Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación
Centro de Ingeniería de Software y Sistemas (ISYS)
Laboratorio de Inteligencia Artificial
Opción Profesional: Inteligencia Artificial

Aplicación de la minería de textos para el desarrollo de un sistema de asignación automática de jurados a trabajos especiales de grado

Trabajo Especial de Grado presentado ante la Ilustre
Universidad Central de Venezuela
Por las Bachilleres

Anabel Cristina Alves Ferreira
C.I. V-19.499.413
alves.f.anabel.c@gmail.com

Mercedes Gabriela Rodríguez Cruz
C.I. V-19.379.403
meche.grc@gmail.com

Para optar al título de Licenciado en Computación

Tutores:
Prof.ª Haydemar Nuñez Prof.ª Esmeralda Ramos

Caracas, mayo 2014

Universidad Central de Venezuela

Facultad de Ciencias
Escuela de Computación

ACTA DEL VEREDICTO

Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación, para dictaminar sobre el Trabajo Especial de Grado titulado: "**Aplicación de la Minería de Textos para el Desarrollo de un Sistema de Asignación Automática de Jurados a Trabajos Especiales de Grado**" y presentado por las bachilleres Anabel Cristina Alves Ferreira, Cédula de Identidad V-19.499.413 y Mercedes Gabriela Rodríguez Cruz, Cédula de Identidad V-19.379.403 para optar al título de Licenciado en Computación, dejan constancia de lo siguiente:

Leído como fue, dicho trabajo por cada uno de los miembros del jurado, se fijó el día 19 de mayo del 2014 a la 1:00 PM**,** para que sus autoras lo defendieran en forma pública, lo que hizo en la Sala PB III de la Escuela de Computación, mediante una presentación oral del contenido del Trabajo Especial de Grado, luego de lo cual respondieron a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió **APROBARLO** con una nota de _____ puntos.

En fe de lo cual se levanta la presente Acta, en la Ciudad Universitaria de Caracas a los diecinueve días del mes de mayo del año dos mil catorce, dejándose también constancia de que actuaron como Coordinadores del Jurado las Profesoras Tutoras Haydemar Nuñez y Esmeralda Ramos.

Jurado Principal

_____	_____
Prof.ª Haydemar Nuñez	Prof.ª Esmeralda Ramos
(Tutora)	(Tutora)
_____	_____
Prof. Carlos Acosta	Prof.ª Zenaida Castillo
(Jurado)	(Jurado)
_____	_____
Prof. Iván Flores	Prof.ª Vanessa Leguízamo
(Suplente)	(Suplente)

AGRADECIMIENTOS

Primero que nada agradezco a todos los dioses de todas las religiones de forma equitativa, no sé cuál(es) habrá(n) ayudado en la elaboración de este trabajo así que mejor darles las gracias a todos por igual y no invocar la furia divina, ya Murphy me da suficientes problemas.

A la UCV, y específicamente a la maravillosa Facultad de Ciencias, donde pude aprender muchísimas cosas. Gracias por darme esa increíble oportunidad, espero que ahora y siempre, puedas seguir siendo la casa que vence la sombra.

A Internet por existir (y a Google por poner a nuestra disposición información que de otra forma será mucho más difícil de adquirir).

A todas las iniciativas de aprendizaje (y a sus respectivos creadores) por ayudarme a complementar mis estudios. Una mención especial a Codecademy[1] y a StackOverflow[2] donde se fomenta el trabajo en lugar de simplemente buscar respuestas fáciles.

A todas las personas que contribuyeron con la elaboración de este trabajo, gracias por su colaboración.

A todos aquellos amigos que me ofrecieron su apoyo a lo largo de la carrera, incluso a aquellos con los que ya no mantengo comunicación. Gracias a todos y cada uno de ustedes, espero que todos ustedes también cumplan las metas que se propongan. No los menciono a todos porque ya me estoy quedando si espacio y mi memoria es mala.

A todas las series, libros y juegos que me acompañaron y me llenaron de ánimos cada vez que me sentía deprimida. También a todos mis waifus y husbandos por hacer mi vida mejor.

A mi compañera de tesis Mercedes por su apoyo y por todos estos años de amistad.

A cuchi por ser una de las mejores personas que he conocido (y eso que no es una persona).

Para finalizar, quiero darle las gracias a mi madre quien siempre ha sido la persona que más apoyo me ha brindado a lo largo de toda mi vida, no me alcanzarían las páginas de este documento para agradecerte todo lo que te debo. Gracias por tu apoyo incondicional, por todos los sacrificios que has hecho a lo largo de tu vida por mi bienestar, por inculcarme esa motivación de dar siempre lo mejor de mí misma… Gracias por todo. ¡Te quiero como no tienes idea! <3

Anabel

[1] http://www.codecademy.com/
[2] http://stackoverflow.com/

Agradecida con ese ser superior que me ha guiado para tomar las decisiones que me han traído hasta el día de hoy. Agradecida con mi amada madre, Gladys, que a pesar de no estar físicamente conmigo me enseñó todos los valores y creencias que forjaron el ser humano que soy, todo el amor que me dio y toda la fe que puso en mi es el combustible principal para seguir avanzando.

Agradecida con la UCV, especialmente con la Facultad de Ciencias, sus profesores y preparadores por enseñarme las bases para comenzar mi vida profesional y por todas las bonitas experiencias.

Agradecida con mi padre Guillermo, mi hermana Vanessa y mi tía Carmen, quienes siempre han estado ahí apoyándome y dándome ánimo para que yo pudiera llegar hasta aquí. Gracias a mi madrina Liliana, ella fue mi consejera y guía en los momentos de angustia y oscuridad, cuando quería renunciar y dejar todo atrás ella siempre me ayudó a recuperar la confianza en mí misma.

Agradecida con el profesor Néstor Méndez, quien desde el inicio de mi (por ahora) corta experiencia profesional ha sido mi orientador y conductor, siempre impulsándome a seguir adelante y a superarme a mí misma.

Agradecida con Anabel, mi amiga desde que comencé la universidad y compañera de tesis, sin ti hubiera tirado la toalla hace muchísimo tiempo, gracias por el apoyo, gracias por escuchar siempre. ¡Las trasnochadas, las loqueras que nos daban, el helado y la Golden de naranja funcionaron! ¡Aquí estamos :D! Gracias a la madre de Anabel, Ana María, ella también nos dio las palmaditas e impulsos necesarios para completar este trabajo.

Agradecida con mis siempre amigas Marisela y Oriana, en los problemas familiares, amorosos, monetarios, escolares, lo que fuera; un rato con ellas hacia que nada pareciera tan grave, sus consejos y ocurrencias siempre me dan aliento.

Agradecida con Alexis, el chico que labora en la biblioteca Alonso Gamero de la Facultad de Ciencias, él se esforzó mucho para ayudarnos a conseguir todas las tesis que fueran necesarias y que finalmente hicieron posible la realización de este trabajo.

Agradecida con Irena, Audel, Fernando y Darwing, sin ellos la universidad hubiera sido muy aburrida. Daniel, Geybe, Julio, Lily, Andreyna, Tayvi, José Rafael, Fidias, muchas gracias a ellos también.

Finalmente, agradecida con todas aquellas personas que de una u otra forma aportaron y fueron importantes para mí durante este camino, no importa si forman o no parte de él actualmente, gracias…

Mercedes

APLICACIÓN DE LA MINERÍA DE TEXTOS PARA EL DESARROLLO DE UN SISTEMA DE ASIGNACIÓN AUTOMÁTICA DE JURADOS A TRABAJOS ESPECIALES DE GRADO

RESUMEN

Este trabajo tiene como objetivo el desarrollo de un sistema basado en minería de textos para la asignación automática de jurados a Trabajos Especiales de Grado (TEG) mediante la clasificación de los documentos dentro de las opciones profesionales (OP) ofrecidas en la Escuela de Computación de la UCV. Ya existe una primera versión de este sistema que sólo realiza la clasificación de TEG bajo las cuatro OP más demandadas de la Escuela (Aplicaciones con Tecnología Internet, Inteligencia Artificial, Base de Datos, y Tecnologías en Comunicaciones y Redes de Computadoras). En este trabajo se toma como base dicho sistema y se crea una nueva versión que clasifica TEG pertenecientes a cualquiera de las once OP que se ofertan en la Escuela de Computación.

Para la creación del clasificador se aplicó el proceso de minería de textos sobre una recopilación de TEG, a los cuales se le realizó un pre-procesamiento eliminando signos de puntuación, palabras no informativas y aplicando lematización, luego fueron representados mediante el modelo de espacio vectorial indexados con pesado por entropía y finalmente se utilizaron diferentes medidas de relevancia para la reducción de la dimensionalidad del espacio de atributos. Para la estimación del modelo de clasificación se utilizó el algoritmo Bayes Ingenuo, alcanzando un 82,0513% de clasificaciones correctas. A través de la recomendación de hasta dos opciones profesionales, se logró incrementar el porcentaje de clasificaciones correctas a 87,5%. El módulo de asignación de opciones profesionales se integró con el módulo de asignación de jurados, éste último considera a los profesores especializados en las OP recomendadas en base a ciertos criterios de elección definidos. En conclusión, clasificando los TEG bajo cualquiera de las once OP de la Escuela, fue posible obtener un rendimiento similar al presentado en la versión previa de este sistema. Además se logró automatizar el proceso de asignación de jurados a TEG, lo que a su vez permite llevar un registro histórico de las asignaciones, contribuyendo con la automatización de procesos de la Escuela de Computación.

Palabras clave: Minería de textos, Modelos de clasificación, Bayes Ingenuo, *Naïve Bayes*, algoritmo de *Porter Stemming*, Weka.

Autores:

Br. Anabel Alves.
alves.f.anabel.c@gmail.com

Br. Mercedes Rodríguez.
meche.grc@gmail.com

Tutores:

Prof.ª Haydemar Nuñez.
esmeralda.ramos@ciens.ucv.ve

Prof.ª Esmeralda Ramos.
haydemar.nunez@ciens.ucv.ve

Mayo 2014

TABLA DE CONTENIDO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE ANEXOS

INTRODUCCIÓN

La revolución digital ha hecho posible que la información sea fácil de capturar, procesar, almacenar y transmitir. Sin embargo, los datos por sí solos no producen beneficio directo, su verdadero valor radica en la posibilidad de extraer conocimiento útil para la toma de decisiones o la comprensión del fenómeno que los produjo. En muchos dominios, el análisis de los datos ha sido realizado tradicionalmente de manera manual: uno o más analistas con la ayuda de técnicas estadísticas, proporcionaban resúmenes y generaban informes. Tal enfoque cambió como consecuencia del crecimiento del volumen de datos. Cuando la escala de manipulación de datos, exploración e inferencia va más allá de la capacidad humana, se necesita la ayuda de las tecnologías informáticas para automatizar el proceso. Todo apunta a la necesidad de metodologías de análisis inteligente que permitan procesar automáticamente grandes cantidades de datos crudos, identificar los patrones más significativos y presentarlos como conocimiento apropiado para satisfacer los objetivos planteados. Este proceso se conoce como Proceso de Extracción de Conocimiento a partir de Datos, KDD (*Knowledge Discovery in Databases*) o simplemente Minería de Datos. El avance de la tecnología para la gestión de base de datos hace posible integrar diferentes tipos de datos, siendo uno de los más utilizados el texto, por ello surge el interés de aplicar el proceso KDD a colecciones de documentos no estructurados. A este proceso que se le denomina Minería de Textos (MT).

Actualmente, la Escuela de Computación de la Universidad Central de Venezuela realiza la asignación de jurados a Trabajos Especiales de Grado (TEG) siguiendo un proceso subjetivo y manual, en el que se deciden los jurados más convenientes de acuerdo al área profesional del mismo. Como solución a este problema, se propuso en un primer trabajo (Torres Navarro, 2011) la automatización de este proceso, desarrollando un sistema de recomendación que permitiera clasificar un TEG de acuerdo a las opciones profesionales ofrecidas en la Escuela aplicando minería de textos y seleccionar los jurados más pertinentes. La primera versión de dicho sistema realiza este proceso sólo para las cuatro áreas más demandadas de la Escuela de Computación (Aplicaciones con Tecnología Internet, Inteligencia Artificial, Base de Datos, y Tecnologías en Comunicaciones y Redes de Computadoras) utilizando la técnica de K-Vecinos. En éste trabajo se plantea tomar como base dicho sistema y crear una nueva versión a partir de un clasificador Bayesiano Ingenuo (*Naïve Bayes*) que sea capaz de asignar automáticamente jurados a TEG pertenecientes a cualquiera de las once opciones profesionales que se ofertan en la Escuela de Computación. A continuación, se describe la estructura general de este documento:

- **Capítulo 1**: Contiene una síntesis de la revisión bibliográfica que se llevó a cabo para la realización del presente trabajo. Se da una visión general del proceso KDD y se presenta una adaptación del mismo a la Minería de Textos, orientado a la tarea de Clasificación de Textos.

- **Capítulo 2**: Se expone el problema a resolver, los objetivos propuestos y la solución. Igualmente, se detalla el proceso de Minería de Textos aplicado para estimar el modelo de clasificación y los resultados de su evaluación. También, se señalan los aspectos relacionados con el desarrollo de la aplicación: el módulo de recomendación, de asignación y de administración. Se da una breve descripción de las tecnologías utilizadas, las consideraciones de diseño, y se muestran los resultados obtenidos.

- Por último, se presentan las conclusiones y las recomendaciones para trabajos futuros.

CAPÍTULO 1: MARCO TEÓRICO

En este Capítulo se explican de manera breve los conceptos teóricos sobre los cuales se fundamenta el trabajo realizado. Primeramente se describen los conceptos básicos de minería de datos para luego abordar el proceso de minería de textos y los algoritmos que se utilizan para la aplicación del mismo.

1.1. CONCEPTOS BÁSICOS DE LA MINERÍA DE DATOS

Se define la minería de datos como el proceso de extraer conocimiento útil y comprensible, desde grandes cantidades de datos almacenados en distintos formatos (Witten & Frank, 2005). El objetivo fundamental de la minería de datos es encontrar modelos inteligibles a partir de los datos. Para que este proceso sea efectivo debería ser automático o semi-automático y el uso de los patrones descubiertos debería ayudar a la toma de decisiones. Se entiende por patrón una expresión en algún lenguaje que describe a un subgrupo de los datos o un modelo aplicable al subgrupo.

Por lo tanto, son dos los retos de la minería de datos: trabajar con grandes volúmenes de datos afrontando los problemas que ello conlleva (ruido, datos ausentes, etc.) y usar técnicas adecuadas para analizarlos y extraer conocimiento útil. En muchos casos, la utilidad de dicho conocimiento está profundamente relacionada con la comprensibilidad del modelo inferido; generalmente el usuario final no es un experto en las técnicas de minería de datos, por ello en muchas aplicaciones es importante hacer que la información descubierta sea fácilmente comprensible por humanos (representaciones gráficas, lenguaje natural, etc.) (Hernández, Ramírez, & Ferri, 2005).

1.1.1. PROCESO DE EXTRACCIÓN DE CONOCIMIENTO

El proceso de extracción de conocimiento a partir de datos (KDD, por las siglas en inglés de "Knowledge Discovery in Databases") es aquel cuyo objetivo es identificar patrones (modelos) significativos en los datos que sean válidos, novedosos, potencialmente útiles y comprensibles, la minería de datos es un paso de este proceso, dedicado específicamente a realizar la transformación de los datos (posiblemente modificados en etapas previas) en estos patrones, que posteriormente servirán para la extracción de conocimiento (Hernández, Ramírez, & Ferri, 2005).

El proceso de extracción de conocimiento se organiza en torno a cinco fases ilustradas en la Figura 1.2. Como se observa, la minería de datos es parte de este proceso y es en esa fase donde se realiza la extracción de conocimiento; por esto en ocasiones al proceso de KDD también se le llama Proceso de Minería de Datos. A continuación, se describen brevemente cada una de estas fases:

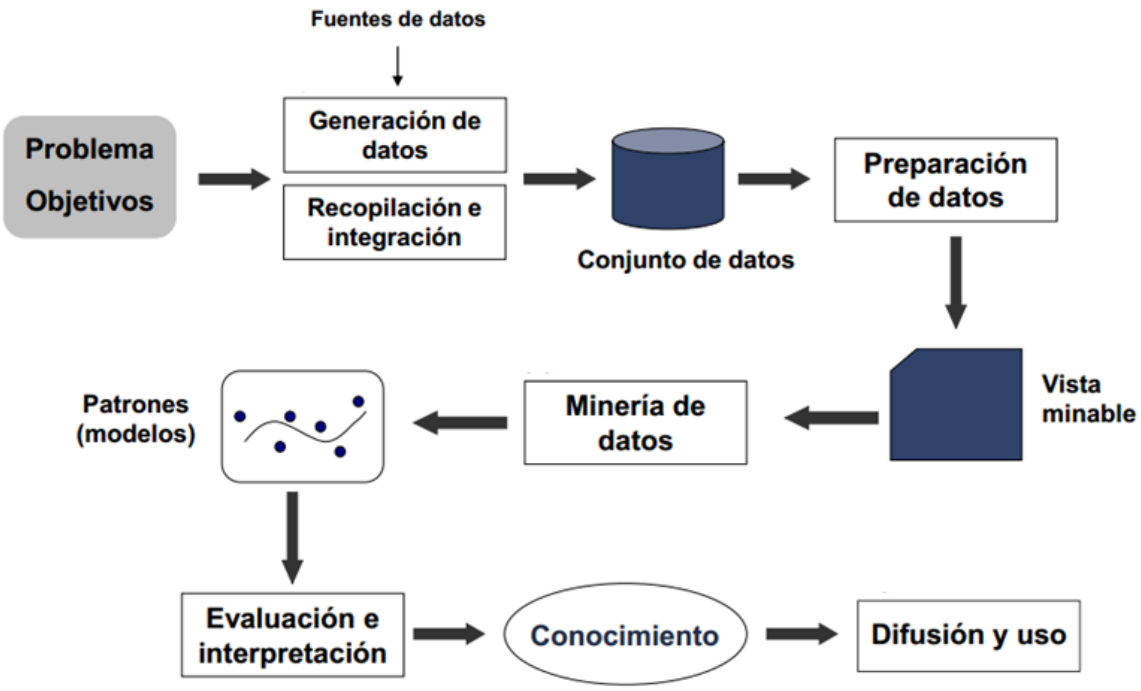

FIGURA 1.1. PROCESO DE EXTRACCIÓN DE CONOCIMIENTO (ELABORACIÓN PROPIA)

- **Recopilación e integración**: Como su nombre lo indica, en esta fase se realiza la recolección de los datos. La recolección puede llevarse a cabo utilizando: datos existentes, realizando experimentos, utilizando programas dedicados a la generación de datos o cualquier otro instrumento que se considere pertinente para el problema. Las fuentes de datos necesarias, pueden utilizar diferentes formatos de representación, en consecuencia, luego de realizada la recolección de los datos, se debe hacer la integración sobre los mismos para almacenarlos bajo un esquema unificado.

- **Limpieza y preparación de datos**: La calidad del conocimiento descubierto depende en gran medida de la calidad de los datos. Por ello, después de la recopilación e integración, la siguiente fase en el proceso KDD es preparar el subconjunto de datos que se va a minar. Esta fase incluye la limpieza de valores ausentes o anómalos, transformación de datos, selección o construcción de nuevos atributos y selección de datos, con el objetivo de reducir su tamaño y obtener los más informativos. Al finalizar este proceso se obtiene un conjunto de datos que conformarán una vista minable, sobre la cual se aplicarán las diferentes técnicas de generación de modelos.

- **Minería de datos**: El objetivo de esta fase es producir nuevo conocimiento construyendo un modelo basado en los datos recopilados. Para ello es necesario tomar un par de decisiones antes de empezar el proceso: determinar qué tarea de minería es la más apropiada para el problema planteado y elegir el algoritmo de minería que se adapte a la tarea.

  Las tareas pueden ser de dos tipos principales según las características de los modelos que producen: predictivas o descriptivas. Se denomina predictivas a aquellas tareas cuyos modelos

estiman valores futuros o desconocidos de variables de interés usando los valores de los atributos que conforman los datos; mientras que, los modelos de las tareas descriptivas identifican patrones que explican o describen los datos.

Entre las tareas predictivas se encuentra la clasificación y la regresión, mientras que el agrupamiento (*clustering*), las reglas de asociación y la detección de anomalías son tareas descriptivas.

- **Evaluación e interpretación**: Para que los modelos puedan utilizarse sobre nuevos datos es importante conocer su rendimiento de generalización. Esta estimación se realiza mediante la aplicación de una medida de evaluación la cual determinará el rendimiento del modelo. Sin embargo, la forma en la que se utilicen los datos (técnica de evaluación) a la hora de aplicar la medida es lo que brindará una mejor aproximación al rendimiento real (rendimiento sobre nuevos datos) del modelo obtenido.

  Las medidas de evaluación utilizadas dependerán de la tarea de minería de datos que haya sido escogida, sin embargo, las técnicas de evaluación se realizan independientemente de la tarea siguiendo alguna de las siguientes variantes:

  - Utilizar todo el conjunto de datos como entrenamiento y prueba. Fallas: posible sobreajuste, por ende fallas en la generalización para datos nuevos.

  - Separación aleatoria del espacio de datos en dos grupos: conjunto de entrenamiento y conjunto de prueba. Los resultados estarán estrechamente relacionados a la partición. Posible falta de datos.

  - Dividir el espacio en k subconjuntos disjuntos de igual tamaño (comúnmente siendo $k = 10$). Estos se utilizarán durante i iteraciones donde se tomarán $k - 1$ de ellos como conjunto de entrenamiento y el restante como conjunto de prueba. El error estimado será la media aritmética de los errores obtenidos en cada iteración. A este procedimiento se le conoce como *validación cruzada*, y a pesar de que provee una mayor independencia de los resultados, posee un costo computacional que puede impedir su aplicación si el espacio de datos es muy extenso.

  - Técnicas basadas en probabilidad como Bootstrap e intervalos de confianza.

- **Difusión y uso**: Una vez validado el modelo éste puede usarse para que un analista recomiende acciones basándose en él, o bien para aplicar el modelo a diferentes conjuntos de datos. También es posible su incorporación a otras aplicaciones.

1.2. PROCESO DE MINERÍA DE TEXTOS

En la actualidad, la mayor parte de la información (más de un 80%) se encuentra almacenada en forma de texto. Es por ello que surge la necesidad de poder realizar el proceso de minería de datos sobre grandes cantidades de información textual.

La minería de textos (*Text Mining – TM* por sus siglas en inglés) también conocida como KDT (*Knowledge Discovery in Texts*), es un proceso cuyo objetivo es la obtención de conocimiento desde textos (Lucas, 2000; Tan, 1999).

En el caso de la minería de textos se tienen cinco fases, todas ellas análogas a alguna de las fases presentes en el proceso de minería de datos (Tabla 1.1.). Estas serán desarrolladas en el apartado 1.3.

TABLA 1.1. COMPARACIÓN DE PROCESOS: MINERÍA DE DATOS VERSUS MINERÍA DE TEXTOS

Minería de datos	Minería de textos
Integración y recopilación	Recolección
Limpieza y preparación de datos	Pre-procesamiento
Minería de datos	Minería de textos
Evaluación e interpretación	Evaluación e interpretación
Difusión y uso	Difusión y uso

- **Recolección**: El objetivo de esta fase es definir el contenido de la base textual (también denominada base de documentos o corpus), un repositorio de datos que servirá como alimento para todo el proceso. Este repositorio puede ser definido una única vez (estático) o puede ser de actualización constante (dinámico), definiéndose actualización como el acto de agregar, modificar o eliminar, parte o toda la base textual.

- **Pre-procesamiento**: Consiste en un conjunto de transformaciones realizadas sobre una colección de textos con el objetivo de hacer que estos pasen a estar estructurados. Esta estructura dependerá de las características del problema y la técnica que posteriormente se va a aplicar sobre los datos.

  Pre-procesar también significa dividir el texto en unidades que puedan ser manejadas por las fases posteriores: lexemas (también conocidos como tokens). Un lexema puede ser equivalente a una palabra o elementos que agrupen un conjunto de palabras. Por otra parte, un texto puede contener gran cantidad de palabras que son simples auxiliares gramaticales del idioma o palabras muy comunes que no aportan ninguna información útil, las cuales normalmente son eliminadas.

  Luego de haber realizado los procedimientos anteriormente descritos, resta definir la representación a utilizar (vectores, matrices, etc.). Finalmente, en caso de considerarse necesario, se procede a la selección de aquellas características que se consideren de mayor relevancia para el problema, en lo que se conoce como reducción de la dimensionalidad.

- **Minería de textos**: Esta fase tiene por objetivo derivar un modelo a partir de los datos que se obtuvieron como salida en la etapa de pre-procesamiento. Las tareas de minería más habituales sobre los datos textuales son las de asociación, agrupación y categorización, siendo esta última en la que se fundamenta el presente trabajo y sobre la cual se profundizará en la siguiente sección.

- **Evaluación, difusión y uso**: Las técnicas y medidas de evaluación utilizadas en minería de textos son similares a las usadas en la minería de datos y siempre dependerán de la tarea elegida. Finalmente, luego de realizadas todas estas actividades, lo que resta es la validación de los resultados obtenidos por parte de los expertos y los usuarios. También es importante medir la evolución del modelo y realizar los ajustes necesarios de así requerirse.

1.3. CATEGORIZACIÓN DE TEXTOS

La categorización de textos es una tarea de minería de textos encargada de resolver el problema de asignar automáticamente una o más categorías o clases (definidas previamente) a un conjunto de documentos (Aas & Eikvil, 1999). Otra forma de ver este problema es como el de la asignación de un valor booleano a cada par $a(c_i, d_i) \in CxD$, donde D es el dominio de documentos y C el conjunto predefinido de clases. Un valor positivo en $a(c_i, d_i)$ indica que d_i será clasificado como perteneciente a la clase c_i, mientras que el valor negativo indicará lo contrario: d_i no es perteneciente a la clase c_i (Sebastiani, 2001). Las clases son sólo etiquetas simbólicas, el conocimiento adicional acerca de su significado no está disponible.

En las secciones siguientes se explica cómo se aplican las fases de la minería de texto en la categorización de textos. Se obvian las fases de recolección y difusión y uso, debido a que no presentan ninguna variación resaltante que no haya sido mencionada con anterioridad.

1.3.1. PRE-PROCESAMIENTO

La idea principal de esta fase es definir algunos términos y variables que son frecuentemente utilizados en la colección de documentos; dichos términos pueden ser considerados características que distinguen cada categoría o clase de las demás. El conjunto de características obtenidas a partir de los textos de todos los documentos de la colección es llamado diccionario. Documentos que posean ciertas características tendrán un factor de posibilidad de pertenecer a una determinada clase, de modo que la acumulación de dichas cantidades puede arrojar un resultado consistente en un coeficiente asociado a cada una de las clases existentes. Este coeficiente lo que expresa en realidad es el grado de confianza o certeza de que el documento en cuestión pertenezca a una determinada categoría (Figuerola, Zazo, & Alonso, 2000).

El pre-procesamiento se compone de tres sub-fases: En la primera fase se extraen del texto las características o palabras representativas, esto es lo que se conoce como proceso de filtrado. En la segunda fase se representan los textos en una estructura que pueda ser utilizada en el resto del proceso de minería, esto se logra por medio de la indexación. En la última fase se realiza la reducción del número de características disponibles, dejando sólo aquellas que provean algún aporte al proceso, esto se conoce como reducción de la dimensionalidad. A continuación se explica con mayor detalle cada una de las mismas.

A. FILTRADO

Los métodos de filtrado son utilizados para eliminar cierto tipo de palabras de los documentos que pertenecen al conjunto de datos. Hay dos tipos de filtrado que deben realizarse antes de proceder con el resto de las tareas:

- **Eliminación de palabras no informativas (*stop words*)**, que consiste en remover las palabras que no aportan información útil (artículos, preposiciones, conjunciones, identificadores, etc.)

- **Lematización:** Los métodos de lematización (*stemming*) se encargan de la eliminación de afijos para llevar las palabras a su forma básica. Al culminar este proceso se tienen un conjunto de palabras con el mismo significado conceptual, por ejemplo: caminar, caminante, caminado, caminando, etc. (Figuerola, Zazo, & Alonso, 2000).

  El proceso de lematización por lo general resulta muy lento y propenso a errores, sin embargo varios expertos y lingüistas se han encargado de diseñar algoritmos para automatizarlo. Estos algoritmos son conocidos como lematizadores. Entre los más conocidos se tienen el lematizador de Lovins (Lovins, 1968), el lematizador de Porter (Porter, 1980) y el lematizador de Paice-Husk (Paice, 1990).

B. INDEXACIÓN

Dado que los textos son datos complejos es necesaria la utilización de algún tipo de representación interna para así poder realizar el proceso de minería. La estructura de datos más utilizada para la representación de la colección de textos en la tarea de categorización, y la que se utilizará a lo largo de este trabajo, es conocida como Modelo de Espacio Vectorial (*Vector Space Model*), la cual, además de ser muy simple, no utiliza información semántica explícita para su construcción.

El modelo vectorial representa cada documento como un vector $D = (t_1 t_2 \dots t_k)$, donde cada uno de los t_i son los términos o palabras dentro del documento D. Por lo tanto, una colección de documentos se representa por una matriz W, en la cual cada entrada representa la ocurrencia de una palabra en un documento:

$$W = (w_{ik})$$

Donde w_{ik} es el peso de la palabra i en el documento k. Dado que cada palabra normalmente no aparece en todos los documentos, la matriz W suele ser esparcida. El número de columnas M de la matriz corresponde al número de palabras en el diccionario (Aas & Eikvil, 1999).

El valor numérico de cada uno de los componentes de la matriz debe obedecer a cálculos que tengan en cuenta otros factores para mejorar el rendimiento del clasificador. Normalmente se utilizan esquemas de pesado de términos que reflejen la importancia de una palabra en un documento específico de la colección.

Se han propuesto diversos esquemas para calcular el peso w_{ik} de la palabra i en el documento k, los más utilizados se pueden apreciar en la Tabla 1.2. y se basan en dos observaciones empíricas con respecto al texto:

- Entre más veces aparezca la palabra en un documento, es más relevante para definir el tópico del documento.

- Entre más veces aparezca la palabra a través de todos los documentos de la colección, ésta discrimina más pobremente entre documento.

TABLA 1.2. PRINCIPALES PROPUESTAS DE PESADO PARA LA INDEXACIÓN DE TÉRMINOS

Propuestas de pesado	Fórmula	
Booleano	w_{ik} = 1 Si $f_{ik} > 0$	w_{ik}= 0 En otro caso
Frecuencia de la palabra	$w_{ik} = f_{ik}$	
TFxIDF	$w_{ik} = f_{ik} * \log\left(\frac{N}{n_i}\right)$	
TFC	$w_{ik} = \frac{f_{ik} * \log\left(\frac{N}{n_i}\right)}{\sqrt{\sum_{j=1}^{M}\left[f_{jk} * \log\left(\frac{N}{n_j}\right)\right]^2}}$	
LTC	$w_{ik} = \frac{\log(f_{ik} + 1.0) * \log\left(\frac{N}{n_i}\right)}{\sqrt{\sum_{j=1}^{M}\left[\log(f_{jk} + 1.0) * \log\left(\frac{N}{n_j}\right)\right]^2}}$	
Entropía	$w_{ik} = \log(f_{ik} + 1.0) * \left(1 + \frac{1}{\log(N)} * \sum_{j=1}^{N}\left[\left(\frac{f_{ij}}{n_i}\right) * \log\left(\frac{f_{ij}}{n_i}\right)\right]\right)$	
Terminología	$\boldsymbol{f_{ik}}$ **:** Frecuencia de la palabra i en el documento k. $\boldsymbol{N}$**:** Número de documentos en la colección. $\boldsymbol{M}$**:** Número de palabras en el diccionario. $\boldsymbol{n_i}$**:** Número de veces que la palabra i ocurre en la colección completa.	

C. REDUCCIÓN DE LA DIMENSIONALIDAD

Debido al número de características contenidas en los documentos uno de los problemas centrales que se presenta en la clasificación de textos es la alta dimensionalidad que puede alcanzar el Modelo de Espacio Vectorial. Por lo tanto, se hace necesaria la reducción del conjunto de características original.

Las técnicas para selección de características toman t' términos a partir de la colección original de t características, tal que $t' \ll t$, procurando que la reducción de efectividad sea la menor posible;

para ello se aplica a cada término en t una función de evaluación de términos y se selecciona el conjunto t' que maximice dicha función.

Algunas de las funciones más utilizadas para reducir características se encuentran representadas en la Tabla 1.3.

TABLA 1.3. PRINCIPALES FUNCIONES PROPUESTAS PARA LA SELECCIÓN DE CARACTERÍSTICAS

Función	Denotado por	Fórmula matemática
Frecuencia en documentos	$\#(t_k, c_i)$	$P(t_k, c_i)$
Ganancia de información	$IG(t_k, c_i)$	$P(t_k, c_i) \cdot \log \frac{P(t_k, c_i)}{P(c_i) \cdot P(t_k)} + P(\overline{t_k}, c_i) \cdot \log \frac{P(\overline{t_k}, c_i)}{P(c_i) \cdot P(\overline{t_k})}$
Chi-cuadrado	$X^2(t_k, c_i)$	$\frac{g \cdot [P(t_k, c_i) \cdot P(\overline{t_k}, c_i) - P(t_k, \overline{c_i}) \cdot P(\overline{t_k}, c_i)]^2}{P(t_k) \cdot P(\overline{t_k}) \cdot P(c_i) \cdot P(\overline{c_i})}$
Razón de ganancia	$GR(t_k, c_i)$	$\frac{\sum_{c \in \{c_i, \overline{c_i}\}} \sum_{t \in \{t_k, \overline{t_k}\}} P(t,c) \log_2 \frac{P(t,c)}{P(t)P(c)}}{-\sum_{c \in \{c_i, \overline{c_i}\}} P(c) \log_2 P(c)}$

Donde $P(\overline{t_k}, c_i)$ indica la probabilidad de que, dado un documento d, el término t_k no ocurra en d y d pertenezca a la categoría c_i.

1.3.2. MINERÍA DE TEXTOS

En la fase de minería, como se mencionó en secciones anteriores, es donde se realiza la construcción del modelo que resuelva el problema planteado. Existen diferentes técnicas aplicables en esta fase, a continuación se describen brevemente tres de las más utilizadas para categorización de texto (Sebastiani, 2001).

A. BAYES INGENUO (*NAÏVE BAYES – NB*)

Bayes ingenuo es un tipo de red bayesiana que se basa en el principio llamado "independencia condicional de clase" para simplificar sus cálculos. La independencia condicional de clase supone que una clase C tiene relación con un subconjunto de los términos existentes, sin embargo, la aparición de un término es independiente de la aparición de los demás (Zhang H. , 2004). Esta premisa hace que el clasificador de Bayes ingenuo sea más eficiente que otros clasificadores, debido a que carece de la complejidad exponencial generada por la combinación de los predictores de palabras.

Un documento D_i pertenecerá a la clase C_i que maximice la probabilidad a posteriori de C_i dado D_i.

$$C = arg\,max(\, P(C_i|D_i)\,)$$

$$C = arg\,max\left(\frac{P(D_i|C_i) * P(C_i)}{P(D_i)}\right)$$

$$C = arg\,max(\,P(D_i|C_i) * P(C_i)\,) \qquad \text{[a]}$$

Donde se ha eliminado de la ecuación $P(D_i)$ dado que no varía entre las clases. Asimismo se tiene que D_i está formado por un conjunto de atributos de la forma:

$$D_i = a_1, a_2, \dots, a_n$$

Con lo que se obtiene:

$$P(D_i|C_i) = P(a_1, a_2, \dots, a_n|C_i)$$

Quedando la formula [a]:

$$C = arg\,max(\,P(a_1, a_2, \dots, a_n|C_i) * P(C_i)\,)$$

Sin embargo, por la hipótesis de independencia condicional de clase, sólo es necesario calcular la probabilidad $P(a_i|C_i)$ para cada atributo y la probabilidad a priori de la clase:

$$C = \arg\max(\;P(C_i)\;*\prod_{j=1}^{M} P(a_j|C_i)\,)$$

Una estimación $P'(C_i)$ para $P(C_i)$ puede ser calculada a partir de la fracción de documentos de entrenamiento asignada a la clase C_i:

$$P'(C_i) = \frac{N_i}{N}$$

Una suposición común, aunque no intrínseca a la aproximación bayesiana ingenua pero aplicada muy a menudo, es que para cada clase, los valores numéricos de los atributos están normalmente distribuidos. Se puede representar una distribución de este tipo en términos de su desviación estándar y su media, y de esta forma calcular eficientemente la probabilidad de un valor observado. Para atributos continuos se tiene:

$$P'(a_j|C_i) \;= g(a_j;\,\mu_{c_i}\,,\sigma_{c_i})$$

Dónde:

$$g(a;\,\mu\,,\sigma) = \frac{1}{\sqrt{2\pi\sigma}}\;e^{-\frac{(a-\mu)^2}{2\sigma^2}}$$

Es la función de densidad probabilística para una normal (Distribución Gaussiana).
Existen variaciones del algoritmo base de Bayes Ingenuo, como la propuesta de (John & Langley, 1995) donde para la estimación de las probabilidades atributo – clase $P(a_j|C_i)$ se utiliza la estimación de densidad con kernel, la cual puede utilizar distribuciones normales u otro tipo de

función. Lo que resulta es la obtención de una función de distribución por cada posible combinación de atributo – clase. Por ello, aunque en Bayes Ingenuo sólo se requiere el número de atributos (a_j), la suma de los mismos y la suma de sus valores al cuadrado para el cálculo de la media (μ_{c_i}) y la desviación (σ_{c_i}), en esta variante se deben guardar los diferentes valores aj, así al clasificar una nueva instancia, en lugar de hacerse una única evaluación por atributo deben realizarse n evaluaciones, una por cada valor observado del atributo a en la clase C, esto implica un mayor coste computacional pero provee la posibilidad de una aproximación más certera.

B. K-NN (*K – NEAREST NEIGHBOURS*)

El algoritmo K-NN, también conocido como el algoritmo de los K vecinos más cercanos, se basa en el concepto de aprendizaje perezoso *(Lazy Learning):* no se realiza ningún cálculo ni se construye un modelo durante la fase de entrenamiento, la información sobre el conjunto de datos es almacenada hasta que se requiere la clasificación de un elemento nuevo. La predicción se basará en una aproximación local utilizando esta información. En el caso específico del algoritmo K-NN se toman en cuenta los k elementos más "cercanos" para decidir a qué clase C_i pertenece un documento D_i. La cercanía entre cada elemento y el elemento a procesar vendrá dada por la similitud o la distancia que entre ellos exista.

La similitud es una medida numérica que indica el grado al cual dos objetos se parecen, mientras más alto sea este valor más parecidos son los objetos. Una de las medidas de similitud más utilizada es la fórmula del Coseno. Asimismo, la distancia es otra medida numérica, sin embargo, ésta indica el grado al cual dos objetos son diferentes. Entre las medidas de distancia más utilizadas se encuentran la distancia Euclídea, la distancia Manhattan y la distancia de Chebychev (Rodríguez, Rojas Blanco, & Franco Camacho, 2007).

Entre las ventajas de esta técnica se encuentran que el coste de aprendizaje es nulo (todo el computo se realiza en la etapa de clasificación), no se necesita hacer ninguna suposición sobre los datos y su tolerancia al ruido (García & Gómez, 2009).

La principal dificultad del método K-NN consiste en determinar el valor de k que mejor se adapte al problema, ya que si se toma un valor muy grande se corre el riesgo de hacer la clasificación tomando en cuenta datos muy poco relacionados y si se considera uno muy pequeño, se tendrá poca información para hacer una clasificación correcta.

C. ÁRBOLES DE DECISIÓN

Esta técnica consiste en comparar las características del vector del documento D_i con respecto a un árbol de decisión, el cual definirá a cuál de las clases C_i será asignado. Éste árbol utiliza como base las características del grupo de entrenamiento para construir una serie de reglas que definirán los criterios de clasificación para futuros elementos de entrada. Existen variedad de algoritmos para la generación de estas reglas, algunos de los más utilizados son ID3, C4.5 y CART.

El método de aprendizaje está basado en el principio de "divide y vencerás", en primer lugar se verifica si todos los elementos de entrenamiento poseen la misma clase, en caso contrario se selecciona un término t_i el cual dividirá el conjunto de acuerdo a sus valores en ese término. Este proceso se repite hasta obtener grupos para todos los elementos. Las reglas obtenidas serán las

que servirán para la evaluación de nuevos elementos: para clasificar un nuevo documento se recorrerá el árbol, evaluando las condiciones necesarias hasta llegar a un nodo hoja.

Los árboles de decisión son rápidos y escalables en cuanto al número de variables como al tamaño del conjunto de entrenamiento que pueden manejar. A pesar de ello, es la que se considera menos apta para minería de textos (de las mencionadas en este apartado) debido a que la decisión final suele estar basada en un conjunto relativamente pequeño de términos (Hotho, Nürnberger, & Paaß, 2003).

D. OTROS ALGORITMOS

Investigaciones en el área de minería de textos mencionan buenos resultados para la tarea de categorización de textos con los siguientes algoritmos:

- Redes Neuronales (Kohonen, 1988), (Zhang & Zhou, 2006).
- Máquinas de Soporte Vectorial (Joachims , 2002), (Venegas, 2007).

1.3.3. EVALUACIÓN E INTERPRETACIÓN

Al ser la categorización de textos una tarea dentro de la minería de textos, las técnicas de evaluación utilizadas son las mismas explicadas en los apartados anteriores. Con respecto a las medidas de evaluación utilizadas en la categorización de textos, las que serán empleadas en este trabajo son las siguientes:

- **Matriz de confusión** Una matriz de confusión NxN representa de manera detallada la distribución de los resultados de un clasificador de N clases (ver Tabla 1.4.). Cada celda de la matriz identifica el número de instancias que fueron clasificadas bajo una determinada clase versus la clase a la que realmente pertenecen.

TABLA 1.4. MATRIZ DE CONFUSIÓN PARA UN CLASIFICADOR DE DOS CLASES

		Clase predicha	
		Clase = 0	**Clase = 1**
Clase verdadera	**Clase = 0**	F_{00}	F_{01}
	Clase = 1	F_{10}	F_{11}

Donde F_{ij} = Número de instancias de la clase *i* predichas como clase *j*

Para entender mejor el resto de las medidas a utilizar en este trabajo, es necesario aclarar los conceptos presentados a continuación. Los ejemplos serán aplicados sobre la Clase 0 de la matriz de confusión representada en la Tabla 1.4.

- **Verdaderos positivos:** Son instancias pertenecientes a la clase 0 que se clasifican correctamente en la clase 0 (F_{00}).

- **Verdaderos negativos:** Son instancias que no forman parte de la clase 0 y que no se clasifican como clase 0 (F_{11}).

- **Falsos positivos:** Son instancias que no forman parte de la clase 0 pero que se clasifican como clase 0 (F_{10}).

- **Falsos negativos:** Son instancias pertenecientes a la clase 0 pero que no se clasifican como clase 0 (F_{01}).

- **Precisión:** Mide la probabilidad de que si un sistema clasifica un documento en una cierta categoría, el documento realmente pertenezca a esa categoría. La precisión de la clase 0 se define como:

$$P\ (Clase = 0) = \frac{\mathrm{F}_{00}}{(\mathrm{F}_{00} + \mathrm{F}_{10})}$$

  Ésta medida aumenta cuando hay pocos falsos positivos.

- **Sensibilidad:** También llamada *Recall*. Mide la probabilidad de que si un documento pertenece a cierta categoría, el sistema lo asigne a la categoría correspondiente. El *recall* de la clase 0 se define como:

$$R\ (Clase = 0) = \frac{\mathrm{F}_{00}}{(\mathrm{F}_{00} + \mathrm{F}_{01})}$$

  El valor del *recall* aumenta cuando hay pocos falsos negativos.

- ***Medida F:*** Es la medida armónica de la precisión y *recall*. Para la clase 0 se calcula de la siguiente manera:

$$F1\ (Clase = 0) = \ 2 * \frac{(P\ (Clase = 0) * R\ (Clase = 0))}{(P\ (Clase = 0) + R\ (Clase = 0))}$$

  Un valor alto de la *Medida F* indica que la clase 0 fue bien clasificada en la mayoría de las ocasiones, es decir, hay pocos falsos negativos y pocos falsos positivos.

CAPITULO 2: MARCO APLICATIVO

En este Capítulo se expone el problema a resolver, los objetivos propuestos y la solución. Igualmente, se detalla el proceso de minería de textos aplicado para estimar el modelo de clasificación y los resultados de su evaluación. Finalmente, se señalan los aspectos relacionados con el desarrollo de la aplicación.

2.1. PLANTEAMIENTO DEL PROBLEMA

Actualmente, el proceso de asignación de jurados a Trabajos Especiales de Grado (TEG) de la Escuela de Computación de la Universidad Central de Venezuela (UCV) se realiza de la siguiente manera:

1. Los tutores de cada TEG sugieren los jurados para los mismos.

2. Dicha sugerencia es sometida a consideración del Consejo de Escuela.

3. El Consejo de Escuela analiza cada caso según ciertos criterios, principalmente: las opciones profesionales en las que se desempeñan los profesores y las abarcadas por el TEG, la cantidad de veces que un profesor ha sido asignado como jurado, la disponibilidad de los profesores, entre otros.

4. El Consejo de Escuela toma la decisión de mantener la sugerencia o realizar cambios y finalmente, se realiza la asignación de jurados.

El proceso descrito presenta ciertas características desfavorables, entre las cuales se pueden señalar:

- Es un proceso subjetivo, puesto que el Consejo de Escuela realiza la toma de decisiones en base a su criterio sobre quiénes de los profesores disponibles, resultan más convenientes como jurados para cierto tema.

- El proceso no se encuentra debidamente regulado.

- No hay información actualizada sobre los profesores que han sido asignados como tutores y la disponibilidad de los mismos.

- La información disponible se encuentra en papel, sin una organización que facilite su consulta.

En la Escuela de Computación de la UCV se han realizado esfuerzos para solventar estos inconvenientes. El más destacable de ellos fue el desarrollo de una aplicación Web para la asignación automática de jurados (Torres Navarro, 2011), sin embargo, ésta sólo era capaz de clasificar (utilizando el algoritmo de K-Vecinos) bajo cuatro etiquetas que representan las opciones profesionales (OP) con mayor demanda y cantidad de TEG disponibles en la Licenciatura en Computación, a saber: Aplicaciones con Tecnologías en Internet, Tecnologías en Comunicaciones y Redes de Computadoras, Bases de Datos e Inteligencia Artificial. Sin embargo, en la Licenciatura

en Computación se disponen de once OP, por ende, con este clasificador quedan por fuera los TEG pertenecientes a las siete restantes. Igualmente

2.2. PROPUESTA DE SOLUCIÓN

Como solución al problema planteado, se propone la automatización del proceso descrito anteriormente, desarrollando un sistema de recomendación basado en la aplicación de (Torres Navarro, 2011) que permita, mediante técnicas de minería de textos, realizar la clasificación de los TEG bajo las once OP ofertadas en la Escuela de Computación de la UCV, para seguidamente asignar como jurados a los profesores especializados en la(s) OP recomendada(s) según ciertos criterios establecidos. Entre los criterios a considerar están aquellos que fueron implementados en el sistema de (Torres Navarro, 2011):

- El profesor no puede encontrarse de permiso.
- El profesor no debe exceder cierta cantidad de asignaciones semestrales como jurado principal.
- El profesor debe encontrarse entre los docentes del área que hayan sido jurado principal de menor cantidad de TEG para el semestre dado.

En la Figura 2.1, se muestran los componentes básicos del Sistema de Asignación de Jurados a TEG propuesto. Principalmente, se plantea la construcción de un clasificador que permita asignar una o dos OP con las que se encuentra más relacionado un TEG, aplicando el proceso de minería de textos con un algoritmo basado en redes bayesianas (específicamente Bayes ingenuo). Una vez clasificado el documento, se procederá a la selección de los diferentes tipos de jurados (principales y suplentes) entre los profesores del área o áreas en cuestión, según los criterios mencionados. Sin embargo, de ser necesario, será posible la modificación de los resultados obtenidos para que se adapten a los resultados requeridos por el usuario.

2.3. OBJETIVO GENERAL

Desarrollar una aplicación basada en técnicas de minería de textos, que permita la asignación de jurados a Trabajos Especiales de Grado pertenecientes a las 11 opciones profesionales de la Licenciatura en Computación de la Facultad de Ciencias de la Universidad Central de Venezuela.

2.4. OBJETIVOS ESPECÍFICOS

- Analizar los requerimientos para el desarrollo del sistema de asignación automática de jurados a TEG.
- Recolectar la información necesaria para la creación de la aplicación, principalmente sobre los TEG, las diversas OP de Computación y el personal docente y de investigación de la Escuela.
- Aplicar el proceso de minería de textos para construir el modelo de clasificación de documentos de TEG.

- Implementar la solución planteada en una aplicación Web que asigne de manera automática los jurados según los criterios establecidos.

- Realizar las pruebas necesarias para comprobar el correcto funcionamiento del sistema desarrollado.

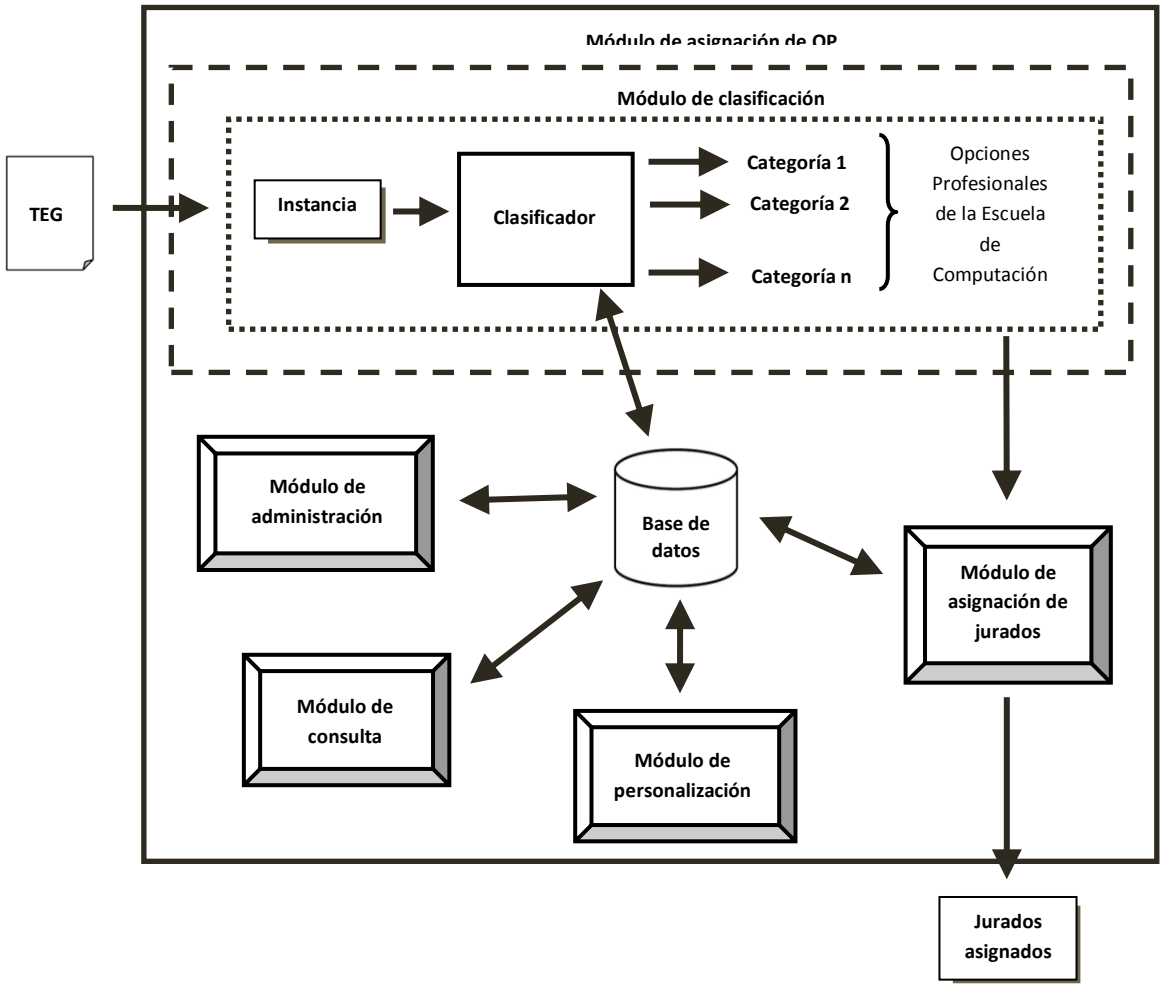

FIGURA 2.1. COMPONENTES BÁSICOS DE LA APLICACIÓN DE ASIGNACIÓN DE JURADOS A TEG

2.5. CONSTRUCCIÓN DEL MODELO DE CLASIFICACIÓN

Siguiendo los pasos especificados en la Sección 1.3., a continuación se detalla el proceso de minería de textos que se llevó a cabo para generar el modelo de clasificación.

2.5.1. RECOLECCIÓN DE LOS DATOS

Los datos sobre los cuales se va a aplicar la minería de texto son los documentos de Trabajos Especiales de Grado, para cada uno de ellos se recolectó la siguiente información:

- Título.
- Resumen (o introducción en su defecto).
- Palabras clave (si se encontraran disponibles)
- Opción Profesional a la cual pertenece el TEG.

Adicionalmente, se recolectó la siguiente información requerida para el desarrollo del sistema:

- Nombres de las opciones profesionales ofertadas en la Escuela de Computación de la UCV.
- Nombres y áreas de especialización de los profesores de la Escuela de Computación de la UCV.

La recolección de los documentos se llevó a cabo en varias etapas: primeramente se utilizó el portal de búsqueda BUSCONEST[3] de la Facultad de Ciencias, el cual tiene disponibles los TEG presentados por los estudiantes de la Escuela de Computación entre los años 2008 y 2012. Sin embargo, la mayoría de los documentos pertenecían a las opciones profesionales más demandadas de la Escuela de Computación (Aplicaciones con Tecnología Internet, Tecnologías en Comunicaciones y Redes de Computadoras, Inteligencia Artificial y Base de Datos). Por esta razón, se decidió recolectar documentos de años anteriores al 2008 (que no pertenecieran a las OP con más demanda) acudiendo directamente a la Biblioteca Alonso Gamero. Además, para complementar la búsqueda, se visitaron los centros de investigación que componen la Escuela de Computación y fueron solicitados a los profesores TEG disponibles en su área de especialización.

Al finalizar la recolección, el conjunto de documentos quedó distribuido según lo especificado en la Tabla 2.1.

TABLA 2.1. DISTRIBUCIÓN POR OP DE LOS DOCUMENTOS RECOLECTADOS PRESENTADOS ENTRE LOS AÑOS 2000 Y 2012

OP de la Licenciatura en Computación UCV	Cantidad de TEG
Aplicaciones con Tecnología Internet	52
Base de Datos	48
Calculo Científico	15
Computación Gráfica	39
Ingeniería de Software e Interacción Humano-Computador	19
Inteligencia Artificial	47
Modelos y Programación Matemática	10
Sistemas de Información	34
Sistemas Distribuidos y Paralelos	12
Tecnologías Educativas	23
Tecnologías en Comunicaciones y Redes de Computadoras	52
	N = 351

N = Número de documentos de la colección

Debido a las diferencias de formato existentes entre los documentos se realizó un proceso de extracción, transformación y carga (*Extraction, Transformation and Loading* o ETL) realizando el almacenamiento de la información relevante (título, resumen/introducción y palabras clave) en

[3] https://busconest.ciens.ucv.ve/

archivos de texto plano con formato UTF-8 sin BOM, separados en diferentes carpetas según la OP a la que pertenecían para facilitar su posterior procesamiento.

2.5.2. PRE-PROCESAMIENTO

Para este paso se desarrolló un programa en lenguaje java encargado de realizar automáticamente las tareas detalladas a continuación:

A. FILTRADO

En primer lugar, se eliminaron los signos de puntuación y demás caracteres especiales. Las vocales acentuadas y las diéresis permanecieron intactas ya que son importantes para el posterior proceso de lematización.

Seguidamente, se realizó un filtrado de aquellas palabras que no aportaran información representativa para la clasificación de textos con la ayuda de un diccionario de palabras frecuentes en el idioma español. Este diccionario fue construido recopilando palabras que por su naturaleza son frecuentes en el idioma español como lo son las preposiciones, artículos, pronombres, conjunciones, etc., además de un conjunto de palabras listadas por la RAE como palabras de alta frecuencia (Española, 2013).

TABLA 2.2. DICCIONARIO DE PALABRAS NO INFORMATIVAS EN EL IDIOMA ESPAÑOL

Preposiciones	"a", "ante", "bajo", "cabe", "con", "contra", "de", "desde", "en", "entre", "hacia", "hasta", "para", "por", "sin", "según", "so", "sobre", "tras", "mediante", "durante", "excepto", "salvo"
Artículos	"el", "la", "lo", "los", "las", "un", "una", "unos", "unas", "del", "al"
Adjetivos determinativos: demostrativos, posesivos, indefinidos e interrogativos	"este", "esta", "estos", "estas", "ese", "esa", "esos", "esas", "aquel", "aquella", "aquellos", "aquellas", "mío", "mía", "míos", "mías", "tuyo", "tuya", "tuyos", "tuyas", "suyo", "suya", "suyos", "suyas", "nuestro", "nuestra", "nuestros", "nuestras", "vuestro", "vuestra", "vuestros", "vuestras", "mi", "mis", "tu", "tus", "su", "sus", "cuyo", "cuya", "cuyos", "cuyas", "otro", "otra", "otros", "otras", "todo", "toda", "todos", "todas", "cierto", "cierta", "ciertos", "ciertas", "semejante", "semejantes", "tal", "diferente", "diferentes", "diverso", "diversa", "diversos", "diversas", "varios", "varias", "cada", "mismo", "misma", "mismos", "mismas", "tanto", "tanta", "tantos", "tantas", "cualquier", "cualquiera", "cualesquier", "cualesquier", "algún", "algunos", "algunas", "ningún", "poco", "poca", "pocos", "pocas", "mucho", "mucha", "demasiado", "demasiada", "demasiados", "demasiadas", "determinado", "determinada", "determinados", "determinadas", "cual", "cuales", "que", "quien", "quienes", "cuanto", "cuantos", "cuanta", "cuantas", "bastante"
Pronombres	"yo", "mi", "me", "conmigo", "nosotros", "nosotras", "nos", tu", "usted", "ti", "te", "contigo", "ustedes", "vosotros", "vosotras", "os", "el", "ella", "ello", "si", "se", "consigo", "le", "ellos", "ellas", "les", "vos", "esto", "eso", "aquello", "alguien", "nadie", "algo", "nada", "alguno", "alguna", "ninguno", "ninguna", "muchos", "muchas", "bastantes"

Adverbios Determinativos: de lugar, tiempo, modo, cantidad, duda, afirmación, negación	"aquí", "allí", "ahí", "acá", "allá", "cerca", "lejos", "fuera", "afuera", "dentro", "adentro", "encima", "debajo", "arriba", "abajo", "delante", "adelante", "alrededor", "detrás", "donde", "mientras", "luego", "temprano", "antes", "después", "pronto", "tarde", "ya", "ahora", "entonces", "hoy", "mañana", "ayer", "nunca", "jamás", "siempre", "todavía", "cuando", "así", "apenas", "como", "mas", "menos", "tanto", "casi", "muy", "quizá", "quizás", "acaso", "vez", "ciertamente", "también", "no", "tampoco", "solo"
Conjunciones	"y", "e", "ni", "o", "u", "ya", "ora", "sea", "fuera", "pero", "sino", "embargo", "obstante", "excepto", "salvo", "porque", "pues", "puesto", "pues", "supuesto", "dado", "luego", "conque", "consiguiente", "manera", "modo", "fin", "aunque", "aun", "siquiera"
Interjección	"ay", "ah", "oh", "huy", "uy", "bah", "hurra", "uf", "ojala", "ea", "puf", "hola", "chao", "caramba"
Verbos más usados en español	"es", "fue", "ha", "son", "ser", "tiene", "hay", "han", "estan", "estado", "nos", "estados", "uno", "fueron", "habia", "estar", "haber", "estaba", "estamos", "estoy","estamos","estais","estan","estes","estemos","esteis","esten", "estare", "estaras", "estara", "estaremos", "estareis", "estaran", "estaria", "estarias","estariamos","estariais","estarían","estaba","estabas", "estabamos", "estabais", "estaban", "estuve", "estuviste", "estuvo", "estuvimos", "estuvisteis", "estuvieron", "estuviera", "estuvieras", "estuvieramos", "estuvierais", "estuvieran", "estuviese", "estuvieses", "estuviesemos", "estuvieseis", "estuviesen", "estando", "estado", "estada", "estados", "estadas", "estad", "he", "has", "ha", "hemos", "habeis", "han", "haya", "hayas", "hayamos", "hayais" , "hayan", "habre", "habras", "habra", "habremos", "habreis", "habran", "habria", "habrias", "habriamos", "habriais", "habrian", "habia", "habias", "habiamos", "habiais", "habian", "hube", "habido", "habida", "habidos", "habidas", "soy", "eres", "es", "somos", "sois", "son", "seas", "seamos", "seais", "sean", "sere", "seras", "sera", "seremos", "sereis", "seran", "seria", "serias", "seriamos", "seriais", "serian", "era", "eras", "eramos", "erais", "eran", "fui", "fuiste", "fue", "fuimos", "fuisteis", "fueron", "fueras", "fueramos", "fuerais", "fueran", "fuese", "fueses", "fuesemos", "fueseis", "fuesen", "siendo", "sido", "tengo", "tienes", "tiene", "tenemos", "teneis", "tienen", "tenga", "tengas", "tengamos", "tengais", "tengan", "tendre", "tendras", "tendra", "tendremos", "tendreis", "tendran", "tendria", "tendrias", "tendriamos", "tendriais", "tendrian", "tenia", "tenias", "teniamos", "teniais", "tenian", "tuve", "tuviste", "tuvo", "tuvimos", "tuvisteis", "tuvieron", "tuviera", "tuvieras", "tuvieramos", "tuvierais", "tuvieran", "tuviese", "tuvieses", "tuviesemos", "tuvieseis", "tuviesen", "teniendo", "tenido", "tenida", "tenidos", "tenidas", "tened", "ser", "hay", "puede", "nos", "bien", "hace", "tan", "dijo", "hacer", "ademas", "debe", "va", "tener", "dice", "parece", "haber", "estar", "dar", "da", "iba"

Una vez eliminadas las palabras no informativas se procedió a realizar el proceso de lematización sobre el conjunto resultante utilizando el algoritmo de *Porter Stemming* (Porter, 1980) para el español. Se implementó una versión del mismo en lenguaje Java basado en el algoritmo publicado en la página de Snowball[4].

Al culminar el pre-procesamiento se obtuvo un total de 4373 raíces informativas, los cuales constituyeron el conjunto de términos a utilizar en las fases posteriores:

$$T = \{T_1, .., T_M\} \text{ con } M = 4373$$

B. INDEXACIÓN

Para la representación de los textos se utilizó el Modelo de Espacio Vectorial, calculando el peso w_{ik} del término *i* en el documento *k* aplicando cuatro de las propuestas de pesado sugeridas en la Sección 1.3.1. (apartado B): TFxIDF, TFC, LTC y entropía. La selección de estas técnicas se basó en su capacidad para tomar en cuenta factores como la diferencia de tamaño de los documentos y reducir el efecto de las diferencias de frecuencias, además de ser recurrentemente referenciados en los trabajos consultados.

Para el cálculo de los pesos se generó una matriz de frecuencias W donde cada fila representa un documento *d* perteneciente a la colección D, las columnas representan cada uno de los términos pertenecientes a T por tanto, cada posición de la matriz representa el peso de un término *t* dentro de un documento *d*.

TABLA 2.3. MATRIZ DE REPRESENTACIÓN DE LOS DOCUMENTOS RECOLECTADOS

	t_1	…	t_j	…	w_{4373}
d_1	$w_{1\,1}$	…	w_{1j}	…	$w_{1\ 4373}$
…	…	…	…	…	…
d_i	w_{i1}	…	w_{ij}	…	$w_{i\ 4373}$
…	…	…	…	…	…
d_N	$w_{N\,1}$	…	w_{Nj}	…	$w_{N\ 4373}$

C. REDUCCIÓN DE LA DIMENSIONALIDAD

La alta dimensionalidad del espacio de términos puede acarrear problemas en las fases posteriores del proceso de minería de textos. Múltiples términos no informativos pueden disminuir sustancialmente el rendimiento del clasificador al dificultar la división de los espacios de decisión, sobre todo para aquellos algoritmos que no estén preparados para enfrentarse a esos escenarios. Igualmente, entre mayor sea el espacio de términos mayor será la cantidad de cálculos necesarios tanto para la generación del modelo como para su clasificación y por tanto, se requerirá mayor poder de cómputo para obtener los resultados en un tiempo razonable para el usuario.

[4] http://snowball.tartarus.org/

Para solventar esta problemática se emplearon las funciones para selección de características señaladas en la Sección 1.3.1. (apartado C): Ganancia de Información (*Info Gain*), Chi-Cuadrado (X^2) y Razón de Ganancia (*Gain Ratio*). Existen diversas herramientas que permiten aplicar estas técnicas, en este trabajo se utilizó el sistema multiplataforma, gratuito y cuya licencia es de software libre, Weka[5] en su versión 3.6.4.

Cada una de las matrices obtenidas luego de aplicar las cuatro propuestas de pesado fueron almacenadas en un archivo con el formato aceptado como entrada por Weka: ".arff" (Anexo 1), en el cual cada uno de los términos en T se toman como atributos de tipo numérico.

Para definir el número de características finales se elaboró un sistema de votación eligiendo aquellos atributos que fueran votados por los menos en dos de las tres funciones de selección. La cantidad de atributos obtenidos luego de evaluar el sistema de votación fueron 126 (Anexo 3), teniendo como resultado cuatro posibles vistas minables (una por cada propuesta de pesado), lo que implica que la dimensionalidad fue reducida en un 97%.

2.5.3. ESTIMACIÓN Y EVALUACIÓN DEL MODELO DE CLASIFICACIÓN

La tarea de minería de datos escogida para la realización de este trabajo fue la de categorización (clasificación). Para cada matriz de indexación, se realizaron un conjunto de experimentos aplicando los algoritmos de K-Vecinos (IBK en Weka) con K = 3, 5 y 9, árboles de decisión (específicamente C4.5, denominado J48 en Weka) y Bayes Ingenuo (*Naïve Bayes* en Weka), siendo éste último el que presentó mayor rendimiento con la vista minable pesada con entropía, logrando un 82,0513 % de exactitud.

Como método de evaluación se utilizó validación cruzada con diez particiones (*10 fold cross-validation*) y como medida de evaluación se utilizaron la matriz de confusión, la precisión, la sensibilidad o *recall* y la medida F. En la Tabla 2.4, pueden observarse los resultados de los experimentos realizados sobre la matriz de indexación pesada con entropía. Los resultados obtenidos con otros pesados se encuentran en el Anexo 2.

TABLA 2.4. RESULTADOS DE EXPERIMENTOS CON WEKA EN LA VISTA MINABLE PESADA CON ENTROPÍA

Algoritmos	Clasificaciones	Cantidad	Porcentaje
K-NN ($K = 3$)	Correctas	217	61,8234%
	Incorrectas	134	38,1766%
K-NN ($K = 5$)	Correctas	219	62,3932%
	Incorrectas	132	37,6068%
K-NN ($K = 7$)	Correctas	239	68,0912%
	Incorrectas	112	31,9088%
C4.5	Correctas	210	59.8291%
	Incorrectas	141	40.1709 %
Bayes Ingenuo	Correctas	288	**82,0513%**
	Incorrectas	63	17,9487%

[5] http://www.cs.waikato.ac.nz/ml/weka/

El detalle de los resultados y evaluación obtenidos con Bayes Ingenuo puede apreciarse en la Tabla 2.5.

TABLA 2.5. EVALUACIÓN MODELO DE BAYES INGENUO

Bayes Ingenuo con pesado por Entropía		
Clasificaciones Correctas	288	82,0513%
Clasificaciones Incorrectas	63	17,9487%
Nro. Total de Instancias	351	100,00%

Medidas de Rendimiento			
Opción Profesional	**Precisión**	**Recall**	**Medida F**
a = Aplicaciones con Tecnología Internet	0.764	0.808	0.785
b = Base de Datos	0.74	0.771	0.755
c = Cálculo Científico	0.929	0.867	0.897
d = Computación Gráfica	0.841	0.949	0.892
e = Ingeniería de Software e Interacción Humano-Computador	0.667	0.526	0.588
f = Inteligencia Artificial	0.957	0.957	**0.957**
g = Modelos y Programación Matemática	1	0.7	0.824
h = Sistemas de Información	0.727	0.706	0.716
i = Sistemas Distribuidos y Paralelos	**0.5**	**0.417**	**0.455**
j = Tecnologías Educativas	0.84	0.913	0.875
k = Tecnologías en Comunicaciones y Redes de Computadoras	0.922	0.904	0.913

Matriz de Confusión

	a	b	c	d	e	F	g	h	i	j	K
a	42	0	0	0	2	1	0	4	0	1	2
b	3	37	0	0	2	0	0	2	3	0	1
c	0	0	13	0	0	0	0	0	0	2	0
d	1	0	0	37	0	0	0	0	1	0	0
e	3	2	0	2	10	0	0	2	0	0	0
f	0	0	0	1	0	45	0	1	0	0	0
g	0	0	0	1	0	0	7	0	1	1	0
h	4	5	0	0	0	0	0	24	0	0	1
i	1	3	1	2	0	0	0	0	5	0	0
j	0	1	0	0	1	0	0	0	0	21	0
k	1	2	0	1	0	1	0	0	0	0	47

Al observar la medida F de cada clase se evidencia que Inteligencia Artificial es la OP que presenta un valor más alto, lo que indica que es el área mejor clasificada por el modelo; Sistemas Distribuidos y Paralelos tiene la medida F con el valor más bajo, por ende es la OP que tiende a ser peor clasificada por el modelo. Al detallar la matriz de confusión se puede notar que la OP Sistemas Distribuidos y Paralelos tiende a ser confundida con Base de Datos y viceversa; ocurre algo similar con la OP que presenta la segunda medida F más baja, Ingeniera de Software e Interacción Humano-Computador, la cual tiende a clasificarse como Aplicaciones con Tecnología Internet; estos errores se atribuyen a la fuerte relación que tienen las OP unas con otras, respectivamente.

2.6. DESARROLLO DEL SISTEMA

A continuación, se desglosan las principales tecnologías empleadas para el desarrollo del sistema, la arquitectura del mismo y cada uno de los módulos que lo componen (los cuales fueron especificados en la Figura 2.1.), así como los diagramas de casos de uso.

2.6.1. TECNOLOGÍAS UTILIZADAS

Las principales tecnologías que se emplearon para el desarrollo del Sistema de Asignación de Jurados a TEG fueron las siguientes:

A. PLATAFORMA JAVA

La Plataforma Java es el nombre de un entorno capaz de ejecutar aplicaciones desarrolladas usando el lenguaje de programación Java y un conjunto de herramientas de desarrollo. La plataforma no es un hardware específico o un sistema operativo, sino más bien una máquina virtual encargada de la ejecución de las aplicaciones, y un conjunto de bibliotecas estándar que ofrecen una funcionalidad común.

Existen diversas plataformas Java muy populares, pero en específico para la implementación de este sistema se utilizó como *kit* de desarrollo de software o SDK (por sus siglas en inglés, *software development kit*) el Java EE 6 SDK.

B. JSP Y SERVLETS

Las *Java Server Pages* (JSP) y los *Servlets* son tecnologías Java EE utilizadas en aplicaciones Web. Los *Servlets* son clases del lenguaje de programación Java que procesan solicitudes dinámicamente y construyen respuestas, generalmente para páginas HTML. Las JSP son documentos basados en texto que son compilados en *Servlets* y que permiten crear páginas Web dinámicas a partir de los parámetros de petición que envíe el navegador Web. Debido a esto, las especificaciones de las JSP van ligadas a una especificación de los *Servlets*.

Específicamente, se utilizaron las especificaciones JSP 2.1 y *Servlet* 2.5.

C. APACHE TOMCAT

Tomcat es un contenedor Web o motor de *Servlets* (*Servlet Engine,* que implementa las especificaciones de los *Servlets* y JSP), por lo que se puede decir que es un servidor Web que funciona bajo la tecnología JEE de Java. Ofrece una solución para la ejecución de páginas Web dinámicas desarrolladas bajo la tecnología JSP, al ofrecer un entorno donde habitan los JSP y *Servlets* (contenedor). Por defecto se presenta en combinación con el servidor web Apache.

En este trabajo se utilizó la versión 7.0.34.0 de Apache Tomcat.

D. MySQL Server

MySQL es uno de los sistemas de gestión de bases de datos relacionales SQL más populares. Una base de datos relacional almacena datos en tablas separadas en lugar de poner todos los datos en un gran almacén, lo que añade velocidad y flexibilidad. Para añadir, acceder, y procesar los datos almacenados en una base de datos, se necesita un sistema de gestión de base de datos como MySQL Server.

MySQL es soportado por una gran cantidad de lenguajes y aplicaciones, como el lenguaje Java utilizado en este trabajo, mediante JDBC. En esta ocasión se desarrolló específicamente con la versión MySQL Server 5.5.32.

E. NetBeans

El IDE NetBeans es una herramienta pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Es de código abierto y gratuito sin restricciones de uso.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases escritas para interactuar con las APIs de NetBeans y un archivo especial (*Manifest File*). Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

Para el desarrollo del sistema presentado en este trabajo se ha utilizado NetBeans en su versión 7.3.4.

2.6.2. Arquitectura del sistema

El diseño del sistema se basa en el patrón de arquitectura de software llamado **Modelo-Vista-Controlador** (MVC), puesto que permite separar la lógica de negocio, el manejo de los datos y la presentación de los mismos. Este patrón plantea tres elementos básicos para la estructura de una aplicación:

- **Modelo:** Representación de la información en el sistema, también define las reglas de acceso a los objetos y cómo será su manipulación; en este sistema los modelos son representados por las clases JavaBeans.

- **Controlador:** Responde a eventos (usualmente acciones del usuario) e invoca peticiones al Modelo cuando se hace alguna solicitud sobre la información. También puede enviar comandos a su Vista asociada si se solicita un cambio en la forma en que se presenta el Modelo. En este sistema el Controlador está representando por los *Servlets* quienes reciben las peticiones de la aplicación y realizan las operaciones haciendo uso de los elementos provistos por el Modelo.

- **Vista:** Presenta el Modelo en un formato adecuado para la interacción con el usuario. En este sistema las Vistas son representadas por los diferentes JSP, quienes se encargan de mostrarle al usuario la información con las reglas de presentación establecidas, esto se logra accediendo a la data del Modelo gracias al lenguaje de expresión (*EL – Expression Language*).

Cada elemento es vital y cumple una función específica en el sistema como puede apreciarse en la Figura 2.2.

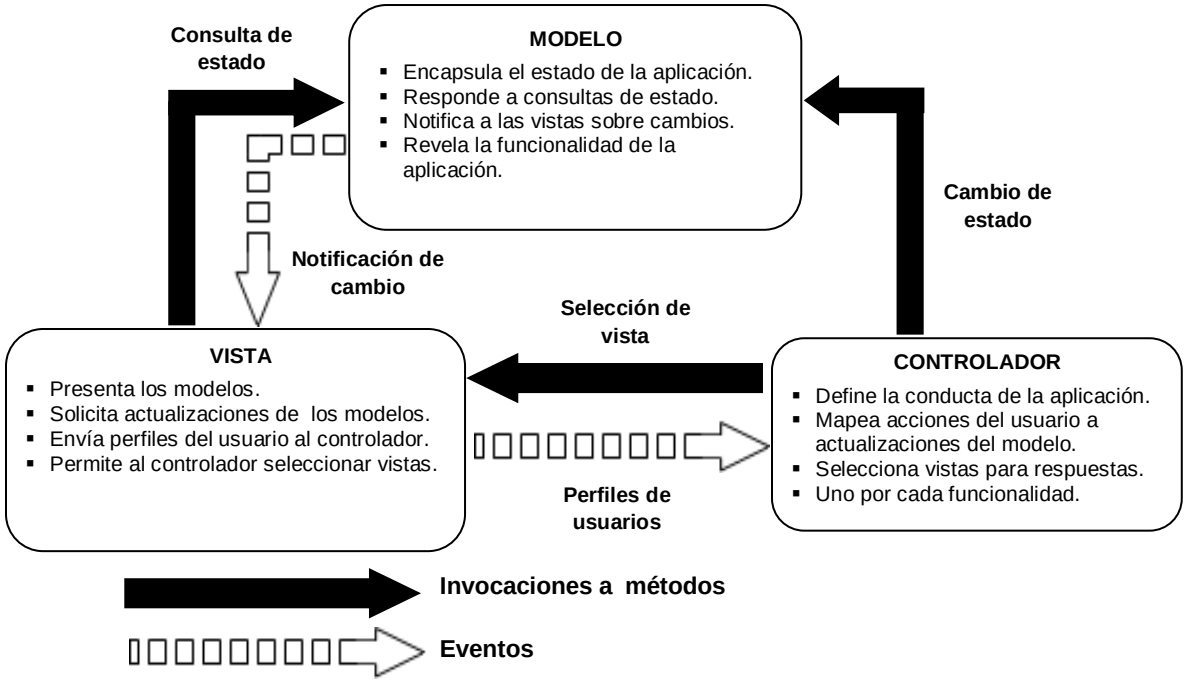

FIGURA 2.2. ESTRUCTURA BÁSICA DEL PATRÓN DE DISEÑO MVC (TORRES NAVARRO, 2011)

Existen muchas variaciones de implementación para este patrón de arquitectura, en el presente trabajo se utilizó **MVC basado en peticiones** el cual trabaja directamente con los objetos *HttpServletRequest* y *HttpServletResponse* que corresponden a la información del *Request* y *Response* HTTP respectivamente.

Asimismo, se utilizaron patrones de diseño para implementar los sub-elementos del sistema. Para el Controlador se utilizó el patrón *Front Controller* mediante la implementación de un único *Servlet,* el cual provee un punto centralizado para la captación de peticiones facilitando la compartición de información en el sistema (evita redundancia de datos que de otra forma tendrían que estar replicados en múltiples *Servlets*) y permitiendo un control de acceso sencillo con menor posibilidad de fallas. Cuando se aplica este patrón el *Servlet* suele recibir el nombre de *Redirector* (o Redireccionador).

El Modelo trabaja con la información disponible en el Servlet tales como el *PathInfo*, *ServletPath* y los parámetros específicos de las peticiones que reciba. La lógica de negocio es desarrollada en clases llamadas *Actions,* cada uno de estos *Actions* sigue el patrón *Strategy* donde el comportamiento de la clase se determina en el momento de ejecución; esto se logra mediante la implementación de una interfaz que luego es extendida para cumplir la lógica requerida por los

diferentes módulos del sistema. Estos *Actions* son utilizados por el *Servlet* mediante una clase llamada *ActionFactory* la cual se encarga de hacerle llegar al *Servlet* la implementación adecuada de la clase *Action* según el *PathInfo* provisto (ver Figura 2.3.).

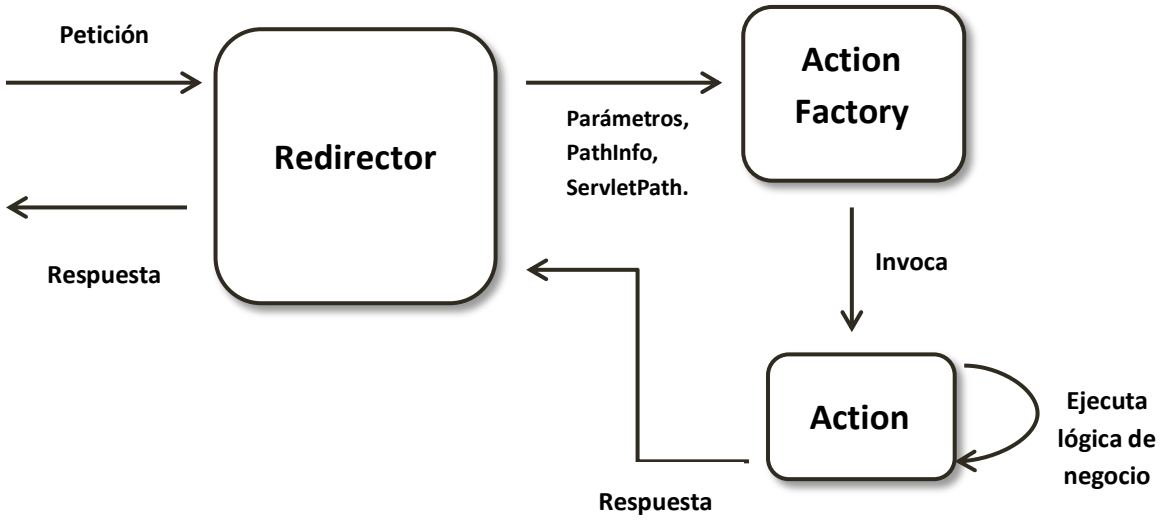

FIGURA 2.3. ARQUITECTURA DEL SISTEMA

Al iniciar el sistema (específicamente cuando el *Redirector* es inicializado) se procede a tomar la vista minable disponible en el sistema y se construye el modelo de clasificación realizando los cálculos necesarios para el algoritmo Bayes Ingenuo (cálculo de las probabilidades de cada clase y generación de las distribuciones clase-atributo); además se guarda cualquier dato necesario para cálculos futuros, como el número de instancias que conforman el modelo y el número de atributos (además de la lista de los mismos). Posterior a su construcción, el modelo queda disponible en el contexto de la aplicación para su uso en cualquier petición.

Aunque la construcción del modelo es un proceso computacionalmente costoso, su creación en tiempo de ejecución permite reemplazar la vista minable asociada sin que esto implique un cambio en la codificación del sistema. Sin embargo, si se desea realizar modificaciones que impliquen un cambio en la lógica, como es el caso de cambio de pesado o de algoritmo de clasificación, se requerirá la modificación de las clases correspondientes según sea el caso.

2.6.3. DESARROLLO DE LOS MÓDULOS DEL SISTEMA

El sistema está divido en 6 módulos: Módulo de asignación de opciones profesionales, módulo de clasificación, módulo de asignación de jurados, módulo de consulta, módulo de administración y módulo de personalización (ver Figura 2.1). Solamente el módulo de consulta está disponible para usuarios que no sean administradores.

A. MÓDULO DE ASIGNACIÓN DE OPCIONES PROFESIONALES

Este módulo recibe los siguientes datos correspondientes al TEG en texto plano: tutores, título, resumen o introducción y palabras claves; y los datos del autor o autores del TEG: nombre,

apellido, cédula de identidad y opcionalmente el correo electrónico. Con estas entradas el módulo de asignación de OP prepara los objetos necesarios para su almacenamiento final en base de datos luego de concluir la asignación de jurados. También se encarga de preparar la información del TEG para su procesamiento por el sub-módulo de clasificación y al finalizar éste, el módulo de asignación de OP recibe los resultados para su confirmación por parte del usuario y seguidamente pasar al módulo de asignación de jurados.

El sistema registra si el usuario utilizó o no la recomendación de OP realizada por el clasificador para llevar estadísticas del sistema.

La verificación de la correctitud y coherencia de los datos introducidos no está contemplado dentro del alcance de éste trabajo, por ende, esto queda a conciencia y expensas del usuario del sistema. La introducción de datos incoherentes puede afectar el comportamiento del clasificador, las estadísticas del sistema y crear historiales de TEG embasurados.

B. MÓDULO DE CLASIFICACIÓN

El módulo de clasificación se encuentra dentro del módulo de asignación de OP y es invocado por éste cuando ya ha recibido y preparado los datos pertenecientes a un TEG. En este escenario se realizan dos actividades separadas, primero se invoca a la función de construcción de instancias la cual se encarga de transformar los datos del TEG (título, resumen y palabras clave) en una estructura de datos válida para su procesamiento por el clasificador mediante el pre-procesamiento de los textos (limpieza, lematización e indexación mediante el pesado por entropía de los lemas que contenga el documento que pertenezcan al conjunto de términos aceptados por el clasificador). La segunda actividad consiste en realizar el proceso de clasificación de la instancia recién creada, para ello se procede a calcular por cada clase el valor de la fórmula de Bayes Ingenuo:

$$P(C_i) \ * \prod_{j=1}^{M} P(a_j|C_i)$$

Donde $P(C_i)$ corresponde a la probabilidad de la clase a evaluar (calculada previamente al construirse el modelo de clasificación según la vista minable disponible al momento de iniciar el sistema) y $P(a_j|C_i)$ es la probabilidad condicional que relaciona la aparición del atributo a_j en un documento con la pertenencia de éste a la clase C_i, la cual se calcula mediante la función de densidad de la distribución clase-atributo correspondiente. Las distribuciones clase-atributo siguen una función normal.

Finalmente se retornan las dos clases con la mayor probabilidad obtenida (si la segunda clase posee una probabilidad muy baja sólo se retorna la primera). Este proceso se encuentra ilustrado en la Figura 2.4.

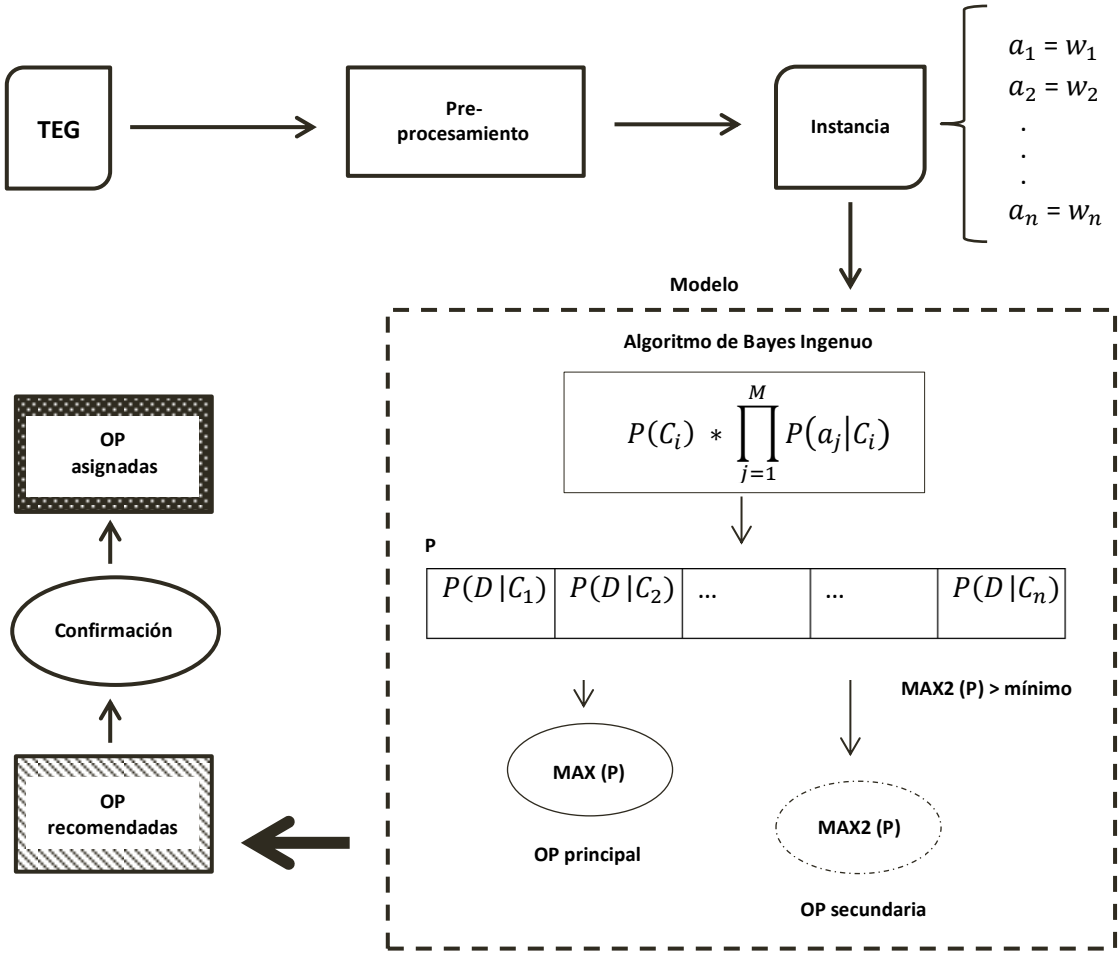

FIGURA 2.4. ESQUEMA DE CLASIFICACIÓN

C. MÓDULO DE ASIGNACIÓN DE JURADOS

Posterior a la confirmación por parte del usuario de las OP relacionadas al TEG, el módulo de asignación de OP llama al módulo de asignación de jurados, el cual busca en el sistema los profesores necesarios para conformar el jurado del TEG (cuatro en total, dos como principales y dos como suplentes) que cumplan con las siguientes reglas:

- Los jurados asignados no pueden ser los tutores del TEG.
- El profesor tiene que estar activo en el sistema.
- Un profesor no será asignado como jurado si ya ha alcanzado el máximo de asignaciones por semestre (el cual es definido en la opción de máximo de asignaciones del módulo de administración)

El proceso de asignación tanto para jurados principales como para suplentes se realiza tomando en cuenta lo siguiente:

- Si el TEG tiene una única OP asignada, los cuatros jurados se seleccionan dentro de los docentes especializados en dicha OP. De no encontrarse disponible ningún profesor, se le notifica al usuario.

- Si el TEG tiene dos OP asignadas, se seleccionan dos profesores de cada área, uno como principal y uno como suplente. Si no se encuentran disponibles más docentes de un área, se selecciona a un docente de la otra área. Si tampoco se encuentra disponible un docente de la otra área, se le notifica al usuario.

- Siempre se busca el profesor con menor cantidad de asignaciones como jurado principal.

- En caso de empate se busca el profesor con menor cantidad de asignaciones como jurado suplente.

- Si nuevamente existe un empate se elige un profesor aleatoriamente.

- Cuando se elige un profesor, éste se elimina de la lista de posibles opciones a jurado del TEG actual antes de proceder a la elección del siguiente.

Estas reglas fueron tomadas del sistema desarrollado por (Torres Navarro, 2011), las cuales fueron creadas consultando con expertos en el proceso de asignación de jurados dentro de la Escuela de Computación (el pseudoformal de estas reglas está disponible en el Anexo 4).

Finalmente, se le muestra al usuario el resultado de la asignación junto a todos los datos del TEG los cuales pueden ser exportados a un documento de texto o impresos.

Sólo al concluir exitosamente el proceso de asignación es que los objetos correspondientes al TEG (los estudiantes, los tutores y los jurados) son almacenados en la base de datos del sistema.

Es posible seleccionar los jurados que hayan quedado sin asignar por no cumplir con alguna regla o modificar los previamente asignados, en el módulo de administración.

D. MÓDULO DE CONSULTA

El módulo de consulta permite a todo usuario consultar el histórico de TEG registrados en el sistema. La búsqueda puede ser filtrada mediante los siguientes parámetros:

- **Semestre:** Semestre en el cual fue registrado el TEG.

- **Profesor:** Nombre del profesor involucrado en el TEG. En caso de no ser suministrada mayor información mediante el campo de "tipo de profesor" la búsqueda devolverá los TEG donde el profesor haya sido tutor o jurado.

- **Tipo de Profesor:** Indica si el profesor seleccionado fue jurado o tutor.

- **Tipo de Tutor:** Sólo se activa en caso de ser seleccionado el valor "tutor" en el campo "tipo de profesor". Indica si el profesor fue tutor principal o suplente.

- **Tipo de Jurado:** Sólo se activa en caso de ser seleccionado el valor “jurado” en el campo “tipo de profesor. Indica si el profesor fue jurado principal o suplente.

- **Estudiante:** Nombre del autor del TEG.

Los resultados obtenidos podrán ser ordenados por ID, título, o semestre, también es posible la configuración del número de resultados a mostrar por página. Igualmente se puede ver en detalle los datos del TEG con la posibilidad de ser exportados o impresos.

E. MÓDULO DE ADMINISTRACIÓN

El módulo de administración permite la manipulación de los diferentes elementos que componen el sistema, además de la visualización de las estadísticas que se llevan dentro del mismo. Se encuentra divido en las siguientes opciones:

- **Semestre:** Proporciona las opciones de creación, modificación y eliminación de semestres. También permite la activación de nuevos semestres en la opción de modificación. Solamente puede existir un semestre activo a la vez, para cambiar de semestres primero se deberá cerrar el actual en la sección cerrar semestres.

- **Cerrar semestre:** Cierra el semestre actual lo cual reinicia el número de asignaciones para todos los profesores, además de inhabilitar el módulo de recomendación de áreas hasta que un nuevo semestre sea activado.

- **Opciones profesionales:** Permite la creación, modificación y eliminación de opciones profesionales.

- **Profesores:** Se encuentran disponibles las opciones creación, modificación y eliminación de profesores. Además permite asociarle a un profesor una o más áreas de conocimiento para las cuales podrás ser elegidos como jurados de TEG.

- **Estudiantes:** Admite la modificación o eliminación de estudiantes (los estudiantes sólo son creados luego de la asignación de jurados de su TEG).

- **Usuarios:** Hace posible la creación, modificación y eliminación de usuarios, así como el reinicio de contraseñas y los cambios de rol.

- **Máximo de asignaciones:** Opción para la modificación del máximo de asignaciones como tutor principal que un profesor puede tener por semestre.

- **Trabajos de grado:** Pone a disposición las opciones de eliminación de TEGs, visualización de los datos detallados de los mismos, su exportación e impresión, además de la reasignación de jurados (siempre y cuando el TEG sea del semestre actual).

- **Estadísticas:** Gráficas comparativas entre las asignaciones realizadas por el módulo de clasificación y las aceptadas por el usuario.

En cada sección es posible buscar registros específicos, ordenar los registros y manejar el número de resultados a mostrar por página.

F. MÓDULO DE PERSONALIZACIÓN

Módulo que le permite a todo usuario modificar los datos específicos para su perfil:

- Nombre.
- Apellido.
- Email.
- Login.
- Contraseña.

Para hacer efectiva cualquier modificación será necesario que el usuario introduzca su contraseña actual.

2.6.4. DIAGRAMAS DE CASOS DE USO

Los casos de usos forman parte del análisis de un sistema, al modelar sus funcionalidades ayuda a describir qué es lo que el sistema debe hacer y cómo el usuario interactúa con él. A continuación, se presentan los casos de uso modelados para el desarrollo del Sistema de Asignación de Jurados a TEG.

A. CASOS DE USO – NIVEL 0

En este nivel se modela el sistema a nivel general, con sus respectivos actores (Ver Figura 2.5.). Los actores se describen en la Tabla 2.6.

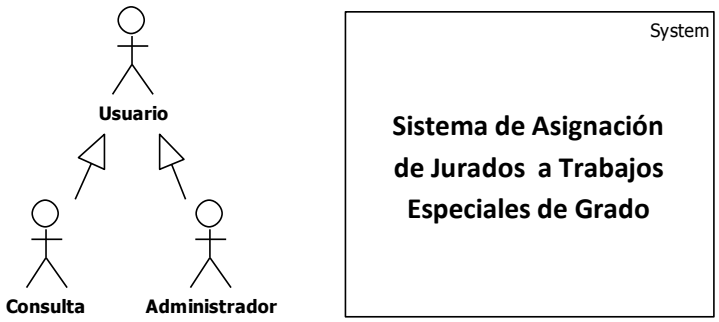

FIGURA 2.5. CASOS DE USO – NIVEL 0

TABLA 2.6. ACTORES DEL SISTEMA DE ASIGNACIÓN DE JURADOS A TEG

ACTORES	DESCRIPCIÓN
Usuario	Actor que representa a los participantes que interactúan con el Sistema de Asignación de Jurados a TEG. Se distinguen dos tipos de usuario: Administrador y Consulta.
Administrador	Usuario que tiene pleno acceso al sistema. Se encarga de su configuración y mantenimiento.
Consulta	Usuario registrado que utiliza la aplicación con fines informativos. Puede visualizar los TEG que han sido registrados en el sistema y sus respectivos jurados asignados.

B. CASOS DE USO – NIVEL 1

En este nivel se refleja en términos generales la interacción que tiene cada actor con el sistema. En la Figura 2.6., se observa que el usuario Consulta interactúa con las funcionalidades generales del mismo, a excepción de la funcionalidad administrativa y la de asignación de jurados; en cambio, el usuario Administrador interactúa con todas las funcionalidades. La descripción de los casos de uso del nivel 1 se expone en la Tabla 2.7, en la cual se indican los elementos que interactúan en cada uno de estos.

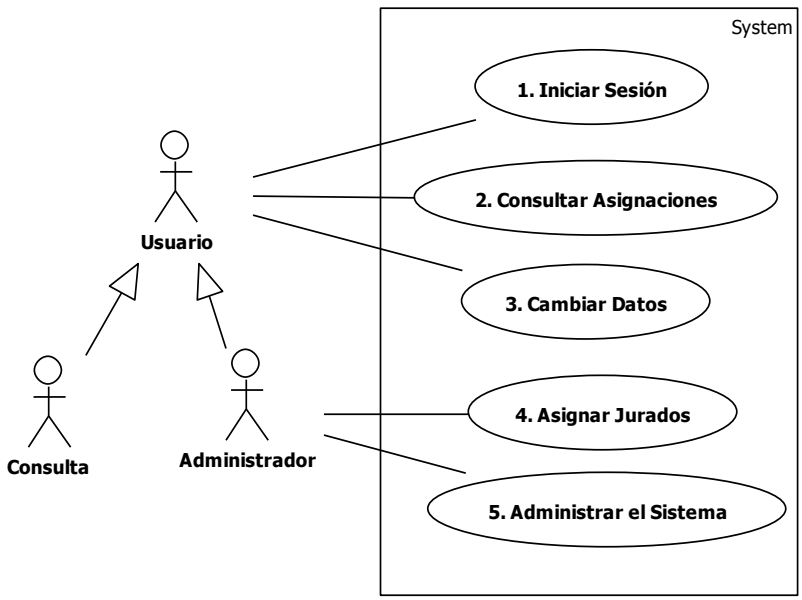

FIGURA 2.6. CASOS DE USO – NIVEL 1

TABLA 2.7. DESCRIPCIÓN CASOS DE USO – NIVEL 1

CASO DE USO 1	
Nombre	Iniciar sesión.
Actor	Consulta y Administrador.
Descripción	Permite a los usuarios ingresar al sistema.
Pre-condición	El usuario debe estar registrado en el sistema.
Flujo Básico	- El usuario ingresa a la página de inicio de sesión. - Se solicita su nombre de usuario y contraseña. - Se verifica la validez de los datos. - De tratarse de un usuario registrado, se le da acceso al sistema, de lo contrario, se le notifica la invalidez de los datos introducidos.
Post-condición	El usuario puede acceder a las funcionalidades del sistema permitidas según su rol.
CASO DE USO 2	
Nombre	Consultar Asignaciones.
Actor	Consulta y Administrador.
Descripción	Según diferentes criterios de búsqueda, se muestran los datos de los TEG registrados en el sistema junto con sus jurados asignados.
Pre-condición	El usuario debe haber iniciado sesión y debe seleccionar los parámetros de búsqueda.
Flujo Básico	- El usuario escoge los parámetros de búsqueda de TEG: por semestre, por profesor y por tipo de jurado o tipo de tutor. - El sistema muestra los resultados filtrados de acuerdo a los parámetros seleccionados. - Si en el sistema no se encuentran TEGs que cumplan con los parámetros de búsqueda especificados, la lista de resultados se mostrará vacía con un mensaje informativo.
Post-condición	Listado con TEG que cumplieron con el filtro de los parámetros seleccionados.
CASO DE USO 3	
Nombre	Cambiar Datos.
Actor	Consulta y Administrador.
Descripción	Se permite al usuario cambiar los datos de su cuenta.
Pre-condición	El usuario debe haber iniciado sesión.
Flujo Básico	- El usuario modifica cualquiera de sus datos (nombre, apellido, email, login y contraseña) y para ser guardados los mismos en el sistema debe introducir adicionalmente su contraseña actual. - Se validan los datos y la introducción de la contraseña actual. - De ser válidos los datos, se actualizan los mismos, de lo contrario, se notifica al usuario con un mensaje de error.
Post-condición	Datos de usuario cambiados.
CASO DE USO 4	
Nombre	Asignar Jurados.
Actor	Administrador.

Descripción	Permite al administrador clasificar y asignar los jurados a un TEG.
Pre-condición	El usuario debe haber iniciado sesión en el sistema y su rol debe ser de administrador.
Flujo Básico	- El administrador introduce los datos requeridos del TEG. - El sistema le recomienda una o dos áreas para la clasificación del mismo. - El administrador confirma las áreas recomendadas o las modifica. - El sistema asigna los jurados al TEG acorde al área o a las áreas escogidas.
Post-condición	TEG con jurados asignados.
CASO DE USO 5	
Nombre	Administrar el Sistema.
Actor	Administrador.
Descripción	Se permite al usuario administrador manipular los elementos que afectan el funcionamiento del sistema.
Pre-condición	El usuario debe haber iniciado sesión con rol de administrador.
Flujo Básico	-El usuario administrador ingresa a la sección Administración y el sistema despliega una lista de todas las áreas configurables y modificables (semestres, opciones profesionales, profesores, estudiantes, trabajos especiales de grado, usuarios del sistema, etc.) -El administrador selecciona la sección que desea modificar o configurar.
Post-condición	El sistema es configurado y/o modificado.

C. CASOS DE USO – NIVEL 2

En este nivel se especifican las funcionalidades del sistema ofrecidas a los actores del mismo, las cuales se especifican en la Figura 2.7. Dichas funcionalidades se describen en la Tabla 2.8.

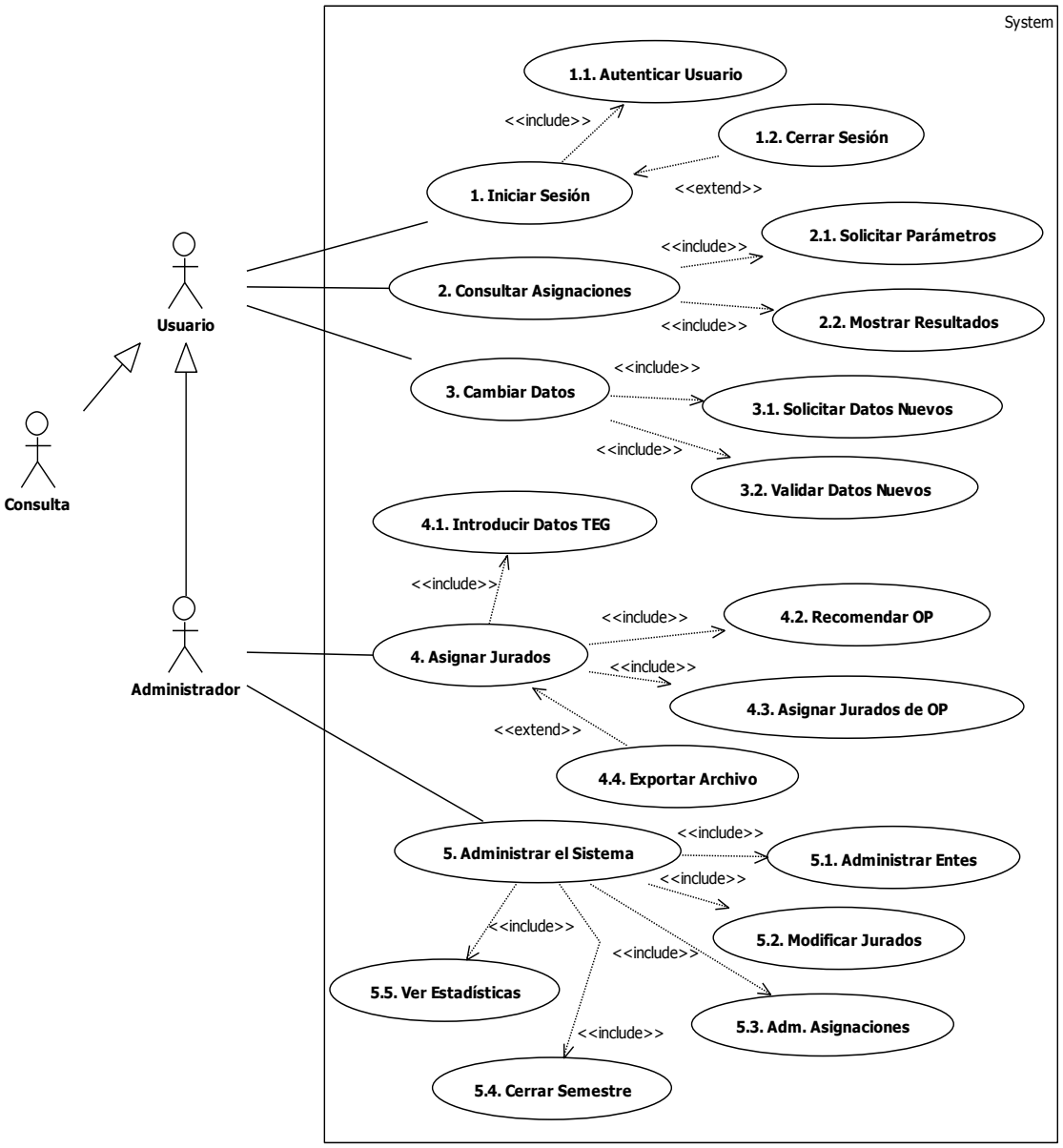

FIGURA 2.7. CASOS DE USO – NIVEL 2

TABLA 2.8. DESCRIPCIÓN CASOS DE USO – NIVEL 2

CASO DE USO 1.1	
Nombre	Autenticar Usuario.
Actor	Consulta y Administrador.
Descripción	Se verifica que el usuario se encuentra registrado en el sistema para así permitir el acceso al mismo.
Pre-condición	Ninguna.
Flujo Básico	- El usuario ingresa a la página de inicio de sesión e introduce su nombre de usuario y contraseña.

	- Se verifica que los datos sean válidos (no vacíos). - Se verifica que exista una coincidencia en base de datos entre el nombre de usuario y la contraseña introducidas. - De tratarse de un usuario registrado, se le da acceso al sistema, de lo contrario, se le notifica la invalidez de los datos introducidos.
Post-condición	Si el usuario se encuentra registrado en el sistema puede acceder a las funcionalidades del sistema permitidas según su rol, sino se permanecerá en la página de inicio de sesión.
CASO DE USO 1.2	
Nombre	Cerrar Sesión.
Actor	Consulta y Administrador.
Descripción	El usuario realiza esta acción para salir del sistema y eliminar su data almacenada en sesión.
Pre-condición	Haber iniciado sesión.
Flujo Básico	- El usuario se dirige a la sección en donde se encuentran sus datos y elige la opción de cerrar sesión. - Los datos del usuario almacenados en sesión son borrados. - Se redirige a la pantalla de inicio de sesión.
Post-condición	Sesión cerrada.
CASO DE USO 2.1.	
Nombre	Solicitar Parámetros.
Actor	Consulta y Administrador.
Descripción	Se solicitan los parámetros necesarios para realizar la búsqueda de asignaciones a TEG.
Pre-condición	El usuario debe haber iniciado sesión.
Flujo Básico	- Se solicita al usuario la información necesaria para consultar las asignaciones de jurados, como: semestre, estudiante (autor del TEG) y profesor. - Si se escoge un profesor se puede seleccionar qué tipo de profesor buscar (jurado o tutor), sino se escoge ningún tipo de profesor se buscarán todos los TEG que tengan coincidencias con el profesor seleccionado.
Post-condición	Parámetros de búsqueda configurados.
CASO DE USO 2.2.	
Nombre	Mostrar Resultados.
Actor	Consulta y Administrador.
Descripción	Se consultan las asignaciones a TEG y se muestra una lista filtrada de acuerdo a los parámetros de búsqueda establecidos.
Pre-condición	EL usuario debe haber establecido los parámetros de búsqueda.
Flujo Básico	- En base a los parámetros de búsqueda seleccionados por el usuario, el sistema realiza la consulta a la base de datos. - Se muestra una lista con las coincidencias encontradas con los TEG

	registrados en el sistema.
Post-condición	Lista de TEG registrados en el sistema y sus asignaciones o lista vacía (si no hay coincidencias para los parámetros de búsqueda).
CASO DE USO 3.1.	
Nombre	Solicitar Datos Nuevos
Actor	Consulta y Administrador.
Descripción	Se solicitan los datos que el usuario quiera cambiar en su cuenta.
Pre-condición	El usuario debe haber iniciado sesión.
Flujo Básico	- En la sección Cambiar Datos, el usuario modificará los atributos de su cuenta que desee. -Para realizar los cambios el usuario debe ingresar en el respectivo campo su contraseña anterior como medida de seguridad.
Post-condición	Validación de los datos introducidos.
CASO DE USO 3.2.	
Nombre	Validar Datos Nuevos
Actor	Consulta y Administrador.
Descripción	Verificación y validación de los datos modificados por el usuario en su cuenta.
Pre-condición	El usuario debe haber iniciado sesión.
Flujo Básico	- Se verifica que los datos ingresados no sean vacíos y coincidan con el formato pertinente (por ejemplo, el nombre y apellido deben contener sólo letras). -Se comprueba que la contraseña actual introducida sea la que este asociada a la cuenta del usuario, si es así se modifican los datos de la cuenta, sino se muestra un mensaje con el error correspondiente.
Post-condición	Datos de usuario modificados.
CASO DE USO 4.1.	
Nombre	Introducir Datos TEG.
Actor	Administrador.
Descripción	Se solicitan los datos pertinentes del TEG para realizar su clasificación y posterior asignación de jurados.
Pre-condición	El usuario debe haber iniciado sesión como administrador.
Flujo Básico	- El usuario introduce los datos solicitados correspondientes al TEG. - Se verifica que los datos ingresados coincidan con el formato pertinente y los datos obligatorios no sean vacíos. - En caso de ocurrir algún inconveniente se muestra el respectivo mensaje de error. Sólo se procederá a la recomendación de áreas cuando todos los datos introducidos sean correctos.
Post-condición	Recomendación de opciones profesionales relacionadas al TEG.
CASO DE USO 4.2.	
Nombre	Recomendar Opciones Profesionales.
Actor	Administrador.
Descripción	Se recomienda una o dos opciones profesionales bajo las cuales puede ser clasificado un TEG luego de haber procesado los datos

	solicitados del mismo.
Pre-condición	- El usuario debe haber iniciado sesión como administrador. - El administrador debe haber ingresado los datos requeridos del TEG (título, resumen, palabras clave, tutor(es) y autor(es))
Flujo Básico	- El sistema procesa los datos del TEG aplicando el modelo definido por el clasificador del sistema, el cual produce como salida la recomendación de una o dos opciones profesionales. Dichas áreas pueden ser confirmadas o ser editadas de así considerarlo necesario.
Post-condición	Asignación de jurados al TEG tomando en cuenta las opciones profesionales seleccionadas.
CASO DE USO 4.3.	
Nombre	Asignar Jurados de Opciones Profesionales.
Actor	Administrador.
Descripción	Se asignan jurados al TEG tomando en cuenta las opciones profesionales seleccionadas.
Pre-condición	- El usuario debe haber iniciado sesión como administrador. - El TEG debe haber sido clasificado y confirmadas el área o áreas a considerar.
Flujo Básico	- El sistema asigna de acuerdo a los criterios de elección preestablecidos, profesores de las opciones profesionales confirmadas para funcionar como jurados (principales y suplentes) en la presentación del TEG.
Post-condición	Pantalla con los datos del nuevo TEG que fue registrado en sistema, sus opciones profesionales asociadas y los jurados asignados. Además se muestra la opción de exportar un archivo con la información mostrada.
CASO DE USO 4.4.	
Nombre	Exportar Archivo.
Actor	Administrador.
Descripción	Se genera un archivo .txt con la información del TEG registrado en el sistema.
Pre-condición	- El usuario debe haber iniciado sesión como administrador. - Debe haberse completado la asignación de jurados al TEG.
Flujo Básico	- El usuario puede visualizar los datos del TEG registrado luego de haber realizado la asignación de jurados o en la sección de Consulta de Asignaciones (Luego de haber realizado la respectiva búsqueda). -En la pantalla de visualización de datos del TEG se hace click en el botón Exportar y dicha información es respaldada en un archivo .txt.
Post-condición	Archivo contenedor de los datos del TEG y su jurado asignado.
CASO DE USO 5.1.	
Nombre	Administrar Entes
Actor	Administrador.

Descripción	El administrador dispone de varias secciones para realizar todas las operaciones crear, obtener, actualizar y borrar (CRUD) sobre los entes que conforman el sistema (semestres, profesores, estudiantes y usuarios)
Pre-condición	El usuario debe haber iniciado sesión como administrador.
Flujo Básico	- El administrador se dirige a la respectiva sección de los entes que quiera configurar o modificar. - En cada sección, cada ente puede ser tratado con las operaciones CRUD, sin embargo, los Usuarios no pueden ser creados directamente en esta sección, ellos son creados cuando se realiza la asignación de jurados. - Al realizar las operaciones de Crear y Actualizar se verifica la validez y formato de los datos introducidos y se muestran mensajes de específicos en caso de error o éxito. - Al realizar operaciones de Borrar se muestra un mensaje de confirmación al usuario.
Post-condición	Entes creados, actualizados o borrados.
CASO DE USO 5.2.	
Nombre	Modificar Jurados.
Actor	Administrador.
Descripción	El administrador modifica los jurados asignados a un TEG.
Pre-condición	- El usuario debe haber iniciado sesión como administrador. - El TEG debe haber sido registrado en el sistema. - El TEG a modificar debe pertenecer al semestre que se encuentre en curso.
Flujo Básico	- El administrador ingresa a la sección de administración de Trabajos de Grado, allí dispondrá de cuatro listas desplegables donde podrá escoger o cambiar cada uno de los jurados asignados al TEG (principales y suplentes). - Si se han escogido todos los jurados se guardan los cambios en el sistema, sino se muestra el respectivo mensaje de error.
Post-condición	Jurados modificados para el TEG.
CASO DE USO 5.3.	
Nombre	Administrar Asignaciones.
Actor	Administrador.
Descripción	Modificación del número de asignaciones como jurado para un profesor.
Pre-condición	- El usuario debe haber iniciado sesión como administrador.
Flujo Básico	- El administrador ingresa a la sección de administración Máximo de Asignaciones, allí dispondrá de un campo en donde deberá introducir un número que representa la máxima cantidad de veces que un profesor puede ser asignado como jurado principal. - Se verifica que el valor proporcionado es un número natural, si lo es se guarda el máximo de asignaciones en el sistema, sino se muestra un mensaje de error.
Post-condición	Número máximo de asignaciones modificado.

CASO DE USO 5.4.	
Nombre	Cerrar Semestre.
Actor	Administrador.
Descripción	Reinicio de todas las asignaciones de profesores como jurados.
Pre-condición	- El usuario debe haber iniciado sesión como administrador.
Flujo Básico	- El administrador ingresa a la sección de administración Cerrar Semestre. Se muestra la información del semestre en curso y un botón para finalizar el mismo. - Al dar click en el botón se mostrará un mensaje alertando que se reiniciarán todas las asignaciones de jurados. Si es confirmada, el semestre se cierra y debe ser escogido otro semestre como actual en la sección Semestres.
Post-condición	Todas las asignaciones de profesores como jurados (principal y suplente) vuelven a ser cero (0).
CASO DE USO 5.5.	
Nombre	Ver Estadísticas
Actor	Administrador.
Descripción	Muestra un gráfico de comparación entre las opciones profesionales de los TEG seleccionadas por el administrador (por semestre) con respecto a las opciones profesionales de los TEG sugeridos por el modelo de clasificación (por semestre).
Pre-condición	- El usuario debe haber iniciado sesión como administrador.
Flujo Básico	- El administrador ingresa a la sección de administración Ver Estadísticas. Allí podrá observar las comparaciones explicadas en la descripción de este caso de uso y podrá realizar filtros de visualización por semestre haciendo click en su respectiva sección de la leyenda.
Post-condición	Ninguna.

D. CASOS DE USO – NIVEL 3

En este nivel se detallan las funcionalidades del caso de uso 5.1. Administrar Entes, las cuales se especifican en la Figura 2.8. Los casos de uso 5.1.1, 5.1.2 y 5.1.3 consisten en las operaciones CRUD del ente especificado, por ello no se darán mayores detalles sobre los mismos. En la Tabla 2.9 se detallan los casos de uso restantes.

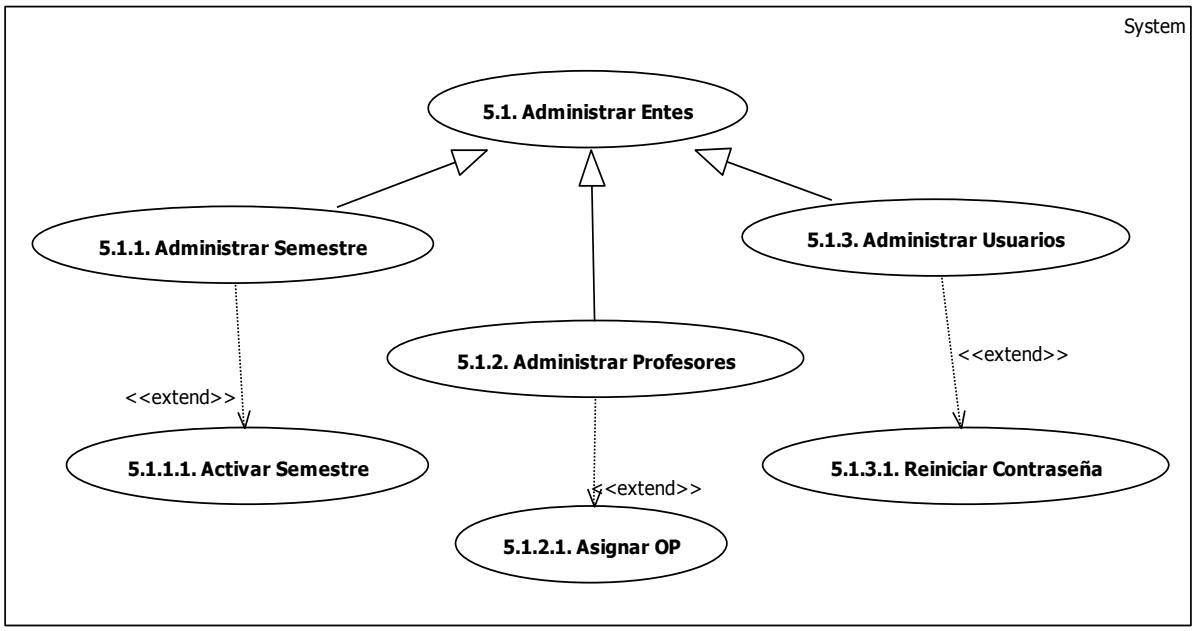

FIGURA 2.8. CASOS DE USO – NIVEL 3

TABLA 2.9. DESCRIPCIÓN CASOS DE USO – NIVEL 3

CASO DE USO 5.1.1.1.	
Nombre	Activar Semestre.
Actor	Administrador.
Descripción	Se coloca un nuevo semestre activo.
Pre-condición	- El usuario debe haber iniciado sesión como administrador. - No debe haber ningún semestre activo.
Flujo Básico	- El administrador ingresa a la sección de administración Semestres, allí se muestra una lista con los semestres registrados en el sistema. Se escoge alguno de dichos semestres y accede a su opción de edición, y se cambia el campo de Estatus a Activo. -Si se encuentra activo algún otro semestre al momento de realizar esta modificación, se muestra un mensaje de error indicando cuál es el semestre activo actual y explicando que debe cerrarse antes de poder activar otro.
Post-condición	Nuevo semestre activado.
CASO DE USO 5.1.2.1.	
Nombre	Asignar Opción Profesional.
Actor	Administrador.
Descripción	Se asigna a un profesor una o más opciones profesionales relacionadas con su especialidad académica.
Pre-condición	- El usuario debe haber iniciado sesión como administrador. - El profesor debe haber sido creado en el sistema. - Debe existir al menos una opción profesional creada en el sistema.

Flujo Básico	- El administrador ingresa a la sección de administración Profesores, allí se muestra la opción de visualizar las opciones profesionales de un profesor. Esto despliega una sub-lista con las áreas profesionales asociadas y la opción de hacer operaciones CRUD sobre las mismas. - El administrador agrega áreas al profesor seleccionándolas de una lista desplegable que contiene las opciones profesionales activas en el sistema. - Así mismo el administrador puede eliminar opciones profesionales asociadas a un profesor en esta sección.
Post-condición	Asignación o modificación de las áreas profesionales asociadas a un profesor.
CASO DE USO 5.1.3.1.	
Nombre	Reiniciar Contraseña
Actor	Administrador.
Descripción	Se reestablece la contraseña a un determinado usuario.
Pre-condición	- El usuario debe haber iniciado sesión como administrador. - El usuario debe haber sido creado en el sistema.
Flujo Básico	- El administrador ingresa a la sección de administración Usuarios, allí se muestra la opción de reiniciar la contraseña de un determinado usuario. Si se selecciona dicha opción se mostrará un mensaje de confirmación y en caso de aceptar se muestra por pantalla la nueva contraseña generada por el sistema. - El administrador deberá contactar con el usuario para hacerle llegar la nueva contraseña por el medio que considere pertinente (correo electrónico, teléfono, etc.) -El usuario con la nueva contraseña deberá ingresar en el sistema y luego en la sección de Cambiar Datos podrá establecer una contraseña más acorde a sus preferencias.
Post-condición	Usuario con contraseña reiniciada.

2.6.5. INTERFAZ DE USUARIO

A continuación las Figuras 2.9., 2.10., 2.11., 2.12. y 2.13. presentan las pantallas de la interfaz de usuario, correspondientes a las funcionalidades de asignar jurados y consultar asignaciones.

A. ASIGNAR JURADOS

En ésta sección la primera pantalla que visualizará el usuario será la que le permitirá introducir los datos correspondientes al TEG para realizar el proceso de clasificación y posterior asignación de jurados (Figura 2.9.). La vista está dividida en dos secciones, datos del documento (parte izquierda de la vista) y datos de los autores (a la derecha), asimismo tendrá la opción de reiniciar el formulario o continuar con el proceso.

FIGURA 2.9. ASIGNACIÓN DE JURADOS– PASO 1 (INTRODUCCIÓN DE DATOS RELACIONADOS AL TEG)

Luego que el usuario haya introducido los datos y presione el botón de siguiente el sistema se encargará de realizar la clasificación del documento y le mostrará al usuario los resultados obtenidos (Figura 2.10.). En esta interfaz el usuario podrá revisar los resultados obtenidos y modificarlos de ser necesario, también podrá cancelar el proceso de asignación de jurados si así lo considera necesario. La asignación de jurados se realizará cuando el usuario presione el botón de confirmación.

FIGURA 2.10. ASIGNACIÓN DE JURADOS – PASO 2 (CONFIRMACIÓN DE ÁREAS)

Luego de que el usuario confirme que desea proceder con la asignación de los jurados el sistema le mostrará de manera detallada los resultados obtenidos (Figura 2.11.). Estos resultados se dividen en varias secciones:

- Datos del Trabajo de Grado
- Opciones profesionales (áreas) asignadas al TEG
- Datos de los autores del TEG
- Datos de los tutores del TEG
- Datos de los jurados asignados por el sistema al TEG

Se ofrece al usuario la posibilidad de exportar estos datos a un documento de texto plano o imprimirlos. Culminado el proceso, el usuario puede salir de esta pantalla haciendo click en el botón Volver para ir a la página de inicio o utilizar el menú para acceder a otra sección del sistema.

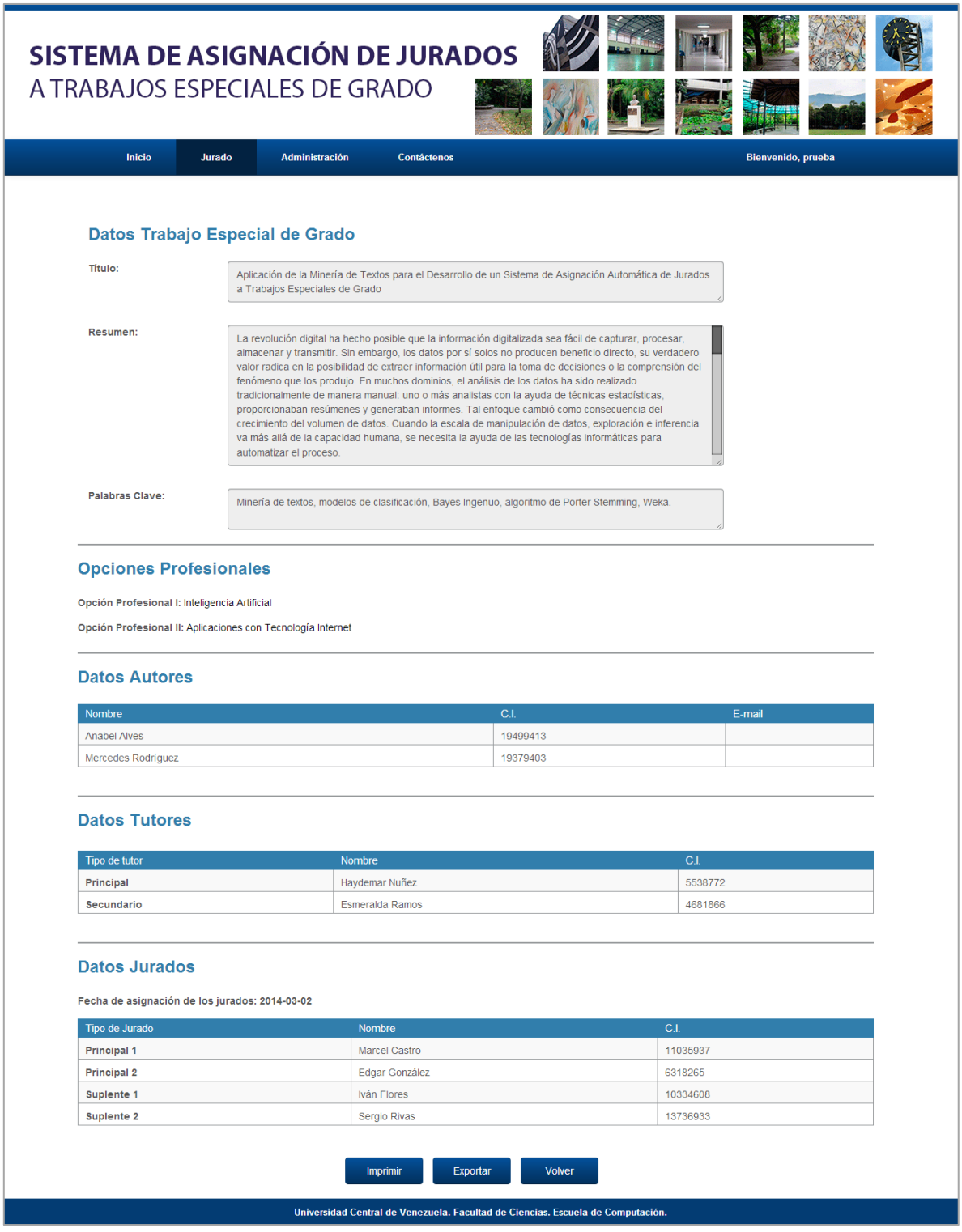

FIGURA 2.11. ASIGNACIÓN DE JURADOS – PASO 3 (INFORMACIÓN FINAL)

B. CONSULTAR JURADOS

En la primera pantalla de la sección de consulta se podrán introducir los datos en los cuales se basará la búsqueda de TEG en el sistema (Figura 2.12.). Los datos por los cuales se puede realizar el filtrado son:

- Semestre
- Profesor
- Tipo de profesor (jurado o tutor)
- Tipo de tutor (principal o secundario)
- Tipo de jurado (principal o suplente)
- Estudiante

Todos los campos son listas desplegables desde las cuales el usuario podrá elegir valores que son válidos dentro del sistema.

FIGURA 2.12. CONSULTA DE ASIGNACIONES – PASO 1 (APLICACIÓN DE FILTROS)

Después de introducir los datos y presionar el botón Consultar el usuario visualizará todos los TEG que cumplan con los parámetros introducidos (Figura 2.13.). En el listado el usuario visualizará el id, título y semestre del TEG, sin embargo, tiene a su disposición un botón que le permite ver la información detallada del TEG con el mismo formato mostrado en la Figura 2.11.

FIGURA 2.13. CONSULTA DE ASIGNACIONES – PASO 2 (RESULTADOS)

2.7. PRUEBAS Y RESULTADOS

Los dos módulos principales del Sistema de Asignación de Jurados a TEG son el módulo de asignación de OP (que contiene el módulo de clasificación) y el módulo de asignación de jurados, por ello las pruebas realizadas se centran en el cálculo del error de clasificación presentado por el primero, y en verificar que las asignaciones de jurados realizadas por el segundo se apeguen a las reglas establecidas (ver sección 2.6.3.C. módulo de asignación de jurados).

Para evaluar el rendimiento que ofrece el módulo de asignación de OP se utilizó un conjunto de 24 documentos de TEGs que no fueron usados durante la estimación del modelo de clasificación; Los TEG en cuestión, pertenecen a sólo 9 de las 11 OP ofertadas por la Escuela de Computación (no fue posible encontrar documentos pertenecientes a las áreas de Modelos y Programación Matemática y Cálculo Científico).

La distribución de los documentos por OP se puede observar en la Tabla 2.10.

Los resultados de la evaluación del módulo de asignación de OP se describen en la Tabla 2.11, en ella se comparan las OP asociadas a los documentos de TEG con las dos clases recomendadas por el sistema. Se consideran correctas las clasificaciones que colocan la OP real del TEG en alguna de las OP recomendadas.

TABLA 2.10. DISTRIBUCIÓN DE LOS DOCUMENTOS DEL CONJUNTO DE PRUEBA POR OP

Opción Profesional	Cantidad de Documentos
Aplicaciones con Tecnología Internet	5
Base de Datos	2
Computación Gráfica	5
Ingeniería de Software e Interacción Humano-Computador	1
Inteligencia Artificial	1
Sistemas de Información	3
Sistemas Distribuidos y Paralelos	2
Tecnologías Educativas	2
Tecnologías en Comunicaciones y Redes de Computadoras	3
	Total = 24

TABLA 2.11. RESULTADOS DE LAS PRUEBAS REALIZADAS AL SISTEMA

OP TEG	N°	OP Recomendada 1	OP Recomendada 2	Clasificación
Aplicaciones con Tecnología Internet	1	Aplicaciones con Tecnología Internet	Base de Datos	Correcta
	2	Aplicaciones con Tecnología Internet	Base de Datos	Correcta
	3	Aplicaciones con Tecnología Internet	Ingeniería de Software e Interacción Humano-Computador	Correcta
	4	Aplicaciones con Tecnología Internet	Sistemas de Información	Correcta
	5	Sistemas de Información	Base de Datos	Incorrecta
Base de Datos	1	Aplicaciones con Tecnología Internet	Ingeniería de Software e Interacción Humano-Computador	Incorrecta
	2	Inteligencia Artificial	Base de Datos	Correcta
Computación Gráfica	1	Sistemas Distribuidos y Paralelos	Computación Gráfica	Correcta
	2	Computación Gráfica	Ingeniería de Software e Interacción Humano-Computador	Correcta
	3	Computación Gráfica	-	Correcta

	4	Computación Gráfica	-	Correcta
	5	Computación Gráfica	Ingeniería de Software e Interacción Humano-Computador	Correcta
Ingeniería de Software e Interacción Humano-Computador	1	Ingeniería de Software e Interacción Humano-Computador	-	Correcta
Inteligencia Artificial	1	Inteligencia Artificial	Tecnologías en Comunicaciones y Redes de Computadoras	Correcta
Sistemas de Información	1	Tecnologías en Comunicaciones y Redes de Computadoras	Sistemas de Información	Correcta
	2	Sistemas de Información	Tecnologías en Comunicaciones y Redes de Computadoras	Correcta
	3	Base de Datos	Sistemas de Información	Correcta
Sistemas Distribuidos y Paralelos	1	Sistemas Distribuidos y Paralelos	Computación Gráfica	Correcta
	2	Sistemas Distribuidos y Paralelos	Computación Gráfica	Correcta
Tecnologías Educativas	1	Tecnologías Educativas	-	Correcta
	2	Tecnologías en Comunicaciones y Redes de Computadoras	Base de Datos	Incorrecta
Tecnologías en Comunicaciones y Redes de Computadoras	1	Tecnologías en Comunicaciones y Redes de Computadoras	-	Correcta
	2	Tecnologías en Comunicaciones y Redes de Computadoras	-	Correcta
	3	Tecnologías en Comunicaciones y Redes de Computadoras	Computación Gráfica	Correcta

Para evaluar los resultados se midió el rendimiento del sistema desarrollado utilizando una y dos clases (Tabla 2.12.). El Sistema de Asignación de Jurados a TEG alcanzó un porcentaje de documentos clasificados correctamente del 87,5% con la utilización de dos clases; con esto se supera el rendimiento alcanzado por el modelo estimado en éste trabajo (82,0513% con una clase) y el alcanzado por el modelo estimado en (Torres Navarro, 2011) (82,32%). También se evidencia

que las OP de Aplicaciones con Tecnología Internet y Base de tienden a ser confundidas por el clasificador, ya que son áreas relacionadas y complementarias entre sí.

TABLA 2.12. COMPARACIÓN DE LOS RESULTADOS OBTENIDOS AL ESTIMAR EL MODELO Y LOS ALCANZADOS POR EL SISTEMA

	Modelo de clasificación estimado		Sistema de Asignación de Jurados a TEG (Una OP)		Sistema de Asignación de Jurados a TEG (Dos OP)	
Clasificaciones Correctas	288	82,0513%	17	70,8%	21	**87,5%**
Clasificaciones Incorrectas	63	17,9487%	7	29,2%	3	12,5%
N° de Instancias	351		24			

A continuación se presentan dos de las pruebas realizadas al módulo de asignación de jurados, las cuales consistieron en reiniciar la cantidad de asignaciones de los profesores de cada área y realizar el proceso de asignación de jurados a cada documento del conjunto de prueba. Los resultados completos de las pruebas al módulo de asignación de jurados se presentan en el Anexo 5. Los resultados mostrados en la Tablas 2.13 y 2.14 corresponden al documento de prueba n° 3 de la OP Sistemas de Información. Los resultados de las Tablas 2.15 y 2.16 corresponden al documento de prueba n° 4 de la OP Computación Gráfica.

TABLA 2.13. ESTADO DE LOS PROFESORES DE LAS OP RECOMENDADAS PARA EL TEG DE PRUEBA N° 3 DE SISTEMAS DE INFORMACIÓN

PRUEBA DOCUMENTO 3						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Sistemas de Información	Base de Datos, Sistemas de Información	Francisco Castillo	Paola Saputelli	Pedro Bonillo	Rossana Díaz	Concettina Di Vasta

TABLA 2.14. JURADOS ASIGNADOS AL TEG DE PRUEBA N° 3 DE SISTEMAS DE INFORMACIÓN

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	**0**	**Paola Saputelli**	**Base de Datos**
1	**1**	**Rossana Díaz**	
2	0	Antonio Silva	
1	2	Concettina Di Vasta	
1	3	Mercy Ospina	
1	1	Alejandro Durán	
1	**0**	**Pedro Bonillo**	**Sistemas de Información**
2	0	Antonio Silva	
1	**2**	**Concettina Di Vasta**	
1	2	Wilfredo Rangel	

TABLA 2.15. ESTADO DE LOS PROFESORES DE LAS OP RECOMENDADAS PARA EL TEG DE PRUEBA N° 4 DE COMPUTACIÓN GRÁFICA

PRUEBA DOCUMENTO 4						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Computación Gráfica	Computación Gráfica	Ernesto Coto	Esmitt Ramírez	Omaira Rodríguez	Rhadames Carmona	Héctor Navarro

TABLA 2.16. JURADOS ASIGNADOS AL TEG DE PRUEBA N° 4 DE COMPUTACIÓN GRÁFICA

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	**1**	**Omaira Rodríguez**	**Computación Gráfica**
1	3	Robinson Rivas	
1	**1**	**Rhadames Carmona**	
1	**2**	**Héctor Navarro**	
2	0	Walter Hernández	
1	**0**	**Esmitt Ramírez**	

Haciendo un seguimiento de la cantidad de asignaciones de los docentes, tanto como jurados principales como suplentes, se corroboró que los jurados seleccionados son los que corresponden según las reglas de elección establecidas (Ver Anexo 5).

CONCLUSIONES Y RECOMENDACIONES

A partir del sistema realizado por (Torres Navarro, 2011), se desarrolló una nueva aplicación que permite la asignación automática de jurados a TEG con base en un clasificador bayesiano ingenuo, que recomienda un máximo de dos de las once opciones profesionales ofertadas por la Escuela de Computación de la UCV. El clasificador fue construido aplicando el proceso de minería de textos lográndose el objetivo general propuesto en este Trabajo Especial de Grado.

El porcentaje de instancias clasificadas correctamente por el sistema con la recomendación de dos OP relacionadas es de 87,5%, mayor al obtenido durante la estimación del modelo con Weka dónde se tomaba en cuenta una sola clase (82,0513%) y también superior al alcanzado en el modelo estimado por (Torres Navarro, 2011) (82,32%); con ello, se concluye que en este contexto, resulta más efectiva la consideración de más de un área para clasificar automáticamente un TEG de la escuela de computación de la UCV, puesto que en muchos casos los documentos tienen relación con más de un área de conocimiento.

La principal ventaja obtenida con el desarrollo de este sistema reside en que la elección de los profesores como jurados pasará de ser un proceso subjetivo y discrecional a ser un proceso automatizado y objetivo, que garantiza la escogencia de estos de acuerdo a reglas establecidos que buscan garantizar una elección justa con igualdad de condiciones. Debido a que el sistema desarrollado también lleva un histórico de las asignaciones de jurados a TEG, su uso facilitará la recolección futura de información en digital relacionada con los TEG de las diferentes áreas. Igualmente, la forma en la que se desarrolló la arquitectura de este sistema permite la sustitución del modelo de clasificación, vista minable y propuesta de pesado sin afectar el funcionamiento del resto de los componentes, esto con el objetivo de aportar extensibilidad si en algún momento el modelo bayesiano ingenuo quedara obsoleto debido a cambios en la población.

Uno de los inconvenientes encontrados durante la realización de este trabajo fue la distribución desigual de los documentos entre áreas profesionales, lo que causaba que el modelo sesgara hacia las OP más probables. Igualmente, existen OP que se encuentran muy relacionadas entre sí; a pesar de que el sistema recomienda dos OP, se presentan casos en los que el área del documento no está bien delimitada y el clasificador puede llegar a fallar. Para sopesar ambos inconvenientes, se recomienda para futuros trabajos utilizar esquemas que permitan trabajar con conjuntos de datos no balanceados, así como aplicar la lógica difusa en el módulo de clasificación del sistema para manejar la Interdisciplinariedad presente en los documentos. Asimismo, se recomienda la incorporación de una funcionalidad que permita navegar por los directorios del sistema operativo para seleccionar el documento TEG que vaya a ser clasificado, esto evitaría la tarea por parte del usuario de transcribir resúmenes de TEG muy extensos.

REFERENCIAS

Aas, K., & Eikvil, L. (1999). Text Categorisation: A Survey. *Reporte 941*. Oslo, Norruega: Norsk Regnesentral.

Española, R. A. (Mayo de 2013). *Corpus de Referencia del Español Actual (CREA) - Listado de frecuencias.* Obtenido de http://corpus.rae.es/lfrecuencias.html

Figuerola, C., Zazo, A., & Alonso, J. (2000). Categorización automática de documentos en español: algunos resultados experimentales. *Primeras Jornadas de Bibliotecas Digitales. JBIDI'2000. 6 y 7 de Noviembre. Valladolid (España)* (págs. 149-159). Valladolid, España: Universidad de Valladolid.

García, C., & Gómez, I. (2009). *Algoritmos De Aprendizaje: Knn & Kmeans.* Obtenido de Inteligencia en Redes de Comunicaciones - Universidad Carlos III de Madrid: http://www.it.uc3m.es/jvillena/irc/practicas/08-09/06.pdf

Hernández, J., Ramírez, M., & Ferri, C. (2005). *Introducción a la Minería de Datos.* Madrid, España: Pearson.

Hotho, A., Nürnberger, A., & Paaß, G. (2003). A Brief Survey of Text Mining. *(20), 1*, 19-62.

Joachims , T. (2002). *Learning to Classify Text Using Support Vector Machines - Methods, Theory and Algorithms.* Kluwer Academic Publishers.

John, G. H., & Langley, P. (1995). Estimating Continuous Distributions in Bayesian Classifiers. *Proceedings of the Eleventh Conference on Uncertanty in Artificial Intelligence* (pág. 2). San Mateo: Morgan Kaufmann Publishers.

Kohonen, T. (1988). An introduction to neural computing. *Neural Networks, 1(1)*, 3-16.

Lovins, J. B. (Marzo y Junio de 1968). Mechanical Translation and Computational Linguistics. *Development of a Stemming Algorithm, 11(1 y 2)*.

Paice, C. (1990). Another Stemmer. *ACM SIGIR Forum*, *24*, págs. 56-61. New York.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14(3)*, 130-137.

Rodríguez, J. E., Rojas Blanco, E. A., & Franco Camacho, R. O. (2007). Clasificación de datos usando el método k-nn. *Vínculos, 4*(1).

Sebastiani, F. (2001). Machine Learning in Automated Text Categorization. Roma, Italia: Consiglio Nazionale delle Ricerche.

Torres Navarro, D. R. (2011). *Minería de Textos para la Asignación Automática de Jurados a Trabajos Especiales de Grado.* Caracas, Venezuela: Universidad Centra de Venezuela. Centro de Ingeniería de Software y Sistemas (ISYS).

Venegas, R. (2007). Clasificación de textos académicos en función de su contenido léxico-semántico. *Revista Signos, 40*(63), 239-271.

Witten, I. H., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations.* San Diego, Estados Unidos: Elsevier.

Zhang, H. (2004). *The Optimality of Naive Bayes.* Fredericton, Canada: University of New Brunswick. Faculty of Computer Science.

Zhang, M.-L., & Zhou, Z.-H. (2006). *Multi-Label Neural Networks with Applications to Functional Genomics and Text Categorization.* Nanjing, China: IEEE Transactions on Knowledge and Data Engineering.

ANEXOS

Anexo 1. Descripción del formato de archivos ".arff".

Nativamente Weka trabaja con un formato denominado arff, acrónimo de Attribute-Relation File Format. Este formato está compuesto por una estructura claramente diferenciada en tres partes:

1. **Cabecera**. Se define el nombre de la relación. Su formato es el siguiente:

   ```
   @relation <nombre-de-la-relación>
   ```

   Donde <nombre-de-la-relación> es de tipo String[6]. Si dicho nombre contiene algún espacio será necesario expresarlo entrecomillado.

2. **Declaraciones de atributos**. En esta sección se declaran los atributos que compondrán nuestro archivo junto a su tipo. La sintaxis es la siguiente:

   ```
   @attribute <nombre-del-atributo> <tipo>
   ```

   Donde <nombre-del-atributo> es de tipo String teniendo las mismas restricciones que el caso anterior. Weka acepta diversos tipos, estos son:

   a) **NUMERIC** Expresa números reales[7].

   b) **INTEGER** Expresa números enteros.

   c) **DATE** Expresa fechas, para ello este tipo debe ir precedido de una etiqueta de formato entrecomillada. La etiqueta de formato está compuesta por caracteres separadores (guiones y/o espacios) y unidades de tiempo:

      - dd Día.
      - MM Mes.
      - yyyy Año.
      - HH Horas.
      - mm Minutos.
      - ss Segundos.

   d) **STRING** Expresa cadenas de texto, con las restricciones del tipo String comentadas anteriormente.

[6] Entendiendo como tipo String el ofrecido por Java.

[7] Debido a que Weka es un programa Anglosajón, la separación de la parte decimal y entera de los números reales se realiza mediante un punto en vez de una coma.

e) **ENUMERADO** El identificador de este tipo consiste en expresar entre llaves y separados por comas los posibles valores (caracteres o cadenas de caracteres) que puede tomar el atributo. Por ejemplo, si tenemos un atributo que indica el tiempo podría definirse:

```
@attribute tiempo {soleado,lluvioso,nublado}
```

3. **Sección de datos**. Declaramos los datos que componen la relación separando entre comas los atributos y con saltos de línea las relaciones.

```
@data
4,3.2
```

Aunque éste es el modo “completo" es posible definir los datos de una forma abreviada (*sparse data*). Si tenemos una muestra en la que hay muchos datos que sean 0 podemos expresar los datos prescindiendo de los elementos que son nulos, rodeando cada una de las filas entre llaves y situando delante de cada uno de los datos el número de atributo[8].

Un ejemplo de esto es el siguiente

```
@data
{1 4, 3 3}
```

En este caso hemos prescindido de los atributos 0 y 2 (como mínimo) y asignamos al atributo 1 el valor 4 y al atributo 3 el valor 3.

En el caso de que algún dato sea desconocido se expresará con un símbolo de cerrar interrogación (“?"). Es posible añadir comentarios con el símbolo “%”, que indicará que desde ese símbolo hasta el final de la línea es todo un comentario. Los comentarios pueden situarse en cualquier lugar del fichero.

Un ejemplo de un archivo de weka.

prueba.arff

```
1 % Archivo de prueba para Weka.
2 @relation prueba
3
4 @attribute nombre STRING
5 @attribute ojo_izquierdo {Bien,Mal}
6 @attribute dimension NUMERIC
7 @attribute fecha_analisis DATE "dd-MM-yyyy HH:mm"
8
9 @data
10 Antonio,Bien,38.43,"12-04-2003 12:23"
11 Manuel,?,34.53,"14-05-2003 13:45"
12 Juan,Bien,43,"01-01-2004 08:04"
13 Maria,?,?,"03-04-2003 11:03"
```

[8] La numeración de atributos comienza desde el 0.

Anexo 2. Resultados de las pruebas realizadas a diferentes algoritmos de clasificación sobre las cuatro matrices de indexación mencionadas en este trabajo.

Pesado	Algoritmos	Clasificaciones	Cantidad	Porcentaje
TFxIDF	K-NN (K=3)	Correctas	218	62,1083%
		Incorrectas	133	37,8917%
	K-NN (K=5)	Correctas	219	62,3932%
		Incorrectas	132	37,6068%
	K-NN (K=7)	Correctas	224	63,8177%
		Incorrectas	127	36,1823%
	C4.5	Correctas	211	60,114%
		Incorrectas	140	39,886%
	Bayes Ingenuo	Correctas	262	74,6439%
		Incorrectas	89	25,3561%
TFC	K-NN (K=3)	Correctas	206	58,6895%
		Incorrectas	145	41,3105%
	K-NN (K=5)	Correctas	216	61,5385%
		Incorrectas	135	38,4615%
	K-NN (K=7)	Correctas	217	61,8234%
		Incorrectas	134	38,1766%
	C4.5	Correctas	206	58,6895%
		Incorrectas	145	41,3105%
	Bayes Ingenuo	Correctas	275	78,3476%
		Incorrectas	76	21,6524%
LTC	K-NN (K=3)	Correctas	228	64,9573%
		Incorrectas	123	35,0427%
	K-NN (K=5)	Correctas	225	64,1026%
		Incorrectas	126	35,8974%
	K-NN (K=7)	Correctas	226	64,3875%
		Incorrectas	125	35,6125%
	C4.5	Correctas	210	60,114%
		Incorrectas	140	39,886%
	Bayes Ingenuo	Correctas	274	78,0627%
		Incorrectas	77	21,9373%
Entropía	K-NN (K=3)	Correctas	217	61,8234%
		Incorrectas	134	38,1766%
	K-NN (K=5)	Correctas	219	62,3932%
		Incorrectas	132	37,6068%
	K-NN (K=7)	Correctas	239	68,0912%
		Incorrectas	112	31,9088%
	C4.5	Correctas	210	59,8291%
		Incorrectas	141	40.1709 %
	Bayes Ingenuo	Correctas	288	**82,0513%**
		Incorrectas	63	17,9487%

Anexo 3. Esquema de votación para la propuesta de pesado utilizado en este trabajo, Entropía.

INFO GAIN	CHI-CUADRADO	GAIN RATIO	Ranking
inalambr	Enseñ	matriz	1
enseñ	Inalambr	matric	2
inform	3d	inalambr	3
red	Matriz	3d	4
metod	Matric	enseñ	5
comput	Metod	volumetr	6
imagen	Aprendizaj	clasificacion	7
inteligent	Educ	pmbok	8
implement	Volumen	textur	9
aprendizaj	Red	sem	10
virtual	Inform	volumen	11
3d	Inteligent	project	12
simul	Imagen	tridimensional	13
educ	Clasificación	neuronal	14
gestion	Volumetr	escen	15
informacion	Simul	metod	16
protocol	Paralel	iee	17
volumen	Neuronal	agent	18
administr	Virtual	siti	19
empres	Volum	artificial	20
clasificacion	Protocol	body	21
capitul	Comput	bluetooth	22
human	Agent	kimball	23
volum	Escen	volum	24
agent	Capitul	realid	25
apoy	Sem	conest	26
paralel	Implement	ruby	27
disen	Gestión	educ	28
siti	Pmbok	simul	29
volumetr	tridimensional	aprendizaj	30
trav	Siti	paralel	31
movil	Textur	geometr	32
expert	Profesional	knowledg	33
ruby	Móvil	virtual	34
sem	Project	gerenci	35
profesional	Ruby	autovalor	36
neuronal	Iee	wlan	37
artificial	Artificial	precision	38

escen	Didact	diferencial	39
negoci	realid	inteligent	40
segur	negoci	movil	41
tridimensional	diferencial	profesional	42
iee	expert	discut	43
implementacion	informacion	interconexion	44
realid	autovalor	imagen	45
textur	body	protocol	46
bluetooth	bluetooth	capitul	47
report	human	didact	48
matriz	empres	report	49
matric	precision	trav	50
project	gerenci	expert	51
pmbok	multimedi	desplieg	52
didact	apoy	rails	53
conest	desplieg	red	54
desplieg	conest	of	55
tecnologi	knowledg	inform	56
rails	kimball	gestion	57
grafic	administr	gui	58
multimedi	trav	informacion	59
servici	geometr	seminal	60
of	segur	transaccional	61
imag	imag	negoci	62
gerenci	of	multimedi	63
geometr	report	disen	64
body	disen	implementacion	65
d	grafic	human	66
kimball	reutiliz	empres	67
knowledg	gui	segur	68
interconexion	rails	comput	69
xp	implementacion	problem	70
facult	wlan	administr	71
local	tecnologi	implement	72
wlan	visualiz	reutiliz	73
problem	numer	ipv6	74
visualiz	interconexion	superfici	75
diferencial	problem	identificacion	76
metodolog	xp	apoy	77
automatiz	local	local	78
discut	servici	grafic	79

numer	d	visualiz	80
reutiliz	Discut	xp	81
comunicacion	Automatiz	management	82
gui	Transaccional	consult	83
cre	Seminal	imag	84
grad	Cre	numer	85
model	Comunicación	model	86
precision	Oracl	tecnologi	87
oracl	Cas	d	88
product	Metodolog	oracl	89
autovalor	Superfici	comunicacion	90
ipv6	ipv6	cre	91
superfici	Codig	automatiz	92
seminal	Product	grad	93
cas	Model	cas	94
codig	Clust	codig	95
transaccional	Grad	electron	96
distanci	Electron	product	97
identificacion	Identificación	facult	98
clust	Management	clust	99
consult	Distanci	servici	100
electron	Facult	iter	101
region	Consult	metodolog	102
hardwar	Región	regl	103
regl	Hardwar	deriv	104
algoritm	Regl	distanci	105
management	Deriv	empresarial	106
poder	Empresarial	region	107
empresarial	Poder	poder	108
iter	Resolu	hardwar	109
resolu	Algoritm	algoritm	110
deriv	Iter	apendic	111
apendic	Apendic	resolu	112
configur	Implant	control	113
interact	Describ	implant	114
describ	Configur	describ	115
implant	Control	compar	116
control	Interact	configur	117
10	Patrón	clasif	118
patron	Comprar	interact	119
migracion	Clasif	migracion	120

compar	Pas	10	121
clasif	prototip	pas	122
pas	jav	prototip	123
prototip	metric	jav	124
jav		metric	125
metric			126

Anexo 4. Reglas de elección de profesores como jurados de TEG.

```
cant_jurados_principales_asignados_a_TEG = 2;
cant_jurados_suplentes_asignados_a_TEG = 2;
cant_jurados_asignados_a_TEG = cant_jurados_principales_asignados_a_TEG +
cant_jurados_suplentes_asignados_a_TEG;
jurados_asignados_a_TEG = {principal1, principal2, suplente1, suplente2};

Si ( (cant_areas_TEG == 1) AND (area1_TEG == a1) )
ENTONCES
        principal1_area = a1;
        principal2_area = a1;
        suplente1_area = a1;
        suplente2_area = a1;
FSI

Si ( (cant_areas_TEG == 2) AND (area1_TEG == a1) AND (area2_TEG == a2) )
ENTONCES
        principal1_area = a1;
        principal2_area = a2;
        suplente1_area = a1;
        suplente2_area = a2;
FSI

SI (tipo_jurado_a_elegir == “principal1”)
ENTONCES
        SI ( (profesor_ci != tutor1_ci ) AND (profesor_ci != tutor2_ci ) AND (profesor_area ==
        principal1_area) AND (profesor_disponible==“si”) AND (profesor_cant_asig_semestral <
        max_asig_semestral ) AND profesor_cant_asig_princ <=
        menor_cant_asig_princ_profs_de_principal1_area )
        ENTONCES
                SI (cant_profs_cumplen_condicion > 1)
                ENTONCES
                    SI (profesor_cant_asig_sup <= menor_cant_asig_sup_profs_de_principal1_area )
                    ENTONCES
                        SI (cant_profs_cumplen_condicion > 1)
                        ENTONCES
                            principal1 = selección_aleatoria_profesor();
                        SINO
                            principal1 = profesor;
                        FSI
                    SINO
                        principal1 = profesor;
                    FSI
                FSI
        FSI
FSI
```

```
SI (tipo_jurado_a_elegir == “principal2”)
ENTONCES
    SI ( (profesor_ci != tutor1_ci ) AND (profesor_ci != tutor2_ci ) AND (profesor_area ==
    principal2_area) AND (profesor_disponible==“si”) AND (profesor_cant_asig_semestral <
    max_asig_semestral ) AND profesor_cant_asig_princ <=
    menor_cant_asig_princ_profs_de_principal1_area )
    ENTONCES
        SI (cant_profs_cumplen_condicion > 1)
        ENTONCES
            SI (profesor_cant_asig_sup <= menor_cant_asig_sup_profs_de_principal2_area )
            ENTONCES
                SI (cant_profs_cumplen_condicion > 1)
                ENTONCES
                    principal2 = selección_aleatoria_profesor();
                SINO
                    principal2 = profesor;
                FSI
            SINO
                principal2 = profesor;
            FSI
        FSI
    FSI
FSI

SI (tipo_jurado_a_elegir == “suplente1”)
ENTONCES
    SI ( (profesor_ci != tutor1_ci ) AND (profesor_ci != tutor2_ci ) AND (profesor_area ==
    suplente1_area) AND (profesor_disponible==“si”) AND profesor_cant_asig_princ <=
    menor_cant_asig_princ_profs_de_suplente1_area )
    ENTONCES
        SI (cant_profs_cumplen_condicion > 1)
        ENTONCES
            SI (profesor_cant_asig_sup <= menor_cant_asig_sup_profs_de_suplente1_area )
            ENTONCES
                SI (cant_profs_cumplen_condicion > 1)
                ENTONCES
                    suplente1 = selección_aleatoria_profesor();
                SINO
                    suplente1 = profesor;
                FSI
            SINO
                suplente1 = profesor;
            FSI
        FSI
    FSI
FSI
```

```
SI (tipo_jurado_a_elegir == “suplente2”)
ENTONCES
    SI ( (profesor_ci != tutor1_ci ) AND (profesor_ci != tutor2_ci ) AND (profesor_area ==
    suplente2_area) AND (profesor_disponible==“si”) AND profesor_cant_asig_princ <=
    menor_cant_asig_princ_profs_de_suplente2_area )
    ENTONCES
        SI (cant_profs_cumplen_condicion > 1)
        ENTONCES
            SI (profesor_cant_asig_sup <= menor_cant_asig_sup_profs_de_suplente2_area )
            ENTONCES
                SI (cant_profs_cumplen_condicion > 1)
                ENTONCES
                    suplente2 = selección_aleatoria_profesor();
                SINO
                    suplente2 = profesor;
                FSI
            SINO
                suplente2 = profesor;
            FSI
        FSI
    FSI
FSI
```

Anexo 5. Resultados de las pruebas realizadas al módulo de asignación de jurados separados por OP.

Jurado Principal
Jurado Suplente

APLICACIONES CON TECNOLOGÍA INTERNET

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Aplicaciones con Tecnología Internet	Aplicaciones con Tecnología Internet y Base de Datos	Jossie Zambrano	Sergio Rivas	Concettina Di Vasta	Antonio Leal	Rossana Díaz

Asignaciones como Principal	**Asignaciones como Suplente**	**Profesor**	**Área**
0	0	Edgar González	**Aplicaciones con Tecnología Internet**
0	**0**	**Antonio Leal**	
0	**0**	**Sergio Rivas**	
0	0	Jossie Zambrano	
0	0	Eugenio Scalise	
0	0	Paola Saputelli	**Base de Datos**
0	**0**	**Rossana Díaz**	
0	0	Antonio Silva	
0	**0**	**Concettina Di Vasta**	
0	0	Mercy Ospina	
0	0	Alejandro Durán	

PRUEBA DOCUMENTO 2						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Aplicaciones con Tecnología Internet	Aplicaciones con Tecnología Internet y Base de Datos	Jossie Zambrano	Edgar González	Alejandro Durán	Eugenio Scalise	Mercy Ospina

Asignaciones como Principal	**Asignaciones como Suplente**	**Profesor**	**Área**
0	**0**	**Edgar González**	**Aplicaciones con Tecnología**
0	1	Antonio Leal	
1	0	Sergio Rivas	

0	0	Jossie Zambrano	**Internet**
0	**0**	**Eugenio Scalise**	
0	0	Paola Saputelli	**Base de Datos**
0	1	Rossana Díaz	
0	0	Antonio Silva	
1	0	Concettina Di Vasta	
0	**0**	**Mercy Ospina**	
0	**0**	**Alejandro Durán**	

PRUEBA DOCUMENTO 3						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Aplicaciones con Tecnología Internet	Aplicaciones con Tecnología Internet e Ingeniería de Software e Interacción Humano-Computador	Antonio Silva	Jossie Zambrano	Iván Flores	Antonio Leal	Andrés Sanoja

Asignaciones como Principal	**Asignaciones como Suplente**	**Profesor**	**Área**
1	0	Edgar González	**Aplicaciones con Tecnología Internet**
0	**1**	**Antonio Leal**	
1	0	Sergio Rivas	
0	**0**	**Jossie Zambrano**	
0	1	Eugenio Scalise	
0	0	Alfredo Matteo	**Ingeniería de Software e Interacción Humano-Computador**
0	0	Nora Montaño	
0	0	Norelva Niño	
0	1	Eugenio Scalise	
0	**0**	**Iván Flores**	
0	**0**	**Andrés Sanoja**	
0	0	Jossie Zambrano	

PRUEBA DOCUMENTO 4						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Aplicaciones con Tecnología Internet	Aplicaciones con Tecnología Internet y Sistemas de Información	Andrés Castro	Eugenio Scalise	Antonio Silva	Antonio Leal	Wilfredo Rangel

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	0	Edgar González	**Aplicaciones con Tecnología Internet**
0	**2**	**Antonio Leal**	
1	0	Sergio Rivas	
1	0	Jossie Zambrano	
0	**1**	**Eugenio Scalise**	
0	0	Pedro Bonillo	**Sistemas de Información**
0	**0**	**Antonio Silva**	
1	0	Concettina Di Vasta	
0	**0**	**Wilfredo Rangel**	

PRUEBA DOCUMENTO 5						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Aplicaciones con Tecnología Internet	Sistemas de Información, Base de Datos	Jossie Zambrano, Sergio Rivas	Pedro Bonillo	Mercy Ospina	Wilfredo Rangel	Alejandro Durán

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
0	**0**	**Pedro Bonillo**	**Sistemas de Información**
1	0	Antonio Silva	
1	0	Concettina Di Vasta	
0	**1**	**Wilfredo Rangel**	
1	0	Paola Saputelli	**Base de Datos**
1	1	Rossana Díaz	
1	0	Antonio Silva	
1	0	Concettina Di Vasta	
0	**3**	**Mercy Ospina**	
1	**0**	**Alejandro Durán**	

BASE DE DATOS

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Base de Datos	Aplicaciones con Tecnología Internet, Ingeniería de Software e Interacción Humano-Computador	Mercy Ospina	Antonio Leal	Alfredo Matteo	Jossie Zambrano	Norelva Niño

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	0	Edgar González	**Aplicaciones con Tecnología Internet**
0	**3**	**Antonio Leal**	
1	0	Sergio Rivas	
1	**0**	**Jossie Zambrano**	
1	1	Eugenio Scalise	
0	**0**	**Alfredo Matteo**	**Ingeniería de Software e Interacción Humano-Computador**
0	1	Nora Montaño	
0	**0**	**Norelva Niño**	
1	1	Eugenio Scalise	
1	0	Iván Flores	
0	1	Andrés Sanoja	
1	0	Jossie Zambrano	

PRUEBA DOCUMENTO 2						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Base de Datos	Inteligencia Artificial, Base de Datos	Antonio Silva, Iván Flores	Marcel Castro	Rossana Díaz	Esmeralda Ramos	Mercy Ospina

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
0	**0**	**Esmeralda Ramos**	**Inteligencia Artificial**
0	0	Haydemar Nuñez	
1	0	Iván Flores	
0	**0**	**Marcel Castro**	

1	0	Paola Saputelli	**Base de Datos**
0	**1**	**Rossana Díaz**	
1	0	Antonio Silva	
1	0	Concettina Di Vasta	
0	**2**	**Mercy Ospina**	
1	0	Alejandro Durán	

COMPUTACIÓN GRÁFICA

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Computación Gráfica	Sistemas Distribuidos y Paralelos, Computación Gráfica	Esmitt Ramírez	Andrés Sanoja	Rhadames Carmona	Claudia León	Héctor Navarro

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	0	Carlos Acosta	**Sistemas Distribuidos y Paralelos**
0	**1**	**Claudia León**	
0	2	Robinson Rivas	
0	**1**	**Andrés Sanoja**	
1	0	Jaime Parada	
1	0	Omaira Rodríguez	**Computación Gráfica**
0	2	Robinson Rivas	
0	**1**	**Rhadames Carmona**	
0	**1**	**Héctor Navarro**	
2	0	Walter Hernández	
1	0	Esmitt Ramírez	

PRUEBA DOCUMENTO 2						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Computación Gráfica	Computación Gráfica, Ingeniería de Software e Interacción Humano-Computador	Ernesto Coto	Héctor Navarro	Nora Montaño	Robinson Rivas	Alfredo Matteo

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	0	Omaira Rodríguez	**Computación Gráfica**
0	**2**	**Robinson Rivas**	
1	1	Rhadames Carmona	
0	**2**	**Héctor Navarro**	
2	0	Walter Hernández	
1	0	Esmitt Ramírez	

1	**1**	**Alfredo Matteo**	**Ingeniería de Software e Interacción Humano-Computador**
1	**1**	**Nora Montaño**	
1	1	Norelva Niño	
1	1	Eugenio Scalise	
2	0	Iván Flores	
1	1	Andrés Sanoja	
1	2	Jossie Zambrano	

PRUEBA DOCUMENTO 3						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Computación Gráfica	Computación Gráfica, Tecnologías de Comunicaciones y Redes de Computadoras	Ernesto Coto	Robinson Rivas	Daniel Villavicencio	Omaira Rodríguez	David Pérez

Asignaciones como Principal	**Asignaciones como Suplente**	**Profesor**	**Área**
1	**0**	**Omaira Rodríguez**	**Computación Gráfica**
0	**3**	**Robinson Rivas**	
1	1	Rhadames Carmona	
1	2	Héctor Navarro	
2	0	Walter Hernández	
1	0	Esmitt Ramírez	
1	1	María Villapol	**Tecnologías de Comunicaciones y Redes de Computadoras**
2	0	Walter Hernández	
1	2	Ana Morales	
1	3	Rafael Angulo	
1	**1**	**David Pérez**	
2	0	Karima Velásquez	
1	**1**	**Daniel Villavicencio**	
1	1	Eric Gamess	

PRUEBA DOCUMENTO 4						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Computación Gráfica	Computación Gráfica	Ernesto Coto	Esmitt Ramírez	Omaira Rodríguez	Rhadames Carmona	Héctor Navarro

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	**1**	**Omaira Rodríguez**	**Computación Gráfica**
1	3	Robinson Rivas	
1	**1**	**Rhadames Carmona**	
1	**2**	**Héctor Navarro**	
2	0	Walter Hernández	
1	**0**	**Esmitt Ramírez**	

PRUEBA DOCUMENTO 5						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Computación Gráfica	Computación Gráfica, Ingeniería de Software e Interacción Humano-Computador	Esmitt Ramírez	Rhadames Carmona	Eugenio Scalise	Robinson Rivas	Norelva Niño

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
2	1	Omaira Rodríguez	**Computación Gráfica**
1	**3**	**Robinson Rivas**	
1	**2**	**Rhadames Carmona**	
1	3	Héctor Navarro	
2	0	Walter Hernández	
2	0	Esmitt Ramírez	
1	2	Alfredo Matteo	**ISW w IHC**
2	1	Nora Montaño	
1	**1**	**Norelva Niño**	
1	**1**	**Eugenio Scalise**	
2	0	Iván Flores	
1	1	Andrés Sanoja	
1	2	Jossie Zambrano	

INGENIERÍA DE SOFTWARE E INTERACCIÓN HUMANO-COMPUTADOR

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Ingeniería de Software e Interacción Humano-Computador	Ingeniería de Software e Interacción Humano-Computador	Andrés Sanoja	Nora Montaño	Norelva Niño	Alfredo Matteo	Jossie Zambrano

Asignaciones como Principal	**Asignaciones como Suplente**	**Profesor**	**Área**
1	**0**	**Alfredo Matteo**	**Ingeniería de Software e Interacción Humano-Computador**
0	**1**	**Nora Montaño**	
0	**1**	**Norelva Niño**	
1	1	Eugenio Scalise	
2	0	Iván Flores	
0	1	Andrés Sanoja	
1	**1**	**Jossie Zambrano**	

INTELIGENCIA ARTIFICIAL

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Inteligencia Artificial	Inteligencia Artificial, Tecnologías en Comunicaciones y Redes de Computadoras	Haydemar Nuñez, Esmeralda Ramos	Iván Flores	Karima Velásquez	Marcel Castro	Ana Morales

Asignaciones como Principal	**Asignaciones como Suplente**	**Profesor**	**Área**
0	1	Esmeralda Ramos	**Inteligencia Artificial**
0	0	Haydemar Nuñez	
1	**0**	**Iván Flores**	
1	**0**	**Marcel Castro**	
0	0	María Villapol	**Tecnologías en Comunicaciones y Redes de Computadora**
0	0	Walter Hernández	
0	**0**	**Ana Morales**	
0	1	Rafael Angulo	
0	0	David Pérez	
0	**0**	**Karima Velásquez**	
0	0	Daniel Villavicencio	
1	0	Eric Gamess	

SISTEMAS DE INFORMACIÓN

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Sistemas de Información	Tecnologías en Comunicaciones y Redes de Computadoras, Sistemas de Información	Paola Saputelli	Walter Hernández	Wilfredo Rangel	María Villapol	Concettina Di Vasta

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	**0**	**María Villapol**	**Tecnologías en Comunicaciones y Redes de Computadoras**
1	**0**	**Walter Hernández**	
1	2	Ana Morales	
1	3	Rafael Angulo	
1	1	David Pérez	
1	0	Karima Velásquez	
1	1	Daniel Villavicencio	
1	0	Eric Gamess	
1	0	Pedro Bonillo	**Sistemas de Información**
1	0	Antonio Silva	
1	**0**	**Concettina Di Vasta**	
0	**2**	**Wilfredo Rangel**	

PRUEBA DOCUMENTO 2						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Sistemas de Información	Sistemas de Información, Tecnologías en Comunicaciones y Redes de Computadoras	Pedro Bonillo	Antonio Silva	Karima Velásquez	Concettina Di Vasta	Eric Gamess

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	0	Pedro Bonillo	**Sistemas de Información**
1	**0**	**Antonio Silva**	
1	**1**	**Concettina Di Vasta**	
1	2	Wilfredo Rangel	
1	1	María Villapol	**Tecnologías en Comunicaciones**
2	0	Walter Hernández	

1	2	Ana Morales	**y Redes de Computadoras**
1	3	Rafael Angulo	
1	1	David Pérez	
1	**0**	**Karima Velásquez**	
1	1	Daniel Villavicencio	
1	**0**	**Eric Gamess**	

PRUEBA DOCUMENTO 3						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Sistemas de Información	Base de Datos, Sistemas de Información	Francisco Castillo	Paola Saputelli	Pedro Bonillo	Rossana Díaz	Concettina Di Vasta

Asignaciones como Principal	**Asignaciones como Suplente**	**Profesor**	**Área**
1	**0**	**Paola Saputelli**	**Base de Datos**
1	**1**	**Rossana Díaz**	
2	0	Antonio Silva	
1	2	Concettina Di Vasta	
1	3	Mercy Ospina	
1	1	Alejandro Durán	
1	**0**	**Pedro Bonillo**	**Sistemas de Información**
2	0	Antonio Silva	
1	**2**	**Concettina Di Vasta**	
1	2	Wilfredo Rangel	

SISTEMAS DISTRIBUIDOS Y PARALELOS

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Sistemas Distribuidos y Paralelos	Sistemas Distribuidos y Paralelos, Computación Gráfica	Rina Surós	Carlos Acosta	Esmitt Ramírez	Claudia León	Héctor Navarro

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
0	**0**	**Carlos Acosta**	**Sistemas Distribuidos y Paralelos**
0	**0**	**Claudia León**	
0	0	Robinson Rivas	
0	1	Andrés Sanoja	
0	0	Jaime Parada	
0	0	Omaira Rodríguez	**Computación Gráfica**
0	0	Robinson Rivas	
0	0	Rhadames Carmona	
0	**0**	**Héctor Navarro**	
0	0	Walter Hernández	
0	**0**	**Esmitt Ramírez**	

PRUEBA DOCUMENTO 2						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Sistemas Distribuidos y Paralelos	Sistemas Distribuidos y Paralelos, Computación Gráfica	Carlos Acosta	Jaime Parada	Walter Hernández	Robinson Rivas	Rhadames Carmona

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	0	Carlos Acosta	**Sistemas Distribuidos y Paralelos**
0	1	Claudia León	
0	**0**	**Robinson Rivas**	
0	1	Andrés Sanoja	
0	**0**	**Jaime Parada**	
0	0	Omaira Rodríguez	**Computación Gráfica**
0	0	Robinson Rivas	
0	**0**	**Rhadames Carmona**	

0	1	Héctor Navarro	
0	**0**	**Walter Hernández**	
1	0	Esmitt Ramírez	

TECNOLOGIAS EDUCATIVAS

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Tecnologías Educativas	Tecnologías Educativas	Yusneyi Carballo	Nancy Zambrano	Ana Leguízamo	Johnny Sepúlveda	Nora Montaño

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
0	**0**	**Johnny Sepúlveda**	**Tecnologías Educativas**
0	**0**	**Nancy Zambrano**	
0	0	Yusneyi Carballo	
0	**0**	**Ana Leguízamo**	
0	**0**	**Nora Montaño**	
0	0	Yosly Hernández	
0	0	Vanessa Miguel	

PRUEBA DOCUMENTO 2						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Tecnologías Educativas	Tecnologías en Comunicaciones y Redes de Computadora y Base de Datos	Yosly Hernández y Vanessa Miguel	Eric Gamess	Paola Saputelli	Rafael Angulo	Mercy Ospina

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
0	0	María Villapol	**Tecnologías en Comunicaciones y Redes de Computadora**
0	0	Walter Hernández	
0	0	Ana Morales	
0	**0**	**Rafael Angulo**	
0	0	David Pérez	
0	0	Karima Velásquez	
0	0	Daniel Villavicencio	
0	**0**	**Eric Gamess**	
0	**0**	**Paola Saputelli**	**Base de Datos**
0	1	Rossana Díaz	
1	0	Antonio Silva	
1	0	Concettina Di Vasta	
0	**1**	**Mercy Ospina**	
1	0	Alejandro Durán	

TECNOLOGÍAS EN COMUNICACIONES Y REDES DE COMPUTADORAS

PRUEBA DOCUMENTO 1						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Tecnologías en Comunica-ciones y Redes de Computadoras	Tecnologías en Comunica-ciones y Redes de Computadoras	María Villapol	David Pérez	Daniel Villavicencio	Ana Morales	Rafael Angulo

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
0	0	María Villapol	**Tecnologías en Comunicaciones y Redes de Computadoras**
1	0	Walter Hernández	
0	**1**	**Ana Morales**	
0	**1**	**Rafael Angulo**	
0	**0**	**David Pérez**	
1	0	Karima Velásquez	
0	**0**	**Daniel Villavicencio**	
1	0	Eric Gamess	

PRUEBA DOCUMENTO 2						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Tecnologías en Comunica-ciones y Redes de Computadoras	Tecnologías en Comunica-ciones y Redes de Computadoras	Eric Gamess	María Villapol	Ana Morales	Rafael Angulo	Daniel Villavicencio

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
0	**0**	**María Villapol**	**Tecnologías en Comunicaciones y Redes de Computadoras**
1	0	Walter Hernández	
0	**2**	**Ana Morales**	
0	**2**	**Rafael Angulo**	
1	0	David Pérez	
1	0	Karima Velásquez	
1	**0**	**Daniel Villavicencio**	
1	0	Eric Gamess	

PRUEBA DOCUMENTO 3						
Área del TEG	**Área Recomendada**	**Tutor(es)**	**Principal1**	**Principal2**	**Suplente1**	**Suplente2**
Tecnologías en Comunicaciones y Redes de Computadoras	Tecnologías en Comunicaciones y Redes de Computadoras, Computación Gráfica	María Villapol	Rafael Angulo	Omaira Rodríguez	David Pérez	Robinson Rivas

Asignaciones como Principal	Asignaciones como Suplente	Profesor	Área
1	0	María Villapol	**Tecnologías en Comunicaciones y Redes de Computadoras**
1	0	Walter Hernández	
1	2	Ana Morales	
0	**3**	**Rafael Angulo**	
1	**0**	**David Pérez**	
1	0	Karima Velásquez	
1	1	Daniel Villavicencio	
1	0	Eric Gamess	
0	**0**	**Omaira Rodríguez**	**Computación Gráfica**
0	**0**	**Robinson Rivas**	
0	1	Rhadames Carmona	
0	1	Héctor Navarro	
1	0	Walter Hernández	
1	0	Esmitt Ramírez	

Anexo 6. Manual para el desarrollador.

1. Estructura del Sistema

El Sistema está compuesto por 5 paquetes:

- com.saj.classifier: clases concernientes al modelo de clasificación.
- com.saj.controller: Redirector, Action Factory y Actions.
- com.saj.dao: clases de acceso a base de datos.
- com.saj.model: clases java bean que representan los objetos del sistema.
- com.saj.utils: clases varias.
- com.saj.validator: clases de validación de objetos.

Además de los diferentes archivos JSP, CSS y Javascript que se encuentran en la carpeta de recursos web del sistema.

2. Agregar una nueva funcionabilidad al sistema

La vista deberá ser un archivo HTML o JSP el cual deberá ser agregado al menú del sistema y podrá estar localizada en cualquier de las subcarpetas del directorio WEB-INF/views.

La lógica de negocios deberá desarrollarse implementando la interfaz Action de la siguiente manera:

```
package com.saj.controller;

import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class AsignarJuradosLoadAction implements Action {

	public String execute(HttpServletRequest request, HttpServletResponse response) {
		/* Lógica de negocio */
		return "/views/ruta de la vista a cargar";
	}
}
```

Se desarrollarán tantos actions como funcionalidades nuevas se deseen agregar, cada una de ellas con su ruta asociada la cual será agregada en el constructor de la clase ActionFactory:

```
actions.put("/ruta", new NombreAction());
```

Esto le indicará al sistema que Action ejecutar cuando se llame a la URL SAJ/pages/ruta.

3. Cambiar la distribución probabilística que usa el sistema

De manera predeterminada el sistema trabajará con una función de distribución normal (también conocida como distribución Gaussiana), sin embargo, esta distribución puede ser cambia modificado la lógica de la clase Distribución.java localizada en el paquete com.saj.classifier.
Los métodos que deben componer la clase son los siguientes:

- agregarValor: recibe como entrada un double X_i. Se encarga de agregar un nuevo valor al conjunto de puntos que conforman la distribución. No devuelve ningún valor.

- fdp: recibe como entrada un double X_i. Devolverá el resultado de evaluar la función de densidad en el punto solicitado.

4. Cambiar el algoritmo de lematización

De manera predeterminada el sistema trabajará con el algoritmo de Potter Steamming, sin embargo, este algoritmo puede ser modificado en la clase Lematizador.java localizada en el paquete com.saj.classifier. Debe existir una función llamada "lematizar" con la siguiente forma:

```
public static String lematizar(String palabra){
       /* lógica de lematización */
}
```

5. Cambiar la vista minable que es cargada por el sistema

El sistema utilizará como vista minable el contenido del archivo VM.arff localizado en la carpeta resources que está en la raíz del sistema. Este archivo seguirá el formato tradicional para un archivo arff.

Dependiendo del esquema de pesado utilizado se pueden requerir archivos adicionales, en el caso del pesado por entropía se necesitará un archivo con todos factores de entropía de los atributos que conforman la vista minable que se encuentra en el archivo factoresEntropia.txt.

6. Cambiar el modelo de clasificación o esquema de pesado

La lógica del clasificador se encuentra alojada en la clase Clasificador localizada en el paquete com.saj.classifier. Tiene tres partes principales:

- El constructor, el cual tomará como parámetro el archivo arff con la vista minable y opcionalmente el archivo con los factores del esquema de pesado a utilizar. Este constructor es llamado exclusivamente en el init del servlet (clase Redirector.java del paquete com.saj.controller), actualmente se llama al constructor con los siguientes parámetros:

```
clasificador = new Clasificador(new File(pathVM), new File(pathFactores));
```

- El método obtenerInstancia, el cual recibe todo el texto que conforma un TEG (título, resumen/introducción y palabras clave) y devuelve un objeto del tipo instancia que tiene la siguiente estructura:

```
public class Instancia implements Serializable{



    private static final long serialVersionUID = 1L;



    /** Clase primaria */

    public String clase1;
```

  Este método deberá encargarse de procesar el texto plano y convertirlo en un conjunto de valores procesables por el clasificador. Asimismo, la lógica del esquema de pesado se encuentra alojada en este método ya que afecta cómo se calculan los valores de los atributos que conforman la instancia (con la salvedad que modificar el esquema de pesado implica la modificación de la vista minable).

- El método clasificar que recibirá una instancia y devolverá la misma pero con las opciones profesionales asignadas (las cuales corresponden a los atributos clase1 y clase2).

Toda la lógica del clasificador y del esquema de pesado puede ser modificada sin que afecte los otros módulos del sistema si se respeta la estructura de la clase Clasificador.java. Si se desean hacer cambios adicionales pueden realizarse en el Action dedicado al clasificador: AsignarJuradosClasificarAction.