubrimiento de conocimiento: algoritmos para seleccionar variables relevantes

Existe una diversidad de algoritmos de aprendizaje automático, los cuales se explotan con distintos fines. El uso adecuado de técnicas de Minería de Datos requiere aplicar, en primer término, aquellas que sirvan para determinar las variables relevantes o contribuyentes, a partir del conocimiento explicitado por el EDC y el IC, y así simular el modo de proceder del experto del dominio. Lo expuesto, permitiría mejorar el desempeño predictivo de los algoritmos de MD (Rosado Gómez et al., 2015; Venkatesh & Anuradha, 2019).

Los algoritmos de selección de variables consideran el conjunto de datos (data set) —definido por el EDC y explicitado por el IC— para obtener una regularidad representativa del dominio definido. Este conocimiento es rebatible, dado que se trabaja con los datos disponibles en un determinado tiempo y espacio.

Por lo expuesto, los métodos para la selección de variables simulan al experto en la determinación de aquellas evidencias relevantes y necesarias para proponer una solución ante un problema de conocimiento planteado.

Una taxonomía los distingue en dos tipos, según se evalúen los atributos, denominados Filtros y Envoltorios (Wrappers). Los primeros seleccionan y evalúan los atributos en forma independiente del algoritmo de aprendizaje; los segundos utilizan el desempeño de un clasificador —que actúa como algoritmo de aprendizaje— para determinar el peso de un subconjunto de atributos.

Para ilustrar el contexto de innovación en referencia al descubrimiento de conocimiento de las variables evidenciales relevantes, se aplicaron 4 algoritmos evaluadores de subconjuntos de atributos, disponibles en una herramienta de análisis inteligente de datos. Para la selección de los atributos se opta por utilizar como Filtros los denominados:

- *CfsSubsetEval*: Evalúa un subconjunto de atributos considerando la habilidad predictiva individual de cada variable, así como el grado de redundancia entre ellas. Se prefieren los subconjuntos de atributos altamente correlacionados con la variable objetivo (especie a identificar en el caso de estudio) y con baja intercorrelación.
- *ConsistencySubsetEval*: Evalúa un subconjunto de atributos según el nivel de consistencia en los valores de la variable objetivo al proyectar las instancias de entrenamiento sobre el subconjunto de atributos.

y, se utilizan como métodos Envoltorios:

- *ClassifierSubsetEval*: Evalúa los subconjuntos de atributos en el conjunto de patrones de entrenamiento o aplicado a un conjunto de prueba independiente, utilizando un clasificador. Se opta por el algoritmo J48 (evolución del algoritmo C4.5 para construir un árbol de decisión).

- *WrapperSubsetEval*: Evalúa los subconjuntos de atributos utilizando como clasificador el algoritmo J48. Se aplica el método de validación cruzada para estimar la exactitud del esquema de aprendizaje en cada conjunto de variables.

Particularmente en esta propuesta, el contexto de descubrimiento se asocia a los procesos de determinación de las variables relevantes y contribuyentes, y se aplican métodos sobre los conjuntos de datos representativos del dominio explicitado por el EDC, comprendidos y capturados por el IC y almacenados en un archivo de trabajo.

1.4 Descubrimiento de conocimiento, generación automática de reglas de decisión

En los problemas de clasificación, diagnóstico o identificación —similares al planteado en este trabajo a fines de validar la propuesta— una de las formas clásicas más utilizadas para la representación y simulación del conocimiento son las reglas de decisión (RD). Una RD es una estructura condicionada IF...THEN en la que se distinguen dos componentes:

- El antecedente, condición que filtra los elementos para los cuales la regla es válida.
- El consecuente, conclusión que establece un hecho sobre todos los elementos que satisfacen el antecedente de la regla.

Existen distintos tipos de condiciones dependiendo de la estructura lógica con que se combinan los componentes (selectores) del antecedente. Las reglas conjuntivas disponen de un antecedente en el que sus selectores se combinan utilizando el "y" lógico, cuya sintaxis general se representa como:

REGLA: A1 y A2 y ... y An ENTONCES Var Obj (x)

Una de las técnicas que demuestra su fiabilidad, para inferir reglas de decisión, son los algoritmos basados en Árboles de Decisión (AD). Este consiste en un grafo dirigido que representa el conocimiento obtenido en el proceso de aprendizaje. La estructura de árbol permite elegir alguna de las hojas —que exhiben diferentes alternativas— considerada como una posible decisión en la resolución de un problema planteado.

En cada nodo se realiza una consulta a una de las características, atributos o variables evidenciales de un ejemplo o caso con el objetivo de asignar una categoría. Es decir, cada nodo interior del árbol contiene una pregunta sobre un determinado atributo, y cada hoja del árbol representa uno de los posibles valores que asume la variable de salida (en este dominio se refiere al nombre científico de una especie). Se parte desde el nodo raíz hasta algún nodo hoja, y se consideran las posibilidades de que el ejemplo en

evaluación pertenezca a una u otra clase. La decisión final depende del nodo hoja en el que se termine, dado que cada uno tiene asociada una clase.

El aprendizaje de Árboles de Decisión se puede ejecutar a través de una diversidad de algoritmos como ID3 (por sus siglas de Interactive Dichotomizer, propuesto por Quinlan, 1986), ASSISTANT y C4.5. Estos algoritmos también se denominan TDIDT (Top-Down Induction of Decision Trees). Se ejecutan con la finalidad de buscar un espacio de hipótesis completamente expresivo y que evite las dificultades de los espacios de hipótesis restringidos. El sesgo inductivo es de preferencia por árboles pequeños sobre aquellos grandes. El algoritmo selecciona los atributos que reducen al mínimo la entropía de clases en el conjunto.

El algoritmo C4.5 (Quinlan, 1993) es una mejora del algoritmo ID3, pues permite clasificar ejemplos con atributos que toman valores continuos. El criterio para seleccionar la variable evidencial con mayor información se fundamenta en el concepto de cantidad de información mutua entre dicha variable y la variable objetivo.

El Árbol de Decisión, es un método de aprendizaje supervisado no paramétrico. Es decir, se conoce el valor que puede asumir la hipótesis, clase o variable objetivo; así la meta del algoritmo clasificador es determinar el valor de dicha clase o variable objetivo para nuevos casos (Truex et al., 2017; Ying et al., 2015; Weka, 2017).

Un AD puede definir una estructura jerárquica compuesta de un conjunto de condiciones o reglas. Ejecuta una evaluación del recorrido de las hojas hasta lograr una decisión. En otras palabras, la decisión se genera al seguir las condiciones que se cumplen desde la raíz hasta alguna de sus hojas. El AD se compone de ramas y nodos:

- Las ramas, indican los posibles caminos generados automáticamente de acuerdo a la decisión tomada.
- El nodo interno contiene una evaluación en referencia a algún valor de una de las propiedades.
- El nodo de probabilidad indica que debe ocurrir un evento aleatorio de acuerdo a la naturaleza del problema.
- El nodo hoja representa el valor que devolverá el árbol de decisión.

Originalmente un Árbol de Decisión se construía aplicando los algoritmos ID3 y C4.5 desarrollados por Quinlan (1993).

El algoritmo C4.5 construye un Árbol de Decisión, que determina los campos significativos. Es decir, el algoritmo realiza particiones en forma recursiva aplicando la estrategia *primero en profundidad* o *depth first* (Quinlan, 1993). En sucesivas pruebas, donde se comparan los ejemplos del conjunto de datos, el algoritmo busca aquellos ejemplos que tienen la mayor ganancia de información. En el caso de atributos discretos, la prueba considera

una cantidad de resultados teniendo en cuenta el número de valores posibles que puede tomar el atributo.

Por ello, la ejecución del algoritmo brinda las ramas que contienen los atributos significativos; es decir, los más representativos del conjunto de datos y que aportan más información para clasificar al atributo objetivo (que implica identificar la especie, en el presente estudio). La Figura 1 ilustra el pseudocódigo original del algoritmo C4.5.

En el trabajo de Rosado Gomez y Verjel Ibañez (2015), se menciona que entre los árboles podados posibles se debe seleccionar el mejor o solución. Este será aquel que obtenga "el menor error en el ajuste de los registros utilizados en su proceso de aprendizaje"; además debe ajustar la base de datos utilizada en su aprendizaje y aquellos registros definidos para el proceso de validación o testeo.

Cada una de las ramas finaliza en un nodo, que representa el valor de una variable objetivo; es decir, la especie en este caso de estudio. Cada rama se transforma en una regla de decisión, y éstas pueden utilizarse para conformar la Base de Conocimiento de un sistema experto.

En la construcción de árboles de decisión la herramienta WEKA proporciona el algoritmo J48 basado en el algoritmo C4.5. Este algoritmo amplía las funcionalidades del algoritmo C4.5, entre las que se mencionan: realizar el proceso de post-poda del árbol mediante un método para la reducción del error (*reducedErrorPruning*); las divisiones sobre las variables discretas son siempre binarias (*binarySplits*). Entre algunas propiedades de la implementación se mencionan (Quinlan, 1993; Weka, 2017): i) Admite atributos simbólicos y numéricos, aunque la clase debe ser simbólica. ii) Se permiten ejemplos con valores desconocidos. iii) El criterio de división se basa en la entropía y la ganancia de información. iv) Se define la entropía (incertidumbre, desorden, impureza) de un conjunto de ejemplos S. Para problemas de múltiples clases con *c* categorías, se generaliza la entropía a:

$$Entropia(S) = \sum_{i=1}^{c} -p_i \log_2(p_i)$$

Ecuación 1. Entropía

donde

p_i es la fracción de ejemplos positivos en S de la clase i

Si todos los ejemplos están en una categoría, la entropía es 0

Si los ejemplos están igualmente mezclados, la entropía es una máxima de 1.

La ganancia o el incremento de la información *Ganancia(S, F)* ocurre por reducción en la entropía esperada al ordenar el conjunto *S* basado en el atributo *F:*

$$Ganancia(S,F) = \text{Entropia (S)} - \sum_{v=valores(F)} \frac{|S_v|}{|S|} \text{Entropia } (S_v)$$

Ecuación 2. Ganancia de la información

Se define *Sv* al subconjunto de *S* que tiene el valor *v* para la característica *F*.
Es decir, la entropía de cada subconjunto resultante se pondera por su tamaño relativo.

```
(R=Conjunto de atributos no clasificados, C=atributo clasificador, S=conjunto de entrenamiento)
Comienzo
   Si S = vacío,
        Devolver un único nodo con Valor Falla;
   Si todos los registros de S tienen el mismo valor para C,
        Devolver un único valor con el valor más frecuente de C en los registros de S;
   Si R = vacío,
        D <- atributo con mayor proporción de ganancia (D;S) entre los atributos de R;
        Sean {dj | j=1,2,….., m} los valores del atributo D;
        Sean {dj | j=1,2,….., m} los subconjuntos de S correspondientes a los valores de dj respectivamente;
        Devolver un árbol con la raíz nombrada como D y con los arcos nombrados d1,d2,…..,dm, que van respectivamente a los árboles
        C4.5 (R-{D}, C, S1), C4.5 (R-{D}, C, S2), C4.5 (R-{D}, C, Sm);
Fin.
```

Figura 1. Pseudocódigo original del algoritmo C4.5 (Fuente: otros)

Algoritmo RTREE

El algoritmo RandomTree o RTree, construye un árbol dibujado al azar a partir de un juego de árboles posibles. En este contexto "al azar" significa que cada árbol en el juego de árboles tiene igual posibilidad de ser probado; es decir, la distribución de árboles es "uniforme". Por lo expuesto, este algoritmo construye un árbol que considera un número aleatorio de características dadas en cada nodo.

Metodología

El enfoque metodológico adoptado en el trabajo consistió en seleccionar un dominio de conocimiento y ejemplares botánicos representativos del mismo. Se adaptó el método CRISP-DM (Chapman et al., 2000) para seleccionar las variables relevantes, y simular y validar el proceso de identificación taxonómica utilizando Árboles de Decisión. A continuación se indican las fases seguidas:

Fase 1. Definición, análisis y comprensión del negocio

En esta fase se determinó el dominio de aplicación. Se plantearon como interrogantes que orientan la indagación utilizando la técnica elegida: ¿Qué se desea obtener de la información recopilada?, y particularmente ¿Cuáles son las reglas que permiten determinar la pertenencia de un ejemplar a un taxón?

Fase 2. Identificación de la fuente de información y conjunto de entrenamiento

El experto en el dominio proporcionó el conjunto de datos para el entrenamiento y comprobación de los modelos inferenciales. En el caso de estudio, un número importante de esos ejemplares botánicos han sido identificados por especialistas en la familia, lo que hace que las nuevas identificaciones puedan ser corroboradas por comparación con testigos fidedignos. Se adoptó como criterio, incorporar la mayor cantidad de caracteres posibles para facilitar la identificación de ejemplares en los que, por no poseer frutos, se desconoce el tipo de embrión, carácter de gran importancia para la identificación en esta familia.

En esta fase se eligió el conjunto de datos sobre el cual se modela y simula el problema. Se trabajó con un especialista del dominio, quien seleccionó los ejemplares de datos y determinó las variables evidenciales representativas. Se consideraron, como variables evidenciales, los caracteres seleccionados por el especialista de conocimiento. La variable objetivo asume los distintos valores de 31 especies de Mirtáceas y las variables evidenciales y sus posibles valores, así como los valores que puede asumir la variable objetivo (nombre científico de la especie a identificar) (Mariño & Alfonzo, 2016; Mariño, 2019).

Se prepararon los datos según el formato de procesamiento requerido por la herramienta software de modelado y simulación computacional. Esta actividad también se denomina pre-procesamiento de los datos (Rusell & Norving, 2004). Se consideraron actividades concernientes a:

- *Construir el conjunto de datos.* Esta fase involucró actividades relacionadas con la obtención del conjunto de datos final, el cual se utiliza como datos de entrada. Las tareas se aplicaron múltiples veces y sin un orden pre-establecido. Incluyeron extracción, transformación y carga, proceso conocido como ETL. Extraídos los datos de la fuente de la información, se procedió a su transformación o conversión a un formato legible por la herramienta de MD.
- *Estimar los estadísticos sobre los atributos.* Registrados los datos, desde la herramienta de MD, se reconocieron los atributos y computaron algunas estadísticas básicas sobre cada atributo en el análisis de los datos. Dado que el conjunto de datos elegidos son

atributos continuos/numéricos, se visualizaron valores mínimo, máximo, media, desviación estándar, entre otros.

- *Seleccionar variables relevantes.* Los expertos del dominio brindaron conocimiento de aquellas variables requeridas para modelar y simular la problemática. Se aplicaron métodos automáticos para restringir la dimensionalidad del problema; en este sentido se eligieron evaluadores de atributos asociados a los métodos de búsqueda. Los resultados de estos procesamientos se muestran en las Tablas 1 y 2.

Fase 3. Modelado

Para modelar una abstracción de la realidad se seleccionaron y aplicaron distintas técnicas considerando los dos pasos propuestos. El primero para descubrir conocimiento utilizando algoritmos de selección de variables relevantes, y el segundo para descubrir conocimiento utilizando técnicas de Minería de datos, en particular Árboles de Decisión.

Es por ello que el Modelado de las tareas de Minería de Datos se aplicó para disminuir la dimensionalidad de variables evidenciales y, a continuación para determinar las reglas que componen un Árbol de Decisión. En ambos pasos del método propuesto:

- Se plantearon diversos experimentos, en los cuales se modificaron algunos de los parámetros.
- Se simularon, interpretaron y valoraron los resultados, confrontándose los valores obtenidos con los especificados por el especialista.
- Se aplicaron las siguientes medidas de calidad para la elección del modelo. Estas métricas aportan información para decidir respecto a aquel modelo que representa una mejor aproximación del problema simulado:
- Estadístico Kappa, es una medida de concordancia entre las categorías pronosticadas por el clasificador y las categorías observadas, considera las posibles concordancias debidas al azar. La valoración del índice Kappa está dada: i) Si el valor es 1: Concordancia perfecta; ii) Si el valor es 0: Concordancia debida al azar; iii) Si el valor es negativo: Concordancia menor que la que cabría esperar por azar.

$$K = \frac{Pr(a) - \Pr(e)}{1 - \Pr(e)}$$

Ecuación 3 Índice Kappa

- Error Raíz Cuadrada de la media (Root Mean Squared Error o RMSE). También, se puede denominar RMSD o *Root Mean Squared*

Desviation. Es una medida de precisión que establece las diferencias entre los valores calculados por un modelo o un estimador y los valores observados. Así, permite comparar diferentes errores de predicción de un mismo conjunto de datos, dado que es dependiente de la escala muestra. En la Ecuación 4, y_i es el valor observado, $\hat{y}$ es la salida de la red para el vector de entrada, y n es el número de residuales.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \|y_i - \hat{y}_i\|^2}{n}}$$

Ecuación 4 Medida de calidad Error Cuadrático Medio

- Error Absoluto Medio (Mean Absolute Error o MAE), mide la diferencia entre el valor medio obtenido y el hallado en esa media. El promedio de error absoluto, es la suma de los errores absolutos de clasificación en cada uno de los elementos llevados a promedio. El clasificador que proporciona la mayor cifra (superior a 0.1) define un error de clasificación alto, por lo cual se debe considerar sobre aquellos que arrojen una cifra menor (Ecuación 5).

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \|f_i - y_i\| = \frac{1}{n}\sum_{i=1}^{n} \|e_i\|$$

Ecuación 5 Medida de calidad Error Absoluto Medio

- Se establecieron los parámetros de selección de los modelos. Se definieron los siguientes valores como criterios de elección de los modelos: Clasificación correcta > 90 % instancias; Clasificación incorrecta < 10 % instancias; MAE <0.1 ideal; estadístico de Kappa >0.79, >0.9 ideal; RMSE < 0.3, <1 ideal. El clasificador que cumpla con estas especificaciones se considera representativo para su integración en un sistema inteligente. Si los modelos proporcionan un mismo valor en la métrica MAE, se opta por el menor valor de RMSE en la elección del modelo más representativo.
- Se comprobaron los modelos. Entrenado el conjunto de datos, se verificó utilizando la técnica denominada Porcentaje de división o *Percent Split* en la herramienta de Minería de Datos seleccionada. Esta opción divide los datos en dos grupos, el porcentaje especificado representa las instancias utilizadas para construir el modelo, y éste se evalúa respecto a las restantes. Si el número de instancias es suficientemente elevado, esta opción es suficiente para estimar con precisión las prestaciones del clasificador en el dominio.

Fase 4. Implementación

Numerosas herramientas de análisis de datos aplican procesos de minería de datos para descubrir y validar conocimientos automáticamente. En este trabajo se optó por Weka.

También esta fase implica el despliegue de la solución tecnológica. Los modelos entrenados y validados son accesibles desde la herramienta para simular nuevas situaciones y proponer alternativas ante nuevos casos no entrenados para apoyar la toma de decisiones.

Resultados

A continuación, se relacionan los contextos del proceso de la investigación científica-tecnológica con los momentos que involucran descubrir y validar el conocimiento en un dominio botánico (Mariño, 2019).

En referencia al contexto de descubrimiento, éste se desarrolla en dos pasos, los cuales se incorporan a la Fase 3 Modelado de la metodología: i) en la detección automática de las variables relevantes, y ii) en la simulación del proceder de los EDC, aplicando algoritmos que descubren el conjunto de reglas de comportamiento para un proceso de identificación taxonómica. Se correspondería a una de las primeras fases de un proceso de Ingeniería de Conocimiento, donde los expertos (EDC) explicitan su conocimiento al ingeniero en conocimiento (IC), quien los captura y representa en un conjunto de ejemplos de datos, patrones o reglas.

El contexto de validación se ejecuta para cada uno de los procesamientos aplicando los métodos de validación cruzada, con la finalidad de contrastar los resultados obtenidos y los definidos por el especialista.

Paso 1. Detección automática de las variables relevantes

El proceso de aprendizaje automático se aplica para reducir el número de variables evidenciales intervinientes. Como producto del procesamiento de la información se obtienen distintos modelos —constituidos por las variables evidenciales relevantes seleccionadas— útiles para lograr el aprendizaje automático (contexto de innovación) y la identificación de nuevos ejemplares botánicos, en posteriores procesos de reconocimiento (contextos de evaluación y aplicación).

Los modelos obtenidos se resumen en la Tabla 1. La primera columna indica los algoritmos evaluadores de atributos seleccionados y ejecutados en combinación con el método de búsqueda (segunda columna) *Best First forward* (primero mejor hacia adelante), que localiza los atributos más relevantes en el espacio de los subconjuntos, utilizando la estrategia greedy hillclimbing con backtracking. Se parte del conjunto vacío de atributos, se establece que el proceso debe ampliar la búsqueda del nodo en 5 niveles de

profundidad. Se opta, como modo de selección, por la evaluación sobre el conjunto total de los datos de entrenamiento.

El resultado del procesamiento de los datos se especifica en las dos últimas columnas (Tabla 1), donde la columna Nro V E representa el número de evidencias seleccionadas. La columna Evidencias resume las variables evidenciales o atributos obtenidos por cada método de selección. Una descripción completa de las mismas se observa en la Tabla 2. Los códigos de las variables evidenciales expresadas en la última columna de la Tabla 1, indican los atributos que mejor explican los posibles valores que asume la variable objetivo (nombre de la Especie) en un proceso de identificación botánica.

Un análisis de la Tabla 1, evidencia cómo el conjunto de variables evidenciales se reduce notablemente en el 75% de los modelos creados. Se detecta que la diferencia entre los modelos 2 y 3 comparado con el modelo 4 reside en el número de subconjuntos de atributos evaluados.

Además, la frecuencia con la cual se seleccionan automáticamente las variables permite inferir cuáles son aquellas más contribuyentes para la distinción de la variable objetivo, que representa a los taxones botánicos. Estos subconjuntos de evidencias se podrían emplear como modelos representativos para simular el razonamiento de los especialistas.

La Tabla 2 sintetiza los resultados obtenidos al validar los modelos creados y explicitados en la Tabla 1 utilizando un conjunto de ejemplos de prueba. Se indica para cada conjunto de variables evidenciales el nivel de probabilidad que aporta para definir el nivel de certeza de estos modelos.

Tabla 1. Modelos de mínimas evidencias (Evaluación conjunto entrenamiento)

Evaluadores de Atributos	**Métodos de Búsqueda**	**Nº V.E**	**Variables Evidenciales Seleccionadas**
CfsSubsetEval	BestFirst	31	id_porte, id_lamina1a, id_lamina1b, id_lamina1c, id_lamina1e, id_lamina2b, id_lamina2c, id_lamina2d, id_epifilo1, id_hipofilo, id_lam_apice1, id_lam_apice2, id_lam_apice3, id_lam_base1, id_lam_base3, id_lam_base4, id_lam_base5, id_caliz1, id_caliz4, id_sepalos1, id_sepalos2, id_sepalos3, id_sepalos4, id_hipanto1, id_hipanto2, id_inflor2, id_inflor4a, id_inflor4b, id_inflor4c, id_inflor4d, id_inflor7a
ConsistencySubsetEval	BestFirst	8	id_lamina1b, id_ lamina1c, id_lamina2c, id_lam_apice2, id_lam_base4, id_caliz1, id_inflor2, id_inflor4a
ClassifierSubsetEval.J48	Best first.	9	id_lamina1a, id_ lamina1b, id_lamina1c, id_lam_apice2, id_lam_base4, id_caliz1,

Evaluadores de Atributos	Métodos de Búsqueda	Nº V.E	Variables Evidenciales Seleccionadas
			id_sepalos4, id_inflor2, id_inflor4a
WrapperSubsetEval.J48	Best first	11	id_hoja1, id_lamina1a, id_lamina1e, id_lamina2b, id_lamina2c, id_lamina3, id_lam_apice2, id_lam_base4, id_caliz1, id_inflor2, id_inflor4a

Tabla 2. Modelos de mínimas evidencias, Evaluación cruzada

Evaluadores de Atributos	Métodos de Búsqueda	Nº V.E	Variables Evidenciales Seleccionadas
CfsSubsetEval	BestFirst	27 (100%)	id_porte, id_lamina1b, id_lamina1e, id_lamina2b, id_lamina2c, lamina2d, id_epifilo1, id_hipofilo, id_lam_apice1, id_lam_apice2, id_lam_apice3, id_lam_base1, id_lam_base3, id_lam_base4, id_lam_base5, id_caliz1, caliz4, id_sepalos1, id_sepalos3, id_sepalos4, id_hipanto1, id_hipanto2, id_inflor2, id_inflor4b, id_inflor4c, id_inflor4d, id_inflor9a
		—90%—	
		5	id_lamina1a, id_lamina1c, id_l_base6, id_sepalos2 id_inflor7a
ConsistencySubsetEval	BestFirst	8	id_lamina1b, id_inflor4a, id_caliz1 id_lamina1c (80%), id_lam_apice2 (80%), l_base4 (80%), id_lamina2c (70%), id_inflor2 (60%),
ClassifierSubset Eval	J48 Best first.	8	id_lamina1b, id_caliz id_lam_apice2, id_lam_base4 (80%) id_sepalos, id_inflor2, id_lamina1c(60%) id_inflor3 (50%)
WrapperSubsetEval	J48 Best first	6	id_caliz1, id_inflor4a (80 %) id_lamina1b (60%), id_lamina1e (60%), id_lam_apice2 (60%), id_sepalos3 (50%),

Paso 2. Descubrimiento del conjunto de reglas de pertenencia

Seleccionadas las variables relevantes evidenciales que intervienen en la modelización y simulación de la situación planteada, se procede a aplicar algoritmos para descubrir el conjunto de reglas de pertenencia de cada grupo denotado por la clase correspondiente —nombre científico del taxón—.

En el algoritmo J48, un parámetro importante es el factor de confianza para la poda. Éste influye en la capacidad de predicción del árbol construido. En cada operación de poda, se define la probabilidad de error admitida para la hipótesis (variable objetivo) dado que el empeoramiento por esta operación es

significativo. Si se define una probabilidad menor, se determinará que la diferencia en los errores de predicción antes y después de podar sea más significativa para no podar. Por defecto se establece como valor del parámetro: 25%. Si se disminuye este valor, se permiten más operaciones de poda (Weka, 2017). En el estudio se establece como valor del factor de confianza para la poda 10%.

El Árbol de Decisión genera reglas —representativas de un razonamiento lógico o causal— que se infieren automáticamente a partir de un conjunto de datos que caracterizan el dominio (Rusell & Norving, 2004; Zhong, 2016). El algoritmo se puede ejecutar sobre el conjunto compuesto por todas la variables evidenciales (MTA) o a un subconjunto de estas y seleccionado como relevante por el algoritmo.

La Tabla 3 resume los modelos obtenidos utilizando como técnica de descubrimiento de conocimiento los algoritmos basados en AD. En la columna 1 se especifican los algoritmos elegidos, la columna 2 indica si se utilizan las variables seleccionadas por el experto (Mariño, 2019) o las inferidas por el método de aprendizaje automático (Tabla 1) debido a que el número se reduce considerablemente.

En el contexto de validación –situado en el proceso de investigación científica-tecnológica-, se verifica el comportamiento de los modelos mediante la aplicación del método de Validación Cruzada con 10 pliegos (Tabla 4). Esta técnica permite evaluar los resultados del análisis estadístico y garantiza la independencia de la partición entre los datos de entrenamiento y prueba. En las tablas: la primera columna indica el algoritmo elegido para la construcción del árbol; la segunda el número de variables evidenciales seleccionadas para el entrenamiento de los modelos; la tercera el número de hojas o reglas resultantes si correspondiera; la cuarta el tamaño del árbol y las tres últimas métricas que facilitan la elección del mejor modelo de representación: Kappa, MAE, RMSE.

La Figura 2, muestra el árbol de decisión completo obtenido al aplicar el algoritmo J.48 sobre el modelo reducido y compuesto por 31 variables (variables indicadas en la Fila 1 de la Tabla 1). Este despliega el conjunto de reglas inferidas automáticamente y que compondrían la Base de Conocimiento de un sistema experto simbólico.

La Figura 3 presenta el árbol de decisión, creado automáticamente, con la ejecución del algoritmo RTree y aplicado al conjunto de mínimas variables (Fila 1 de la Tabla 1), información que sustenta la determinación de las reglas que componen una Base de Conocimiento.

Un análisis de los Árboles de Decisión generados automáticamente (Figuras 2 y 3) permite inferir que cada algoritmo selecciona como relevantes distintas variables evidenciales para iniciar la construcción de los modelos y simular el conocimiento experto.

Tabla 3. Entrenamiento de modelos basados en Árboles de Decisión

Algoritmo	Nro variables	Tamaño del árbol	Corr.	Corr. %	Ka	MAE	RMSE
J.48	65	59	422	99.52	0.9948	0.0005	0.0152
RTree	65	64	423	99.76	0.9974	0.0002	0.0087
J.48	31	39	422	99.52	0.9948	0.0005	0.0152
Rtree	31	71	423	99.76	0.9974	0.0002	0.0087

Tabla 4. Validación Cruzada de los modelos (número de pliegues 10)

Algoritmo	Nro variables	Tamaño del árbol	Corr.	Corr. %	Ka	MAE	RMSE
J.48	66	59	416	98.11	0.9793	0.0014	0.0336
Rtree	66	62	418	98.58	0.9845	0.0009	0.0297
J.48	31	59	414	97.64	0.9742	0.0017	0.0381
Rtree	31	71	420	99.05	0.9897	0.0007	0.0254

Para fundamentar la elección del modelo aprendido, se establece como métrica de evaluación de la calidad o nivel de confianza el estadístico Kappa (Ka). Éste mide el nivel de predicción respecto a la variable objetivo. Además, se consideran las instancias clasificadas correctamente y los errores asociados al clasificador como son las métricas MAE y RMSE.

Por ejemplo, para validar el modelo indicado en la fila 3 de la Tabla 3 (paso 2 de la propuesta), se utilizan las 31 variables seleccionadas y explicitadas en la fila 1 de la Tabla 1 (paso 1). Al contrastar los resultados, se determina el correcto comportamiento de los métodos de selección, dado que minimizando el número de variables se obtienen resultados que superan el valor del índice Kappa en un 99%. Lo expuesto, evidencia distintos momentos de descubrimiento y validación de conocimiento asociados a los contextos científicos – tecnológicos.

Las Figuras 2 y 3 presentan los árboles de decisión descubiertos automáticamente, aplicando como métodos de Aprendizaje Automático los algoritmos J.48 y RTree. En las figuras, se observa que los algoritmos seleccionan diferentes variables relevantes para iniciar el descubrimiento de conocimiento y generan distintas ramas componentes del árbol de decisión.

Con el propósito de ilustrar la transformación explícita del conocimiento, encapsulado en un árbol de decisión en reglas de una Base de Conocimiento de un sistema experto, a continuación se analiza la semántica correspondiente a distintas ramas generadas automáticamente (Figura 4).

```
LAMINA1B <= 1
|  LAMINA2D <= 1
|  |  L_BASE3 <= 1: 34 (54.0)
|  |  L_BASE3 > 1
|  |  |  SEPALOS1 <= 1
|  |  |  |  LAMINA1A <= 1: 60 (17.0)
|  |  |  |  LAMINA1A > 1: 56 (3.0/1.0)
|  |  |  SEPALOS1 > 1
|  |  |  |  LAMINA1A <= 1: 48 (18.0)
|  |  |  |  LAMINA1A > 1: 58 (2.0)
|  LAMINA2D > 1
|  |  LAMINA1A <= 1
|  |  |  LAMINA1C <= 1: 26 (6.0)
|  |  |  LAMINA1C > 1
|  |  |  |  LAMINA1E <= 1: 40 (6.0)
|  |  |  |  LAMINA1E > 1: 6 (3.0)
|  |  LAMINA1A > 1
|  |  |  LAMINA2B <= 1
|  |  |  |  LAMINA1E <= 1: 14 (7.0/1.0)
|  |  |  |  LAMINA1E > 1
|  |  |  |  |  HIPOFILO <= 1
|  |  |  |  |  |  PORTE <= 1: 46 (3.0)
|  |  |  |  |  |  PORTE > 1: 20 (9.0)
|  |  |  |  |  HIPOFILO > 1: 42 (6.0)
|  |  |  LAMINA2B > 1
|  |  |  |  LAMINA2C <= 1: 12 (3.0)
|  |  |  |  LAMINA2C > 1: 32 (18.0)
LAMINA1B > 1
|  LAMINA2B <= 1
|  |  LAMINA2C <= 1
|  |  |  LAMINA2D <= 1: 8 (9.0)
|  |  |  LAMINA2D > 1
|  |  |  |  LAMINA1E <= 1
|  |  |  |  |  L_APICE2 <= 1: 10 (81.0)
|  |  |  |  |  L_APICE2 > 1: 16 (3.0)
|  |  |  |  LAMINA1E > 1: 2 (6.0)
|  |  LAMINA2C > 1
|  |  |  LAMINA1A <= 1
|  |  |  |  LAMINA1C <= 1
|  |  |  |  |  L_APICE1 <= 1: 30 (18.0)
|  |  |  |  |  L_APICE1 > 1: 18 (2.0)
|  |  |  |  LAMINA1C > 1: 44 (6.0)
|  |  |  LAMINA1A > 1: 54 (18.0)
|  LAMINA2B > 1
|  |  LAMINA1A <= 1
|  |  |  LAMINA2C <= 1: 36 (9.0)
|  |  |  LAMINA2C > 1
|  |  |  |  L_APICE2 <= 1
|  |  |  |  |  INFLOR4A <= 1
|  |  |  |  |  |  LAMINA1C <= 1: 38 (3.0)
```

```
| | | | | | LAMINA1C > 1: 22 (27.0)
| | | | | INFLOR4A > 1: 24 (54.0)
| | | | L_APICE2 > 1
| | | | | L_APICE1 <= 1: 62 (6.0)
| | | | | L_APICE1 > 1: 28 (3.0)
| | LAMINA1A > 1
| | | L_APICE1 <= 1: 50 (18.0)
| | | L_APICE1 > 1: 4 (6.0)
```

Figura 2. Árbol de Decisión descubierto al aplicar el algoritmo J48, 31 var. (Fuente: Elaboración propia)

```
INFLOR4A < 1.5
| INFLOR4D < 1.5
| | LAMINA2D < 1.5
| | | LAMINA2C < 1.5 : 36 (9/0)
| | | LAMINA2C >= 1.5 : 34 (54/0)
| | LAMINA2D >= 1.5
| | | INFLOR4B < 1.5
| | | | L_APICE3 < 1.5
| | | | | LAMINA1A < 1.5
| | | | | | LAMINA1B < 1.5 : 26 (6/0)
| | | | | | LAMINA1B >= 1.5 : 16 (3/0)
| | | | | LAMINA1A >= 1.5
| | | | | | SEPALOS4 < 1.5 : 14 (6/0)
| | | | | | SEPALOS4 >= 1.5 : 46 (3/0)
| | | | L_APICE3 >= 1.5 : 22 (27/0)
| | | INFLOR4B >= 1.5
| | | | L_BASE4 < 1.5
| | | | | LAMINA1B < 1.5 : 20 (9/0)
| | | | | LAMINA1B >= 1.5 : 30 (18/0)
| | | | L_BASE4 >= 1.5 : 12 (3/0)
| INFLOR4D >= 1.5
| | INFLOR2 = 0 : 2 (0/0)
| | INFLOR2 = 1
| | | HIPOFILO < 1.5 : 32 (18/0)
| | | HIPOFILO >= 1.5 : 42 (6/0)
| | INFLOR2 = 2
| | | LAMINA2B < 1.5
| | | | LAMINA1B < 1.5
| | | | | CALIZ4 < 1.5 : 40 (2/0)
| | | | | CALIZ4 >= 1.5
| | | | | | PORTE < 1.5
| | | | | | | CALIZ4 < 2.5
| | | | | | | | LAMINA1A < 1.5 : 40 (1/0)
| | | | | | | | LAMINA1A >= 1.5 : 52 (1/0)
| | | | | | | CALIZ4 >= 2.5 : 40 (1/0)
| | | | | | PORTE >= 1.5 : 40 (2/0)
| | | | LAMINA1B >= 1.5
| | | | | LAMINA1E < 1.5
| | | | | | L_APICE1 < 1.5 : 44 (6/0)
```

```
| | | | | | L_APICE1 >= 1.5 : 18 (2/0)
| | | | | LAMINA1E >= 1.5 : 2 (6/0)
| | | LAMINA2B >= 1.5
| | | | SEPALOS2 < 1.5 : 4 (6/0)
| | | | SEPALOS2 >= 1.5
| | | | | LAMINA1C < 1.5 : 38 (3/0)
| | | | | LAMINA1C >= 1.5 : 50 (18/0)
INFLOR4A >= 1.5
| INFLOR2 = 0
| | LAMINA1B < 1.5
| | | SEPALOS1 < 1.5 : 60 (17/0)
| | | SEPALOS1 >= 1.5
| | | | LAMINA1A < 1.5 : 6 (3/0)
| | | | LAMINA1A >= 1.5 : 58 (2/0)
| | LAMINA1B >= 1.5
| | | CALIZ1 < 1 : 8 (9/0)
| | | CALIZ1 >= 1 : 10 (81/0)
| INFLOR2 = 1
| | LAMINA2D < 1.5 : 48 (18/0)
| | LAMINA2D >= 1.5 : 28 (3/0)
| INFLOR2 = 2
| | L_BASE5 < 1.5
| | | PORTE < 1.5
| | | | LAMINA2C < 1.5 : 56 (2/1)
| | | | LAMINA2C >= 1.5 : 24 (27/0)
| | | PORTE >= 1.5
| | | | HIPANTO2 < 1.5 : 56 (1/0)
| | | | HIPANTO2 >= 1.5 : 24 (27/0)
| | L_BASE5 >= 1.5
| | | L_APICE2 < 1.5 : 54 (18/0)
| | | L_APICE2 >= 1.5 : 62 (6/0)
```

Figura 3. Árbol de decisión inferido ejecutando el algoritmo RTree, 31 variables (Fuente: Elaboración propia)

```
LAMINA1B <= 1
| LAMINA2D <= 1
| | L_BASE3 <= 1: 34 (54.0)
| | L_BASE3 > 1
| | | SEPALOS1 <= 1
| | | | LAMINA1A <= 1: 60 (17.0)
| | | | LAMINA1A > 1: 56 (3.0/1.0)
| | | SEPALOS1 > 1
| | | | LAMINA1A <= 1: 48 (18.0)
| | | | LAMINA1A > 1: 58 (2.0)
```

A continuación se explicita la regla que se deriva (de las expresiones indicados en letras negritas) como resultado de ejecutar el algoritmo de árbol de decisión sobre el conjunto de datos seleccionados

Si id_lamina1b (lámina coriácea) es menor o igual a 1
Y LAMINA2D (lámina oblonga) es menor o igual a 1
Y L_BASE3 (base no obtusa) es menor o igual a 1
Entonces el ejemplar corresponde a la especie *Eugenia myrcianthes*

Figura 4. Ejemplo de una rama del AD y su expresión en un conjunto de reglas (Fuente: Elaboración propia)

Conclusión

La literatura expone una diversidad de métodos para modelar y simular el razonamiento humano en procesos decisorios. Uno de ellos, involucra a los expertos en dominios del conocimiento científico como es el botánico.

En este trabajo se expusieron los resultados de aplicar un método desarrollado en dos pasos: en primer lugar para reducir el número de variables relevantes y, a continuación, determinar las reglas aplicando Árboles de Decisión como estrategia inferencial. Los métodos computacionales seleccionados se aplicaron a los registros representativos y contenidos en una matriz de datos que encapsula el conocimiento y experticia de un especialista del dominio.

Las métricas definidas para determinar la efectividad en la identificación de especies: el estadístico Kappa, el Error Medio Absoluto y la Raíz del Error Cuadrático Medio, brindaron resultados que sostienen la elección de este método para modelar al experto del dominio. Los resultados de la Tabla 3 ilustran el alto grado de certeza (superior al 90%) obtenido en los procesos de aprendizaje y validación. Es decir, aplicar los algoritmos J.48 y RTree a un mínimo conjunto o al conjunto total de variables evidenciales, produce similares valores de los estadísticos y en los porcentajes de aciertos, aseverando el buen comportamiento de estos métodos.

Lo expuesto también demuestra la validez de los algoritmos para detectar las variables relevantes de un dominio y así contrastar con la experticia del especialista del dominio, sosteniendo procesos metodológicos como el descrito. Por lo expuesto, estas experimentaciones permitieron comprobar que los algoritmos seleccionados funcionaron adecuadamente tanto en procesos de descubrimiento como en procesos de validación para disminuir el número de variables y, a continuación, para inferir las reglas aplicando Árboles de Decisión.

Los Árboles de Decisión proporcionaron el conjunto de ramas, en el cual cada una de ellas finaliza en un nodo que representa el valor de la variable objetivo; es decir, en este caso de estudio, la especie a la que corresponde. Cada rama se transforma en una regla de decisión, que conformaría la Base de Conocimiento de un sistema experto.

Desde un enfoque interdisciplinario, se observa la analogía entre las reglas de decisión construidas automáticamente y las reglas inferibles de una

Clave Dicotómica. Esta última, es una herramienta de apoyo a la toma de decisiones ampliamente utilizada en procesos de identificación botánicos, por lo que la presente propuesta se podría considerar como una estrategia alternativa para apoyar la toma de decisiones.

Se continuará con estudios similares al expuesto, con la finalidad de diseñar soluciones utilizando diversos algoritmos y abordar estudios comparativos en torno al proceso de identificación taxonómica, mediado por tecnologías de la Inteligencia Artificial.

Agradecimiento

Se agradece a la Lic. Sara G. Tressens quien como experta en el dominio botánico brindó la fuente de datos para la elaboración del presente trabajo.

References:

1. Cardona Taborda, C., Gelvez García, N., & Palacios Rozo, J. (2016). Análisis de datos mediante el algoritmo de clasificación J48, sobre un cluster en la nube de AWS. *Redes de Ingeniería*, 3-15.
2. Chapman, P., Clinton, J., Keber, R., Khabaza, T., Reinartz, T., Shearer, C. & Wirth, R. (2000). CRISP-DM 1.0 Step by step BIguide. Edited by SPSS. Documento en línea. Disponible en: http://www-staff.it.uts.edu.au/~paulk/teaching/dmkdd/ass2/readings/methodology/CRISPWP-0800.pdf
3. Díaz, E. & Rivera, S. (2013). *Algunas consideraciones para una ética aplicada a la investigación científica*. Disponible en: www.estherdiaz.com.ar/textos/etica_investigacion.htm [Consultado 12-10-2016]
4. Echeverría, J. (1995). *Filosofía de la ciencia*, Madrid- España: Ed. Akal.
5. Godoy Viera, A. F. (2017). Técnicas de aprendizaje de máquina utilizadas para la minería de texto. *Investigación Bibliotecológica: archivonomía, bibliotecología e información***,** [S.l.], 31(71), pp. 103-126, Disponible en: <http://rev-ib.unam.mx/ib/index.php/ib/article/view/57812/51821>. doi:http://dx.doi.org/10.22201/iibi.0187358xp.2017.71.57812.
6. Mariño, S. I. & Alfonzo, P. L. (2016). Simulación del razonamiento en el proceso de identificación botánica basado en redes bayesianas, *Investigación Operativa*, 24(39), pp. 55-72.
7. Mariño, S. I. (2019). *Modelo de gestión de conocimiento como apoyo a la toma de decisiones basado en una tecnología inteligente. Descubrimiento y validación en dominios botánicos*, Tesis de

Doctorado para acceder al título de Doctor en Ciencias Cognitivas, Universidad Nacional del Nordeste, Inédito.

8. Rosado Gomez, A. A. & Verjel Ibanez, A. (2015). Minería de datos aplicada a la demanda del transporte aéreo en Ocaña, Norte de Santander. *Tecnura* [online]. 19(45), pp.101-113. http://dx.doi.org/10.14483/udistrital.jour.tecnura.2015.3.a08.
9. Rudin, C. & Wagstaff, K. L. (2014). Machine learning for science and society, *Mach Learn*, 95, pp. 1–9.
10. Russell, S. & Norvig, P. (2004). Inteligencia Artificial. Un Enfoque Moderno. 2da edición, Ed. Prentice–Hall Hispanoamericana.
11. Somvanshi, M. & Chavan, P. (2016). A review of machine learning techniques using decision tree and support vector machine, *2016 International Conference on Computing Communication Control and automation (ICCUBEA)*, Pune, 2016, pp. 1-7.
12. Truex, S., Liu, L., Gursoy, M. E. & Yu, L. (2017). Privacy-Preserving Inductive Learning with Decision Trees, *2017 IEEE International Congress on Big Data (BigData Congress)*, Honolulu, HI, 2017, pp. 57-64.
13. Venkatesh, B. & Anuradha, J. (2019). A Review of feature selection and its methods. *Cybernetics and Information Technologies*, 19(1), pp. 1-26.
14. WEKA (2017). Disponible en: http://www.cs.waikato.ac.nz/ml/weka/.
15. Ying, K., A. Ameri, Trivedi, A., Ravindra, D., Patel, D. & Mozumdar, M. (2015). Decision tree-based machine learning algorithm for in-node vehicle classification. *IEEE Green Energy and Systems Conference (IGESC)*, Long Beach, CA, 2015, pp. 71-76.
16. Zhong, Y. (2016). The analysis of cases based on decision tree. *2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, Beijing, 2016, pp. 142-147.