

MINERÍA DE DATOS

Dr. José A. Olivas Varela

MÁSTER UNIVERSITARIO EN BIG DATA Y CIENCIA DE DATOS

Módulo II. Ciencia de Datos

viu

Universidad
Internacional
de Valencia



Universidad
Internacional
de Valencia

Este material es de uso exclusivo para los alumnos de la Universidad Internacional de Valencia. No está permitida la reproducción total o parcial de su contenido ni su tratamiento por cualquier método por aquellas personas que no acrediten su relación con la Universidad Internacional de Valencia, sin autorización expresa de la misma.

Edita

Universidad Internacional de Valencia

Máster Universitario en
Big Data y Ciencia de Datos

Minería de Datos
Módulo II. Ciencia de Datos
6ECTS

Dr. José A. Olivas Varela

Leyenda

abc Los términos resaltados a lo largo del contenido en color **naranja** se recogen en el apartado GLOSARIO

Índice

TEMA 1. APRENDIZAJE ESTADÍSTICO Y MINERÍA DE DATOS.....	7
1.1. Origen: Los datos	8
1.1.1. Datos, información, conocimiento	8
1.1.2. Tipos de Datos.....	8
1.1.3. Cómo se consiguen y dónde residen los datos	10
1.2. Analítica de Datos y Minería de Datos.....	11
1.2.1. ¿Qué se entiende habitualmente por analítica de datos?.....	12
1.2.2. Algunas críticas sobre la percepción habitual del análisis de datos	13
1.2.3. Tipos de salidas: predicción, pronóstico.....	14
1.2.4. El científico de datos y el KDD (Knowledge Discovery in Databases)	16
1.2.5. Métodos basados en estadística para la analítica de Datos	17
1.2.6. Métodos basados en Inteligencia Artificial (Machine Learning) para la analítica de Datos.....	18
1.2.7. Adecuación de los métodos a los problemas	20
1.2.8. Sistemas Basados en Conocimiento (Conocimiento Experto...)	22
1.3. El proceso KDD (Knowledge Discovery in Databases).....	26
1.3.1. Selección: Datos objetivo/Tarjeta de Datos	26
1.3.2. Preproceso: Limpieza de Datos	27
1.3.3. Transformación: del Clustering a la Clasificación	27
1.3.4. Minería de Datos.....	28
1.3.5. Conocimiento: formalización de patrones.....	28
1.3.6. La metodología CRISP-DM	30
1.3.7. Herramientas actuales (en entornos Big Data) para implementar estas soluciones ..	33
1.4. Algunas aplicaciones/ejemplos.....	35
1.4.1. Minería de Datos: Prevención de Incendios Forestales.....	35
1.4.2. Minería de Texto: Búsqueda y Recuperación de Información (en la Web)	70
1.4.3. Minería de Opiniones: Análisis de Sentimientos	94
1.5. Conclusiones	101
GLOSARIO.....	102

ENLACES DE INTERÉS.....	105
BIBLIOGRAFÍA.....	106
Referencias bibliográficas.....	106
Bibliografía recomendada:.....	107



Tema 1.

Aprendizaje Estadístico y Minería de Datos

En este documento se presenta una introducción al aprendizaje estadístico y la Minería de Datos. Se comienza por el origen, los datos, estableciendo inicialmente las diferencias entre datos, información y conocimiento, qué tipos de datos se suelen manejar, cómo se consiguen y dónde se almacenan habitualmente, introduciendo el concepto de “**lago de datos**”. Se prosigue con una clasificación de los diferentes tipos de “minerías”: datos, textos, opiniones, gráficos... y se enmarcan dentro de lo que hoy se denomina “**Inteligencia de Negocio**”. Se presenta una crítica sobre cómo se suele afrontar habitualmente el análisis de datos, que nos permita reflexionar sobre en qué grado se suelen hacer las cosas de una forma adecuada. A continuación se describen los diferentes tipos de análisis, distinguiendo por ejemplo entre conceptos como “predicción” y “pronóstico”. Se presenta el papel del “**científico de datos**” con taxonomías exhaustivas tanto de los métodos basados en la estadística como los basados en Inteligencia artificial (**aprendizaje automático**) para el análisis de datos y se hace una descripción de qué métodos resultan más adecuados para afrontar cada uno de los diferentes tipos de retos que suelen presentarse. Esta parte se concluye con una descripción detallada de la “**Ingeniería de Conocimiento**”, por su importancia a la hora de establecer los criterios que deben guiar el proceso de análisis de datos o la relación con los expertos en los temas a analizar, que nos pueden guiar en el análisis.

La segunda parte está dedicada a describir en profundidad la metodología genérica comúnmente usada para el proceso de descubrimiento de conocimiento en bases de datos y la minería de datos (**KDD: Knowledge Discovery in Databases**). También se describe más superficialmente la metodología CRISP-DM. Esta parte se finaliza con una introducción a las técnicas y herramientas que se suelen usar actualmente para estos propósitos en entornos Big Data.

Por último, se describen con mucho detalle tres ejemplos de aplicaciones sofisticadas de los conceptos presentados. El primero tiene que ver con un sistema para la prevención de incendios forestales (**Minería de datos**). El segundo con diferentes aplicaciones en acceso y recuperación de información (**Minería de textos**). El tercero con la importancia actual del análisis de sentimientos (**Minería de opiniones**).

Esta visión panorámica pretende ser un mapa completo de los objetivos, orientación y técnicas (tanto estadísticas como provenientes de la **Inteligencia Artificial**) para afrontar el análisis de datos en su estado actual desde el punto de vista de las Ciencias de la Computación. No se presentarán herramientas concretas debido a la imposibilidad de profundizar mínimamente en el gran número de las disponibles actualmente y no haber un criterio robusto para elegir una u otra para estudiar en detalle. Es por ello que queda fuera del alcance de esta asignatura. Se verán en profundidad tres ejemplos de minería de datos, textos y opiniones que permitirán profundizar en diversos temas relevantes en este campo, como por ejemplo el uso de técnicas de **Soft-computing** para la minería de textos o el papel cada vez más importante del análisis de datos no estructurados.

1.1. Origen: Los datos

1.1.1. Datos, información, conocimiento

Debemos distinguir claramente entre **datos** (contenido bruto), **información** (resúmenes, datos “presentables”) y **conocimiento** (mayor nivel de abstracción que nos permite tomar decisiones ante nuevas situaciones). Habitualmente hoy en día la gestión de datos se queda en un análisis descriptivo, es decir, la visualización de gráficos o lo que se suelen llamar “cuadros de mandos” o “dashboards” que ayudan a la persona responsable a tomar decisiones. Por ejemplo, esto puede venir muy bien en un entorno médico, porque permite a los profesionales tener una información más completa y procesada de los pacientes, historias clínicas, pruebas... para poder tomar decisiones. Pero no debemos quedarnos ahí, se debe ir hacia la analítica predictiva y prescriptiva, modelos que permitan extraer conocimiento a partir de los datos y la información que podamos usar para mejorar sustancialmente la práctica clínica.

1.1.2. Tipos de Datos

Las diferentes organizaciones y empresas suelen clasificar los datos según su origen y tipo de la siguiente forma:

Fuentes de datos:

Donde se originan los datos, habitualmente se consideran las siguientes:

- **Externas:** no dependen directamente de la organización. Muchas veces puede

haber problemas y limitaciones para su uso o acceso, y en otros casos pueden resultar caros.

- *Internas:* dependen directamente de la organización, no suele haber problemas y limitaciones para su uso o acceso, y normalmente no suponen ningún tipo de coste para la organización.

Tipos de datos:

Normalmente se suele hablar de datos:

- **Estructurados:** datos con formato o esquema fijo que poseen campos fijos. Por ejemplo las clásicas bases de datos relacionales. Suelen ser los más usados como origen de los procesos de Minería de datos.
- **Semi-estructurados:** no tienen formato fijo pero contienen etiquetas u otros marcadores que permiten separar los diferentes elementos, por ejemplo: los Weblogs o algunos campos de tipo texto en muchas bases de datos.
- **No estructurados:** también se suelen denominar desestructurados. Datos sin tipo predefinido. Se almacenan como documentos u objetos sin una estructura uniforme, por ejemplo: documentos de texto o multimedia.

Por lo tanto, se puede establecer una taxonomía básica inicial de los tipos de datos que se suelen usar para los procesos de Análisis y Minería:

- **Datos Internos estructurados:**

Suele ser la categoría más abundante y mejor entendida actualmente en las organizaciones, pero cada vez parece más necesario cambiar el enfoque hacia los datos externos desestructurados. Como ejemplos se pueden considerar los Perfiles Web, registros de clientes, usuarios, inventario, recursos humanos, financieros, ventas...

- **Datos Externos estructurados:**

Extensión de los anteriores pero con restricciones de existencia, disponibilidad y accesibilidad. Como ejemplos se pueden considerar los historiales de créditos o viajes, registros de morosos, censales, telefónicos...

- **Datos Internos no estructurados:**

Pueden suponer una excelente fuente para extraer conocimiento y un buen campo de pruebas para las técnicas y herramientas analíticas. Como ejemplos se pueden considerar cualquier tipo de documentos de texto o multimedia, datos de sensores, foros, comentarios en las propias webs...

- **Datos Externos no estructurados:**

La mayor área de estudio y oportunidad actualmente, sobre todo para recoger las opiniones de los consumidores y usuarios. Como ejemplos se pueden considerar los sensores externos, blogs y todas las consideradas “redes sociales” (*social media*): *Twitter*, *Facebook*, *Instagram*, *Google+*...

También hoy en día es frecuente hablar de *Bases de datos relacionales* o *SQL* frente a las *No SQL*. En las primeras, el volumen de los datos si crece lo hace poco a poco y de forma previsible y controlada, las necesidades de proceso se suelen poder asumir en un único servidor y no suele haber picos de uso del sistema imprevistos. En las segundas el volumen de los datos puede crecer puntualmente de forma descontrolada o no prevista, las necesidades de proceso tampoco se pueden prever y se pueden tener picos de uso del sistema en diferentes ocasiones.

1.1.3. Cómo se consiguen y dónde residen los datos

Los datos se pueden conseguir o generar de muy diferentes formas. Entre las más habituales podríamos destacar las siguientes:

- **Sistemas Transaccionales:** operadores que recogen las peticiones a través de Call-Centers...
- **Transacciones** que se generan en las Webs (ficheros *weblogs*)...
- **Los sensores** permiten capturar las magnitudes físicas o químicas y convertirlas en datos, por ejemplo: temperatura, luz, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, humedad, sonido, movimiento o el pH.
- **Redes sociales, etc.**

Es interesante destacar la relevancia actual de la adquisición, manejo, transmisión, uso y almacenamiento de lo que se suelen denominar **Datos en Tiempo Real** (*Real Time Data*). Se producen en tiempo inmediato, algunos provienen de sensores y muchos suelen ser no estructurados (por ejemplo los provenientes de redes sociales), lo que provoca diversas necesidades específicas y problemas para su tratamiento.

Hasta hace muy pocos años, cuando se hablaba de “bases de datos”, los profesionales del ámbito podían tener una idea bastante aproximada y conocimiento sobre casi todos los soportes y herramientas existentes, porque eran relativamente pocas y conocidas. En los últimos años ha habido una explosión en tipos, herramientas y productos disponibles, los que imposibilita un conocimiento completo de los mismos y obliga a los profesionales a una actualización permanente y muy rápida en estas tecnologías. Cada día aparecen nuevos sistemas de soporte para diferentes bases de datos. Sirvan como ejemplos:

- **Bases de datos documentales (*Document Databases*):**

MarkLogic, *Couchdatabase*, *MongoDB*...

- **Bases de datos de columna ancha (Wide Column Stores):**

Aerospike, Redis, Riak, Amazon DynamoDB...

- **Bases de datos gráficas (Graph Databases):**

Neo4j, Infinite Graph...

- **Bases de datos Valor-Llave (Key-Value Databases):**

Accumulo, Hypertable, Cassandra, Apache Hbase, Amazon SimpleDB...

Esta gran diversidad de sistemas de almacenamiento para los muy variados tipos de datos provoca que cada vez sea más importante y usado el concepto de **Lago de Datos** o **Data Lake**, que podríamos describir como un repositorio para grandes cantidades y variedades de datos, tanto estructurados como no estructurados. El lago acepta entradas desde diversas fuentes y puede preservar tanto la fidelidad de los datos originales como las diversas transformaciones que se les van haciendo, incluso simultáneamente desde diferentes departamentos, lo que puede generar diferentes evoluciones en paralelo de los mismos datos originales, por ejemplo desde el departamento de publicidad y del de ventas. Los programadores podrían actuar directamente sobre los datos que se están transmitiendo para una analítica en tiempo real. El lago puede servir como zona de montaje del **Data Warehouse**, la ubicación de los datos tratados con más cuidado para reportes y análisis en modo **Batch**.

Por lo tanto, se debe abandonar la forma clásica de partir únicamente de una base de datos relacional (y habitualmente numérica) para un determinado proceso de análisis de datos, que es lo que se hace casi en la totalidad de los casos en la actualidad, y comenzar a pensar en usar como origen un contenido más amplio y heterogéneo del **Data Lake**.

Esto, desde mi punto de vista, provoca la necesidad de evolucionar desde lo que se suele hacer ahora, un mero análisis estadístico o con una única técnica de aprendizaje automático sobre una única base de datos numérica hacia una Minería de Datos que parte de un conjunto de datos variado y heterogéneo y que use diversas técnicas en diferentes momentos con el objetivo de obtener un conocimiento común, según los objetivos que se pretendan conseguir. Pero esto no es una tarea fácil, requiere un extenso conocimiento sobre tipos de datos y su tratamiento, técnicas de análisis e Ingeniería del Conocimiento. Quizá esta debería ser la formación requerida para un auténtico “*Científico de Datos*” o en común del equipo de análisis. Pero creo que actualmente la realidad en las organizaciones dista mucho de esta concepción.

1.2. Analítica de Datos y Minería de Datos

Como se ha destacado en los epígrafes anteriores el análisis de datos no suele ser una tarea sencilla. Se suelen designar las tareas de “minería” atendiendo al origen de los datos y a la intención del proceso. Según esto, entre otros tipos de minería, los más habituales son:

- **Minería de Datos:** se parte de datos estructurados y casi siempre numéricos. El objetivo es encontrar regularidades (“patrones”) que permitan establecer modelos normalmente de predicción o de clasificación.
- **Minería de Textos:** se parte de colecciones de documentos de texto. El objetivo es encontrar regularidades (“patrones”) semánticos (aunque normalmente solo se manejan desde el punto de vista lexicográfico) que permitan ayudar en tareas como el acceso, la búsqueda y la recuperación de información (se detalla en los ejemplos finales de este documento) o la elaboración automática de resúmenes de dichos textos.
- **Minería de Opiniones:** también se suele denominar “Análisis de Sentimientos” (*Sentiment Analysis*). Se parte de colecciones de documentos de texto, habitualmente pequeños, como los típicos mensajes en redes sociales. El objetivo es manifestarse sobre la “polaridad” (bueno o malo) de un mensaje con respecto a un determinado tema, encontrando regularidades (“patrones”) semánticas (aunque normalmente solo se manejan desde el punto de vista lexicográfico) que permitan ayudar en tareas como la percepción de un producto o un político a través de las opiniones de los usuarios de las redes sociales (se detalla en los ejemplos finales de este documento). Esto permite, por ejemplo, disponer de alertas tempranas (*early warnings*) sobre las críticas de los consumidores a un determinado producto.
- **Minería de Grafos:** se parte de grafos o redes, normalmente compuestas de términos/conceptos y sus relaciones. El objetivo es encontrar regularidades (“patrones”) en dichos grafos que permitan ayudar a describir y entender los fenómenos susceptibles de ser representados por redes o grafos, como las redes sociales, las relaciones lingüísticas entre conceptos, las redes de computadoras, las redes sociales, las estructuras químicas... Esto permite, por ejemplo, el aprovechamiento inteligente de ontologías o tesauros como *Wordnet* o como se pretende habitualmente en la denominada “Web Semántica”.

En la evolución de la profundidad del análisis tanto de datos estructurados como de no estructurados, en el primer caso el nivel más profundo se alcanza con la analítica predictiva y en el segundo con el análisis de sentimientos. A lo largo de este documento se ilustrarán diversos ejemplos, críticas, aplicaciones y particularidades de todas estas variantes.

1.2.1. ¿Qué se entiende habitualmente por analítica de datos?

La “analítica de datos” se suele enmarcar como una parte esencial de lo que se denomina “Inteligencia de Negocio” (*Business Intelligence*, BI) y se suele definir como la capacidad de transformar datos en información para ayudar a gestionar una empresa, que consiste en los procesos, aplicaciones y prácticas que apoyen la toma de decisiones ejecutivas. La BI se suele dividir en dos grandes grupos:

- **BI Operacional:** trata los informes estándar, descripciones de los datos (información), funciones al nivel operacional con los trabajadores, clientes, usuarios, socios...
- **BI Analítica y Táctica/Estratégica:** pretende dar soporte a los ejecutivos y a los gestores en niveles tácticos que contribuyen a la estrategia global de la empresa o institución. Suele

contemplar análisis estadístico, modelos predictivos y de extrapolación, pronósticos, optimización...

Como ejemplos de soluciones BI se pueden mostrar las siguientes en diferentes ámbitos de las instituciones y empresas:

- **Gestión de la Información (Information Management)**: calidad de datos, Gobierno del dato, ETL (Extraer, Transformar y Cargar)...
- **Informes (Reporting)**: cuadros de Mando, visualización, movilidad...
- **Analítica avanzada (Advanced Analytics)**: segmentación, mejor próxima oferta, mantenimiento preventivo, modelos de riesgo...
- **Gestión de proyectos de empresa (Enterprise Project Management)**: presupuestación y planificación, consolidación financiera, rentabilidad, balances, costes...

1.2.2. Algunas críticas sobre la percepción habitual del análisis de datos

Desde mi punto de vista, la visión habitual de las posibilidades y expectativas de la BI es demasiado restringida, y mucho más las prácticas habituales en las empresas e instituciones. Cuando se habla de “transformar datos en información”, “apoyen la toma de decisiones” son expresiones muy imprecisas y ¡hay muchas otras cosas que se pueden hacer!

Si nos fijamos en las posibles salidas propuestas, vemos que de nuevo es demasiado restringido. Normalmente la “información” es solo “visualización”, es decir, una forma de representar datos originales que están en bruto de una forma más ordenada y resumida, por ejemplo en un gráfico. Pero esto es muy poco “inteligente” y al final es el que lo hace el responsable de tomar las decisiones, basándose en esta información de la que dispone, y la calidad de estas decisiones dependerá exclusivamente de la capacidad y preparación del decisor.

Por ello, creo que se debe tender a generar y manejar “conocimiento” en el sentido anteriormente descrito. Se pueden diseñar sistemas sofisticados de muy diversos tipos, como los Sistemas de Ayuda a la Decisión (**DSS**), los Sistemas Recomendadores (**Recommender Systems**) o el Análisis de series temporales (Predicción vs Pronóstico), por ejemplo, para tareas automáticas de Segmentación, recomendación de otros productos, pronóstico de terremotos, incendios u otros fenómenos naturales, puntos de inflexión en la bolsa o el cambio de moneda... y solo las grandes empresas punteras se esfuerzan en diseñar sistemas de este tipo.

Todo ello pasa por encontrar regularidades en los datos o su evolución (¡patrones!), y ser capaz de formalizarlos o expresarlos de alguna forma computable para que puedan ser usados por estos sistemas. Las salidas esperadas deben condicionar todo el proceso de análisis de datos. Por ejemplo, no se diseña de la misma forma un sistema de predicción que uno de pronóstico. Pero la práctica habitual es ir “a ciegas” hacia delante, es decir, partir de los datos más o menos en bruto y aplicarles algún algoritmo o herramienta de análisis (por ejemplo: estadística o de aprendizaje automático) y tratar de interpretar la salida de este procedimiento por si puede ser útil.

Creo que este es un enfoque erróneo del análisis y minería de datos y suele ser el más habitual.

1.2.3. Tipos de salidas: predicción, pronóstico...

Los Sistemas de Ayuda a la decisión (*Decision Support Systems* DSS) son sistemas computacionales que proporcionan consejos para la mejora en la toma de decisiones. Básicamente, estos consejos pueden venir de tres tipos de análisis:

- **Análisis Descriptivo:**

Resumen claro y fácil de entender de una colección de datos. Fundamento y concepto más básico de todas las estadísticas. Visualización de datos para entender el pasado y el presente. Se describen los datos con tablas o gráficos. Descripciones numéricas de la variabilidad y la posición (Figura 1).



Figura 1. Diferentes tipos de visualización de datos. Fuente: elaboración propia.

- **Análisis Predictivo:**

Extrapolación de funciones (tendencia para el futuro, pero no hay capacidad de pronóstico, hechos/cambios puntuales, figura 7). Correlaciones entre variables (demasiado evidente, no suele funcionar de forma muy fina). Encontrar patrones en los datos que puedan ser aplicados a situaciones futuras (KDD: Knowledge Discovery in Databases y Minería de Datos). Métodos de CLUSTERING Y CLASIFICACIÓN.

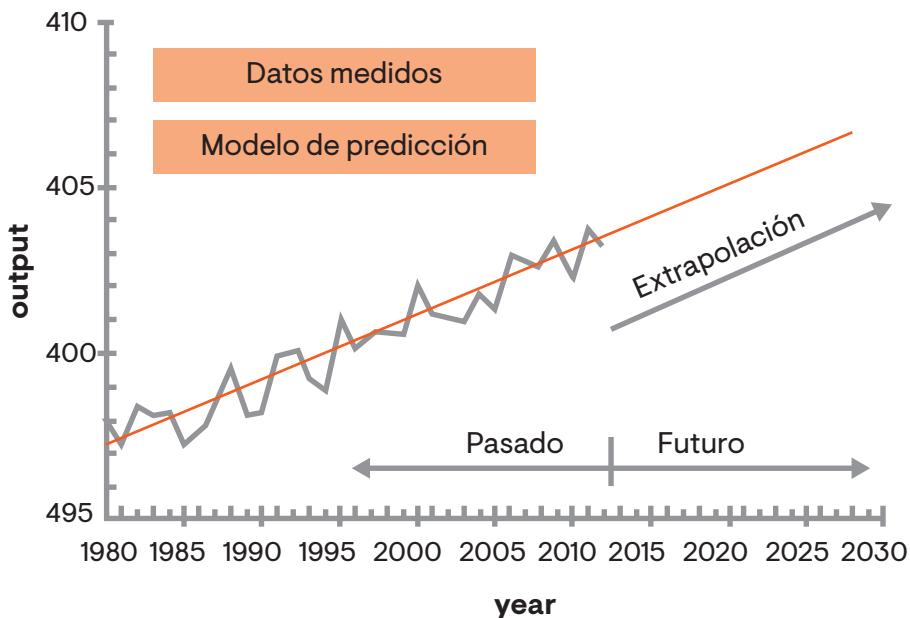


Figura 2. Extrapolación de funciones (Por ejemplo: Estimaciones o Líneas de Tendencia). Fuente: elaboración propia.

- **Análisis Prescriptivo:**

El análisis predictivo se centra en un escenario futuro². El prescriptivo se centra en múltiples alternativas. Por lo tanto, un modelo prescriptivo puede ser considerado como una combinación de modelos predictivos (uno por cada posible escenario), que se ejecutan en paralelo. El objetivo es encontrar la mejor opción posible: OPTIMIZACIÓN, por ejemplo, la elección del tratamiento más adecuado en oncología. Técnicas: técnicas de investigación operativa, algoritmos genéticos, técnicas estocásticas, metaheurísticas, etc.

Conviene distinguir entre **predicción** y **pronóstico**. La primera tiene que ver con la estimación, con la extrapolación, continuidad, por ejemplo, podemos estimar (predecir) cuántos habitantes tendrá Madrid en 2025 en base a la tendencia demográfica. En cambio, el pronóstico consiste en anticiparse a un hecho puntual en base a un conjunto pequeño de alternativas, por ejemplo, en una quiniela de fútbol pone “marque con una X su pronóstico” y las alternativas son 1-X-2, o un terremoto se debe pronosticar en base a las alternativas SI-NO. Los sistemas para abordar una u otra opción son claramente distintos. Los primeros se basan en la extrapolación y los segundos en una serie de características para anticiparse al hecho puntual (por ejemplo, en las quinielas saber si alguno de los equipos se juega algo importante como el descenso si hay un jugador importante lesionado o quién va a ser el árbitro del partido).

Hay una gran cantidad de aplicaciones de este tipo de sistemas. Si nos centramos en sistemas de medicina, podemos citar tres pequeños ejemplos que se pueden estudiar en detalle en las referencias de la bibliografía recomendada, desarrollados en el marco del grupo de investigación liderado por el autor de este tema.

El primero tiene que ver con el concepto de “enfermedades borrosas” (fuzzy deseases), implantado en un sistema para diagnosticar y tratar fibromialgia, que obtuvo el premio al mejor trabajo en el congreso de la Asociación Española para la IA en 2015 (Romero-Cordoba, Olivas, Romero, Alonso-Gonzalez, Serrano-Guerrero, 2017). En este trabajo se propone el concepto de “prototipo deformable borroso” para caracterizar enfermedades que pueden ser confundidas o emascarar otras, incluso que no están perfectamente caracterizadas o asumidas por toda la comunidad médica.

En el segundo ejemplo se muestra el diseño de un sistema de ayuda a la decisión en oncología a partir de la detección, clasificación y uso de las expresiones causales y condicionales en textos médicos (en este caso de Mayo Clinic y Mount Sinai Hospital). En la figura 3 se puede ver el grafo causal de la pregunta ¿qué causa cáncer de pulmón? (Sobrino, Puente, Olivas, 2014). En el tercer caso se presenta un estudio inicial sobre el desarrollo de un sistema para la estimación de probabilidades de sufrir determinados tipos de cáncer en base al estudio del genograma del paciente (Calatrava, Oruezabal, Olivas, Romero, Serrano-Guerrero, 2015).

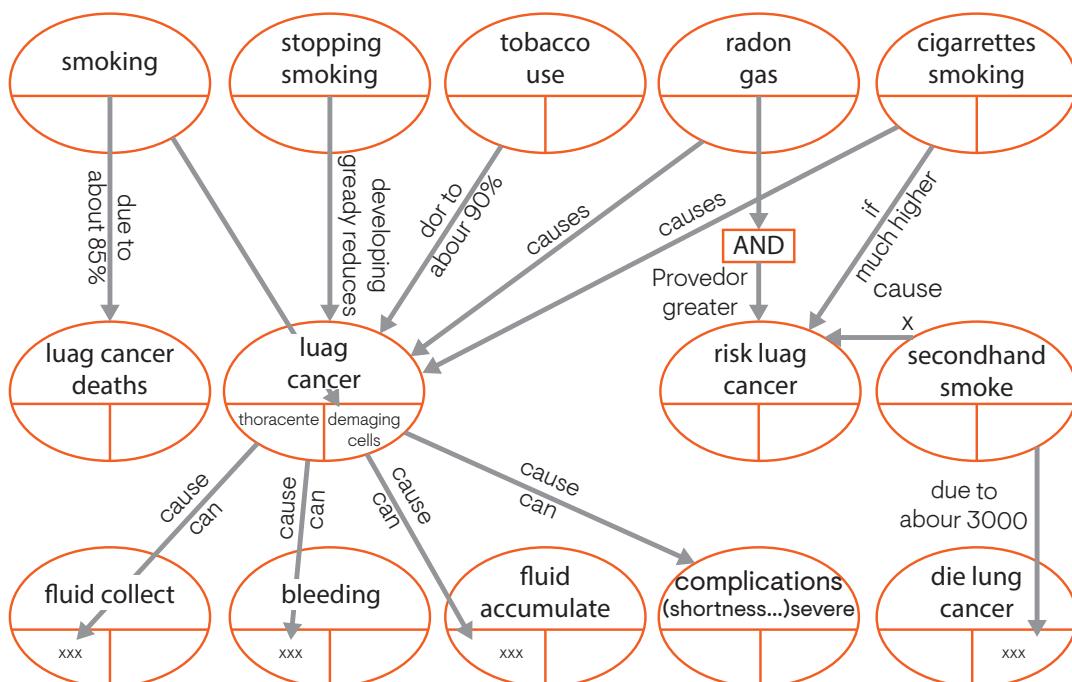


Figura 3. Grafo causal de la pregunta ¿qué causa cáncer de pulmón?. Fuente: Sobrino et al. (2014).

1.2.4. El científico de datos y el KDD (Knowledge Discovery in Databases)

Hoy en día el “científico de datos” (*data scientist*) es una figura muy demandada y escasa en ambientes profesionales, científicos o académicos que, como se describe en este documento, debe poseer conocimientos de computación, bases de datos, Inteligencia Artificial, Aprendizaje Automático, estadística, visualización, reconocimiento de patrones, sociología, psicología, KDD y Minería de Datos... y que debe ser capaz de seleccionar y guiar las herramientas y técnicas más adecuadas para cada problema y objetivos concretos. Es por ello que no es frecuente ni fácil encontrar profesionales con este perfil tan completo. La mejor forma de paliar esta dificultad debería ser la formación de equipos

multidisciplinares que cubran todas estas necesidades, pero esto tampoco es frecuente y por ello gran parte de los proyectos de análisis de datos que llevan a cabo diferentes instituciones y entidades no consiguen los resultados que podrían ser esperados.

Otro problema frecuente e importante a la hora de desarrollar este tipo de proyectos es la tendencia a aplicar las herramientas, métodos o algoritmos que el equipo mejor conoce (“al que solo dispone de un martillo, todos los problemas le parecen clavos”) de una forma “ciega” sobre las bases de datos de las que se dispone (normalmente sólo sobre una), sin importar en principio cuál es el propósito final de ese análisis de datos. Esta aproximación errónea se podría mejorar en gran medida disponiendo de un buen “científico de datos”, en el sentido anteriormente descrito, capaz de guiar todo el proceso con criterios bien sustentados.

1.2.5. Métodos basados en estadística para la analítica de Datos

Las técnicas de **Regresión** expresan la formalización de una relación significativa entre dos o más variables para calcular pronósticos a partir del conocimiento de los valores en un individuo concreto. Entre otros tipos de Regresión, para el análisis de datos se suele usar:

- Lineal (aproximación de la dependencia entre una variable dependiente y variables independientes).
- Múltiple (para predecir el valor de una variable dependiente a partir de variables independientes).
- Logística (para predecir variables categóricas).
- CART (Classification And Regression Trees, Leo Breiman).
- Etc.

Además, también se suelen usar otras técnicas clásicas en el ámbito de la estadística, como:

- Técnicas de **extrapolación** de funciones.
- Técnicas de **aproximación** y **ajuste** de funciones.
- Técnicas de **agrupamiento** basadas en medidas estadísticas (*clustering*).
- Etc.

Es importante reseñar que muchas se pueden englobar tanto en técnicas estadísticas como de Machine Learning, es decir, la mayoría de las técnicas de aprendizaje automático se basan en mecanismos estadísticos.

1.2.6. Métodos basados en Inteligencia Artificial (**Machine Learning**) para la analítica de Datos

La **Inteligencia Artificial** (IA/AI -siglas en inglés-) se puede ver como la disciplina del ámbito de la computación y los sistemas de información que pretende simular computacionalmente comportamientos humanos que pueden ser considerados como inteligentes. Hay diversas ramas dentro de la IA, como la Visión Artificial o la Robótica, pero en este tema nos centraremos en el aprendizaje automático (AA/ML **Machine Learning**) y en la que se suele denominar “Ingeniería del Conocimiento” (IC/KE), encargada del desarrollo de Sistemas Basados en el Conocimiento (SBC/KBS), como pueden ser los Sistemas de Ayuda a la Decisión (DSS **Decision Support Systems**). La tradición de los SBC comenzó con los denominados “Sistemas Expertos”, sistemas computacionales que tratan de emular las capacidades de un experto en un tema basándose en la extracción del conocimiento del propio experto o grupo de expertos y transmitiéndoselo al sistema. Con la proliferación del almacenamiento y uso de datos de forma masiva, los SBC actuales suelen apoyarse en ambos pilares: expertos y datos.

Para la gestión del conocimiento experto hay diversas metodologías que consisten básicamente en la adquisición, representación e implantación de dicho conocimiento (IC). Estos sistemas suelen usar bases de reglas del tipo “si el paciente tiene los síntomas A, B y C entonces con probabilidad o creencia X tiene la enfermedad E” para almacenar y usar este conocimiento para inferir nuevos consejos de ayuda en la decisión.

Cuando se tienen en cuenta datos, por ejemplo historias clínicas de los pacientes, casos anteriormente tratados, registros de incidencias de enfermedades, datos de factores que pueden provocar determinadas dolencias (por ejemplo medioambientales, hábitos sociales...), entonces es necesario recurrir a lo que en IA se llaman técnicas de aprendizaje automático o Machine Learning y, por supuesto, técnicas provenientes de la matemática y la estadística, como por ejemplo las de Regresión (formalización de una relación significativa entre dos o más variables para calcular pronósticos a partir del conocimiento de los valores en un individuo concreto).

Muchas de estas técnicas se pueden englobar tanto en estadísticas como de **Machine Learning**: rama de la Inteligencia Artificial en la que se diseñan mecanismos para dotar a los sistemas computacionales de capacidad de aprendizaje, en el sentido de la capacidad de descubrir regularidades (patrones) en datos o situaciones anteriores y aplicarlos a nuevos problemas o situaciones análogas. Se pueden considerar diversos paradigmas y grupos de técnicas en AA:

- **Paradigma Analógico (Aprendizaje por analogía)**

Se pretende encontrar una solución a un problema que se presenta ahora usando el mismo procedimiento utilizado en la resolución de uno similar que se presentó en otra ocasión anterior. Si dos problemas son similares en algún aspecto de su formulación entonces pueden serlo también en sus soluciones. Nuevos problemas pueden ser abordados reduciéndolos a problemas análogos resueltos. Ejemplos: analogía por transformación, analogía por derivación, razonamiento basado en casos, etc.

- **Paradigma Inductivo**

Árboles de decisión, algoritmos de inducción pura, etc.

- **Paradigma Conexionista**

Redes Neuronales Artificiales, etc.

- **Paradigma Evolutivo**

Algoritmos Genéticos, otros métodos de optimización, colonias de insectos, descenso estocástico del gradiente, etc.

- **Modelos gráficos probabilistas**

Bayesianos, cadenas de Markov, Filtros de Kalman, redes de creencia, Máquinas de Soporte Vectorial (SVM), Metaheurísticas, etc.

Técnicas de **Clustering** (aprendizaje no supervisado) consisten en agrupar los elementos de una colección en subconjuntos (clases, categorías, *clusters*), nítidos o borrosos, en base a su similitud. Es no supervisado porque las clases o categorías no se conocen a priori, las determinaran las propias similitudes entre los elementos. Por lo tanto, se centran en una “medida de similitud” entre elementos, de la que puede haber infinidad de variantes: estadísticas, distancias euclídeas, distancias vectoriales (coseno), distancias borrosas, etc. Ejemplos:

- **Paradigma Conexionista**

Redes Neuronales Artificiales: SOM (*Self Organized Maps*, Mapas de Kohonen). Toolbox de Matlab SOM, etc.

- **Modelos estadísticos y probabilistas**

K-means, *c-means*, *K-nearest neighbours* (KNN), *Mean shift* (ventanas circulares con un centroide), *Dirichlet process* (estocásticos basados en distribuciones de probabilidad), LDA (*Latent Dirichlet Allocation*), Modelos Gaussianos, etc.

- **Extensiones basadas en Lógica Borrosa (fuzzy logic)**

Fuzzy K-means, *Fuzzy c-means*, Isodata, etc.

Técnicas de **Clasificación** (aprendizaje supervisado), consisten en asignar una clase a un nuevo elemento en base a un conjunto de categorías previamente establecidas (supervisado), por ejemplo, evaluar los síntomas de un nuevo paciente y decir que tiene gripe (clase previamente establecida). Se basan en un entrenamiento en base a ejemplos con la solución conocida (supervisado) para crear modelos que permitan clasificar nuevos casos análogos:

- **Paradigma Inductivo**

Árboles de decisión: ID3, CART, C4.5, See5, *Random Forest* (de moda en Big Data, introducidos por Leo Breiman en 2001), etc.

- **Paradigma Conexionista**

Redes Neuronales Artificiales: Perceptrón Multicapa (con backpropagation), Convolutivas, Neocognitrones, Redes de Hopfield, Redes recurrentes, Adaline, *Deep Learning* (de moda en Big Data), etc.

- **Modelos estadísticos y probabilistas**

Redes Bayesianas, Naive-Bayes, Máquinas de Soporte Vectorial (SVM), Metaheurísticas, etc.

1.2.7. Adecuación de los métodos a los problemas

La **analítica descriptiva** está orientada a la generación de un resumen claro y fácil de entender de una colección de datos. Este es el fundamento y concepto más básico de todas las estadísticas. Se centra en la **Visualización** de datos para entender el pasado y el presente. Se describen los datos con tablas o gráficos, mostrando descripciones numéricas de la variabilidad y la posición. También se suele llamar **Modelización descriptiva** (ver figura 6).

Para el **análisis predictivo** se suele usar:

- **Extrapolación** de funciones (tendencia para el futuro, pero no hay capacidad de pronóstico, hechos/cambios puntuales).
- **Correlaciones** entre variables (demasiado evidente, no suele funcionar de forma muy fina).
- Encontrar **patrones** en los datos que puedan ser aplicados a situaciones futuras (**KDD** y Minería de Datos).
- Métodos de **CLUSTERING Y CLASIFICACIÓN**.
- **Extrapolación** de funciones (por ejemplo Estimaciones o Líneas de Tendencia).

Dentro del análisis predictivo juega un papel fundamental el **Análisis de Series temporales**. Se suelen clasificar en:

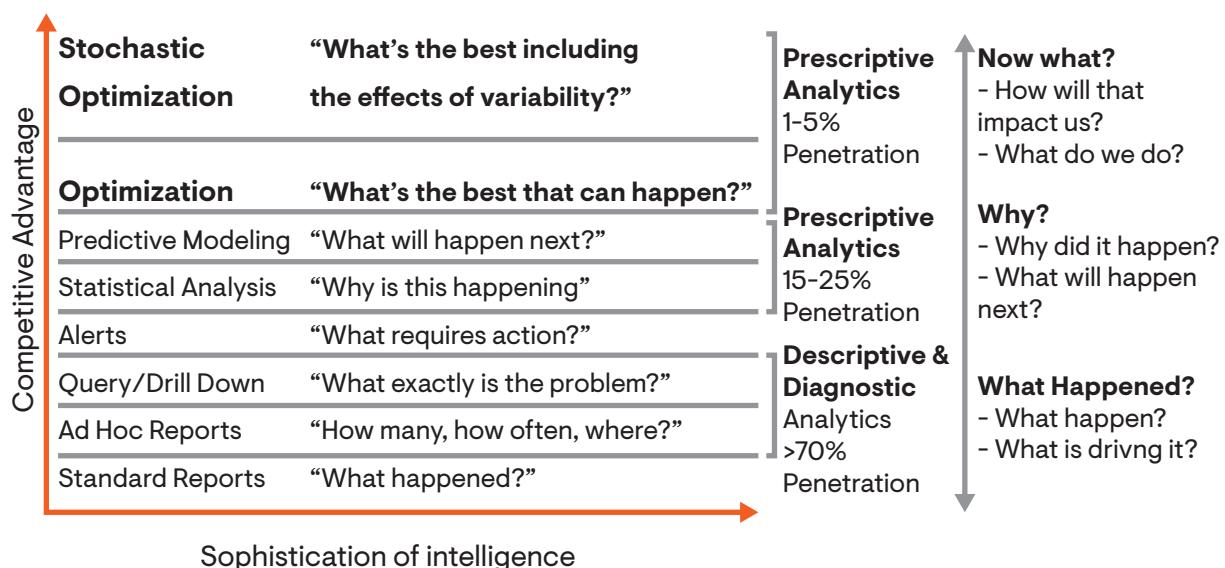
- **Estacionarias** (medias y/o variabilidad se mantienen constantes).
- **No Estacionarias** (medias y/o variabilidad NO se mantienen constantes, cambios de varianza/tendencias).

También es importante señalar qué otros métodos se usan para otras diferentes necesidades en el análisis de series temporales:

- **Tendencias:** método de Mínimos cuadrados. Tendencias evolutivas. Diferenciación estacional.
- **Predicción:** alisadores exponenciales:
 - Alisado exponencial simple
 - Alisado exponencial lineal de Holt
 - Alisado exponencial estacional de Holt-Winters
- **Interpolación:** Predecir datos faltantes.

El análisis predictivo se centra en un escenario futuro. El **prescriptivo** se centra en múltiples alternativas. Por lo tanto, un modelo prescriptivo puede ser considerado como una combinación de modelos predictivos (uno por cada posible escenario) que se ejecutan en paralelo. El objetivo es encontrar la mejor opción posible: **OPTIMIZACIÓN**. Las principales técnicas usadas son:

- Técnicas de Investigación Operativa
- Algoritmos Genéticos
- Técnicas estocásticas
- Metaheurísticas
- Etc.



Source: Competing on Analytics: The New Science of Winning (Davenport/Harris): Gartner

Figura 4. Ventajas competitivas de los diferentes tipos de analítica. Fuente: Davenport & Harris (2007).

1.2.8. Sistemas Basados en Conocimiento (Conocimiento Experto...)

Para analizar datos, casi siempre conviene, si es posible, contar con un experto o varios para que ayuden a “interpretar” la semántica, limitaciones, calidad, objetivos del análisis... de los datos a analizar. Esta interacción con los expertos en los diferentes temas está profusamente estudiada y descrita en la disciplina que suele denominarse “**Ingeniería del Conocimiento**”, que en general se encarga del desarrollo de los Sistemas Basados en Conocimiento (SBC) y en particular de los llamados “Sistemas Expertos”. Por lo general, el desarrollo de un SBC suele ser, en parte, similar al de otros programas convencionales. Por ello, se podría decir que hay una parte donde surgen problemas y dificultades relacionadas con el contexto de los SBC y otra parte en la que las dificultades son comunes al desarrollo convencional de software. No obstante, se va a hacer hincapié en el ámbito que nos ocupa, y así tratar de describir las sucesivas fases que integran el proceso de desarrollo de un SBC, fases que, al estar tan interrelacionadas entre sí, no siempre es posible establecer una clara separación entre las mismas. No existe una metodología única de desarrollo para los distintos tipos de Sistemas Basados en Conocimiento. Sin embargo, puede resultar interesante la elaboración de un esquema genérico aproximado, cuya estructura y orden podría variar dependiendo fundamentalmente del objetivo del programa a diseñar.

En el desarrollo de un SBC siempre se tiene en cuenta la interacción entre los Ingenieros de Conocimiento y los expertos humanos en el dominio. Ahora bien, la actuación de ambos no debería ignorar las aspiraciones y necesidades de una tercera entidad en juego que no es otra que los destinatarios o usuarios finales del programa. El papel protagonista del desarrollo suele corresponder a los dos primeros, pero el éxito o fracaso del sistema dependerá en gran medida de las aportaciones y colaboraciones que se hayan dado entre las tres entidades involucradas. A continuación se describen brevemente las fases genéricas para el desarrollo de un SBC:

Etapa 1: Definición del problema. Identificación

En un primer momento, previo al inicio del proceso de desarrollo de la aplicación informática, es sumamente importante realizar una descripción lo más detallada posible de la cuestión que se va a intentar resolver. Es muy conveniente indicar la misión, objetivos, cuáles van a ser las entradas al sistema, y especificar claramente las salidas del mismo. Es decir, se necesita aclarar desde un principio la situación de partida y el nivel de detalle al que se desea llegar.

Si esta fase no queda definida y acotada con suficiente claridad, es posible que al finalizar el desarrollo el usuario del sistema no quede satisfecho porque lo que está obteniendo como resultados, no le sirven para nada. Por tanto, si desde el principio, se tiene una visión muy clara de cuál es el problema y qué resultados se desean obtener, todo será mucho más fácil tanto para el usuario como para el equipo encargado de realizar el SBC.

Etapa 2: Estudio de viabilidad

Lo que se pretende es comprobar si realmente el SBC va a poder contribuir, de una manera eficiente, a la resolución de un problema permitiendo alcanzar todos los objetivos previstos. No siempre la mejor alternativa para resolver convenientemente un problema tiene que ser de forma obligada un SBC o expresándolo de una manera más genérica, las técnicas de Inteligencia Artificial. Son múltiples los

métodos que se pueden aplicar con el fin de realizar este estudio, pero quizás uno de los más sencillos y precisos es el *método o test de Slagel* de 1989.

Etapa 3: Adquisición del conocimiento

Aunque no de forma exclusiva, casi siempre que dentro de este ámbito se hace referencia expresa a las posibles fuentes de las que se extraerá la experiencia e información, se suele pensar en un especialista o experto como protagonista fundamental del proceso. De hecho, no parece factible plantear la creación de un programa que acumule experiencia y conocimientos pertenecientes a un campo específico del saber sin tener en cuenta, desde el principio, las fuentes de información provenientes de los expertos. La elección de un buen experto como fuente inicial de conocimiento es de suma importancia ya que cuanto mayor sea la calidad de su experiencia mejor tenderá a ser la calidad de los resultados del futuro SBC.

No obstante, se puede disponer del mejor experto del mundo en una materia, pero de nada sirve eso si carece de la capacidad y transparencia necesarias para la transmisión de ideas, argumentos, opiniones, creencias, razonamientos, pensamientos, intuiciones, etc. El Ingeniero del Conocimiento tendrá entonces que ingeníárselas para poder sonsacar la información que necesita del experto. Existen varias técnicas para conseguirlo.

La Adquisición de Conocimientos consiste en la recolección de la información necesaria para construir un Sistema Basado en el Conocimiento a partir de cualquier posible fuente. Esta información puede estar constituida por datos, noticias, conocimientos humanos, etc.

La Adquisición de Conocimientos no debe ser considerada como una etapa dentro de una metodología para la construcción de un Sistema Basado en el Conocimiento sino un proceso paralelo a todas las fases de desarrollo de uno de estos sistemas, ya que cada etapa necesita determinada información, lo que provoca que la recolección de ésta no se haga en un único paso aislado sino en cada una de las etapas. El papel de la Adquisición del Conocimiento en las primeras fases (definición del problema, conceptualización, ...) es fundamental, mientras que en las últimas etapas del desarrollo (implementación, evaluación, mantenimiento) la dedicación a la Adquisición de Conocimiento es mucho menor.

Es posible que la tarea más importante para el desarrollo de un Sistema Basado en el Conocimiento sea la Adquisición del Conocimiento. Pero paradójicamente, este es un campo experimental más que una tecnología, y salvo en el caso de la inducción y aprendizaje automático, la Inteligencia Artificial no aporta métodos completos que solucionen o automaticen esta tarea, solo técnicas para abordar problemas parciales. Por consiguiente, la Adquisición del Conocimiento resulta en la actualidad una labor artesanal, propia para cada caso y dependiente de las personas concretas que estén involucradas en ella. Todo esto provoca que uno de los principales cuellos de botella en el desarrollo de un Sistema Basado en el Conocimiento sea el adquirir los conocimientos necesarios para poder construir sistemas eficientes. La información necesaria puede presentarse de múltiples formas, aunque conviene, en lo que respecta a los Sistemas Basados en el Conocimiento, considerar especialmente una serie de fuentes:

- **Libros y manuales**

Conocimientos básicos, específicos y públicos del dominio y del problema.

- **Documentación formal**

Documentos que contienen políticas, procedimientos, estándares, normas, regulaciones, leyes, etcétera, de un dominio. Este conocimiento también es de carácter público.

- **Documentación informal**

Notas, manuscritos, ayudas de trabajo, etc., proporcionan frecuentemente conocimiento heurístico para la resolución de problemas. Aunque a veces esta documentación es confidencial proporciona conocimientos semipúblicos.

- **Registros internos**

Registros de casos que se presentan, en forma de fichas de clientes, pacientes, estudios, estadísticas, etcétera. Pueden estar en forma escrita o, cada vez más, en forma digital (bases de datos). Además de para la validación y evaluación de los Sistemas Basados en el Conocimiento, esta información debe ser útil para la generación del propio conocimiento del sistema, uno de los objetivos de este curso.

- **Presentaciones**

Material usado para la formación, impartida o recibida. Tiene la ventaja de contener conocimientos expuestos de una forma muy clara.

- **Publicaciones**

Revistas especializadas, actas de congresos, etc.

- **Investigación**

Resultados de las investigaciones que se estén llevando a cabo, en forma de datos empíricos, informes, resultados estadísticos, etc.

- **Visitas**

El Ingeniero de Conocimiento se desplaza a los centros de trabajo del experto y los usuarios, para observar "*in situ*" el *modus operandi*.

- **Conocimiento humano**

Además de las entrevistas con los expertos, resulta imprescindible la interacción con los directivos y usuarios. Los directivos pueden aportar objetivos del proyecto, alcance del sistema, contexto donde irá instalado, etcétera. Los usuarios deben dar claves de interfaces, necesidades, requisitos, etcétera.

El Ingeniero de Conocimiento debe controlar constantemente qué información necesita, con qué profundidad, sobre qué temas, que técnica debe utilizar para adquirir ese conocimiento y otros factores. Muchas veces resulta tentador improvisar, lo que frecuentemente provoca resultados negativos y falta de rigor. Por estas razones, el método que se presenta en este tema contempla básicamente tres grandes bloques:

1. Evaluación de viabilidad, definición del problema, primeras reuniones.
2. Extracción de conocimientos de la documentación (incluidas bases de datos y demás fuentes documentales anteriormente mencionadas).
3. Educción de conocimientos del experto, directivos y usuarios.

De una forma más precisa, los objetivos de la Adquisición de Conocimiento son los siguientes:

1. Comprensión general de la tarea y estructura funcional del Sistema Basado en el Conocimiento.
2. Proceso de razonamiento del experto y pasos en la resolución de problemas.
3. Datos necesarios para resolver un problema determinado, con los valores que pueden tomar.
4. Desarrollo de un modelo conceptual.

Además de los métodos usuales de Adquisición de Conocimiento: Métodos automáticos (*Machine Learning*), a partir de ejemplos, inducción. Análisis estructural de textos. Entrevistas (abiertas, estructuradas). Observación de tareas habituales. Clasificación de conceptos. Cuestionarios. Análisis de protocolos. Emparrillados (*Repertory Grids*). Técnicas para educación en grupo. Método Delphi, se usa KDD y Minería de Datos para manipular grandes volúmenes de datos, con el fin de proporcionar información útil y conocimiento a partir de los mismos.

Etapa 4: Representación del conocimiento

Suele ser una de las tareas que más tiempo y esfuerzo demandan de las personas que se dedican al diseño de SBCs. Ello se debe al hecho de no conocer aún con la suficiente precisión cuáles son los procesos que cualquier ser humano y, en particular un experto, activa en su mente cuando selecciona, examina, sintetiza y transforma los datos iniciales de un problema para alcanzar una solución válida.

La función de cualquier esquema de representación es capturar los rasgos esenciales del ámbito correspondiente a un problema concreto y hacer accesible esta información a un procedimiento de resolución específico. Como se sabe, diferentes tipos de problemas requieren diferentes tipos de razonamiento. A su vez, cada modalidad de razonamiento precisa de una adecuada representación del conocimiento.

Etapa 5: Implementación

En esta fase se lleva a cabo el desarrollo íntegro del SBC en términos de programación. Se suelen construir prototipos que serán confrontados con el experto y rediseñados en numerosas ocasiones. Así se sigue una metodología incremental y cíclica con la que el sistema se va refinando hasta conseguir

optimizarlo. También es importante decidir el lenguaje de programación utilizado para la implementación dependiendo de factores como los siguientes: requerimientos de tiempos de respuesta, requerimientos de Interfaz de usuario, flexibilidad que se quiere aportar a la herramienta, requerimientos de hardware, coste en mantenimiento de la herramienta y facilidades aportadas al usuario.

1.3. El proceso KDD (*Knowledge Discovery in Databases*)

Fayyad y sus colaboradores (Fayyad, Piatetsky-Shapiro, Smyth, 1996) definen el proceso KDD como “*El proceso no trivial de identificar válidos, nuevos, potencialmente útiles y comprensibles patrones en datos*”, utilizando técnicas multidisciplinares y viendo cómo actúan juntas. El término *proceso* implica que hay varios pasos, como la preparación de datos o la búsqueda de patrones. El término *patrón* se puede referir al comportamiento regular de los datos, junto con una descripción y un modelo aplicable al mismo. Los *patrones* descubiertos deben ser válidos para nuevos casos con cierto grado de certeza. Se necesita que estos patrones sean nuevos, al menos para el sistema y preferiblemente para el usuario y potencialmente útiles. Por último, estos patrones deben ser comprensibles, todo ello si no es inmediatamente, lo debe cumplir después de un postprocesado. Esta definición implica que deben definirse medidas de la bondad de los patrones, en muchos casos es posible definir medidas de certeza (capacidad de clasificación de nuevos datos) o utilidad (calidad de las predicciones en base a estos patrones).

El proceso KDD descrito por Fayyad y sus colaboradores, interactivo e iterativo, es el que refleja la figura 5:

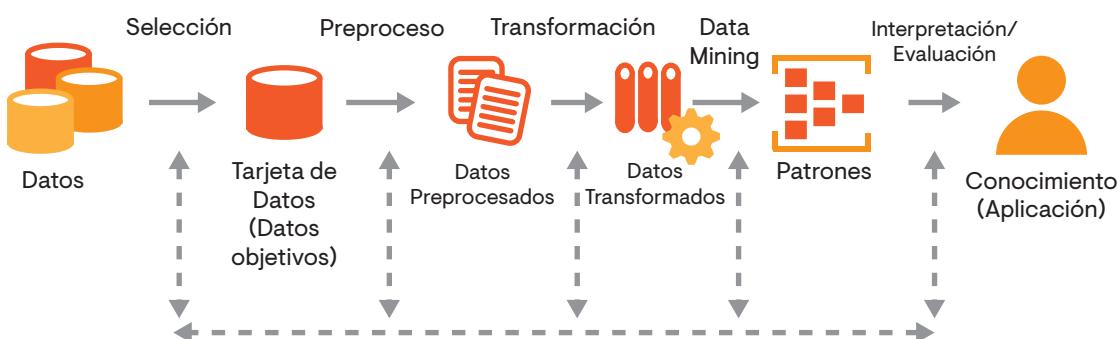


Figura 5. Proceso KDD, *Knowledge Discovery in Databases*. Adaptado de: Fayyad et al. (1996).

1.3.1. Selección: Datos objetivo/Tarjeta de Datos

Aplicando el conocimiento del dominio y el conocimiento relevante *a priori*, teniendo en cuenta los objetivos del proceso global del KDD se crea una tarjeta de datos (datos objetivo) que incluirá conjuntos seleccionados de datos o subconjuntos de variables relevantes o ejemplos.

Habitualmente se lleva a cabo un análisis previo, prospección para verificar la viabilidad de las hipótesis iniciales. Es decir, se comprueba que los datos de los que se dispone son adecuados para buscar los

patrones que se pretende encontrar (hipótesis inicial). Si no se verifica este extremo no tiene sentido continuar con el proceso KDD bajo la hipótesis establecida.

1.3.2. Preproceso: Limpieza de Datos

Limpieza de datos, eliminación de ruidos, manejo de campos vacíos, datos perdidos (se suele hablar de datos ausentes o *missing values* y datos perdidos por el sistema o *system missing values*), valores desconocidos o por defecto, evolución de datos, **outliers** e **inliers**. Se aplican técnicas estándar de bases de datos.

En situaciones de incertidumbre, es decir, cuando faltan datos (en el sentido de valores en campos que hacen que los registros estén incompletos), se debe plantear la posibilidad de rellenarlos con **valores prototípicos** (por ejemplo, si la base de datos es de muebles y un campo es el número de patas, ante la ausencia de valores podría llenarse con “4”, que es un número prototípico de patas para una silla) o **valores por defecto** (en una base de datos de pacientes-síntomas, ante la ausencia de valor para “fiebre” se podría asumir que el paciente NO tenía fiebre como valor por defecto). Esta decisión puede llevarnos a introducir valores erróneos (sillas de 3 patas o pacientes que tenían fiebre y les hemos puesto que no). Es tarea del analista de datos decidir si compensa el nivel de error añadido la ganancia que puede generar este relleno.

En la etapa de preproceso también se llevan a cabo las labores de agregación de datos. Por ejemplo, en una base de datos de clientes de un banco, ante el desarrollo de un sistema de ayuda a la decisión en la concesión de créditos bancarios, es posible que convenga sumar los ingresos mensuales de los clientes (y los gastos) en vez de trabajar con todos los movimientos producidos en las cuentas.

1.3.3. Transformación: del *Clustering* a la Clasificación

Reducción del número de variables. Localización de formas útiles para expresar los datos dependiendo del uso posterior que se les va a dar y de los objetivos del sistema. Se usa el conocimiento experto y técnicas de transformación e informes en bases de datos.

Es quizá la fase más crítica y relevante de todo el proceso. Se suele usar inicialmente un algoritmo de *clustering* para agrupar elementos similares, normalmente buscando un número de *clusters* compatible (o coincidente) con la granularidad esperada en las decisiones a tomar. Por ejemplo, ante una base de datos de clientes de un banco, con el objetivo de desarrollar un sistema de ayuda a la decisión en la concesión de créditos bancarios, es muy probable que lo más adecuado sea establecer tres grupos (*clusters*): el de los “buenos pagadores” a los que se les concedieron los créditos, el de los “malos pagadores”, que no se les concedió y los “dudosos” a los que se les pidieron mayores garantías.

Otro aspecto importante es que en muchos casos es necesaria una transformación grande de la base de datos, convertirla en otra totalmente diferente. Por ejemplo, en sistemas en los que se quiera predecir o pronosticar a través del uso de series temporales. Normalmente la base de datos inicial debe estar compuesta de registros individuales (por ejemplo ocurrencias de terremotos) y nos puede interesar construir series temporales de terremotos anteriores a uno “grande” para tratar de encontrar patrones anteriores que permitan establecer un modelo de pronóstico. Esto supone haber partido de una base de datos donde cada registro era una ocurrencia y haberla transformado en otra en la

que cada registro es una serie temporal. Aquí también se agruparían estas series temporales según su similitud con el fin de encontrar regularidades o patrones de “evolución sísmica previa a un gran terremoto”.

Estos casos se ilustran de una forma muy detallada en el ejemplo de aplicación a la prevención de incendios forestales que se describe al final de este documento.

Por último, es importante señalar que una vez agrupados los elementos, se deben “etiquetar” estos clusters, con el fin de convertir este proceso de *clustering* en uno de clasificación. Es decir, en mi sistema de ayuda a la decisión en la concesión de créditos bancarios, al *cluster3* que me generó el algoritmo de *clustering* usado, le asigno el nombre de “buenos pagadores”, al *cluster2* el de “dudoso”, etc. con lo que podría transformar la base de datos inicial de clientes en otra con un campo más en cada registro que diga si ese cliente es “buen pagador”, “dudoso”... dependiendo del *cluster* al que pertenezca. Así, hemos convertido este proceso de *clustering* (no supervisado) en uno de clasificación (supervisado). Ahora, por ejemplo se podría generar un árbol de decisión que nos permitiese clasificar a un nuevo cliente según sus características en “buen pagador”, “dudoso”...

Esta idea tiene una repercusión importante: habitualmente (casi siempre, salvo raras excepciones) no basta con usar un único método o algoritmo para llevar a cabo un proceso completo de KDD. Es necesario primero usar *clustering*, no supervisado, para agrupar los datos según su similitud (*y transformarlos*) y después usar uno de clasificación supervisado (*data mining*) para poder usarlo en la toma de decisiones (segmentación, predicción...).

1.3.4. Minería de Datos

Elección de los algoritmos de Minería de Datos. Decisiones acerca del modelo que se deriva del algoritmo de Minería de Datos elegido (clasificación, resumen de datos, predicción). Búsqueda de patrones de interés, en cuanto a clasificación, reglas o árboles, regresión, clasificación, dependencia, heurísticas, incertidumbre.

1.3.5. Conocimiento: formalización de patrones

Dentro del proceso KDD descrito, adquiere especial relevancia el paso de Minería de Datos para determinar los patrones de los datos observados. La elección de los modelos a utilizar tiene una componente fundamental de conocimiento de los expertos, supervisado por el Ingeniero de Conocimiento. En la literatura, véase por ejemplo (Berry, Linoff, 1996), se describe un gran número de algoritmos y técnicas de Minería de Datos. En este tema se introducen fundamentalmente los provenientes de la estadística y el aprendizaje automático. En particular, los algoritmos de Minería de Datos consisten en una mezcla específica de tres componentes (Fayyad et al., 1996):

El modelo

Un modelo contiene parámetros que son determinados a partir de los datos. Dos factores relevantes:

1. La función

En la práctica de la Minería de Datos se utilizan funciones de **Regresión** (comparan datos con valores reales de las variables establecidos mediante predicción), **Dependencia** (describen dependencias significativas entre variables), **Análisis de relaciones** entre campos de las bases de datos, pero las más relevantes para la predicción (como evaluación de una situación mediante la comparación de casos reales con situaciones prototípicas) son las siguientes:

- Funciones de **Clasificación**: categorizan un registro dentro de una de varias clases predefinidas.
- Funciones de **Clustering**: categorizan un registro dentro de una de varias clases (*clusters*), pero a diferencia de la clasificación, las clases las determinan los propios datos, mediante agrupamientos naturales basados en medidas de afinidad, similitud o probabilidad.
- Funciones de **Resumen**: generan una descripción compacta de un subconjunto de datos. Se puede usar un simple ejemplo como media y las desviaciones estándar para todos los campos.
- Funciones de **Análisis de secuencias**: modelizan patrones secuenciales, series temporales. El objetivo es generar la secuencia de estados del proceso que se trata de modelizar.

2. La representación

Se utilizan modelos clásicos, como árboles de decisión, reglas, modelos lineales y no lineales, métodos basados en ejemplos (razonamiento basado en casos), redes bayesianas, modelos borrosos, redes neuronales, algoritmos genéticos, etc. El modelo de representación determinará la flexibilidad de representación de los datos e interpretación del modelo desde el punto de vista humano. Normalmente, cuanto más complejo sea el modelo de representación, es posible que maneje mejor los datos, pero hará que sea mucho más difícil entenderlos. Aunque a nivel de investigación se tiende a utilizar modelos complejos, en aplicaciones reales se utilizan mayoritariamente modelos simples debido a su robustez e interpretabilidad.

El criterio de preferencia

Es aquel que permite seleccionar un modelo o conjunto de parámetros en función de unos datos determinados. Es, en cierto modo, una medida de bondad de la relación entre el modelo y los datos. Normalmente este criterio es explícito y cuantitativo en el algoritmo de búsqueda (por ejemplo, el criterio de encontrar los parámetros que maximicen la probabilidad de algunos datos observados). También suele haber un criterio subjetivo (por parte del Científico de datos o del Ingeniero de Conocimiento) implícito, sobre qué modelos se ha de tener en cuenta inicialmente.

El algoritmo de búsqueda

Es necesario especificar un algoritmo para encontrar modelos particulares y parámetros e incluso criterios de preferencia. En los de búsqueda de parámetros es frecuente que, dado un modelo, el problema de encontrar los mejores parámetros se reduzca a un problema de optimización. En Minería

de Datos se suelen utilizar técnicas de optimización sencillas (por ejemplo, el *descenso del gradiente*) que presentan los problemas de máximos y mínimos locales y otros muy estudiados en el ámbito de la Inteligencia Artificial.

1.3.6. La metodología CRISP-DM

Hay otras metodologías de minería de datos muy conocidas y usadas. Por ejemplo: el *SAS Institute* propuso SEMMA (*Sample, Explore, Modify, Model, Assess*) y, mucho más conocida y usada, a comienzos de siglo se propuso el *Cross-Industry Standard Process for Data Mining*, más conocido como CRISP DM, un modelo estándar y abierto muy usado, sobre todo a nivel empresarial para el desarrollo de proyectos de Minería de Datos.

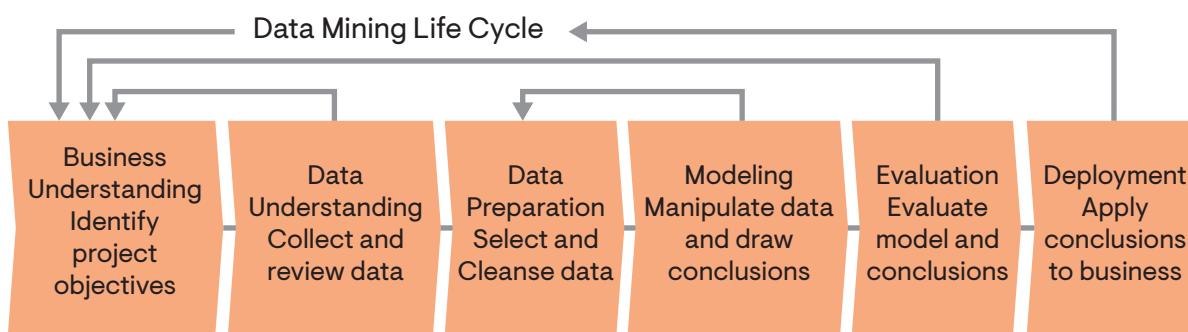


Figura 6. Proceso CRISP-DM, Cross-Industry Standard Process for Data Mining. <https://exde.wordpress.com/2009/03/13/a-visual-guide-to-crisp-dm-methodology/>, 2009.

El estándar incluye un modelo y una guía estructurados en seis fases, algunas de estas fases son bidireccionales, lo que significa que permitirán revisar parcial o totalmente las fases anteriores.

1. **Comprensión del negocio:** objetivos y requerimientos desde una perspectiva no técnica.
 - **Establecimiento de los objetivos del negocio:**
 - Antecedentes
 - Contexto inicial
 - Objetivos del Negocio
 - Criterios de éxito
 - **Evaluación de la situación:**
 - Inventario de requerimientos de Recursos, Hipótesis y Limitaciones
 - Riesgos y Contingencias
 - Terminología

- Costos y Beneficios
 - **Establecimiento de los objetivos de la Minería de Datos:**
 - Objetivos de Minería de Datos
 - Criterio de Éxito de la Minería de Datos
 - **Generación del plan del proyecto:**
 - Plan de proyecto
 - Evaluación inicial de herramientas, equipo y técnicas
- 2. Comprensión de los datos:** familiarizarse con los datos teniendo presente los objetivos del negocio. Identificar las características de calidad de los datos e Identificar los resultados iniciales obvios.
- **Obtener los datos iniciales:**
 - Informe de la obtención de los datos iniciales
 - **Descripción de los datos:**
 - Informe con la descripción de los datos
 - **Exploración de los datos:**
 - Informe de la exploración de datos
 - **Verificación de calidad de datos:**
 - Informe de la calidad de los datos
- 3. Preparación de Datos:** selección y limpieza de datos. Obtener un subconjunto o vista aprovechable.
- **Selección de los datos:**
 - Justificación de la inclusión/exclusión de determinados datos
 - **Limpieza de datos:**
 - Informe de la Limpieza de Datos
 - **Construcción de datos:**
 - Atributos derivados
 - Registros generados

- **Integración de datos:**
 - Datos combinados
 - **Formateo de datos:**
 - Datos Formateados
- 4.** Modelado de Datos: aplicar las técnicas de minería de datos. Implementación en herramientas de Minería de Datos
- **Selección de la técnica de modelado:**
 - Técnica de modelado
 - Modelado
 - Hipótesis
 - **Diseño de la evaluación:**
 - Diseño de pruebas
 - **Construcción del modelo:**
 - Configuración de los parámetros del Modelo
 - Descripción del Modelo
 - **Evaluación del modelo:**
 - Evaluación del Modelo
 - Revisión de la configuración de los parámetros del modelo
- 5. Evaluación:** Determinar si los resultados coinciden con los objetivos del negocio. Identificar los temas de negocio que deberían haberse abordado.
- **Evaluación de resultados:**
 - Hipótesis de Minería de Datos
 - Resultados.
 - Criterio de éxito del negocio
 - Modelos aprobados

- **Revisión del Proceso**
- **Establecimiento de los siguientes pasos o acciones:**
 - Lista de posibles acciones
 - Decisión
- 6. **Despliegue:** explotar la utilidad de los modelos, integrándolos en las tareas de toma de decisiones de la organización. Configuración para minería de datos de forma repetida o continua.
- **Planificación de despliegue**
- **Planificación de la monitorización y del mantenimiento**
- **Generación del informe final**
- **Revisión del proyecto:**
 - Documentación de las experiencias

Esta metodología CRISP-DM es muy similar, en esencia, a la de KDD. Aunque han surgido muchas variaciones e instancias de estas metodologías, en lo esencial siguen siendo muy similares.

1.3.7. Herramientas actuales (en entornos Big Data) para implementar estas soluciones

El crecimiento exponencial de generación de datos en todos los ámbitos de la ciencia y la sociedad ha dado lugar al fenómeno denominado Big Data: capacidad de gestionar (almacenar, acceder, procesar, interpretar, transmitir, analizar...) grandes volúmenes de datos que exceden las capacidades de lo que podríamos denominar “software convencional” para hacerlo. Estos datos pueden proceder de muy diversas fuentes como sensores (IoT *Internet of Things*, *Smart Cities*...), redes sociales, bases de datos de pacientes, estudios epidemiológicos... Son de naturaleza heterogénea, lo que hace necesario usar nuevas herramientas para poder gestionarlos, frente a lo que habitualmente se almacenaba en bases de datos relacionales (estructuradas) tradicionales. Como se ha comentado, se suele hablar de “*data lake*” refiriéndose a un repositorio para grandes cantidades y variedades de datos, tanto estructurados como no estructurados. Consisten en nuevos modelos de bases de datos, como *mongoDB*, *InfiniteGraph*, *amazonDynamoDB*, *apacheHBASE*, *Cassandra*, etc. que permiten cumplir estas funciones.

Para ello, hoy en día hay diversos “ecosistemas” Big Data, como *Spark* (apache) o los de *Google*, IBM, *amazon* o *Microsoft*, que permiten combinar bases de datos avanzadas con herramientas de procesado. Para visualización son muy usadas herramientas como QlikView (enlace 1), Microsoft power BI (enlace 2) o Tableau (enlace 3). Pero, como se ha dicho, se debe tender a la predicción y prescripción, como proponen empresas como Microstrategy (enlace 4). Aunque existen muchas herramientas que lo pretenden (aquí se puede ver un análisis de las 52 mejores)(enlace 5), la única forma de hacerlo “a

medida” para sistemas de ayuda a la decisión tan complejos es acudiendo a la figura del “científico de datos” (**data scientist**), que, como se describe en este documento, debe poseer conocimientos de computación, bases de datos, Inteligencia Artificial, Aprendizaje Automático, estadística, visualización, reconocimiento de patrones, KDD y Minería de Datos... y es una figura muy demandada y escasa en ambientes profesionales, científicos o académicos, que debe ser capaz de seleccionar y guiar las herramientas y técnicas más adecuadas para cada problema y objetivos concretos.

Por último, es importante resaltar que en los entornos de Big Data se suelen usar tanto lenguajes de programación específicos (R, Scala, Python...) como librerías que permiten implementar todas las técnicas que se describen. Entre ellas, podemos destacar Mahout (*Scalable Machine Learning and data mining*) (enlace 6), MLlib (*Apache Spark's scalable machine learning library*) (enlace 7) o H2O (enlace 8).

Se profundiza en estos aspectos en otros temas/asignaturas de este Programa.

**Enlace 1****QlikView**<http://global.qlik.com/>**Enlace 2****Microsoft power BI**<http://powerbi.microsoft.com/>**Enlace 3****Tableau**<http://www.tableau.com>**Enlace 4****Microstrategy**<https://www.microstrategy.com/es>**Enlace 5****Top 52 predictive analytics & prescriptive analytics software**<https://www.predictiveanalyticstoday.com/top-predictive-analytics-software/>**Enlace 6****Mahout**<http://mahout.apache.org/>**Enlace 7****MLlib**<https://spark.apache.org/mllib/>**Enlace 8****H2O**<https://www.h2o.ai/>

1.4. Algunas aplicaciones/ejemplos

A continuación se muestran varios ejemplos de aplicaciones de procesos de **Minería de Datos**, **Minería de Textos** y **Minería de Opiniones**.

1.4.1. Minería de Datos: Prevención de Incendios Forestales

Se describe un proceso completo de KDD-Minería de datos, presentado inicialmente en la tesis doctoral del autor (Olivas, 2000). Se parte de una base de datos de en torno a 100.000 registros con unos 40 campos que contienen datos sobre cada uno de los incendios forestales producidos en Galicia en los años anteriores a la fecha de desarrollo del trabajo, en particular se considerarán unos 12.000 registros de los años 1991 y 1992 (están los de todos los años).

Adquisición de Conocimiento y datos

El proceso de Adquisición del Conocimiento es fundamental para la validez real de los métodos que pretenden servir para la resolución de problemas de ayuda en la toma de decisiones de este tipo. Aunque se describen con precisión estos procesos a lo largo de los diferentes apartados, parece adecuado abordar previamente ciertos aspectos.

1. Normalmente, la aparición de incendios forestales en la comunidad gallega está ligada a **factores socio-económicos**, difíciles de cuantificar de una forma precisa, que no se reflejan en los datos estadísticos, ni en los denominados “informes” o “partes”.
2. La **medida de factores físicos o meteorológicos no es lo precisa que podría ser deseable**. Hay demasiados microclimas, especies arbóreas en poca superficie, pequeños cultivos, etc. como para que características como “humedad relativa” o “estrés hídrico” puedan ser generalizadas a grandes zonas de una forma fiable.
3. **En los datos estadísticos disponibles, existe un sesgo** fundamental: no reflejan la auténtica peligrosidad e importancia de los incendios ocurridos.

Esto es debido a lo siguiente: supóngase la aparición de dos alarmas simultáneas en una misma comarca, una de ellas en una zona de matorrales que habitualmente se utiliza para pastoreo y en la que periódicamente se realizan quemas controladas, sin riesgo por proximidad a una zona de especial valor ecológico, y la otra alarma surge a pocos metros de un bosque con difícil accesibilidad. Es razonable y habitual, en el caso de que los recursos sean limitados, destinar más medios al segundo caso que al primero, lo que provoca en muchas ocasiones que en las estadísticas posteriores se refleje como importante el primer incendio (ej. 200 Ha. de matorral), y el segundo incluso ni aparezca (el umbral de aparición en estadísticas fue de 0,5 Ha. y posteriormente de 0,1 Ha.), porque se ha atajado en sus inicios.

4. Los **criterios de prioridad**, por ejemplo, para el envío de medios aéreos, no siempre están claros, porque en los puntos donde se toman estas decisiones, la información de la que se dispone no es lo suficientemente objetiva ni completa.

Teniendo en cuenta estos criterios, se ha realizado un análisis previo y “superficial” de todos los “ciclos” de incendios de una comarca durante varios años (Ejemplos en fig. 7) y se ha observado que la evolución de la siniestralidad puede ser representada como una función creciente de tipo sigmoidal dividida en tres sectores y que comenzaría en el día posterior a un periodo de lluvia en el que el número de los incendios se ha reducido a cero.

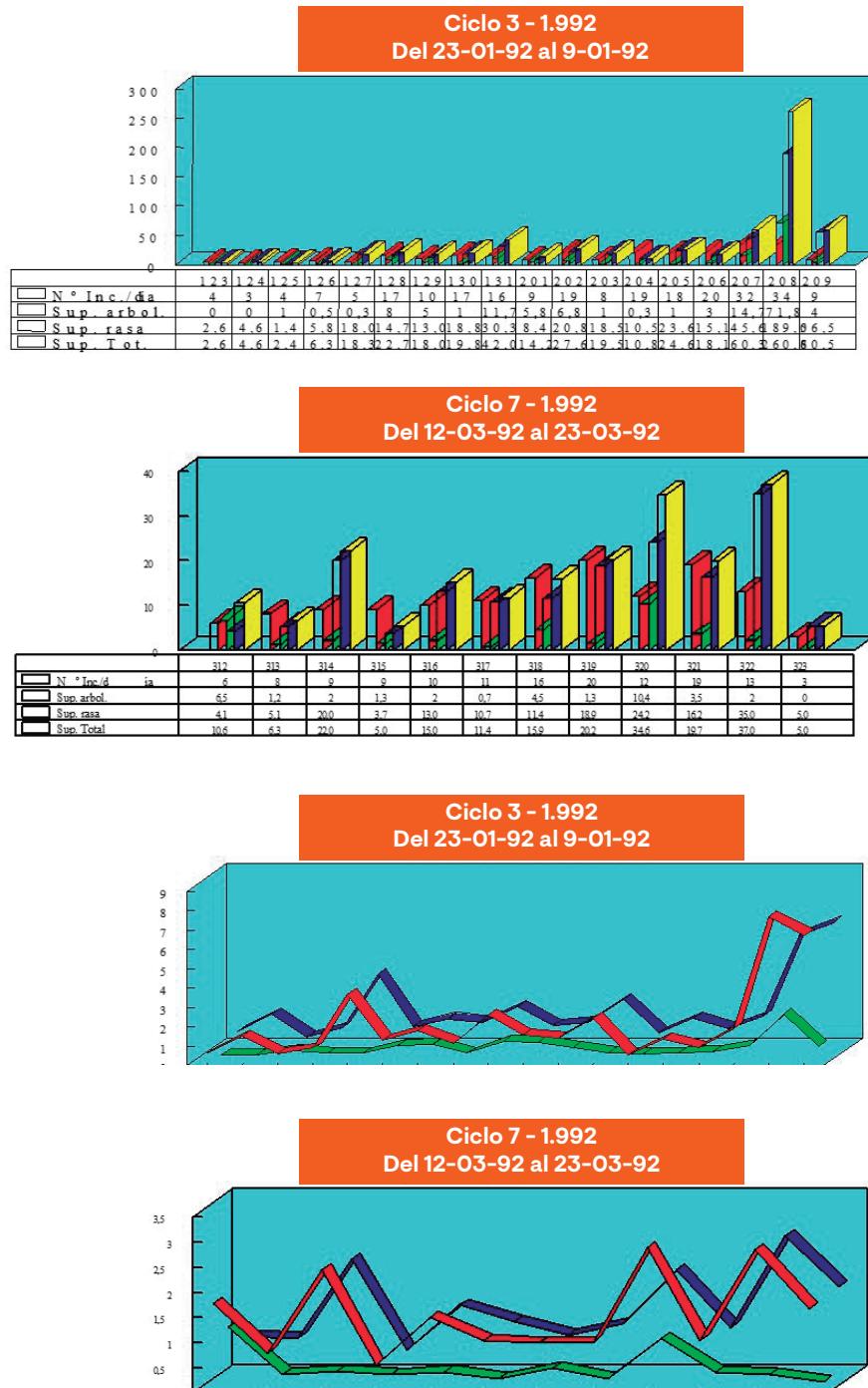


Figura 7. Análisis previo, prospección para verificar la viabilidad de las hipótesis iniciales. Fuente: elaboración propia.

Este patrón de crecimiento se repite de forma cíclica después de cada periodo de este tipo, pero puede sufrir modificaciones debido a factores específicos.

Entiéndase por siniestralidad la combinación de varios factores, como número, peligrosidad de los incendios, etc. La figura 8 muestra la representación del patrón de crecimiento.

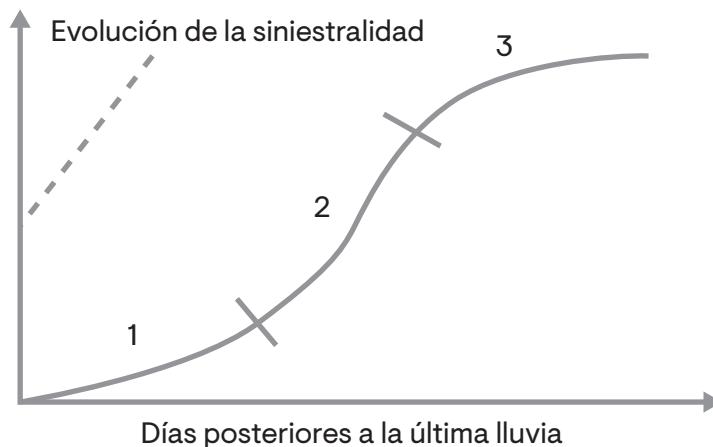


Figura 8. Representación de un posible patrón de evolución de incendios forestales en Galicia. Fuente: elaboración propia.

- **El primer sector representa un lento crecimiento de la siniestralidad en los días inmediatamente posteriores al periodo de lluvias.**
- **El segundo sector expresa un alto crecimiento de la siniestralidad, especialmente en cuanto al número de incendios.**
- **En el tercero se quiere representar una estabilización en cuanto al número, pero un progresivo aumento en la peligrosidad de los fuegos.**

En nuestra vida cotidiana es usual asociar un hecho o conjunto de hechos con un **patrón aprendido**, de tal forma que el patrón interpreta la situación y de él dependen las acciones que llevemos a cabo. Por ejemplo, si estamos conduciendo y empieza a granizar adecuamos nuestra conducción al esquema de conducción bajo condiciones potencialmente peligrosas que nuestra experiencia ha forjado en nuestro conocimiento.

Como se ha visto en el ejemplo, muchas de las acciones que realizamos en nuestra vida dependen de una interpretación. Lo que aquí se plantea es que interpretar una situación es encontrar en los datos los **patrones o prototipos afines** a las circunstancias del problema. En este caso se trata de simular la capacidad del experto para interpretar la situación, es decir, para encontrar el modelo de evolución de la siniestralidad de los incendios más adaptado a las circunstancias reales.

Concepto y Prototipos (la importancia del Científico de Datos)

La importancia del Científico de Datos se pone de manifiesto cuando es necesaria (casi siempre) una visión multidisciplinar del análisis del problema, buscando, en muchos casos, soluciones “**sofisticadas**” (desde el punto de vista de la Inteligencia Artificial), porque con el uso de herramientas y técnicas

estándar simples la aproximación a muchos problemas de análisis de datos reales puede ser “trivial” y no proporcionar resultados relevantes.

Por ejemplo, tomando como marco de referencia la **teoría de prototipos** de la **psicología cognitiva**, podría entenderse que esta representación es prototípica del avance de la siniestralidad de los incendios. Sin embargo, en el proceso de adquisición del conocimiento se pudo observar que esta representación simplifica en exceso las pautas del comportamiento de los expertos. Cuando un técnico se enfrenta a una situación real maneja un abanico de prototipos determinados por una serie de factores, es decir, debe decidir qué tipo de evolución de la siniestralidad es previsible. Dicho de otro modo, el prototipo “**Evolución de la siniestralidad**” no es único, sino que existen diferentes formas de evolución dentro de la misma estructura sigmoidal.

El profesor Lotfi A. Zadeh, creador de la “**Lógica Borrosa**”, aludía a las **teorías clásicas de prototipos** desde el punto de vista de la psicología (Zadeh, 1982), criticando precisamente lo que aquí se ha expuesto: su falta de adecuación a la función que debe cumplir un prototipo. La aproximación de Zadeh a lo que debe entenderse por prototipo es menos intuitiva que las concepciones de las teorías psicológicas, pero más racional, más próxima a lo que en un examen detenido muestra el significado de un concepto prototípico. En el blog de la VIU se puede leer el comentario “Zadeh (D.E.P), la Lógica Borrosa y el Análisis de datos masivos” (enlace 9).

Enlace 9

Zadeh (D.E.P), la Lógica Borrosa y el Análisis de datos masivos

<https://www.universidadviu.es/zadeh-d-e-p-la-logica-borrosa-analisis-datos-masivos/>

En nuestro caso se ha observado que la idea de Zadeh sugiere que un concepto engloba un conjunto de prototipos, los cuales representan la buena, baja o media compatibilidad de los ejemplares con el concepto. Las Categorías Prototípicas Borrosas representan las diferentes clases que se pueden determinar en el dominio.

Desde este punto de vista se puede hablar de “siniestralidad altamente progresiva”, “medianamente progresiva” o “escasamente progresiva”. Esto se puede representar, tal y como refleja la figura 9, con tres sigmoides, entendidas simplemente como representaciones gráficas de los tres prototipos borrosos.

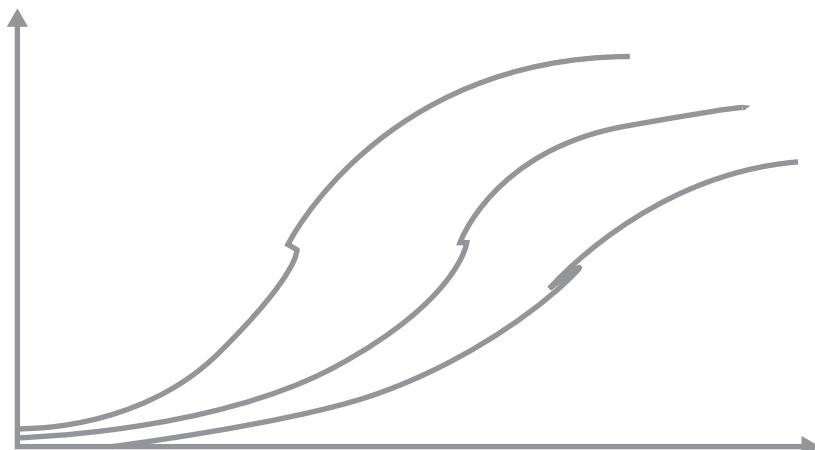


Figura 13. Representación de los tres patrones de evolución. Fuente: elaboración propia.

KDD: Descubrimiento de Conocimiento Prototípico Borroso en datos sobre Incendios Forestales

El proceso KDD, en este caso orientado al Descubrimiento Prototípico para este caso particular de los datos disponibles sobre incendios forestales se lleva a cabo en las siguientes fases:

1. **Selección:** aplicando el conocimiento del dominio y el conocimiento relevante a priori, teniendo en cuenta los objetivos del proceso global de KDD se crea una tarjeta de datos (Datos objetivo) que incluye conjuntos seleccionados de datos o subconjuntos de variables relevantes o ejemplos. Se toma como conjunto de partida una base de datos relacional que contiene aproximadamente 12.000 incendios ocurridos en Galicia durante los años 1991 y 1992. Se seleccionan estos años por ser 1991 un año poco conflictivo y 1992 uno muy conflictivo. Se seleccionan los 3.204 correspondientes a la comarca de Lugo, compuesta por siete municipios, de los años 1991 y 1992. También se separan por comarcas y se eliminan campos no relevantes.
2. **Preproceso:** limpieza de datos, eliminación de ruidos, manejo de campos vacíos, datos perdidos, valores desconocidos o por defecto, evolución de datos. Se aplican técnicas estándar de bases de datos.
3. **Transformación:** reducción del número de variables, localización de ciclos de incendios. Búsqueda de formas útiles para expresar los datos dependiendo del uso posterior que se les va a dar y de los objetivos del sistema. Se usa el conocimiento experto y técnicas de transformación e informes en bases de datos. Ordenación y clasificación de los ciclos según la evolución de la siniestralidad.
4. **Data Mining:** elección de los algoritmos de Data Mining. Decisiones acerca del modelo que se deriva del algoritmo de Data Mining elegido (clasificación, resumen de datos, predicción). Búsqueda de patrones de interés, en cuanto a clasificación, reglas o árboles, regresión, clasificación, dependencia, heurísticas, incertidumbre. Se generan los prototipos de evolución de la siniestralidad en base a los ciclos representativos de cada una de las clases.

A continuación se muestran con detalle las operaciones de Preproceso, Transformación y Data Mining.

Preproceso. Eliminación de ruido

Una vez extraídos de la base de Datos de Galicia los 3.204 incendios correspondientes a los años 1991 y 1992 de la comarca de Lugo, se debe proceder a su estudio completo, antes habrá que hacer unas modificaciones tanto en el diseño como en el contenido de la tabla ya que presenta diversas irregularidades que hacen imposible o muy dificultoso el aprovechamiento de estos datos.

1. Tratamiento de las Fechas y Horas

La base de datos contiene el control de las fechas y las horas en que los incendios se han iniciado, se han controlado y se han extinguido. Estos datos son muy importantes ya que permiten el control de la evolución temporal de los incendios y la obtención de datos estadísticos sobre cada día, semanas... En la base de datos inicial estos campos estaban

tipificados como simples cadenas de caracteres con un cierto formato, lo cual facilitaba en un primer momento su almacenamiento pero dificulta enormemente su posterior tratamiento ya que pierden toda su semántica propia, no pudiéndose efectuar sobre ellos operaciones como comparaciones, restas o distancias.

La solución a este problema es sencilla, siempre y cuando las fechas y horas hayan sido almacenadas con algún formato estándar, lo que en este caso ocurre, de esta forma cambiando el tipo de los campos en el sistema Gestor de Base de Datos utilizado (Microsoft Access '97) se permite que las aplicaciones accedan y operen sobre estos datos con toda la semántica correspondiente a los tipos Fecha y Hora (CTime en Visual C++):

Campo afectado	Formato	Descripción
FECHA_INI	DD/MM/AA	Fecha de inicio del incendio
HORA_INI	HH:MM:SS	Hora de inicio del incendio
FECHA_EXT	DD/MM/AA	Fecha de control del incendio
HORA_EXT	HH:MM:SS	Hora de control del incendio
FECHA_FIN	DD/MM/AA	Fecha de la extinción total
HORA_FIN	HH:MM:SS	Hora de la extinción total

Tabla 1. Gestor de base de datos. Fuente: elaboración propia.

2. Pérdida de datos numéricos

Todos los campos o variables estadísticas que se manejan poseen un conjunto de valores perdidos, es decir, el valor o los valores que no se consideran como válidos para la variable con la que se está trabajando, no existiendo por defecto valores ausentes.

Todos los valores numéricos existentes en esta base de datos son indispensables para el posterior tratamiento, por ello es necesario que estén bien definidos y se pueda procesar de forma correcta estos valores perdidos.

Existen dos tipos de valores perdidos, los valores omitidos por el usuario (**missing value**) códigos que indican que el verdadero valor de una variable es desconocido y que los casos que contengan esos valores deben ser excluidos del análisis y los valores perdidos por el sistema (**system-missing values**) valores asignados por el SGBD correspondiente cuando un valor de los datos resulta indefinido de acuerdo con el tipo de formato que se ha especificado (como por ejemplo un valor identificativo de menos infinito).

En la tabla de incendios que se maneja y debido en su mayor parte a su procedencia, sucede que los valores numéricos han sido introducidos a mano y sin tener en cuenta que debían de tener un valor por defecto, por lo tanto en la primera parte de la base de datos ocurre que todos los campos están totalmente ocupados por valores que suelen tomar el valor 0 (superficie quemada despreciable, categorías de personal que no estuvieron en el incendio...), el problema surge cuando alrededor de la mitad de la tabla esos valores no aparecen y campos numéricos que deberían tener algún valor no lo tienen, lo cual provoca un cierto caos en su tratamiento ya

que el SGBD utilizado y el controlador de ODBC para comunicarlo con la aplicación coloca en estos valores el de menos infinito ($-9,18 * 10E-19$) lo cual hace imposible un cálculo automático sobre estos valores si no se controla esta contrariedad:

Campo afectado	(Min,Max)	Tipo de Dato	Descripción
SUP_ARBO	(0,30)	real doble	Superficie arbolada arrasada por el incendio
SUP_RASA	(0,60)	real doble	Superficie rasa afectada por el incendio
SUP_TOTAL	(0,90)	real doble	Superficie total quemada por el incendio
TECNICOS	(0,10)	entero	Nº de técnicos que intervinieron en el incendio
AGEN_FOR	(0,10)	entero	Nº de agentes forestales implicados
AUT_CIVIL	(0,15)	entero	Autoridad civil destinada al incendio

Tabla 2. Tabla de incendios. Fuente: elaboración propia.

Debido a la experiencia acumulada en la gestión de la información que generan los incendios se sabe que estos valores son nulos, despreciables o desconocidos por su poca relevancia, debido a esto se ha procedido a que todos estos casos pasen a tener un valor por defecto 0.

3. Malos diseños

El diseño de una base de datos que gestiona la información sobre un cierto asunto (en este caso incendios) debe ser cuidadoso ya que posteriormente se debe acceder a ellos para obtener resultados de todo tipo. Esta es una premisa que se aleja mucho de la realidad de la base de datos con la que aquí se trabaja.

Esta base de datos de una única tabla que gestiona todos los datos necesarios está concebida por su fácil construcción y modificación, sin tener en cuenta para nada los conceptos de modelización y normalización de las bases de datos así como las características esenciales de un buen diseño. Este diseño está más destinado a almacenar de alguna forma sencilla los datos sin ningún interés en una posterior utilización de éstos. Estas situaciones son la pesadilla de las personas dedicadas a la analítica de datos, ya que no se trata de valores perdidos, si no de valores que existen pero no se sabe dónde se localizan.

Esta situación se presenta en nuestro caso cuando se quiere reflejar los recursos personales y mecánicos que estuvieron involucrados en la extinción de cierto incendio:

- Para medios humanos: 20 campos de tipo texto con los identificadores de cada uno de los recursos. Solo se ocupan los campos, por orden, según el número de recursos asignados, el resto de los campos hasta 20 se quedan en blanco. De estos 20 campos, sólo son utilizados con frecuencia los cinco primeros, y como máximo son utilizados 14, por lo que existen campos que sobran.

- Para medios mecánicos: de los 20 campos de idénticas características que los anteriores solo son utilizados alguna vez 10, de los cuales solo los primeros son utilizados con frecuencia.

El diseño, aunque posee un cierto margen para poder albergar muchos recursos asignados a incendios genera espacio desaprovechado, y si se produjera una situación con más de 20 recursos (humanos o mecánicos) la base de datos no podría contemplarlos. Además, para obtener datos estadísticos directamente de la base de datos sobre el personal o vehículos utilizados para la obtención del incendio (dato muy relevante) se necesitan hacer comprobaciones que generarían una pérdida de tiempo y quizás también de información que no sería aceptable en un estudio serio de los incendios.

Para no tener que hacer cálculos inútiles e imprecisos sobre estos datos se han creado dos campos nuevos en la tabla: personal y medios; los cuales son, respectivamente, un recuento de los medios humanos y mecánicos que han sido asignados al incendio. Estos campos han sido llenados por un algoritmo que ha contado los campos no vacíos de los indicados anteriormente ofreciendo un resultado acumulado de todos ellos. Posteriormente se han eliminado todos los 40 campos que la base de datos destinaba a indicar el personal y los medios.

Los únicos datos perdidos en este proceso son los identificadores correspondientes a los recursos existentes en la comarca, datos que pueden ser obtenidos fácilmente de otras fuentes. Como contrapartida de 40 campos prácticamente inútiles se ha pasado a dos campos numéricos, sin valores perdidos, que ofrecen valores muy significativos para el posterior estudio de estos datos, es decir, se obtienen unos datos más manejables y más significativos:

Campos Introducidos	Descripción
PERSONAL	Cantidad de personas (medios humanos) no especialistas destinadas al incendio
MEDIOS	Cantidad de medios mecánicos (motobombas, palas...) destinadas al incendio

Campos Eliminados	Descripción
De PER1 a PER20	Campos destinados a los identificadores de los recursos humanos involucrados en la extinción del incendio
De MED1 a MED20	Campos destinados a los identificadores de los recursos mecánicos involucrados en la extinción del incendio

Tabla 3. Resultado de recursos mecánicos y personales. Fuente: elaboración propia.

Para todas las labores que siguen, se ha construido una aplicación que permite que todas las operaciones se realicen paso a paso, comprobándose cada vez los efectos que se tienen sobre la base de datos albergando todos los algoritmos y funciones auxiliares indicadas durante la exposición del preproceso y la primera transformación de los datos.

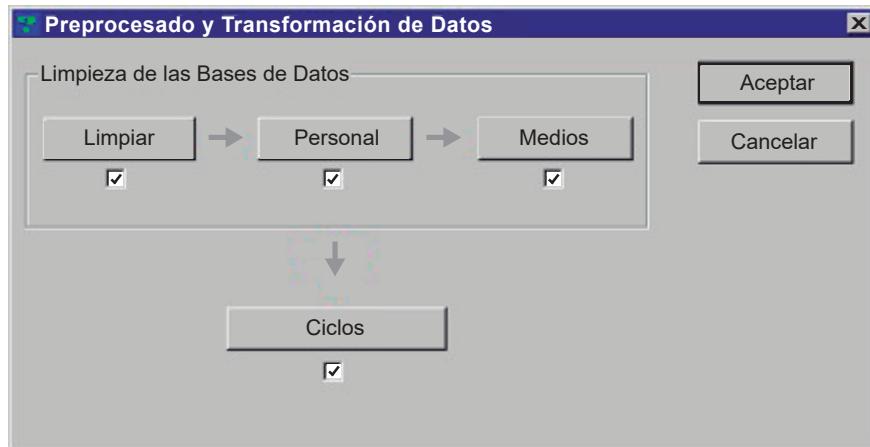


Figura 10. Interfaz de las operaciones descritas. Fuente: elaboración propia.

En este interfaz se pueden ver los diferentes botones que suponen la utilización de cada uno de los algoritmos sobre la base de datos y un control checkbox que les acompaña para indicar que ese proceso se ha culminado con éxito.

El resultado de este proceso han sido 37 ciclos que quedan reflejados en la siguiente tabla:

A	B	C	D	E	F	G	H	I	J	K	L
0	2/01/91	2/01/91	1	0	0	1	0,01	0,5	3	1	1
1	15/01/91	15/01/91	1	1	0	0	0	0,001	1	1	0
2	24/01/91	30/01/91	9	2	4	3	0	0,272	10	10	2
3	23/02/91	25/02/91	12	1	8	3	0	0,395	14	13	2
4	4/03/91	4/03/91	1	0	0	1	0,008	0,03	3	1	1
5	11/03/91	11/03/91	1	0	1	0	0	0,03	1	1	0
6	20/03/91	20/03/91	3	1	0	2	0,032	0,13	2	3	1
7	29/03/91	2/04/91	45	17	19	9	0,213	1,209	31	60	45
8	9/04/91	11/04/91	20	6	10	4	0	0,406	20	23	14
9	14/04/91	24/04/91	48	24	14	10	0,392	0,824	64	71	40
10	27/04/91	27/04/91	1	1	0	0	0	0,01	0	1	1
11	8/05/91	29/05/91	113	59	18	36	0,99	3,641	162	185	126
12	1/06/91	8/06/91	7	4	2	1	0	0,076	9	7	5
13	11/06/91	19/06/91	19	15	0	4	0,169	0	24	30	30
14	22/06/91	6/07/91	57	39	10	8	0	1,841	72	102	87
15	10/07/91	10/09/91	554	375	94	85	0	14,922	744	1306	1196
16	14/09/91	25/09/91	87	58	24	5	0	1,335	94	196	198
17	2/10/91	2/10/91	1	1	0	0	0	0	0	1	0
18	6/10/91	7/10/91	8	6	2	0	0	0,109	10	20	6
19	1/11/91	1/11/91	1	0	1	0	0	0,03	1	2	0
20	24/11/91	24/11/91	1	0	1	0	0	0,02	4	2	1
21	29/11/91	30/11/91	2	1	1	0	0	0,03	2	2	0
22	5/12/91	7/12/91	16	10	5	1	0	0,396	9	25	7
23	14/12/91	16/12/91	6	1	4	1	0	0,245	7	14	3

24	21/12/91	8/01/92	213	98	78	37	63,876	434,756	147	383	136
25	16/01/92	20/01/92	9	8	1	0	1,01	2,71	7	12	4
26	23/01/92	11/02/92	424	233	112	79	206,3	1064,74	234	635	187
27	15/02/92	23/03/92	916	635	189	92	183,1	1074,52	366	1406	515
28	10/04/92	23/05/92	268	195	41	32	139,2	293,4	223	494	305
29	13/06/92	15/06/92	7	6	0	1	0,8	3,24	7	9	5
30	19/06/92	20/06/92	3	2	0	1	1,7	1,8	2	4	3
31	25/06/92	26/06/92	2	2	0	0	0	0,3	1	2	0
32	29/06/92	18/07/92	53	51	1	1	2,08	8,57	38	70	54
33	21/07/92	7/08/92	133	110	9	14	0	82,9	136	266	217
34	12/08/92	27/08/92	80	70	6	4	6,15	0	68	155	142
35	2/09/92	21/09/92	80	74	4	2	3,92	0	87	155	142
36	3/10/92	3/10/92	1	0	0	1	1	0,75	1	2	1

Tabla 4. Ciclos de incendios 1991 - 92. Fuente: elaboración propia.

Leyenda:

- A:** Número de Identificación
- B:** Fecha de Inicio del ciclo
- C:** Fecha de Finalización del ciclo
- D:** Número total de sucesos
- E:** Número de Conatos
- F:** Número de Quemas
- G:** Número de Incendios
- H:** Superficie arbolada quemada
- I:** Superficie rasa quemada
- J:** Número de especialistas (Técnicos, Agentes Forestales...)
- K:** Número de brigadas de personal
- L:** Número de medios mecánicos (Terrestres, aéreos...)

La tabla muestra los 37 ciclos de incendios entre dos períodos de lluvias que han tenido lugar en la zona estudiada durante los años 1991 y 1992. El siguiente paso será comprobar qué estructura de clases determinan los propios datos de estos ciclos (*clustering*), establecer una correlación entre los *clusters* descubiertos y los prototipos sugeridos por los expertos y, por último, encuadrar cada uno de estos ciclos en su correspondiente prototipo.

Es decir, primero se ha realizado un *clustering* sensible al contexto (detección de ciclos) y a continuación se hará uno en un espacio lingüístico inducido (se comenzará con un proceso de *clustering* sobre la tabla de ciclos y se concluirá con uno de clasificación de cada ciclo en un prototipo, tal como se había definido en las etapas del KDD, “convertir *clustering* en clasificación...”).

Clustering jerárquico sobre la tabla de ciclos

Con el fin de detectar las relaciones entre los ciclos, para obtener aquellos de escasa, mediana y alta siniestralidad, se realiza un proceso de *clustering jerarquizado* mediante la técnica de Emparrillados (*Repertory Grids*). El conjunto de elementos es el constituido por los 37 ciclos, y las construcciones son las 7 que se detallan a continuación:

Construcción	Nº DE INCENDIOS, C1		
Valores	Muy pocos	[0 - 20]	1
	Pocos	[21- 30]	2
	Regular	[31- 50]	3
	Nº Alto	[51 - 100]	4
	Muy alto	[101 +]	5

Construcción	SUPERFICIE ARB. QUEMADA, C2		
Valores	Muy poca	[0 - 0,2]	1
	Poca	[0,21 - 0,4]	2
	Regular	[0,41 - 1]	3
	Bastante	[1,1 - 20]	4
	Mucha	[21 +]	5

Construcción	SUPERFICIE RASA QUEMADA, C3		
Valores	Muy poca	[0 - 1]	1
	Poca	[1,1 - 5]	2
	Regular	[5,1 - 100]	3
	Bastante	[101 - 1000]	4
	Mucha	[1001 +]	5

Construcción	Nº DE ESPECIALISTAS, C4		
Valores	Muy pocos	[0 - 10]	1
	Pocos	[11 - 20]	2
	Regular	[21 - 50]	3
	Nº Alto	[51 - 100]	4
	Muy alto	[101 +]	5

Construcción	Nº DE PERSONAL, C5		

Valores	Muy pocos	[0 - 10]	1
	Pocos	[11 - 50]	2
	Regular	[51 - 100]	3
	Nº Alto	[101 - 300]	4
	Muy alto	[301 +]	5
Construcción		Nº DE MEDIOS, C6	
Valores	Muy pocos	[0 - 10]	1
	Pocos	[11 - 30]	2
	Regular	[31 - 50]	3
	Nº Alto	[51 - 100]	4
	Muy alto	[101 +]	5
Construcción		Nº DE DÍAS, C7	
Valores	Muy pocos	[0 - 3]	1
	Pocos	[4 - 10]	2
	Regular	[11 - 20]	3
	Nº Alto	[21 - 30]	4
	Muy alto	[31 +]	5

Tabla 5. Clustering jerárquico sobre la tabla de ciclos. Fuente: elaboración propia.

Con lo que la malla de repertorio queda como se refleja en la tabla 2:

Ciclo	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Ciclo
C1	1	1	1	1	1	1	1	3	1	3	1	5	1	1	4	5	4	1	1	C1
C2	1	1	1	1	1	1	1	2	1	2	1	3	1	1	1	1	1	1	1	C2
C3	1	1	1	1	1	1	1	2	1	1	1	2	1	1	2	3	2	1	1	C3
C4	1	1	1	2	1	1	1	3	2	4	1	5	1	3	4	5	4	1	1	C4
C5	1	1	1	2	1	1	1	3	2	3	1	4	1	2	4	5	4	1	2	C5
C6	1	1	1	1	1	1	1	3	2	3	1	5	1	2	4	5	5	1	1	C6
C7	1	1	2	1	1	1	1	2	1	3	1	4	2	2	3	5	3	1	1	C7

Ciclo	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	Ciclo
C1	1	1	1	1	1	5	1	5	5	5	1	1	1	4	5	4	4	1	C1
C2	1	1	1	1	1	5	4	5	5	5	3	4	1	4	1	4	4	3	C2
C3	1	1	1	1	1	4	2	5	5	4	2	2	1	3	3	1	1	1	C3
C4	1	1	1	1	1	5	1	5	5	5	1	1	1	3	5	4	4	1	C4
C5	1	1	1	2	2	5	2	5	5	5	1	1	1	3	4	4	4	1	C5
C6	1	1	1	1	1	5	1	5	5	5	1	1	1	4	5	5	5	1	C6
C7	1	1	1	1	1	3	2	3	5	5	1	1	1	3	3	3	3	1	C7

Tabla 6. Matriz de entrada al algoritmo de clustering jerárquico. Fuente: elaboración propia.

Para realizar un análisis de clusters (clustering jerárquico) sobre elementos, se construye una matriz de proximidad (malla de repertorio reducida) que representa las diferentes similitudes de los elementos, una matriz de 37x37 elementos (los ciclos) que por encima de la diagonal representan las distancias entre los diferentes ciclos. Pasando estos valores a porcentajes y se crea la tabla reducida a porcentaje.

Como resultado se obtienen los siguientes porcentajes de similitud entre elementos y sus agrupaciones:

(0,1) → 100%

((0,1),4) → 100%

(((0,1),4),5) → 100%

(((0,1),4),5),6) → 100%

(((0,1),4),5),6),10) → 100%

((((0,1),4),5),6),10),17) → 100%

((((((0,1),4),5),6),10),17),19) → 100%

((((((0,1),4),5),6),10),17),19),20) → 100%

((((((0,1),4),5),6),10),17),19),20),21) → 100%

((((((0,1),4),5),6),10),17),19),20),21),31) → 100% (A)

(2,12) → 100% (B)

(18,22) → 100%

((18,22),23) → 100% (C)

(34,35) → 100% (D)

((((((0,1),4),5),6),10),17),19),20),21),31),(2,12)) → 97%

(3,8) → 97%

(14,16) → 97%

(24,26) → 97%

(27,28) → 97%

(29,30) → 97%

((((((0,1),4),5),6),10),17),19),20),21),31),(2,12)),((18,22),23)) → 93%

((29,30),36) → 93%

$(((((((((((0,1),4),5),6),10),17),19),20),21),31),(2,12)),((18,22),23)),(3,8)) \rightarrow 90\%$

$(7,9) \rightarrow 90\%$

$(15,33) \rightarrow 90\%$

$((24,26),(27,28)) \rightarrow 90\%$

$(25,((29,30),36)) \rightarrow 90\%$

$(((((((((((0,1),4),5),6),10),17),19),20),21),31),(2,12)),((18,22),23)),(3,8)),13) \rightarrow 83\%$

$(11,(15,33)) \rightarrow 83\%$

$((14,16),(34,35)) \rightarrow 83\%$

$((7,9),((14,16),(34,35))) \rightarrow 79\%$

$(((((((((((0,1),4),5),6),10),17),19),20),21),31),(2,12)),((18,22),23)),(3,8)),13),(25,((29,30),36))) \rightarrow 75\%$

$((7,9),((14,16),(34,35))),32) \rightarrow 75\%$

$((11,(15,33)),((24,26),(27,28))) \rightarrow 75\%$

$((7,9),((14,16),(34,35))),32),((11,(15,33)),((24,26),(27,28))) \rightarrow 61\%$

$(((((((((((0,1),4),5),6),10),17),19),20),21),31),(2,12)),((18,22),23)),(3,8)),13),(25,((29,30),36)))$

$((7,9),((14,16),(34,35))),32),((11,(15,33)),((24,26),(27,28))) \rightarrow 40\%$

Con lo que se obtiene el siguiente Dendrograma:

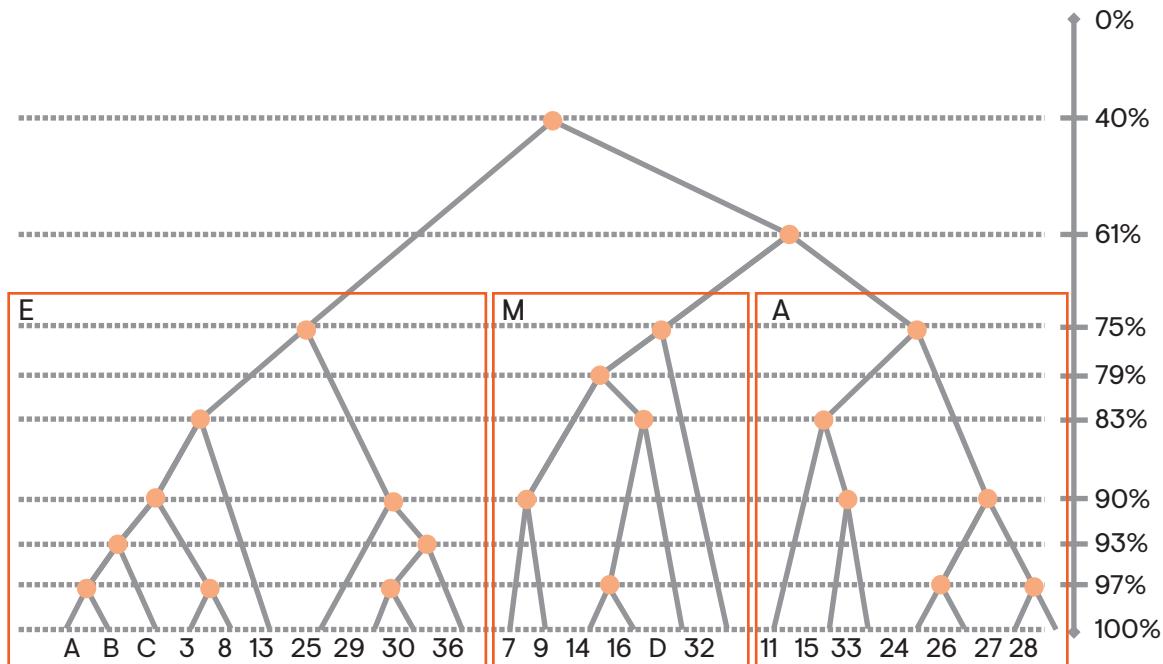


Figura 11. Dendrograma final del proceso de clustering jerárquico. Fuente: elaboración propia.

E: Siniestralidad escasamente progresiva

M: Siniestralidad medianamente progresiva

A: Siniestralidad altamente progresiva

Ordenación de los ciclos según la evolución de su siniestralidad

Una vez realizado este proceso de *clustering* sobre los ciclos, se está en condiciones de ordenarlos en función de sus siniestralidades. Para ello, realizados **test** sobre varios expertos, se ha llegado a la conclusión de que la evolución de la siniestralidad está básicamente vinculada al número de siniestros ocurridos (sin distinguir entre incendios, superficie arbolada mayor a media hectárea, conatos, superficie total menor a media hectárea y quemas, superficie rasa mayor que media hectárea), porque cualquiera de los siniestros, por pequeño que haya sido, podría haber tenido consecuencias peores de no haber actuado a tiempo. El otro factor influyente en la evolución de la siniestralidad es la superficie que se ha quemado, dándole un mayor peso a la superficie arbolada.

Teniendo en cuenta estos factores, se está en condiciones de definir una medida *Heurística* de siniestralidad del ciclo, expresada del siguiente modo:

$$\text{Número total de incendios}/100$$

+

$$[(1 * \text{Superficie arbolada total}) + (0,5 * \text{Superficie rasa total})]/100$$

Presentados casos de ejemplo, los expertos concuerdan en que un ciclo sería de siniestralidad escasamente progresiva cuando esta medida estuviera por debajo de 0,4, entre este valor y 1 sería un ciclo de siniestralidad medianamente progresiva y a partir de 1 la siniestralidad sería altamente progresiva.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	P
17	2/10/91	2/10/91	1	1	0	0	0	0	0	1	0	1	0,01	E
1	15/01/91	15/01/91	1	1	0	0	0	0,001	1	1	0	1	0,0105	E
10	27/04/91	27/04/91	1	1	0	0	0	0,01	0	1	1	1	0,0105	E
20	24/11/91	24/11/91	1	0	1	0	0	0,02	4	2	1	1	0,0101	E
5	11/03/91	11/03/91	1	0	1	0	0	0,03	1	1	0	1	0,0105	E
19	1/11/91	1/11/91	1	0	1	0	0	0,03	1	2	0	1	0,0105	E
4	4/03/91	4/03/91	1	0	0	1	0,008	0,03	3	1	1	1	0,0103	E
0	2/01/91	2/01/91	1	0	0	1	0,01	0,5	3	1	1	1	0,0126	E
21	29/11/91	30/11/91	2	1	1	0	0	0,03	2	2	0	2	0,0205	E
31	25/06/92	26/06/92	2	2	0	0	0	0,3	1	2	0	2	0,0215	E
36	3/10/92	3/10/92	1	0	0	1	1	0,75	1	2	1	1	0,0235	E
6	20/03/91	20/03/91	3	1	0	2	0,032	0,13	2	3	1	1	0,0307	E
30	19/06/92	20/06/92	3	2	0	1	1,7	1,8	2	4	3	2	0,056	E
23	14/12/91	16/12/91	6	1	4	1	0	0,245	7	14	3	3	0,0625	E
12	1/06/91	8/06/91	7	4	2	1	0	0,076	9	7	5	8	0,0708	E
18	6/10/91	7/10/91	8	6	2	0	0	0,109	10	20	6	2	0,0805	E

2	24/01/91	30/01/91	9	2	4	3	0	0,272	10	10	2	7	0,0913	E
29	13/06/92	15/06/92	7	6	0	1	0,8	3,24	7	9	5	3	0,0942	E
25	16/01/92	20/01/92	9	8	1	0	1,01	2,71	7	12	4	5	0,1136	E
3	23/02/91	25/02/91	12	1	8	3	0	0,395	14	13	2	3	0,1219	E
22	5/12/91	7/12/91	16	10	5	1	0	0,396	9	25	7	3	0,1619	E
13	11/06/91	19/06/91	19	15	0	4	0,169	0	24	30	30	9	0,1916	E
8	9/04/91	11/04/91	20	6	10	4	0	0,406	20	23	14	3	0,2020	E
7	29/03/91	2/04/91	45	17	19	9	0,213	1,209	31	60	45	4	0,4581	M
9	14/04/91	24/04/91	48	24	14	10	0,392	0,824	64	71	40	11	0,4880	M
14	22/06/91	6/07/91	57	39	10	8	0	1,841	72	102	87	15	0,5792	M
32	29/06/92	18/07/92	53	51	1	1	2,08	8,57	38	70	54	20	0,5936	M
35	2/09/92	21/09/92	80	74	4	2	3,92	0	87	155	142	20	0,8392	M
34	12/08/92	27/08/92	80	70	6	4	6,15	0	68	155	142	16	0,8615	M
16	14/09/91	25/09/91	87	58	24	5	0	1,335	94	196	198	12	0,8766	M
11	8/05/91	29/05/91	113	59	18	36	0,99	3,641	162	185	126	22	1,1581	A
33	21/07/92	7/08/92	133	110	9	14	0	82,9	136	266	217	17	1,7445	A
24	21/12/91	8/01/92	213	98	78	37	63,876	434,756	147	383	136	18	4,9425	A
28	10/04/92	23/05/92	268	195	41	32	139,2	293,4	223	494	305	44	5,539	A
15	10/07/91	10/09/91	554	375	94	85	0	14,922	744	1306	1196	61	5,6146	A
26	23/01/92	11/02/92	424	233	112	79	206,3	1064,74	234	635	187	19	11,626	A
27	15/02/92	23/03/92	916	635	189	92	183,1	1074,52	366	1406	515	39	16,363	A

Tabla 7. Ordenación de los ciclos según su peligrosidad. Fuente: elaboración propia.

Leyenda

A: Número de Identificación

I: Superficie rasa quemada

B: Fecha de Inicio del ciclo

J: Número de especialistas (Técnicos, Agentes Forestales...)

C: Fecha de Finalización del ciclo

K: Número de brigadas de personal

D: Número total de sucesos

L: Número de medios (Terrestres, aéreos...)

E: Número de Conatos

M: Número de días de duración del ciclo

F: Número de Quemas

N: Valor de la medida Heurística de siniestralidad del ciclo

G: Número de Incendios

P: Prototipo de evolución de la siniestralidad: E (Escasamente progresiva), M (Medianamente progresiva), A (Altamente progresiva)

H: Superficie arbolada quemada

Data Mining

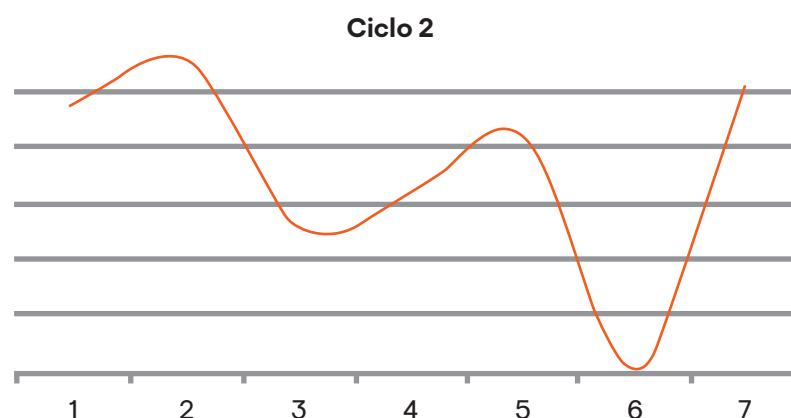
En esta parte final del proceso de KDD, lo que se pretende es aplicar funciones de resumen sobre los ciclos de cada tipo, con el fin de obtener los tres prototipos de evolución de la siniestralidad, para poder evaluar casos reales y llegar a predecir el comportamiento y las necesidades en los siniestros de días sucesivos.

Se presenta cada ciclo con los valores relevantes de cada uno de los días y con un gráfico que representa la evolución de la siniestralidad en el tiempo (en días), tomando como dato la media de todos los valores de cada día (ocurrencia diaria), normalizados en el intervalo [0, 10]. El patrón de este gráfico sería el presentado al principio de este proceso (fig. 13).

- **El primer sector representa un lento crecimiento de la siniestralidad en los días inmediatamente posteriores al periodo de lluvias. Durará desde el inicio del ciclo hasta que la ocurrencia diaria alcance el valor 4.**
- **El segundo sector expresa un alto crecimiento de la siniestralidad, especialmente en cuanto al número de incendios. Será desde que la ocurrencia diaria haya alcanzado el valor 4 por primera vez, hasta que alcance el valor 7.**
- **En el tercero se quiere representar una estabilización en cuanto al número, pero un progresivo aumento en la peligrosidad de los fuegos. Los valores irán desde que la ocurrencia diaria haya alcanzado el valor 7 por primera vez, hasta el final.**

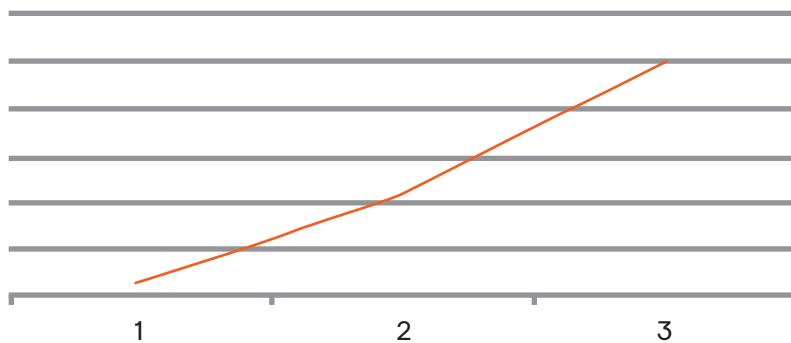
Siniestralidad escasamente progresiva

Los ciclos 0, 1, 4, 5, 6, 10, 17, 18, 19, 20, 21, 23, 29, 30, 31 y 36 no se tienen en cuenta por ser de uno o dos días con incendios y por lo tanto, ser nula su representatividad. Teniendo esto en cuenta, los ciclos que se analizan son los siguientes (en el eje X se representan los días del ciclo y en Y el valor de la Ocurrencia Diaria):

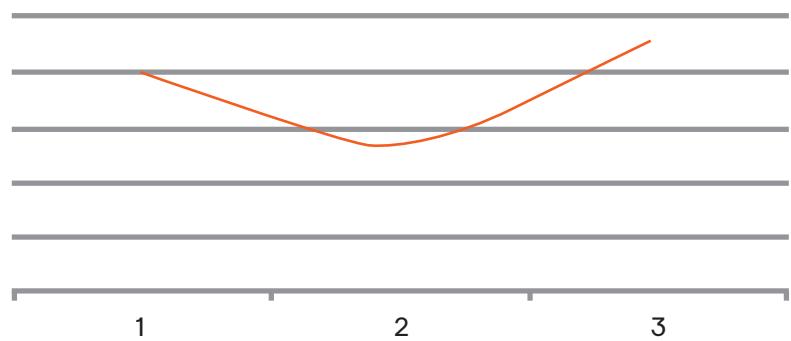


Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
24-ene-91	2	0	0,027	6	2	0	4,7	2

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
25-ene-91	2	0,04	0,05	2	2	0	5,6	2
26-ene-91	1	0,015	0	2	1	0	2,6	2
27-ene-91	1	0	0,01	0	1	1	3,2	2
28-ene-91	2	0,005	0,035	1	3	0	4,2	2
30-ene-91	1	0	0,15	1	1	1	5,0	2

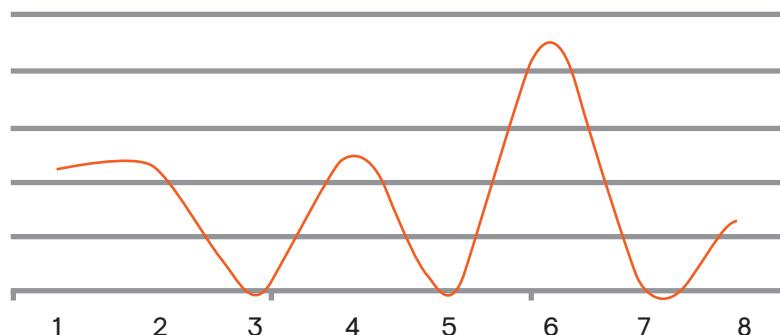
Ciclo 3

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
23-feb-91	1	0	0,02	0	1	0	0,6	1
24-feb-91	4	0,005	0,19	4	4	0	4,4	2
25-feb-91	7	0,018	0,185	13	8	2	10,0	3
27-ene-91	1	0	0,01	0	1	1	3,2	2
28-ene-91	2	0,005	0,035	1	3	0	4,2	2
30-ene-91	1	0	0,15	1	1	1	5,0	2

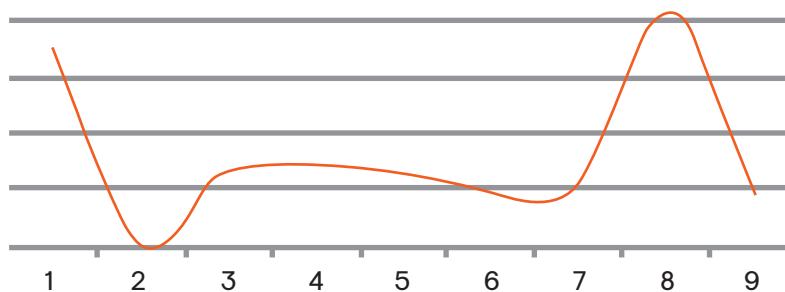
Ciclo 8

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
09-abr-91	9	0,03	0,151	10	10	4	5,5	2
10-abr-91	5	0,01	0,095	6	5	5	3,5	2

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
11-abr-91	6	0,128	0,16	12	8	5	5,2	2
27-ene-91	1	0	0,01	0	1	1	3,2	2
28-ene-91	2	0,005	0,035	1	3	0	4,2	2
30-ene-91	1	0	0,15	1	1	1	5,0	2

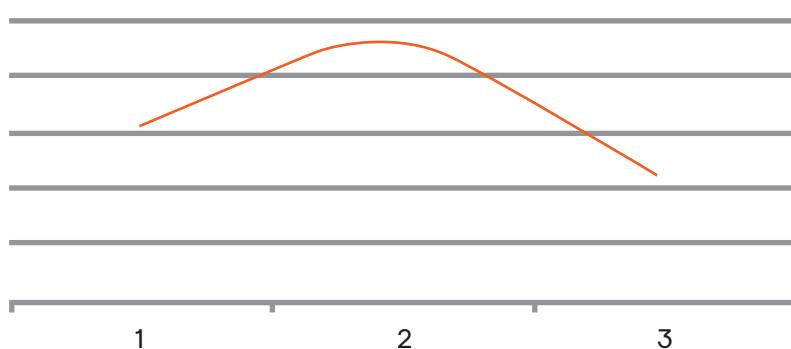
Ciclo 12

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
01-jun-91	1	0	0,03	1	1	1	4,4	2
02-jun-91	1	0	0,03	1	1	1	4,4	2
03-jun-91	0	0	0	0	0	0	0,0	2
04-jun-91	2	0,001	0,006	1	2	1	4,9	2
05-jun-91	0	0	0	0	0	0	0,0	2
06-jun-91	2	0,01	0,01	7	2	2	8,9	3
07-jun-91	0	0	0	0	0	0	0,0	3
08-jun-91	1	0,001	0	3	1	0	2,5	3

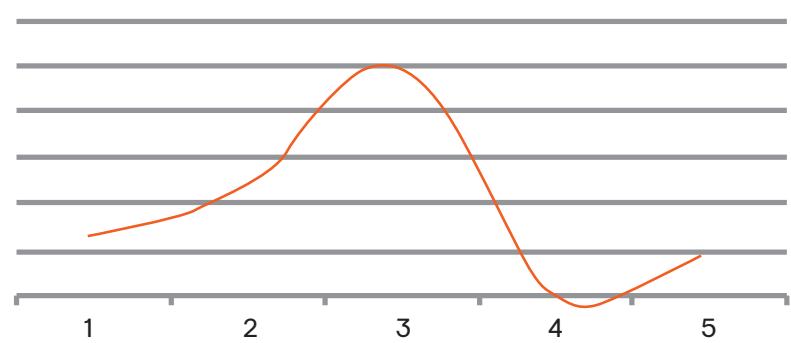
Ciclo 13

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
11-jun-91	3	0,083	0,08	6	4	5	6,9	2
12-jun-91	0	0	0	0	0	0	0,0	2

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
13-jun-91	1	0,02	0,03	4	2	2	2,6	2
14-jun-91	2	0,01	0,001	4	5	2	2,9	2
15-jun-91	3	0	0,009	1	3	3	2,6	2
16-jun-91	2	0,001	0,002	1	3	2	1,9	2
17-jun-91	2	0,005	0	1	3	3	2,1	2
18-jun-91	4	0,05	0,028	14	8	11	8,3	3
19-jun-91	2	0	0,006	2	2	2	1,9	3

Ciclo 22

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
05-dic-91	4	0	0,21	5	9	2	6,2	2
06-dic-91	6	0,01	0,116	9	9	4	9,3	3
07-dic-91	6	0	0,07	3	7	1	4,5	3

Ciclo 25

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
16-ene-92	1	0	0,5	1	2	1	2,6	1
17-ene-92	2	0,51	0,01	2	2	1	4,8	2
18-ene-92	5	0,5	2	3	7	2	10,0	3

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
19-ene-92	0	0	0	0	0	0	0,0	3
20-ene-92	1	0	0,2	2	1	0	1,8	3

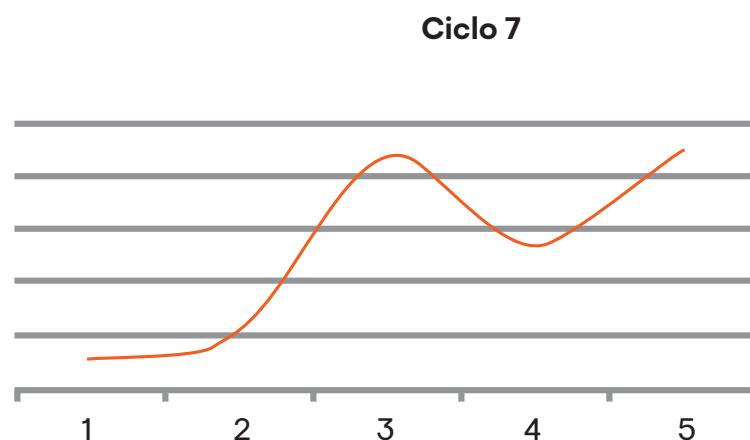
Tabla 8. Ciclos de Siniestralidad Escasamente Progresiva. Fuente: elaboración propia.

Analizando estos ciclos y aplicando funciones que los resuman, se llega a la definición del prototipo **“Siniestralidad Escasamente Progresiva”**:

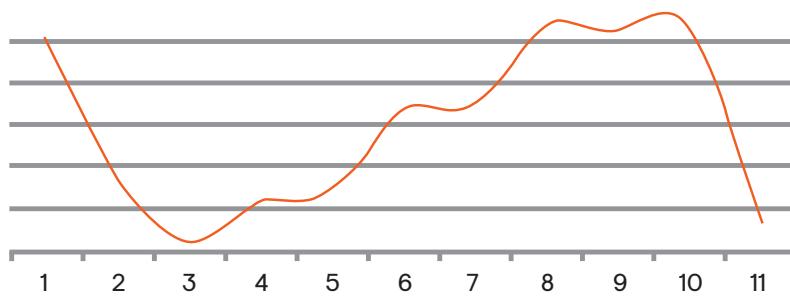
	Sector 1	Sector 2	Sector 3
Media de días:	1	4	2
Media de incendios/día:	1	2	4
Mínimo de inc/día:	1	1	1
Máximo de inc/día:	1	4	9
Nº especialistas/día:	1	2	6
Nº de brigadas/día:	2	2	5
Nº de medios/día:	1	1	3

Siniestralidad medianamente progresiva

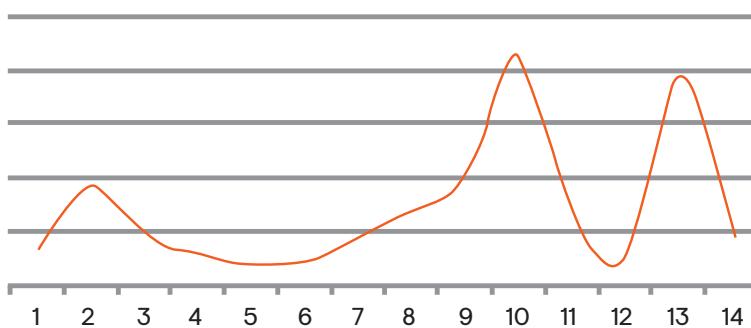
Los ciclos que se analizan son los siguientes (en el eje X se representan los días del ciclo y en Y el valor de la Ocurrencia Diaria):



Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
29-mar-91	3	0	0,1	1	3	1	1,1	1
30-mar-91	4	0,035	0,04	2	4	3	2,1	1
31-mar-91	14	0,079	0,569	11	19	12	8,6	3
01-abr-91	11	0,011	0,23	12	12	11	5,5	3
02-abr-91	13	0,088	0,27	15	22	18	9,0	3

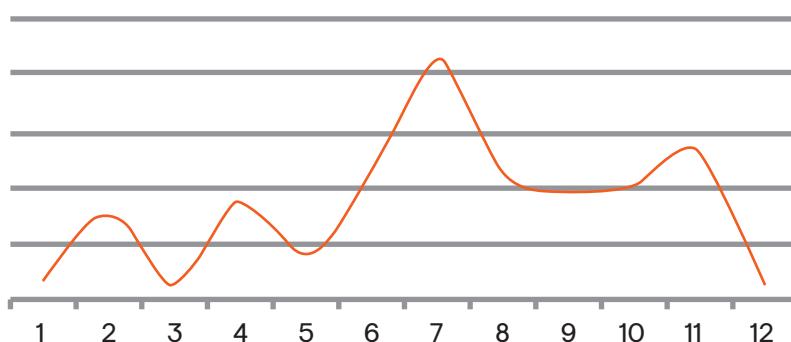
Ciclo 9

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
14-abr-91	4	0,005	0,235	18	8	8	6,9	2
15-abr-91	4	0,005	0,013	0	5	3	2,4	2
16-abr-91	1	0	0,005	0	1	0	0,4	2
17-abr-91	2	0,01	0,005	3	4	1	1,6	2
18-abr-91	2	0,012	0,013	4	4	2	2,0	2
19-abr-91	6	0,04	0,117	6	9	2	4,8	2
20-abr-91	6	0,013	0,07	10	9	4	4,9	2
21-abr-91	7	0,158	0,16	13	9	4	7,5	3
22-abr-91	7	0,074	0,036	21	10	8	7,3	3
23-abr-91	8	0,075	0,11	11	11	8	7,4	3
24-abr-91	1	0	0,06	1	1	0	0,9	3

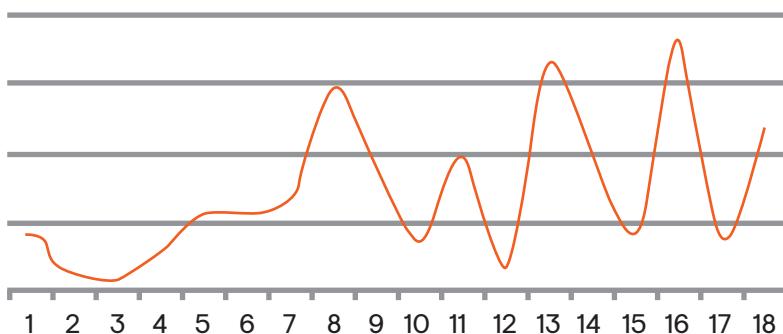
Ciclo 14

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
22-jun-91	2	0,015	0,01	4	5	2	1,5	1
23-jun-91	6	0,01	0,053	8	12	9	3,7	1
24-jun-91	4	0	0,016	4	5	5	1,9	1
25-jun-91	2	0	0,035	6	2	3	1,3	1
26-jun-91	1	0	0,01	1	3	2	0,7	1
27-jun-91	3	0,003	0,003	0	3	0	0,9	1

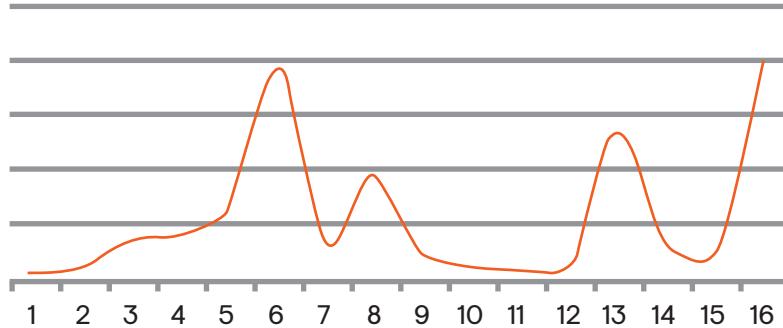
Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
28-jun-91	3	0	0,124	3	5	3	1,7	1
29-jun-91	6	0,009	0,024	7	7	5	2,8	1
30-jun-91	5	0	0,154	14	11	10	4,0	2
01-jul-91	6	0,124	0,505	26	19	15	8,5	3
02-jul-91	5	0	0,051	5	5	8	2,5	3
03-jul-91	2	0	0,006	2	3	2	0,9	3
04-jul-91	9	0,017	0,813	16	17	19	7,7	3
06-jul-91	3	0	0,037	7	5	4	1,9	3

Ciclo 16

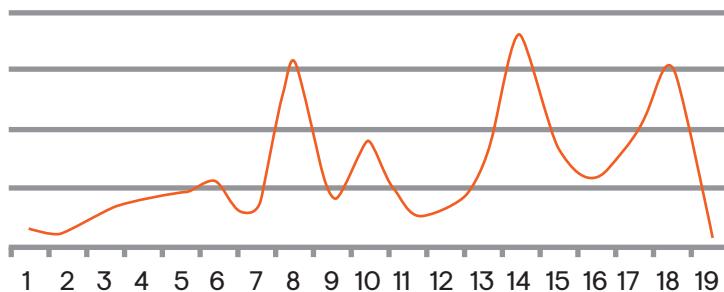
Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
14-sep-91	2	0	0,006	3	3	4	0,6	1
15-sep-91	5	0,003	0,093	13	15	20	3,1	1
16-sep-91	2	0	0,01	1	3	3	0,5	1
17-sep-91	8	0,002	0,123	8	20	23	3,5	1
18-sep-91	4	0,008	0,035	6	7	5	1,5	1
19-sep-91	11	0,012	0,068	18	22	22	4,3	2
20-sep-91	19	0,01	0,286	31	43	42	8,4	3
21-sep-91	6	0,06	0,12	12	14	10	4,4	3
22-sep-91	10	0,001	0,155	13	20	18	3,9	3
23-sep-91	11	0	0,102	13	25	23	4,1	3
24-sep-91	8	0,015	0,322	12	22	25	5,3	3
25-sep-91	1	0	0,015	2	2	3	0,5	3

Ciclo 32

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
29-jun-92	1	0	1	1	1	1	1,7	1
01-jul-92	1	0	0,1	0	1	1	0,6	1
02-jul-92	1	0	0	0	1	0	0,4	1
03-jul-92	2	0	0,11	1	2	1	1,1	1
04-jul-92	2	0	0,51	5	2	2	2,2	1
05-jul-92	4	0	0,4	0	4	2	2,2	1
06-jul-92	3	0	0,14	7	3	3	2,7	1
07-jul-92	4	0,4	1,3	7	6	7	5,8	2
09-jul-92	1	1,2	0	4	5	3	3,7	2
10-jul-92	3	0	0,13	0	3	1	1,4	2
11-jul-92	4	0,25	0,19	6	6	4	3,8	2
12-jul-92	1	0	0,04	1	1	1	0,7	2
13-jul-92	4	0	1,22	13	7	9	6,5	2
14-jul-92	5	0,13	0,18	4	6	4	3,6	2
15-jul-92	3	0	0,34	1	3	2	1,9	2
16-jul-92	8	0,1	1,14	8	11	7	7,0	3
17-jul-92	3	0	0,13	1	3	1	1,5	3
18-jul-92	3	0	1,64	6	5	5	4,7	3

Ciclo 34

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
12-agosto-92	1	0,01	0	1	1	0	0,2	1
13-agosto-92	2	0	0,16	1	2	2	0,5	1
14-agosto-92	6	0,01	0,64	3	8	6	1,6	1
15-agosto-92	4	0	1,57	6	8	4	1,6	1
16-agosto-92	8	0	1,17	9	14	11	2,7	1
17-agosto-92	17	0,25	9,17	27	25	39	7,6	3
18-agosto-92	4	0	0,73	4	6	7	1,4	3
19-agosto-92	7	0,85	1,75	16	16	16	3,7	3
20-agosto-92	2	0,03	0,65	5	3	4	0,9	3
21-agosto-92	2	0	0,01	2	2	1	0,5	3
22-agosto-92	2	0	0,24	0	2	1	0,4	3
23-agosto-92	3	0	0,16	0	3	3	0,6	3
24-agosto-92	9	0	6,91	18	26	23	5,3	3
25-agosto-92	3	0	0,39	8	6	6	1,4	3
26-agosto-92	3	0	0,26	7	3	3	1,1	3
27-agosto-92	7	5	13,65	24	30	16	7,9	3

Ciclo 35

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
02-sep-92	1	0,1	0	3	2	1	0,7	1
04-sep-92	1	0	0,2	4	1	2	0,8	1
05-sep-92	3	0,04	0,54	1	6	4	1,5	1
06-sep-92	5	0	0,1	2	8	5	2,0	1
07-sep-92	3	0,28	0,12	6	9	4	2,2	1
08-sep-92	3	0,13	0,06	15	6	4	2,5	1
09-sep-92	3	0,15	0,05	3	5	4	1,5	1
10-sep-92	9	0	5,41	14	20	21	7,1	3
11-sep-92	2	0,5	1	4	4	3	1,9	3
12-sep-92	6	0	0,86	20	10	9	4,2	3
13-sep-92	3	0,02	0,15	3	4	6	1,5	3
14-sep-92	4	0	0,25	2	5	5	1,6	3
15-sep-92	4	0	2,99	7	7	6	3,0	3
16-sep-92	10	1,87	1,34	22	24	16	8,3	3
17-sep-92	4	0,7	1,25	5	10	13	3,7	3
18-sep-92	4	0,02	1,02	10	8	7	2,8	3
19-sep-92	6	0	1,11	14	12	13	4,2	3
20-sep-92	8	0,11	6,14	18	13	18	6,8	3
21-sep-92	1	0	0,15	1	1	1	0,4	3

Tabla 9. Ciclos de Siniestralidad Medianamente Progresiva. Fuente: elaboración propia.

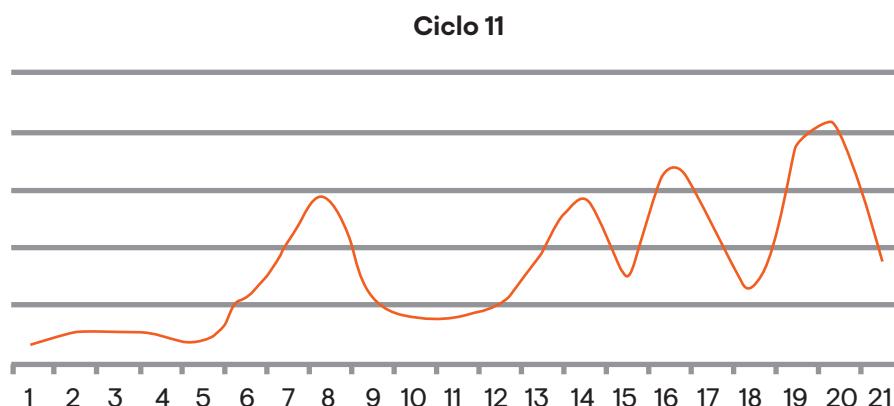
Analizando estos ciclos y aplicando funciones que los resuman, se llega a la definición del prototipo **"Siniestralidad Medianamente Progresiva"**:

	Sector 1	Sector 2	Sector 3
Media de días:	6	2	6
Media de incendios/día:	3	4	6
Mínimo de inc/día:	1	1	1
Máximo de inc/día:	8	11	19
Nº especialistas/día:	4	6	11
Nº de brigadas/día:	5	6	12
Nº de medios/día:	4	5	11

Siniestralidad altamente progresiva

Los ciclos que se analizan son los siguientes (en el eje X se representan los días del ciclo y en Y el valor de la Ocurrencia Diaria):

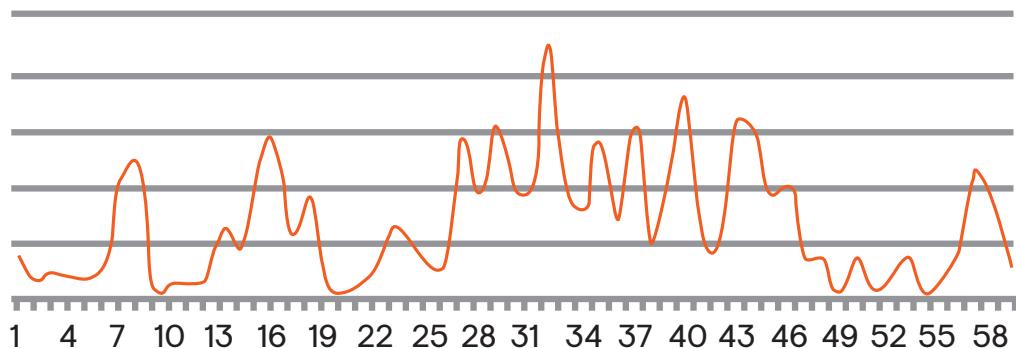
- El *primer* sector durará desde el inicio del ciclo hasta que la ocurrencia diaria alcance el valor 4.
- El *segundo* sector será desde que la ocurrencia diaria haya alcanzado el valor 4 por primera vez, hasta que alcance el valor 7.
- En el *tercero* los valores irán desde que la ocurrencia diaria haya alcanzado el valor 7 por primera vez, hasta el final.



Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
08-may-91	1	0	0,001	5	2	1	0,6	1
09-may-91	1	0,01	0,09	6	3	2	1,1	1
10-may-91	1	0,05	0,01	5	2	1	1,1	1
12-may-91	3	0,007	0,01	1	3	3	1,0	1
13-may-91	2	0,002	0,007	3	2	1	0,7	1
14-may-91	5	0,002	0,032	6	7	5	2,0	1
15-may-91	6	0,03	0,33	13	10	10	3,9	1
16-may-91	7	0,176	0,097	16	15	10	5,6	2

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
17-may-91	4	0,002	0,265	7	6	5	2,2	2
18-may-91	4	0,004	0,011	5	6	2	1,5	2
19-may-91	3	0,03	0,212	4	5	1	1,7	2
20-may-91	5	0,01	0,025	7	6	4	2,0	2
21-may-91	8	0,03	0,28	12	11	4	3,6	2
22-may-91	8	0,052	0,506	18	17	10	5,5	2
23-may-91	5	0,044	0,025	19	7	4	3,0	2
24-may-91	7	0,152	0,275	33	16	10	6,6	2
25-may-91	8	0,106	0,078	17	10	9	4,6	2
26-may-91	7	0,013	0,039	7	8	6	2,6	2
27-may-91	9	0,096	1,063	23	14	15	7,4	3
28-may-91	13	0,16	0,155	25	21	18	8,0	3
29-may-91	6	0,014	0,13	16	14	5	3,5	3

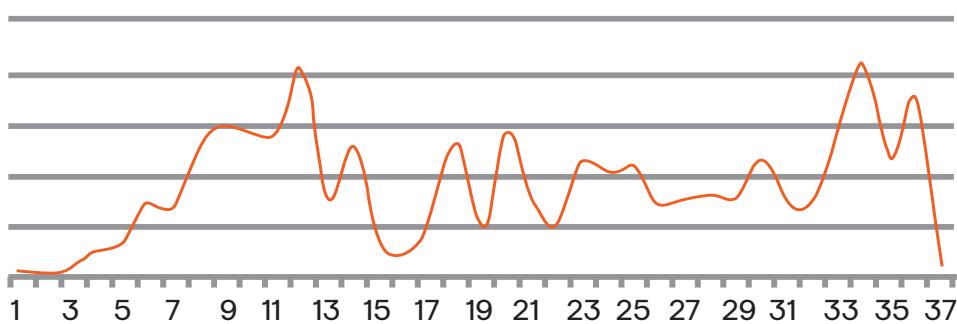
Ciclo 15



Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
10-jul-91	8	0,001	0,039	14	8	7	1,4	1
12-jul-91	3	0,021	0,081	7	4	5	0,7	1
13-jul-91	4	0,022	0,04	9	6	9	1,0	1
14-jul-91	2	0,05	0,005	6	7	7	0,8	1
15-jul-91	3	0	0,03	5	6	9	0,8	1
16-jul-91	4	0,004	0,04	10	11	9	1,2	1
17-jul-91	17	0,004	0,188	32	33	35	4,2	2
18-jul-91	15	0,094	0,495	38	37	35	4,6	2
19-jul-91	2	0	0,004	2	2	3	0,3	2
20-jul-91	2	0,02	0	4	7	6	0,7	2
21-jul-91	3	0	0,046	3	5	4	0,6	2
22-jul-91	5	0,008	0,023	5	7	4	0,8	2
23-jul-91	6	0,011	0,283	26	18	20	2,5	2
24-jul-91	7	0,08	0,022	13	15	14	1,8	2

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
25-jul-91	17	0,01	0,274	33	41	45	4,8	2
26-jul-91	22	0,015	0,16	43	44	43	5,5	2
27-jul-91	9	0,027	0,121	18	20	18	2,4	2
28-jul-91	12	0,02	0,243	40	28	23	3,7	2
30-jul-91	1	0,022	0,015	6	4	2	0,5	2
01-agosto-91	2	0	0,013	2	3	1	0,3	2
02-agosto-91	2	0	0,006	3	4	4	0,5	2
03-agosto-91	4	0	0,036	14	10	10	1,3	2
04-agosto-91	7	0,135	0,098	20	19	23	2,5	2
05-agosto-91	4	0,025	0,222	12	12	20	1,7	2
06-agosto-91	5	0	0,034	4	12	13	1,2	2
07-agosto-91	7	0,003	0,01	8	11	9	1,3	2
08-agosto-91	17	0,065	0,684	46	49	41	5,6	2
09-agosto-91	13	0,061	0,088	32	32	27	3,7	2
10-agosto-91	12	1,622	1,692	27	35	33	6,1	2
11-agosto-91	12	0,083	0,249	32	30	29	3,7	2
12-agosto-91	14	0,021	0,273	38	30	27	3,9	2
13-agosto-91	20	1,101	3,505	53	61	52	9,1	3
14-agosto-91	13	0,161	0,459	31	29	29	3,9	3
15-agosto-91	11	0,013	0,201	24	26	28	3,2	3
16-agosto-91	21	0,109	0,364	34	51	46	5,6	3
17-agosto-91	12	0,005	0,121	13	26	27	2,8	3
18-agosto-91	21	0,048	0,248	48	60	45	6,1	3
19-agosto-91	11	0	0,079	4	21	17	2,0	3
20-agosto-91	19	0,021	0,11	25	40	36	4,3	3
21-agosto-91	21	0,456	1,83	42	65	41	7,1	3
22-agosto-91	7	0,02	0,108	12	17	17	1,9	3
23-agosto-91	9	0	0,061	6	19	15	1,8	3
24-agosto-91	24	0,077	0,877	38	50	51	6,2	3
25-agosto-91	22	0,017	0,463	33	49	53	5,7	3
26-agosto-91	16	0,054	0,099	27	31	26	3,7	3
27-agosto-91	18	0,004	0,194	31	41	21	4,0	3
28-agosto-91	6	0	0,142	12	13	10	1,5	3
29-agosto-91	4	0,01	0,02	9	15	15	1,4	3
30-agosto-91	1	0	0,001	0	2	0	0,1	3
01-sept-91	6	0,005	0,064	5	16	13	1,4	3
02-sept-91	1	0,002	0	2	2	1	0,2	3
03-sept-91	2	0	0,004	5	5	3	0,5	3
04-sept-91	5	0	0,065	6	14	19	1,5	3
05-sept-91	1	0	0,01	0	1	0	0,1	3

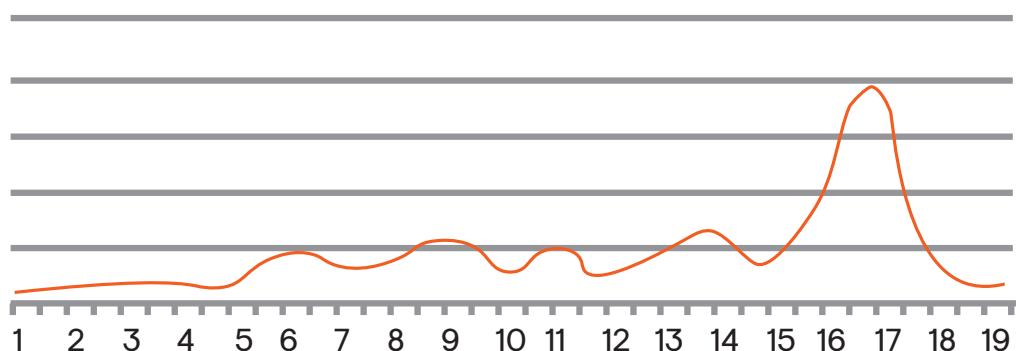
Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
06-sep-91	3	0,01	0,004	9	8	6	0,9	3
07-sep-91	7	0,01	0,03	12	13	11	1,6	3
08-sep-91	16	0,01	0,171	26	43	42	4,4	3
09-sep-91	11	0,033	0,117	22	26	29	3,1	3
10-sep-91	5	0	0,061	6	12	8	1,1	3

Ciclo 24

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
15-feb-92	1	0	1,5	2	2	1	0,2	1
16-feb-92	1	0,2	0	1	2	1	0,2	1
18-feb-92	3	0	2,7	1	4	2	0,3	1
19-feb-92	9	0,1	10,9	4	13	5	1,1	1
20-feb-92	13	0,3	13,13	6	16	5	1,4	1
21-feb-92	19	6,1	27,52	11	29	9	2,9	1
22-feb-92	22	5,9	15,61	12	30	7	2,7	1
23-feb-92	42	2,5	66,7	14	61	18	5,1	2
24-feb-92	43	7,1	78,63	21	62	15	5,9	2
25-feb-92	35	3,5	82,31	30	54	17	5,7	2
26-feb-92	43	4,2	38,11	36	61	18	5,5	2
27-feb-92	60	21,9	39,9	29	84	27	7,9	3
28-feb-92	29	3,8	22,31	9	41	7	3,0	3
29-feb-92	34	7,7	66,1	13	57	15	5,1	3
01-mar-92	9	0,1	19,8	5	12	3	1,2	3
02-mar-92	6	1	7	3	13	4	0,9	3
03-mar-92	15	5,9	8,6	11	25	8	2,3	3
04-mar-92	36	9,1	38,1	20	60	17	5,1	3
05-mar-92	17	0,6	16	9	22	7	1,9	3
06-mar-92	35	1,9	56,3	36	47	26	5,5	3
07-mar-92	20	9,7	15,9	11	26	11	3,0	3
08-mar-92	21	2,7	8,7	4	26	10	2,0	3

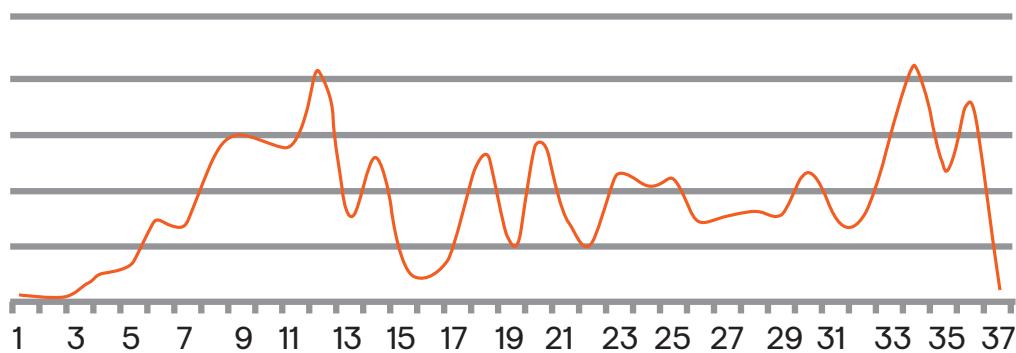
Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
09-mar-92	31	6,2	40,8	17	51	17	4,5	3
10-mar-92	24	4,7	50,8	18	36	17	4,1	3
11-mar-92	24	15	25,2	12	43	17	4,3	3
12-mar-92	15	6,7	12,1	14	30	14	2,8	3
13-mar-92	25	2,9	23,2	9	41	14	3,1	3
14-mar-92	20	2,6	25,45	19	38	13	3,2	3
15-mar-92	24	3,8	9,2	10	37	21	3,1	3
16-mar-92	23	8,6	23,75	24	50	24	4,6	3
17-mar-92	22	2,35	19,1	16	32	13	2,9	3
18-mar-92	23	4,6	14,5	13	34	14	3,0	3
19-mar-92	41	4,9	56,7	27	56	23	5,7	3
20-mar-92	44	15,95	43	45	78	43	8,2	3
21-mar-92	36	6,7	28,4	18	54	21	4,6	3
22-mar-92	47	3,8	61	39	72	30	6,9	3
23-mar-92	4	0	5,5	1	7	1	0,4	3

Ciclo 26



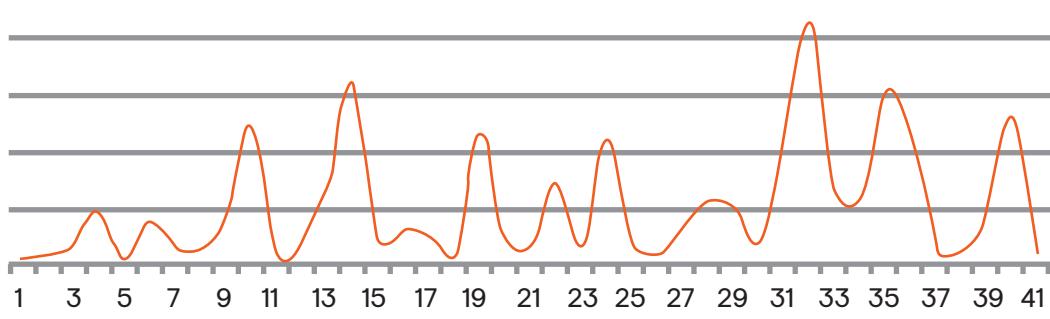
Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
23-ene-92	5	0	3,6	4	6	0	0,3	1
24-ene-92	3	0	4,6	4	5	2	0,3	1
25-ene-92	6	1,1	2,5	7	12	3	0,5	1
26-ene-92	8	0,5	8,8	3	13	4	0,5	1
27-ene-92	6	0,3	19	4	17	3	0,6	1
28-ene-92	28	10,1	52,5	28	39	8	2,2	1
29-ene-92	22	5	33,2	12	32	7	1,5	1
30-ene-92	23	2	26,45	18	33	6	1,5	1
31-ene-92	31	11,7	50,53	21	43	12	2,3	1
01-feb-92	16	5,9	12,4	7	17	5	0,9	1
02-feb-92	27	7,4	25,65	28	40	8	2,0	1
03-feb-92	11	1	34,63	8	15	4	0,8	1
04-feb-92	27	1,4	19,6	18	38	4	1,6	1

05-feb-92	27	1,5	125,8	27	45	8	2,4	1
06-feb-92	30	3	24,21	4	38	9	1,6	1
07-feb-92	53	25,2	128,61	36	87	27	4,5	2
08-feb-92	85	115,1	425,56	88	124	61	10,0	3
09-feb-92	13	14,8	66,5	11	25	11	1,6	3
11-feb-92	3	0,3	0,6	8	6	5	0,4	3

Ciclo 27

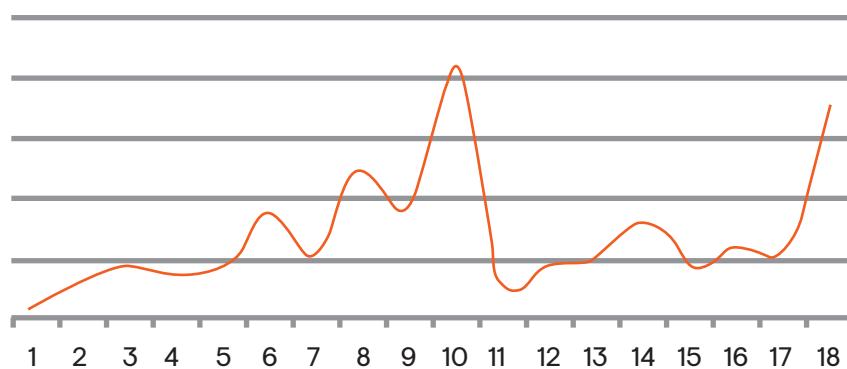
Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
15-feb-92	1	0	1,5	2	2	1	0,2	1
16-feb-92	1	0,2	0	1	2	1	0,2	1
18-feb-92	3	0	2,7	1	4	2	0,3	1
19-feb-92	9	0,1	10,9	4	13	5	1,1	1
20-feb-92	13	0,3	13,13	6	16	5	1,4	1
21-feb-92	19	6,1	27,52	11	29	9	2,9	1
22-feb-92	22	5,9	15,61	12	30	7	2,7	1
23-feb-92	42	2,5	66,7	14	61	18	5,1	2
24-feb-92	43	7,1	78,63	21	62	15	5,9	2
25-feb-92	35	3,5	82,31	30	54	17	5,7	2
26-feb-92	43	4,2	38,11	36	61	18	5,5	2
27-feb-92	60	21,9	39,9	29	84	27	7,9	3
28-feb-92	29	3,8	22,31	9	41	7	3,0	3
29-feb-92	34	7,7	66,1	13	57	15	5,1	3
01-mar-92	9	0,1	19,8	5	12	3	1,2	3
02-mar-92	6	1	7	3	13	4	0,9	3
03-mar-92	15	5,9	8,6	11	25	8	2,3	3
04-mar-92	36	9,1	38,1	20	60	17	5,1	3
05-mar-92	17	0,6	16	9	22	7	1,9	3
06-mar-92	35	1,9	56,3	36	47	26	5,5	3
07-mar-92	20	9,7	15,9	11	26	11	3,0	3

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
08-mar-92	21	2,7	8,7	4	26	10	2,0	3
09-mar-92	31	6,2	40,8	17	51	17	4,5	3
10-mar-92	24	4,7	50,8	18	36	17	4,1	3
11-mar-92	24	15	25,2	12	43	17	4,3	3
12-mar-92	15	6,7	12,1	14	30	14	2,8	3
13-mar-92	25	2,9	23,2	9	41	14	3,1	3
14-mar-92	20	2,6	25,45	19	38	13	3,2	3
15-mar-92	24	3,8	9,2	10	37	21	3,1	3
16-mar-92	23	8,6	23,75	24	50	24	4,6	3
17-mar-92	22	2,35	19,1	16	32	13	2,9	3
18-mar-92	23	4,6	14,5	13	34	14	3,0	3
19-mar-92	41	4,9	56,7	27	56	23	5,7	3
20-mar-92	44	15,95	43	45	78	43	8,2	3
21-mar-92	36	6,7	28,4	18	54	21	4,6	3
22-mar-92	47	3,8	61	39	72	30	6,9	3
23-mar-92	4	0	5,5	1	7	1	0,4	3

Ciclo 28

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
10-abr-92	1	0,4	0,5	1	2	1	0,3	1
11-abr-92	1	0	1,5	1	2	1	0,3	1
12-abr-92	3	0	4,1	2	5	1	0,7	1
14-abr-92	6	3,3	9	9	11	3	1,9	1
15-abr-92	1	0	0,5	0	1	0	0,1	1
16-abr-92	6	2,2	4,8	8	10	3	1,5	1
17-abr-92	3	0	1,2	4	4	1	0,6	1
18-abr-92	3	0,5	0,5	4	4	1	0,6	1
19-abr-92	8	0,7	7,4	2	9	6	1,6	1
20-abr-92	14	1,3	37,6	21	24	13	5,0	2
21-abr-92	1	0,1	0,2	3	1	1	0,3	2

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
22-abr-92	5	0,2	2,5	7	6	4	1,1	2
23-abr-92	9	0,6	6,8	17	21	9	2,8	2
24-abr-92	16	19,1	29,5	20	34	18	6,3	2
25-abr-92	5	0,1	0,6	4	5	4	0,9	2
26-abr-92	5	0,5	2,7	9	7	5	1,3	2
27-abr-92	3	0,1	8,5	6	5	4	1,2	2
28-abr-92	1	0	2	0	3	1	0,3	2
29-abr-92	5	30	28,6	10	14	10	4,6	2
30-abr-92	5	0,1	1,2	3	7	2	0,9	2
02-may-92	2	2	0,1	7	4	2	0,7	2
03-may-92	6	10,1	12,2	13	14	8	2,9	2
04-may-92	3	2	0,7	2	4	1	0,6	2
05-may-92	11	2,7	20,8	23	24	17	4,3	2
06-may-92	4	1	3,6	2	7	2	0,9	2
07-may-92	2	0	0,3	3	2	2	0,4	2
08-may-92	5	0,1	2,7	6	8	5	1,2	2
09-may-92	7	1,3	6	14	13	9	2,2	2
10-may-92	7	0,3	11,6	5	13	9	2,1	2
11-may-92	3	1,6	0,7	3	3	5	0,8	2
12-may-92	15	2,2	7,9	17	29	20	4,2	2
13-may-92	23	22,1	12,9	50	46	34	8,5	3
14-may-92	5	4,1	9,9	14	16	9	2,5	3
15-may-92	8	1,2	1	8	19	14	2,3	3
16-may-92	17	2,2	32,9	19	34	27	6,0	3
17-may-92	22	2	6,6	17	35	22	4,9	3
18-may-92	1	0	2,5	3	2	3	0,5	3
19-may-92	2	0	0,5	3	3	2	0,5	3
20-may-92	6	0,5	6,4	7	10	8	1,7	3
21-may-92	17	24,5	4,4	14	31	16	5,2	3
23-may-92	1	0,1	0	3	2	2	0,3	3

Ciclo 33

Día	NºInc	SupArb	SupRasa	Esp	Brig	MedMec	Oc.D.	Sector
21-jul-92	1	0	0,1	2	2	1	0,3	1
22-jul-92	3	0,45	0,25	6	6	5	1,0	1
23-jul-92	3	0,8	1,8	8	8	11	1,7	1
24-jul-92	5	0	1,22	6	7	9	1,5	1
25-jul-92	5	0,4	1,18	12	6	8	1,7	1
26-jul-92	8	1,15	3,62	16	19	21	3,5	1
27-jul-92	2	0	13,8	7	8	8	2,2	1
28-jul-92	9	1,95	23,39	12	20	20	4,9	2
29-jul-92	10	2,35	6,71	17	18	13	3,6	2
30-jul-92	23	15,4	4,47	37	44	23	8,2	3
31-jul-92	4	0,75	0,08	3	6	7	1,1	3
01-agosto-92	8	1,25	1,15	4	12	8	1,8	3
02-agosto-92	7	0,5	0,72	13	11	7	2,0	3
03-agosto-92	9	0,3	5,21	21	16	13	3,3	3
04-agosto-92	5	0,16	2,62	4	10	9	1,6	3
05-agosto-92	6	0,27	3,41	7	20	10	2,3	3
06-agosto-92	6	0	5,39	9	11	12	2,3	3
07-agosto-92	19	5,07	7,78	29	42	32	7,0	3

Tabla 10. Ciclos de Siniestralidad Altamente Progresiva. Fuente: elaboración propia.

Analizando estos ciclos y aplicando funciones que los resuman, se llega a la definición del prototipo “Siniestralidad Altamente Progresiva”:

	Sector 1	Sector 2	Sector 3
Media de días:	8	13	15
Media de incendios/día:	9	12	19
Mínimo de inc/día:	1	1	1
Máximo de inc/día:	31	53	85
Nº especialistas/día:	8	16	17
Nº de brigadas/día:	13	21	33
Nº de medios/día:	5	13	18

Estas definiciones de los 3 prototipos pueden ser útiles para establecer modelos de prevención de incendios forestales en base a la predicción de como podría evolucionar su posibilidad de ocurrencia en las fechas siguientes a una dada. Además, se puede desarrollar un sistema sofisticado de organización de recursos con reajuste diario como se puede ver en detalle en (Olivas, 2000), <http://hdl.handle.net/10578/18399>.

1.4.2. Minería de Texto: Búsqueda y Recuperación de Información (en la Web)

Como ejemplo de **Minería de Texto**, el más relevante es el que tiene que ver con los Sistemas de **Recuperación de Información**, tema actualmente muy destacado en las ciencias de la computación y la Inteligencia Artificial. Es el mejor ejemplo de tratamiento de **Datos No Estructurados**. Se pueden consultar más detalles en el libro “Búsqueda eficaz de información en la Web” (Olivas, 2011),



Enlace 10

Búsqueda eficaz de información en la web - Resumen

<http://hdl.handle.net/10915/18401>

Habitualmente, un sistema de recuperación de información es definido como el proceso que trata la representación, almacenamiento, organización y acceso de elementos de información. Es decir, es un sistema capaz de almacenar, recuperar y mantener información.

Pero podríamos plantearnos qué representa el concepto de “información” en este contexto. Se entiende por información cualquier elemento susceptible de ser recuperado, lo que incluye principalmente texto (incluidos números y fechas), imágenes, audio, video y otros objetos multimedia. Pero el tipo principal de objeto recuperable, hasta el momento siempre ha sido el texto, motivado especialmente por su facilidad de manipulación en comparación con los objetos multimedia, especialmente en lo que se refiere a capacidad de cómputo. Actualmente están surgiendo muchos sistemas que tratan de gestionar este tipo de objetos (diversos buscadores comerciales incluyen buscadores de imágenes), aunque de momento simplemente buscan en el texto de las etiquetas de dichos objetos multimedia, sin escudriñar realmente su contenido interno, lo que suele dar frecuentemente origen a engaños o falsos etiquetados.

En los sistemas de recuperación de información no se suele trabajar directamente con los documentos de texto sino con representaciones más estructuradas de los mismos. La forma de representar los documentos determina en gran medida las características del resto de elementos del sistema. Los

modelos de representación de documentos clásicos se basan generalmente en el modelo **booleano** o en el modelo **vectorial**. En el primero, cada documento es representado por un vector donde cada posición se corresponde con cada uno de los términos susceptibles de aparecer en el documento y el valor de cada posición será 0 ó 1 según ese término aparezca o no en ese documento. La esencia del modelo **vectorial** es similar, salvo que el contenido de cada componente representa algún valor que tiene que ver con la frecuencia de aparición de ese término en el documento. Claramente, estos modelos de representación de documentos son adecuados para documentos de texto, que pueden ser, por ejemplo, páginas Web u otros objetos (como elementos multimedia) que estén descritos de forma textual.

Quizá el concepto más importante en recuperación de información es el de **relevancia**. Tiene que ver con cómo medir la satisfacción de un usuario con los resultados devueltos por el sistema ante una determinada pregunta (**query**). Esta medida es claramente subjetiva, ya que ante una misma **query** y el mismo resultado (documento o lista ordenada de documentos), la relevancia puede ser totalmente distinta para dos usuarios diferentes, e imposible de medir de forma precisa. Esta es una de las razones por las que cada vez se tiene más en cuenta el papel del usuario en los sistemas de recuperación de información: si se conocen los intereses de los usuarios el sistema puede “guiar” la búsqueda de información hacia los mismos.

Un ejemplo: supongamos que dos usuarios diferentes (U1 y U2) introducen la consulta “monitor barato” en un buscador Web comercial. Si U1 habitualmente hace búsquedas en páginas de gimnasios y deportes y U2 lo hace en páginas de productos informáticos, lo más probable es que U1 esté buscando entrenadores baratos y U2 pantallas de ordenador baratas. Si se hubiesen “almacenado” de alguna forma estos “perfiles de usuario”, estas dos búsquedas podrían haber sido guiadas de formas totalmente diferentes y la relevancia de los resultados para cada usuario hubiera aumentado significativamente. Además, este ejemplo pone de manifiesto uno de los principales problemas de la recuperación de información, que es la propia complejidad del lenguaje natural. La palabra “monitor” es polisémica, lo que dificulta enormemente la tarea de recuperación de información cuando es usada.

Recuperar información no es recuperar datos

Cuando accedemos a una base de datos, por ejemplo la de una biblioteca, usamos un lenguaje muy estructurado y con una semántica muy precisa. Si buscamos libros de Lope de Vega, lo pondremos en el campo autor de la ficha que nos proporcione el sistema, y nunca tendrá en cuenta la acepción que de “vega” que tiene que ver con el paso de un río. El sistema tratará de recuperar aquellos libros de su base de datos cuyo autor es Lope de Vega. Pero si entre sus más de mil obras queremos localizar aquellas que contengan la palabra “rimas” en su título, el sistema puede transformar nuestra pregunta en una sentencia precisa de un lenguaje que entienda la base de datos (por ejemplo **SQL: Structured Query Language**), y que represente algo como “busca las obras de nuestra base de datos cuyo autor es Lope de Vega y cuyo título contenga la palabra rimas”. Con esta especificación, el sistema podría recuperar por ejemplo las obras La Circe con otras rimas y prosas (1624) y Rimas (poesías, 1604). Por el contrario, en un sistema de recuperación de información, el objeto recuperado no tiene porque adaptarse de forma exacta a las peticiones de búsqueda. La razón fundamental es que la información que gestiona un sistema de recuperación de información está en lenguaje natural, sin estructurar, por lo que puede ser semánticamente ambigua.

Etapas en el proceso de recuperación de información

Para un usuario, el proceso de recuperación de información consiste en realizar una pregunta al sistema y obtener como respuesta un conjunto de documentos ordenados. Pero en todo sistema de recuperación de información es necesaria la realización de una serie de pasos previos y diferenciados para poder llegar a sus respuestas, los más relevantes son:

4. **Indexación:** el sistema de recuperación de información crea un índice que contiene los términos que el sistema considera importantes (después de un preprocesado de cada documento) y su ubicación en los documentos.
5. **Consulta:** el usuario formula una pregunta al sistema, en un lenguaje (formalismo) procesable por éste.
6. **Evaluación:** el sistema devuelve los resultados (documentos que satisfacen en cierto grado la demanda de información del usuario), ordenados según su (posible) relevancia con respecto a la consulta formulada.
7. **Retroalimentación del usuario** (opcional): el sistema aprende de las diferentes consultas de un usuario, focalizando la recuperación según este conocimiento adquirido.

Estos procesos suelen ser estándar en todos los sistemas de recuperación de información. Además, esta clasificación no es cerrada, sino que dependiendo del sistema de recuperación pueden ser ejecutados otros nuevos métodos diferentes, por ejemplo los sistemas que utilizan estructuras de conocimiento adicionales a los índices de términos clásicos suelen tener procesos adicionales encargados de construir, mantener o actualizar dichas estructuras. Los sistemas de recuperación de información que se basan en el uso de perfiles de usuario pueden realizar una nueva etapa para la construcción y actualización del perfil de usuario (almacenando los términos que representan las preferencias de los usuarios con la finalidad de mejorar el comportamiento del sistema en futuras consultas).

Hay varias propuestas formales para definir un modelo de sistema de recuperación de información, pero una de las más utilizadas es la de Baeza-Yates (Baeza-Yates, Ribeiro-Neto, 1999) que define un sistema de este tipo como una cuádrupla $[D, Q, F, R(q_i, d_j)]$ donde:

- D es un conjunto de vistas lógicas (o representaciones) de los documentos que forman la colección.
- Q es un conjunto compuesto por vistas lógicas (o representaciones) de las necesidades de información de los usuarios. Estas vistas se denominan consultas (*queries*).
- F es una forma de modelar la representación de los documentos, consultas y sus relaciones.
- $R(q_i, d_j)$ (*ranking*) es una función de evaluación que asigna un número real al par formado por una consulta $q_i \in Q$ y la representación de un documento $d_j \in D$. Este valor determinará el orden de aparición de los documentos de una consulta q_i .

Todos estos elementos se verán reflejados en las etapas en el proceso de recuperación de información, que se detallan a continuación.

Indexación

Para conseguir D (conjunto de vistas lógicas o representaciones de los documentos que forman la colección) es necesaria la construcción de una base documental que contenga la información de los objetos que el sistema es capaz de escudriñar para poder llevar a cabo el proceso de recuperación. Si un objeto no está en esta base de datos no podrá ser recuperado. Pero lo que maneja realmente el sistema no son los propios documentos susceptibles de ser recuperados sino una representación de los mismos. Estos documentos se representan con algún formalismo, habitualmente creando un índice con el conjunto de términos significativos que aparecen en los documentos, que suele ser un subconjunto de todos los términos de los documentos. También es necesario que el sistema de recuperación de información disponga de mecanismos que permitan introducir un nuevo objeto en la base documental, o bien utilizan mecanismos automáticos que se encargan de explorar el espacio de búsqueda (en nuestro caso la Web) añadiendo al índice la información sobre los términos que aparecen en estos nuevos documentos. Esto es necesario entre otras cosas porque los mecanismos de ordenación por relevancia para el usuario (se explicarán con detalle más adelante) suelen seleccionar como documentos más relevantes, entre otros criterios, aquellos que posean con más frecuencia o en determinada posición los términos que están en la consulta del usuario.

La opción más simple sería almacenar los documentos y buscar en cada uno de ellos la existencia o no de los términos, pero este planteamiento es inviable computacionalmente debido a la enorme cantidad de información que sería necesario manejar, lo que imposibilitaría que el sistema fuese eficiente. Por tanto, es necesaria la utilización de estructuras que almacenen información sobre los documentos y que permitan realizar las búsquedas en tiempos razonablemente cortos. Estas estructuras es lo que hoy en día se denominan índices. Pero habitualmente los documentos suelen ser preprocesados antes de ser indexados para reducir el número de elementos (términos, signos de puntuación, ubicación de los términos...) a tener en cuenta y por tanto mejorar la eficiencia del proceso de recuperación. Está claro que esta reducción de elementos supone una pérdida de información, que puede ser muy importante a la hora de buscar los documentos más relevantes ante una consulta, por lo que el secreto del éxito en esta etapa radica en encontrar el equilibrio justo entre la eliminación de elementos a indexar de los documentos y la eficiencia de los procesos de búsqueda en este índice.

Preprocesado de documentos

Las tareas que se utilizan habitualmente en el preprocesado de documentos para su indexación son las siguientes:

- *Eliminación de signos de puntuación:* se eliminan los acentos, comas, puntos y demás signos de puntuación con el fin de tratar los términos de forma uniforme. Este proceso tiene los inconvenientes de que se pierde esta información y no se podrán utilizar signos de puntuación en las consultas de los usuarios y que los signos de puntuación poseen información semántica importante. Por ejemplo:

Documento original: “vivía cerca de León. Furioso con lo que le rodeaba, los hombres de la comarca lo odiaban”

Documento sin signos de puntuación: “vivía cerca de león furioso con lo que le rodeaba los hombres de la comarca lo odiaban”.

- **Eliminación de palabras prohibidas (stop words):** en todos los idiomas hay un conjunto de palabras muy frecuentes que se usan por cuestiones lingüísticas de concordancia sintáctica entre palabras y frases. Si se considera que estas palabras no aportan ningún significado a un documento (cosa que no es cierta), sino que sólo se utilizan para seguir las reglas del idioma, podrían ser eliminadas. Por ejemplo en español podrían eliminarse artículos, preposiciones, conjunciones, algunos adverbios, etc.

Existen listas para los diferentes idiomas (denominadas **stoplists**) con estas palabras, que sirven como referencia para no tener en cuenta estas palabras cuando aparezcan en los documentos a la hora de ser indexados. También es frecuente la construcción dinámica de estas listas cuando se desarrollan sistemas de recuperación de información. En el ejemplo se puede ver claramente cómo cambia la semántica del texto tras eliminar estas palabras:

Documento original: “vivía cerca de León. Furioso con lo que le rodeaba, los hombres de la comarca lo odiaban”.

Documento sin signos de puntuación y sin palabras prohibidas: “vivía león furioso rodeaba hombres comarca odiaban”.

- **Stemming o lematización:** consiste en obtener la raíz léxica (o **stem**) de una palabra. Normalmente se ignoran las diferentes variaciones morfológicas que puede tener a la hora de indexar. En la mayoría de los casos la raíz es una palabra sin significado, como por ejemplo en el caso de “vivía” y “vivencia”, cuya raíz común sería “viv”. Este mecanismo se suele aplicar para eliminar el sufijo de una palabra, pero no se aplica al prefijo. Esto se debe a la idea de que la raíz contiene la fuerza semántica de la palabra, y que los sufijos introducen ligeras modificaciones del concepto o tienen meramente funciones sintácticas.

El objetivo inicial de este proceso de **lematización** fue mejorar el rendimiento de los sistemas de recuperación de información al reducir el número de palabras que un sistema tenía que almacenar en el índice. Otra de las características de la **lematización** es que frecuentemente favorece la exhaustividad (**recall**) en la búsqueda, es decir, se recuperan más palabras relacionadas léxicamente al tener la misma raíz y por tanto se obtiene un conjunto más elevado de términos, evitándose así perder términos potencialmente relevantes. Pero esto es a costa de una reducción en la precisión, porque los lenguajes naturales no suelen ser regulares en sus construcciones y además, hay muchas palabras con la misma raíz cuyo significado no tiene nada que ver.

Por ejemplo, podría suceder que se indexen bajo la raíz “cas” los términos “casa”, “casero” y “casual”. Este fenómeno se denomina **sobrelematización**. También es posible que el mecanismo de **lematización** falle y obtenga raíces distintas para dos palabras semánticamente

similares. A esta situación se la denomina *bajolematización*. Este caso es muy frecuente en los verbos irregulares, y se podría dar por ejemplo con dos variaciones del verbo “haber”, para las palabras “habido” y “hayamos”, indexándose bajo raíces distintas (“habí” y “hay”).

Otro problema es que este método es dependiente del idioma, y por lo tanto sería necesario a la hora de indexar utilizar un mecanismo específico para cada idioma. Esta situación lleva asociada la utilización de una técnica para determinar el idioma. Además, este tipo de métodos funcionan bien con idiomas que tengan una sintaxis no excesivamente complicada, como el inglés, pero en cambio fallan mucho más con otro tipo de idiomas más complejos, como el español. Por tanto, la *lematización* difiere mucho dependiendo de los distintos idiomas.

- Existen muchas técnicas utilizadas en este tipo de métodos, destacando la utilización de reglas y diccionarios. Existen multitud de propuestas de *lematización* basadas en reglas, la mayoría de ellas para el idioma inglés, de los cuales el clásico es el más sencillo, el *lematizador S*, que simplemente quita las terminaciones plurales. El método de *lematización* más famoso es el que se ha implementado en el algoritmo de Porter, desarrollado en la década de los 80, que elimina cerca de 60 terminaciones en cinco etapas, en cada una de las cuales se elimina un tipo concreto. Para eliminar los errores más frecuentes descritos, se han desarrollado mecanismos basados en diccionarios, como *KSTEM*. Hay mucha polémica sobre la efectividad de la *lematización* y algunos autores afirman que esta técnica mejora la precisión y exhaustividad de las búsquedas en tanto las consultas (y también los documentos) sean más cortas.

Finalmente, se puede hablar del uso de los *n-gramas* (subsecuencia de n elementos de una secuencia dada). Al trabajar con *n-gramas* se ignora el aspecto semántico de las palabras y la hipótesis de los mecanismos basados en n-gramas es que dos palabras relacionadas semánticamente suelen contener los mismos caracteres.

Siguiendo con el ejemplo:

Documento original: “vivía cerca de León. Furioso con lo que le rodeaba, los hombres de la comarca lo odiaban”.

Documento sin signos de puntuación, sin palabras prohibidas y tras un proceso de lematización: “viv león furios rodea hombr comarc odiaba”.

- *Eliminación de documentos duplicados*: muchos contenidos de las páginas Web están multiplicados (como mínimo duplicados) en diferentes sitios. La eliminación de estos documentos multiplicados permite mejorar el rendimiento de los programas encargados de la indexación y reducir el espacio de almacenamiento que ocupan los índices generados. Pero la tarea de identificar documentos iguales o similares no es trivial, ya que pueden darse diferentes situaciones que compliquen esta labor, como por ejemplo variaciones en el formato del documento. Dos documentos pueden ser idénticos en contenido, pero estar en diferentes formatos (*html, pdf, Word...*). Una de las formas de detectar la similitud entre documentos consiste en convertirlos a un mismo formato, normalmente texto plano, utilizando alguna herramienta de conversión estándar. Posteriormente cada documento se

divide en una colección de partes o trozos formados por pequeñas unidades de texto (por ejemplo líneas o sentencias). Después, a cada trozo se le aplica una función *hash* para obtener un identificador único. Si dos documentos comparten un número de trozos con igual identificador por encima de un umbral T, entonces se consideran documentos similares.

Estructuras de indexación clásicas

En los primeros sistemas, los índices se limitaban a contener un conjunto de palabras clave representativas del documento, pero actualmente el número de términos ha crecido demasiado. Como se ha dicho, en la indexación no se utilizan todos los términos (aunque hay excepciones), sino que se suele usar un subconjunto de términos y el documento completo se almacena aparte en repositorios o caches si es posible, pero lo más habitual es que solo se almacene su ubicación (normalmente su *URL: Universal Resource Locator*). La estructura más utilizada en la indexación de documentos es el archivo invertido, formada por dos componentes: el vocabulario y las ocurrencias. El vocabulario es el conjunto de todas las palabras diferentes del texto. Para cada una de las palabras del vocabulario se crea una lista donde se almacenan las apariciones de cada palabra en un documento. El conjunto de todas estas listas se llama ocurrencias (Baeza Yates, 1999). Este mecanismo no es el único, sino que existen otros muchos como los ficheros de firmas, basados en técnicas *hash*, *árboles PAT* y *grafos*. Este es un ejemplo de fichero invertido, después de que los documentos iniciales hayan sido preprocesados:

Documento	Texto	
1	“...vivía cerca de León. Furioso con lo que le rodeaba, los hombres de la comarca lo odiaban...”	
2	“... no tiene tanta furia el león como ...”	
3	“...León pertenece a Castilla...”	
4	“...los hombres y las comarcas castellanas...”	
Cada línea es un documento diferente.		
Número de índice	Término	Documento
1	vivía	1
2	león	1, 2, 3
3	furi (furioso, furia)	1, 2
4	rodeaba	1
5	hombres	1, 4
6	comarca (comarca, comarcas)	1, 4
7	odiaban	1
8	cast (castilla, castellanas)	3, 4
Fichero invertido para los documentos de la tabla a, entre paréntesis se presentan las palabras antes de la lematización		

Tabla 11. Ejemplo de fichero invertido. Fuente: elaboración propia.

Consulta

El inicio de un proceso de búsqueda lo origina un problema que requiere información para poder resolverse. La carencia de esta información depende de la amplitud de conocimiento de cada usuario. Un usuario avezado en un tema concreto tendrá más claro que información solucionaría su problema y seguramente lo encontraría en un plazo de tiempo más corto. La aparición de un problema conlleva la demanda de información en el usuario para solucionarlo, y esta carencia de información origina lo que se denomina una “necesidad de información”.

Las personas buscan información basándose en su conocimiento previo, que es muy diferente de unas a otras. La necesidad de información puede ser definida como la representación implícita de un problema en la mente de los usuarios. Se diferencia del problema, ya que cada usuario percibe las cosas de diferente forma, y ante un mismo problema varios usuarios pueden construir necesidades de información distintas.

Las necesidades de información se pueden clasificar en necesidades verificativas, sobre temas conscientes e imprecisas o mal definidas. La primera categoría se refiere a la situación en la que se buscan documentos con propiedades conocidas, por ejemplo se conoce el nombre del autor, el título, etc. En el segundo tipo se conoce el tema y es definible, pero menos exacto que en la primera categoría. En esta categoría una persona que busca información tiene algún nivel de comprensión de lo que busca. La tercera categoría son los casos en los que una persona desea encontrar nuevo conocimiento en dominios que no le resultan familiares.

Una necesidad de información se puede satisfacer de distintas formas. Es decir, el concepto de necesidad de información tiene una naturaleza ambigua. Debido a esta característica, se han comentado distintos problemas cuyo motivo es la inexactitud de la necesidad de información, como el problema ASK (*Anomalous State of Knowledge*), ISK (*Incomplete State of Knowledge*) y USK (*Uncertain State of Knowledge*).

Cuando se aborda el desarrollo de un sistema de recuperación de información se asume la idea de que las necesidades de información pueden describirse. La persona que quiere recuperar la información tiene que ser capaz de expresar la necesidad de información que demanda en forma de una petición o consulta (*query*). La petición es una representación de la necesidad de información del usuario en un lenguaje humano, casi siempre en lenguaje natural (no estructurado, como se ha comentado anteriormente en la recuperación de datos).

Pero esta consulta debe ser también comprensible y procesable para el sistema de recuperación de información. Evidentemente, la representación mental de la información que el usuario necesita para resolver su problema difiere enormemente de la información que recibe el SRI del usuario. Este proceso implica una adaptación de lo que el usuario cree que resolverá su problema a una expresión que represente lo que el usuario necesita encontrar.

Pero no basta con seguir este proceso para obtener la información que resuelva el problema. Si los resultados no satisfacen al usuario puede ser necesario repetir este proceso de forma cíclica. Durante cada ciclo el sistema recibe realimentación del usuario con nueva información, formalizada en forma de nuevas consultas. En este proceso se pueden distinguir a grandes rasgos 4 fases:

1. Fase **explorativa**. El usuario reúne la información que pueda serle útil en el proceso de búsqueda.
2. Fase **constructiva**. Se aprovecha la información adquirida en la fase anterior para reformular una nueva consulta.
3. Fase de **realimentación**. Si los resultados de la consulta formulada en la fase 2 no son satisfactorios es necesario volver a realizar las fases 1 y 2 para refinar el resultado.
4. Fase de **presentación**. Se limita a la forma de representar los resultados.

Evaluación

Los algoritmos de evaluación que utilizan muchos de los buscadores actuales se basan en la estructura de la web para determinar la relevancia de las páginas que deben ordenar. Estos algoritmos de evaluación se denominan “algoritmos basados en enlaces” y los más usados son estos tres: *PageRank* (utilizado en el buscador Google), *HITS* (*Hypertext Induced Topic Selection*) y *SALSA*.

Los algoritmos basados en enlaces consideran la Web como un grafo dirigido de páginas y enlaces: una página con muchos enlaces a ella se supone que es una página de alta calidad, especialmente si (circularmente) los enlaces vienen de páginas que son a su vez de alta calidad. Por tanto, se puede considerar a la Web como un grafo dimitido $G = (P, E)$ donde P son los nodos o páginas web y E los enlaces entre las páginas.

Este tipo de mecanismos sufren el “efecto de la contribución circular”. Este efecto se basa en el hecho de que las páginas Web se pueden enlazar unas a otras, de forma que se produzca un camino circular entre ellas. Por tanto, cada página estimula la evaluación de las que enlaza, y si existe un camino circular, entonces estimula su propia evaluación indirectamente. Para tratar de evitar este problema, es frecuente proponer la aplicación del concepto de “distancia en la Web”, de forma que se asignen pesos a los enlaces en función de la importancia de la página enlazada.

Estos métodos presentan el inconveniente de que son potencialmente vulnerables a ataques del tipo *link spamming*, como por ejemplo (en este caso se suele denominar *Google bombing*). Hace algún tiempo, cuando se introducía en Google el término “ladrones” en la lista de resultados obtenida el primer puesto lo ocupaba, antes de tomarse las medidas oportunas, la página de la SGAE (Sociedad General de Autores Españoles) debido a que mucha gente se había puesto de acuerdo para enlazarla en sus páginas utilizando el término “ladrones”, (por la baja popularidad de la SGAE por el cobro de cánones en la adquisición de material informático).

El algoritmo de ordenación (*ranking*) *PageRank* (nombre propuesto por Larry Page, fundador de Google) define un camino aleatorio con saltos aleatorios sobre la Web (completa). Así, la puntuación *PageRank* de una página se puede interpretar como global, evaluando la importancia de cada página independiente del tema.

En cambio, *HITS* y *SALSA* son específicos a un tema y se pueden considerar como algoritmos de evaluación locales. Estos dos algoritmos funcionan utilizando una pequeña porción de la Web donde los recursos correspondientes de un tema específico es probable que existan, analizando la estructura de enlaces de ese subgrafo Web y asignando a sus páginas puntuaciones *hub* y autoridad. Una página

es una autoridad en un tema si contiene información valiosa y de alta calidad sobre ese tema. Una página es un “*hub*” sobre un tema si enlaza a buenas autoridades sobre el tema, si es por ejemplo una lista de recursos de calidad sobre ese tema.

Además de la estructura de los enlaces, también se suelen tener en cuenta otras características a la hora de evaluar una página. Por ejemplo, Google tiene en cuenta el texto que acompaña a cada enlace, ya que se supone que da una descripción general o el nombre de la página a la que enlaza. Esto tiene varias ventajas ya que permite obtener una descripción bastante exacta de la página, además permite recuperar documentos que no estén basados en texto como por ejemplo imágenes, programas o bases de datos. Otros aspectos que se suelen tener en cuenta son el título de la página, el tamaño de la fuente empleada, etc.

PageRank

La puntuación **PageRank** de una página A, denotada como $PR(A)$, es la probabilidad de visitar A en un camino aleatorio que implique a toda la Web, donde el conjunto de estados del camino aleatorio es el conjunto de páginas, y cada paso aleatorio es de uno de estos tipos: elegir una página Web aleatoriamente, y saltar a ella o desde un estado s dado, elegir aleatoriamente un enlace saliente de s y seguir ese enlace hasta la página de destino. Larry Page y Sergey Brin (Brin, 1998), creadores de Google, describen el cálculo del algoritmo **PageRank** de la siguiente forma: se asume que la página A tiene las páginas $T_1 \dots T_n$ que apuntan a ella. El parámetro d es un factor que puede tomar valores comprendidos entre 0 y 1. Normalmente se establece d con el valor 0.85. Además, $C(A)$ se define como el número de enlaces que salen de la página A. El valor **PageRank** de una página A se determina como sigue:

$$PR(A) = (1 - d) + d(PR(T_1)/C(T_1) + \dots + PR(T_n)/C(T_n))$$

Hay que destacar que PageRank establece una distribución de probabilidad sobre las páginas Web, de tal modo que la suma de todos los valores PageRank de las páginas Web serán uno. El valor **PageRank** o $PR(A)$ se puede calcular utilizando un algoritmo iterativo. La idea sobre la que se basa **PageRank** es bastante intuitiva. Asume que si una página recibe bastantes enlaces provenientes de otras páginas, entonces se supone que esa página merece ser visitada. No obstante, también tiene en cuenta el hecho de que páginas muy importantes enlacen a otra página, lo que implica que es probable que esa página sea digna de ser visitada al ser enlazada por una página de calidad. A grandes rasgos, se podría decir que el algoritmo **PageRank** mide la probabilidad de que un usuario visite una página Web. El factor d es la probabilidad de que un visitante que navega en una página se aburra de ella y solicite otra.

HITS (Hypertext Induced Topic Selection)

HITS se basa en un modelo de la Web que distingue *hubs* y *autoridades*. Cada página tiene asignado un valor “*hub*” y un valor “*autoridad*”. El valor *hub* de la página H está en función de los valores de autoridad de las páginas que enlaza H, el valor autoridad de la página A está en función de los valores *hub* de las páginas que enlazan a A. Por tanto, según **HITS** cada página tiene un par de puntuaciones: una puntuación *hub* (h) y una puntuación *autoridad* (a), basadas en los siguientes principios: La calidad de un *hub* se determina mediante la calidad de las autoridades que le enlazan. La calidad de

una **autoridad** se determina mediante la calidad de los ***hubs*** a los que enlaza. Por tanto, el algoritmo **HITS** establece que una página tiene un alto peso de “**autoridad**” si recibe enlaces de muchas páginas con un alto peso de “**hub**”. Una página tiene un alto peso “**hub**” si enlaza con muchas páginas **autoritativas**.

SALSA (Stochastic Approach for Link Structure Analysis)

SALSA también asigna dos puntuaciones a cada página: la puntuación ***hub*** y **autoridad**. Estas puntuaciones se basan en dos caminos aleatorios realizados en G, el camino autoridad y el camino ***hub***. Intuitivamente, el camino **autoridad** sugiere que las páginas **autoritarias** deberían ser visibles (enlazadas) **desde** muchas páginas. Así, un camino aleatorio de este subgrafo visita aquellas páginas con alta probabilidad.

Formalmente, el estado del camino **autoridad** son los nodos de G con al menos un enlace de entrada. Sea v un nodo, y q_1, \dots, q_k los nodos que enlazan con v . Una transición desde v implica elegir un índice aleatorio i uniformemente sobre $\{1, 2, \dots, k\}$, y seleccionar un nuevo estado desde los enlaces salientes de q_i (de nuevo, aleatoriamente y uniformemente).

Así, la transición implica atravesar dos enlaces Web, el primero de ellos se atraviesa al revés (desde el destino al origen) y el segundo se atraviesa hacia delante. Si π denota la distribución estacionaria del camino aleatorio descrito anteriormente, cuando la distribución inicial es uniforme sobre todos los estados. La puntuación de cada página (=estado) v es πv (las páginas que no tienen enlaces de entrada alcanzarán una puntuación 0).

Cabe destacar el efecto TKC (*Tightly-Knit Community*) que pone de manifiesto importantes diferencias entre los algoritmos **HITS** y **SALSA**. **HITS** favorece a los grupos de páginas que tienen muchas cocitaciones “internas”, mientras **SALSA** prefiere las páginas con muchos enlaces de entrada. Una comunidad estrechamente tejida (*tightly-knit community*) es un conjunto de páginas pequeño pero sumamente interconectado. El efecto TKC se da cuando dichas colecciones de páginas (comunidades estrechamente tejidas) obtienen evaluaciones altas en los algoritmos basados en los enlaces, aunque esas páginas no sean autoridad en el tema, o sólo conciernan a un aspecto de dicho tema.

Las técnicas de Soft Computing en la Recuperación de Información

La Teoría de Conjuntos Borrosos fue introducida por Lotfi A. Zadeh (Azerbaiyán, 1921-2017, en sus últimos años fue profesor emérito de la Universidad de California en Berkeley) a mediados de los años 60. Previamente, Max Black (1909 - 1989), en un artículo de 1937 titulado “*Vagueness: An exercise in Logical Analysis*” y Karl Menger (1902 - 1985) con los artículos de 1942 “*Statistical Metrics*” y los de los años 50 sobre relaciones borrosas de indistinguibilidad, sentaron las bases de lo que hoy es una teoría tan utilizada y con tan buenos resultados. Se puede ver el artículo divulgativo completo “La lógica borrosa y sus aplicaciones” (Olivas, 2002) en (enlace 2)



Enlace 11

La lógica borrosa y sus aplicaciones

<https://docplayer.es/6702064-La-logica-borrosa-y-sus-aplicaciones.html>

Bajo el concepto de Conjunto Borroso (*Fuzzy Set*) reside la idea de que los elementos clave en el pensamiento humano no son números sino etiquetas lingüísticas. Estas etiquetas permiten que los objetos pasen de pertenecer de una clase a otra de forma suave y flexible.

La Lógica Borrosa se puede inscribir en el contexto de la *Lógica Multivaluada*. En 1922 Lukasiewicz cuestionaba la Lógica Clásica bivaluada (valores cierto y falso). Además, adelantaba una lógica de valores ciertos en el intervalo unidad como generalización de su lógica *trivaluada*. En los años 30 fueron propuestas lógicas multivaluadas para un número cualquiera de valores ciertos (igual o mayor que 2), identificados mediante números racionales en el intervalo [0, 1].

Uno de los objetivos de la Lógica Borrosa es proporcionar las bases del razonamiento aproximado que utiliza premisas imprecisas como instrumento para formular el conocimiento.

- **¿Qué son los conjuntos borrosos? y el Soft-Computing**

En un conjunto clásico (*crisp*) se asigna el valor 0 ó 1 a cada elemento para indicar la pertenencia o no a dicho conjunto. Esta función puede generalizarse de forma que los valores asignados a los elementos del conjunto caigan en un rango particular, y con ello indiquen el grado de pertenencia de los elementos al conjunto en cuestión. Esta función se llama “**función de pertenencia**” y el conjunto por ella definida “**Conjunto Borroso**”. La función de pertenencia

A por la que un conjunto borroso A se define, siendo [0, 1] el intervalo de números reales que incluye los extremos, tiene la forma:

$$\mu_A : X \rightarrow [0, 1]$$

Es decir, mientras que en un conjunto clásico los elementos pertenecen o no pertenecen a él totalmente (por ejemplo un número puede pertenecer o no al conjunto de los pares, pero no pertenecerá con un determinado grado), en los conjuntos borrosos hay grados de pertenencia en referencia a un universo local. Por ejemplo, en el contexto de nuestra sociedad actual una persona de 45 años pertenecerá al conjunto borroso “viejo” con un grado supongamos de 0.5. Si en vez de usar de referencia nuestra sociedad actual aludimos a una sociedad donde la esperanza de vida fueran 40 años este grado cambiaría.

Desde que en el año 1965 el profesor **Lotfi A. Zadeh** introdujo la Lógica **Borrosa** o **Difusa** (*Fuzzy Logic*), muchas han sido sus aplicaciones, las más importantes en el campo del control industrial (lavadoras Bosch con sistema ECO-Fuzzy, ABS de Nissan, Aire Acondicionado Mitsubishi...). Pero, ya desde sus inicios, la intención del profesor Zadeh era introducir un formalismo capaz de representar y manipular la incertidumbre e imprecisión inherentes al lenguaje natural.

Por otro lado, la mayor parte de la inmensa cantidad de información contenida en Internet está almacenada en documentos textuales en lenguaje natural en multitud de idiomas.

El profesor Zadeh acuñó el término Soft-computing. La computación suave se diferencia de la computación convencional (dura) en que, a diferencia de ella, es tolerante a la imprecisión, la incertidumbre y la verdad parcial. El modelo a seguir para el soft computing es la mente humana. El principio rector de la computación suave es: Aprovechar la tolerancia a la imprecisión, la

incertidumbre y la verdad parcial para lograr la trazabilidad, la robustez y el bajo coste de las soluciones. Las ideas básicas que subyacen a la computación suave en su estado actual tienen vínculos con muchas influencias anteriores, entre ellas los conjuntos difusos introducidos en los años 60 del siglo pasado, los trabajos de los 70 sobre el análisis de sistemas complejos y procesos de decisión y los de los 80 sobre teoría de posibilidades y análisis de datos blandos. La inclusión de la teoría de redes neuronales en la computación suave llegó en un momento posterior. En esta coyuntura, los principales componentes de la computación suave (SC) son la lógica borrosa (FL), la teoría de redes neuronales (NN) y el razonamiento probabilístico (PR), con este último subsumiendo las redes de creencias, los algoritmos genéticos, la teoría del caos y partes de la teoría del aprendizaje. Lo que es importante tener en cuenta es que el SC no es una mezcla de FL, NN y PR. Se trata más bien de una asociación en la que cada uno de los socios aporta una metodología distinta para abordar los problemas en su ámbito. En esta perspectiva, las principales contribuciones de FL, NN y PR son complementarias y no competitivas.

- **Implicaciones del soft computing**

La complementariedad de FL, NN y PR tiene una consecuencia importante: en muchos casos un problema puede resolverse de forma más eficaz utilizando FL, NN y PR en combinación y no exclusivamente. Un ejemplo llamativo de una combinación particularmente efectiva es lo que se ha llegado a conocer como sistemas *neurofuzzy*. Estos sistemas son cada vez más visibles como productos de consumo, desde acondicionadores de aire y lavadoras hasta fotocopiadoras y videocámaras. Menos visibles, pero quizás aún más importantes, son los sistemas neuroprotectores en aplicaciones industriales. Lo que es particularmente significativo es que tanto en los productos de consumo como en los sistemas industriales, el empleo de técnicas de *soft computing* conduce a sistemas que tienen un alto *MIQ (Cociente de Inteligencia de Máquinas)*. En gran medida, es el alto MIQ de los sistemas basados en SC lo que explica el rápido crecimiento en el número y variedad de aplicaciones de la computación suave y especialmente de la lógica difusa. La estructura conceptual de la computación suave sugiere que los estudiantes deben ser entrenados no solo en teoría de redes neuronales o lógica difusa o razonamiento probabilístico, sino en todas las metodologías asociadas, aunque no necesariamente en el mismo grado. Lo mismo se aplica a las revistas, libros y conferencias. Estamos empezando a ver la aparición de revistas y libros con *soft computing* en su título. Una tendencia similar es visible en los títulos de las conferencias.

A la hora de clasificar las diferentes líneas de investigación relacionadas con la recuperación de información, y más en concreto, con las posibilidades de las técnicas de Soft-Computing en la recuperación de Información en Internet, se pueden utilizar diferentes criterios.

Una posibilidad es realizar una clasificación en función de la parte del proceso de recuperación de información en la que están centradas. En este caso podemos distinguir los siguientes grupos de enfoques:

- Modelos de representación lógica de documentos

- Lenguajes de especificación de consultas
- Sistemas de evaluación de consultas
- Sistemas de presentación y clasificación de resultados
- Sistemas de retroalimentación de consultas

Otra posibilidad es distinguir los enfoques según las técnicas utilizadas:

- Utilización de ontologías
- Estudio de asociaciones y relaciones entre términos
- Construcción y utilización de perfiles de usuarios
- Utilización de algoritmos de *clustering* y clasificación

Atendiendo al objetivo y el ámbito de aplicación de los sistemas de recuperación de información, podemos distinguir diferentes tipos de sistemas:

- Sistemas de búsquedas basados en consultas
- Sistemas basados en directorios dinámicos
- Sistemas de preguntas-respuestas (*Question-Answering systems*)
- Búsquedas conceptuales

Todas estas categorías podrían ser a su vez subdivididas en función de si en su realización se han considerado tecnologías típicas de Soft-computing como Lógica Borrosa y técnicas de Inteligencia Artificial. Las posibilidades de estas técnicas en el campo de la recuperación de información están claramente constatadas en numerosos estudios.

Por último señalar que el propio profesor Zadeh, en los seminarios de BISC (*Berkeley Initiative in Soft-Computing*) siempre resaltaba la necesidad y actualidad de la línea propuesta. Para él, los motores de búsqueda existentes (con Google en la cúspide) tienen muchas capacidades notables; pero lo que no está entre ellos es la capacidad de deducción, la capacidad de sintetizar una respuesta a una consulta desde diferentes repositorios de información. En los últimos años, se han realizado progresos impresionantes en la mejora del rendimiento de los motores de búsqueda mediante el uso de métodos basados en la lógica bivalente y en la teoría de la probabilidad basada en la lógica bivalente. Pero ¿se pueden utilizar estos métodos para añadir una capacidad de deducción no trivial a los motores de búsqueda, es decir, para actualizar los motores de búsqueda a sistemas de respuesta a preguntas? Una opinión que se es que la respuesta es "No". El problema tiene sus raíces en la naturaleza del conocimiento mundial, el tipo de conocimiento que los seres humanos adquieren a través de la experiencia y la educación.

Se reconoce ampliamente que el conocimiento del mundo desempeña un papel esencial en la evaluación de la pertinencia, la síntesis, la búsqueda y la deducción. Pero un tema básico que no se aborda es que gran parte del conocimiento mundial está basado en la percepción, por ejemplo, "es difícil encontrar estacionamiento en París", "la mayoría de los profesores no son ricos" y "es poco probable que llueva en pleno verano en San Francisco". El problema es que (a) la información basada en la percepción es intrínsecamente borrosa y (b) la lógica bivalente es intrínsecamente inadecuada para tratar con la confusión y la verdad parcial.

Para enfrentarse a la imprecisión del conocimiento del mundo, se necesitan nuevas herramientas, como puede ser el Lenguaje Natural Preciso (PNL). PNL se basa en una lógica difusa y tiene la capacidad de tratar con la parcialidad de la certeza, la parcialidad de la posibilidad y la parcialidad de la verdad. Estas son las capacidades que se necesitan para poder aprovechar el conocimiento mundial para la evaluación de la pertinencia y para la síntesis, la búsqueda y la deducción.

A continuación se pretende describir cuál es el papel que puede jugar la *lógica borrosa* como técnica de *Soft-Computing* para mejorar la búsqueda en este tipo de herramientas. La lógica borrosa puede proporcionar herramientas para la extracción y uso de conocimiento procedente de tesauros y ontologías, permite formalizar sentencias e implementar capacidades de deducción en sistemas de tipo *Pregunta - Respuesta*, combinar valores borrosos y diferentes lógicas, mejorar algoritmos de *clustering* y manejar las diferentes arquitecturas de un *Metabuscador*.

Los principales buscadores de la red basan sus criterios de búsqueda en aspectos léxicos y en algunos casos semánticos con respecto a los términos de la consulta. Debido a esto, son muchas las formas por las que se han tratado de mejorar los resultados de las búsquedas Web, y es aquí donde las técnicas de Soft-Computing pueden tener un papel importante. Prueba de ello es que en los últimos años se han propuesto múltiples soluciones basándose en este tipo de técnicas: soluciones dirigidas a la construcción de sitios flexibles y adaptables (basados en patrones Web, perfiles de usuario, patrones de acceso, patrones de comportamiento de usuarios...) que utilizan técnicas de **Data Mining**, otras centradas en la organización por grupos de documentos recuperados (destacando la importancia del uso de los algoritmos de *clustering* en lugar de los algoritmos que se basan en los grupos temáticos predefinidos) o en otro tipo de aproximaciones que incluyen sistemas basados en lenguajes de consulta flexibles, o sistemas basados en reglas de asociación borrosa que ayudan al usuario a encontrar nuevos términos que utilizar en su consulta.

Por otro lado, existen distintos tipos de aproximaciones que se centran en la representación de documentos, para lo que utilizan, en la mayoría de los casos, extensiones del *modelo espacio vectorial* estándar. También es frecuente encontrar sistemas que utilizan las interrelaciones entre términos almacenadas en tesauros y ontologías como *WordNet*, para construir redes semánticas de grupos de palabras.

Los Metabuscadores surgen como una herramienta muy prometedora cuyo objetivo es el de mejorar los resultados de la búsqueda Web, para ello utilizan varios de los mejores motores de búsqueda como Google o Yahoo para después hacer una selección de los mejores resultados dados por dichas fuentes. Todos estos tipos de sistemas son muy diferentes a los que proponía Zadeh que, como hemos visto, se centraban en el desarrollo de sistemas Pregunta-Respuesta, un punto de vista muy interesante en problemas de recuperación de información.

Actualmente no hay muchos buscadores comerciales con propiedades borrosas. Tanto las técnicas de **Soft-Computing** en general como la de la Lógica Borrosa en particular pueden jugar un papel importante en los problemas relacionados con la búsqueda basada en tecnologías Web. A continuación se relatan varios aspectos que podrían ser mejorados a través de las técnicas de **Soft-Computing**.

Si hacemos la consulta “**buscador borroso**” en Google, serán muchos los resultados que aparezcan, pero seleccionando aquellos que consideremos más relevantes, podemos observar que la idea de borrosidad que algunos buscadores comerciales implementan radica únicamente en el uso de funcionalidades de **matching** sintáctico con propiedades borrosas, es decir, tratan de corregir las palabras posiblemente mal escritas por el usuario mediante el uso de algún diccionario específico que contenga el buscador o al cual puede tener acceso, y en el cual aparezcan las palabras que busca el usuario escritas de forma correcta. Como resultado, el buscador mandará una señal de aviso (texto escrito) que diría algo de la forma: *¿quiso usted decir?* Aunque realmente esta funcionalidad es borrosa, es demasiado pretencioso hablar de buscador borroso simplemente por este detalle. Más concretamente, en buscadores famosos, el operador borroso de búsqueda permite al usuario escribir un trozo de la palabra que busca cuando no sabe a ciencia cierta como se deletrea la palabra completa.

Por ejemplo, muchos de los términos médicos o farmacéuticos son difíciles de deletrear, quizás sabes cómo suena la palabra pero realmente no sabes cómo se escribe. Tanto la búsqueda borrosa como los operadores **wild card** permiten encarar este problema. Por ejemplo, el buscador **Netscape Search** implementa funcionalidades borrosas, así si un usuario buscara el famoso antibiótico “amoxicilina” podría hacer uso del operador borroso de búsqueda. Escribiría el trozo de palabra que supiera deletrear con certeza acompañado del símbolo ~ y el resto final de la palabra que ya no sabe si lo está deletreando bien o mal. Así, si hiciéramos la consulta: “**amoxi~lilina**”, el buscador devolvería satisfactoriamente aquellos documentos que contuvieran coincidencias del término buscado. Es similar a la búsqueda basada en caracteres **wild card**. Es posible combinar búsquedas en las que intervengan diferentes variantes de una palabra o deletreos incorrectos. Por ejemplo, muchos buscadores permiten tres caracteres **wild card**: el símbolo del dólar (\$), la interrogación (?) y el asterisco (*).

El papel de la lógica borrosa en los Metabuscadores

Los **Metabuscadores específicos** son hoy en día quizá una de las aplicaciones más usadas en lo que tiene que ver con el **acceso y análisis de datos**. Si observamos la publicidad en televisión nos damos cuenta de que gran parte de los anuncios son de metabuscadores específicos: elección de seguros, vuelos, viajes hoteles... Los metabuscadores de carácter general no son tan abundantes debido a las dificultades que entraña su desarrollo y explotación.

Un **Metabuscador** es un motor de búsqueda que, a diferencia de los buscadores conocidos, no tiene una base de datos propia donde indexar documentos. Otra desventaja con respecto a los motores de búsqueda es que son más lentos debido a que, además de no tener base de datos propia, tienen que realizar un proceso de selección y elaboración de la lista de resultados bastante complejo. Sin embargo, disponen de un interfaz que permite al usuario consultar a la vez en diferentes motores de búsqueda, es decir, el Metabuscador se encarga de recibir la petición del usuario, enviarla a diferentes buscadores y mostrarle después los resultados.

El principal problema que tienen los Metabuscadores consiste en combinar las listas de resultados devueltos por los motores de búsqueda que utiliza, ya que en este proceso se deben clasificar y ordenar los documentos según su relevancia. Sin embargo, este tipo de sistemas tienen ciertas mejoras con respecto a los buscadores tradicionales, solucionan algunos de los problemas que estos tienen, como el del *recall*, aunque hay otro tipo de inconvenientes que no han conseguido resolver, como por ejemplo el de la mejora de la *precisión*. Para llegar a conseguir mayor precisión en la búsqueda, se pueden utilizar cualquiera de estos cuatro mecanismos: basados en el contenido, colaborativos, de conocimiento del dominio y basados en el uso de ontologías.

Los métodos basados en el contenido tratan de obtener una representación tan precisa como sea posible de las preferencias del usuario para después hacer una mejor evaluación y ranking de las páginas devueltas. El método colaborativo se basa en establecer la similitud que puede haber entre usuarios para determinar la relevancia de la información. El método basado en el conocimiento del dominio se caracteriza por utilizar dos fuentes de información para proporcionar una mayor relevancia en los resultados: la ayuda del usuario y el conocimiento del dominio de búsqueda.

Finalmente, el método basado en ontologías establece una jerarquía entre conceptos que permite concretar y mejorar la búsqueda.

Las principales ventajas que encontramos en los Metabuscadores son las siguientes:

- **Facilita la consulta simultánea en múltiples buscadores.** Es decir, permite, por medio de una única consulta, obtener la lista ordenada de los documentos más relevantes que han devuelto los diferentes buscadores, evitándole al usuario la tarea de tener que realizar la misma consulta en cada uno de ellos.
- **Mejora la eficiencia de recuperación.** Dada la existencia de buscadores especializados en ciertos dominios, el Metabuscador puede utilizarlos para mejorar los resultados finales, evitando así la información irrelevante que se pueda obtener de otros buscadores más generales.
- **Resuelve el problema de la escalabilidad de la búsqueda en la Web.**
- **Incrementa la cobertura de búsqueda en la Web.** Debido a la gran cantidad de documentos que hay en Internet, es imposible que un solo motor de búsqueda indexe la totalidad de links de la Web. Por lo tanto, con la combinación de diferentes buscadores, es posible cubrir un mayor número de documentos en las búsquedas.

Así mismo, un Metabuscador también presenta las siguientes ventajas potenciales:

- **Arquitectura modular:** las tecnologías utilizadas en un Metabuscador se pueden dividir en módulos más pequeños y especializados que puedan ser paralelizados y ejecutados colaborativamente.
- **Consistencia:** los buscadores actuales a menudo responden de manera muy diferente a la misma consulta según pasa el tiempo. Sin embargo, el Metabuscador, al utilizar diferentes fuentes, tendrá menos variabilidad en los resultados ya que se verá favorecido por aquellos

buscadores que proporcionen resultados más estables.

- **Mejora el factor recall:** habiendo obtenido los resultados de múltiples buscadores, se puede mejorar el número de documentos relevantes recuperados (el factor *recall*) con respecto al número total de documentos existentes.
- **Mejora la precisión:** diferentes algoritmos recuperan más documentos relevantes iguales, pero diferentes documentos irrelevantes. Basándose en este fenómeno, en caso de ser cierta esta teoría, cualquier algoritmo que dé prioridad a los documentos que aparecen en las primeras posiciones en los resultados de diferentes buscadores obtendrán una mejora en la recuperación. Este fenómeno se conoce como “*efecto coro*”.

A pesar de contar con toda esta serie de ventajas, los Metabuscadores también cuentan con una serie de inconvenientes:

- **La selección de la base de datos:** este problema se asocia con la selección del buscador más adecuado para recibir la consulta introducida, seleccionar el o los buscadores que devolverán buenos resultados para una consulta concreta no es nada sencillo. Por ejemplo, no tiene sentido la consulta “guitarra eléctrica” en un buscador especializado de literatura científica. Para intentar resolver esta desventaja, se propone el uso de medidas que indiquen la utilidad de cada base de datos con respecto a una consulta dada y clasifica estos mecanismos en tres categorías: métodos de *representación amplia*, métodos de *representación estadística* y métodos basados en *aprendizaje*.
- **La selección de documentos:** una vez seleccionado el origen de los documentos, el problema consiste en determinar el número apropiado de documentos que es necesario devolver. Si se consideran demasiados, el coste computacional para determinar los mejores documentos y el coste de comunicación para obtenerlos puede ser excesivo. Se pueden establecer una serie de mecanismos para tratar de resolver este problema: decisión del usuario, pesos (se obtienen mayor número de documentos del buscador que se considere el mejor), *métodos basados en el aprendizaje* (se basa en experiencias pasadas para determinar el número de documentos de cada buscador) y la garantía de recuperación (trata de *garantizar la recuperación* de todos los documentos potencialmente útiles).
- **Combinación de los resultados:** el problema consiste en combinar los resultados de diferentes buscadores, teniendo en cuenta sus características y formas de evaluación, en una lista ordenada por relevancia. Además, existe la posibilidad de encontrar documentos devueltos que estén repetidos en diferentes buscadores. Las técnicas utilizadas para resolver este problema están basadas en un ajuste de semejanza local (se basa en las características del buscador o la semejanza devuelta) y la estimación de una semejanza global (se evalúa o estima la semejanza de cada documento recuperado con la consulta original).

La traducción de una consulta a cada uno de los lenguajes específicos de los buscadores puede ser un factor importante en un Metabuscador dado que cada motor de búsqueda tiene su propio lenguaje de consulta. Así pues, adaptar cada consulta al lenguaje de cada buscador parece necesario.

Hay muchas arquitecturas de Metabuscadores. Habitualmente, esta estructura se descompone en una serie de módulos más o menos específicos:

- **Interfaz de usuario:** es el encargado de recoger la consulta del usuario y posteriormente mostrar los resultados de la búsqueda. En algunos casos, el interfaz también contiene un sistema de refinamiento de la consulta, basada en el uso de algunas estructuras de conocimiento.
- **Selector de la base de datos:** trata de seleccionar los buscadores que darán mejores respuestas a la consulta introducida por el usuario. Trata de evitar el envío masivo de consultas a los buscadores más lentos y con un elevado coste computacional.
- **Selector de documentos:** el objetivo es obtener el mayor número de documentos relevantes, evitando recuperar los no relevantes. Si se recupera un número excesivo de documentos no relevantes, la eficiencia de la búsqueda se verá afectada negativamente.
- **Emisor de Consultas:** es el encargado de establecer la conexión con el buscador y enviarle la consulta (o consultas), así como de obtener los resultados. Se utiliza habitualmente http (*HyperText Transfer Protocol*) por el hecho de utilizar los métodos GET y POST. Sin embargo, existen buscadores que facilitan un interfaz de programación (API) para enviar consultas y que utilizan diferentes protocolos (Google usaba el protocolo SOAP en su API).
- **Agrupador de resultados:** su función principal es combinar los resultados de los diferentes buscadores en una lista. Es esencial el uso de algún criterio de evaluación para establecer un orden en la lista que, finalmente, se muestre al usuario.

Hoy día es normal encontrar referencias bibliográficas acerca de Metabuscadores de tercera generación (o motores de búsqueda de nivel 3). Estos trabajan de la siguiente forma: a petición del usuario se crea una base de datos de nivel 3 a partir de los resultados obtenidos por los Metabuscadores de nivel 2. Los Metabuscadores (que representan el nivel 2 de los motores de búsqueda) utilizan motores de búsqueda estándar (nivel 1) para encontrar los correspondientes resultados. Después se realiza un análisis de relevancia retroalimentada con estos resultados. Con esta colección de direcciones y documentos de texto se desarrolla una base de datos (a la que se denomina de nivel 3) que contiene solo documentos que sean relevantes en ese dominio de búsqueda para el usuario. En otras palabras, se crea una base de datos centrada específicamente en los campos de interés del usuario sobre la que podrá conseguir la información que necesita obteniendo unos resultados de gran calidad. Así pues, se plantea la necesidad de incluir perfiles de usuario y personalizaciones en futuros Metabuscadores.

La idea es que un buscador sea borroso en el momento que implemente búsquedas mediante aproximaciones semánticas, es decir, cuando incluya criterios de aproximación semántica a las consultas, no solo sintácticas. A continuación se presentan algunos aspectos a considerar.

- **El uso de diccionarios de sinónimos y tesauros (ontologías): búsqueda conceptual**

Cuando un usuario realiza una búsqueda usando una única palabra, la búsqueda puede ser completada haciendo uso de un diccionario de sinónimos. El diccionario permite realizar búsquedas no solo teniendo en cuenta la palabra original sino, además, sus sinónimos y pudiendo establecer grados de sinonimia que después serán tenidos en cuenta para calcular el grado de relevancia de los documentos recuperados como respuesta a la consulta realizada por el usuario.

El proceso de búsqueda puede, además, ser mejorado usando tesauros y ontologías. En la actualidad, existen muchas ontologías referentes a diferentes dominios, que pueden mejorar diversos aspectos en ciertas aplicaciones. Todas ellas han sido hechas a mano siguiendo diferentes metodologías, tales como *Methontology*. En la otra cara de la moneda está la construcción automática de ontologías que representa uno de los principales focos de investigación en la actualidad, donde hasta el momento los resultados obtenidos no son demasiados satisfactorios.

Possiblemente el tesauro más usado en la actualidad es WordNet (enlace 3), basado en las relaciones semánticas entre diferentes palabras. Las principales relaciones que gestiona WordNet son las de *sinonimia*, *hiponimia*, *hiperonimia*, *holonimia*, *meronimia*, etc. Así, un conjunto de sinónimos de una palabra puede ser agrupado en grupos llamados *synsets*, y como consecuencia, una palabra polisémica podrá pertenecer a diferentes *synsets*. La relación de hiponimia (y la de hiperónimia) se da entre diferentes términos entre los que se puede establecer una jerarquía. Así por ejemplo, si la palabra perro “es un tipo de” canino, entonces el término canino es un hiperónimo del término perro. Este tipo de relaciones aportan importante información que permite expandir la consulta inicial del usuario incorporando información con carácter semántico al proceso de búsqueda, así como un mecanismo para la identificación del significado adecuado de los términos involucrados en la consulta. Este tipo de sistemas normalmente requiere un mecanismo especial de *matching*, como el algoritmo de *ontomatching* que compara los conceptos asociados a las palabras o los sistemas de búsqueda que permiten desambiguar significados analizando la probabilidad de que coocurran ciertos conceptos. Se puede tratar el problema de la desambiguación estudiando el contexto local de las palabras y comparándolas con el contexto habitual de los distintos significados de las palabras. Este sistema requiere tener almacenado en un repositorio el contexto habitual de las palabras. La desambiguación desde el punto de vista del Soft-Computing podría verse como que las palabras no son desambiguadas en un único sentido, más bien en un conjunto de sentidos con sus correspondientes grados de relevancia. Hay modelos en los que se introduce el concepto de sinonimia relativa para definir un modelo de vectores basados en conceptos. En estos modelos, un término puede ser representado por un vector conceptual. Estos sistemas requieren repositorios de conceptos.



Enlace 12

WordNet

www.wordnet.com

Un ejemplo, FIS-CRM es un modelo introducido por el autor y sus colaboradores para representar los conceptos contenidos en cualquier clase de documentos. Puede ser considerado como una extensión del modelo vectorial (VSM). Su principal característica es que se alimenta de información proveniente de un diccionario sinónímico borroso y varias ontologías temáticas. El diccionario almacena el grado de sinonimia entre cada pareja de sinónimos y las ontologías guardan el grado de generalidad entre una palabra y otras más generales que ésta. La clave del éxito del modelo FIS-CRM (en su aplicación a varios metabuscadores) radica primeramente en la construcción de los vectores base de los documentos teniendo en cuenta el número de ocurrencias de los términos (lo que se denominan vectores VSM) y el posterior reajuste de los pesos de los vectores con el propósito de representar el peso de las ocurrencias de conceptos, haciendo uso de la información disponible en el diccionario de sinónimos y las ontologías temáticas. El proceso de reajuste conlleva el hecho de repartir las ocurrencias de un concepto entre los distintos sinónimos que convergen al mismo y que dan un peso a las palabras que representan un concepto más general que el que ellas mismas representan.

- **Sentencias de búsqueda y capacidades deductivas**

Si la búsqueda incluye frases, además del diccionario de sinónimos, los tesauros y ontologías, será necesario el uso de conectivas borrosas adecuadas, para poder discernir por ejemplo en una consulta de la forma “A y B” donde A y B tienen información común o cuando son totalmente independientes. Algo similar ocurre con la relación “A o B”. Otro aspecto deseable es que se mantenga el significado de las palabras que se tienen en mente mediante la relación de sinonimia, para elegir la mejor función de similitud.

Pero el problema puede ser aún mayor si trabajamos con relaciones causales. Primero, es muy difícil detectar una relación causal escrita (en un texto o una consulta). Por ejemplo, el texto podría ser: “lluvioso y oscuro”, que podría ser entendido por una persona como: “si el tiempo está lluvioso, el cielo se oscurece”. ¿Cómo puede un buscador distinguir la conectiva “y” y la causal? Ahora mismo es prácticamente imposible incluso si hay información relativa al contexto. Segundo, es muy difícil encontrar la función de implicación más adecuada para representar la sentencia (hay una gran variedad de implicaciones borrosas). La detección y gestión de relaciones causales podrían ser muy importantes para desarrollar sistemas Pregunta-Respuesta.

Para detectar las relaciones causales que existen en una colección de documentos, un punto de comienzo podría ser la detección de frases condicionales, pero esto no es tarea fácil. Descartes nunca pensó que su frase “pienso luego existo”, daría lugar a tal cantidad de conjetas e interpretaciones años después. En realidad, lo que él quiso decir, “primero pienso y luego soy persona”, o “como tengo la capacidad de pensar, soy una persona”. Incluso en esta ocasión, donde la intención de Descartes parece clara cuando expresó su máxima, no es tan fácil de interpretar y representar la información expresada en lenguaje natural, especialmente cuando conlleva frases complejas y giros complicados.

Con el fin de encontrar frases condicionales, se han desarrollado en el marco de nuestro grupo de investigación algunos sistemas para la detectar estructuras y clasificar frases causales (Sobrino, 2014), lo cual permite localizar, en términos de componentes básicos (verbos,

adverbios, giros lingüísticos, etc.), algunas formas causales. Para realizar el análisis gramatical, tenemos por un lado que es posible separar ciertas relaciones causales considerando su forma verbal, mientras que por otro lado vemos que es posible separar las relaciones atendiendo a los adverbios que aparecen en las frases. Ambos análisis dan lugar a algunas reglas causales que pueden ser usadas para extraer conocimiento de forma automática. De igual manera, cualquier estructura puede ser dividida en dos subestructuras que corresponden al antecedente y al consecuente de la relación causal, y a un parámetro que mide el grado de certeza, conjectura o conformidad de dicha relación causal. En otras palabras, no es lo mismo una frase de la forma: "si gano la lotería, me comprará un coche", en la cual no hay duda de que si el antecedente es verdadero el consecuente también lo será, que tener una frase que diga "si hubiéramos comprado un boleto en Sacramento, podríamos haber ganado la lotería", la cual deja muchas más dudas y conjecturas, y no se puede asegurar que el cumplimiento del antecedente garantice el cumplimiento del consecuente.

Pero esto sigue siendo un problema de *Procesamiento de Lenguaje Natural* muy complejo. Hay otras aproximaciones muy interesantes para la representación de frases condicionales mediante implicaciones borrosas. Por otro lado, también hay una idea basada en *Procesamiento de Lenguaje Natural* y *protoformas* que podría ser una prometedora línea de investigación, tal y como propuso el Profesor Zadeh.

• **Combinación de valores borrosos**

Un Metabuscador tiene que llevar a cabo una combinación de lógicas (los algoritmos que cada buscador usa) con la intención de combinar las similitudes locales en una similitud global o un orden final. Pero las similitudes locales no están basadas en criterios borrosos. Así, el orden de las páginas relevantes no es aproximado. Normalmente, los Metabuscadores realizan las búsquedas de acuerdo a la importancia que ellos conceden a la pregunta del usuario, y dependiendo de ella, evalúan los resultados para incorporarla a la lista final de resultados devueltos. En este caso, se tienen en cuenta criterios de mercado, y no criterios lingüísticos. Podría resultar interesante aplicar este criterio, primero, como se indicó antes, para conseguir búsquedas semánticas borrosas. Segundo, para lograr que los Metabuscadores creen una lista final de páginas recuperadas conforme a la relevancia proporcionada con los grados de confianza asociados a la lista local obtenida, no solo con las palabras tomadas de la consulta, y además usando términos relacionados como sinónimo, las medidas de similitud usadas en el cálculo de las relaciones lingüísticas y los operadores *booleanos* borrosos utilizados en las búsquedas. Cada búsqueda podría responder a diferentes lógicas borrosas realizadas por los buscadores, las cuales el Metabuscador se encargaría de combinar para establecer una lista final de resultados. El uso de los enlaces proporcionados por el Metabuscador para dar el orden de los resultados devueltos, podría ser útil como banco de pruebas para experimentar distintas combinaciones hipotéticas de lógicas borrosas.

Otro problema que puede aparecer es cuando es necesario agregar varios valores borrosos provenientes de distintas fuentes. Dos palabras (conceptos) pueden tener más de una relación lingüística (cada una con su valor borroso), tales como la hiperonimia o la sinonimia. Por ejemplo "balompié" y "fútbol" son sinónimos pero el primer término es más general que el último. Una relación causal puede existir entre ambas palabras (conceptos). Más aún, una relación borrosa

basada en la situación física de los términos (misma frase, párrafo, capítulo) podría ser tenida en cuenta. Entonces es necesario unir todos los valores borrosos en uno solo para poder ser aplicado en tareas de representación y búsqueda. Cómo agregar estos valores borrosos es un problema que no tiene solución conocida. Actualmente se usan los operadores estándar OWA (o derivados de ellos como los LOWA y los WOWA).

- **Resultados del proceso de clustering borroso**

La clasificación de documentos o la categorización de textos (como se conoce en el mundo de la recuperación de información) es el proceso por el que se asigna un documento a un conjunto predefinido de categorías basándose en el contenido del documento. Sin embargo, las categorías predefinidas en un repositorio real de documentos no son conocidas. Los métodos de *clustering* de textos pueden ser aplicadas para estructurar los conjuntos de documentos resultantes, así el usuario puede interactuar en el proceso de *clustering* de los documentos. En definitiva, el proceso de *clustering* permite lograr la estructuración de la colección de documentos en un número reducido de grupos, donde los documentos de cada grupo tienen el suficiente grado similitud entre ellos. En consecuencia, podemos enumerar los principales elementos que influyen a la hora de organizar un repositorio:

- *Dimensionalidad*: el clasificador puede manejar espacios de elementos de miles de dimensiones, por lo que es necesaria la capacidad de gestionar espacios de datos escasos o métodos de reducción de dimensiones.
- *Eficiencia*: los algoritmos de *clustering* documental deben ser eficientes y escalables. Además, el método debería ser preciso en la clasificación de nuevos documentos entrantes.
- *Entendibilidad*: el método debe proveer una descripción comprensible de los clusters descubiertos.
- *Actualización*: el clasificador debe ser capaz de actualizarse con cada nuevo documento que es archivado en el repositorio.

Existen muchos algoritmos de *clustering* basados en técnicas de Soft-Computing, por ejemplo el fuzzy C-means, los mapas autoorganizados basados en arquitecturas de redes neuronales, como por ejemplo los Mapas de Kohonen, etc. Actualmente los algoritmos de *Soft-Clustering* y los interfaces de *clustering dinámico* son muy utilizados en las tareas de clasificación de los Metabuscadres.

- **Arquitectura de un Metabuscadador**

Como se ha comentado ampliamente, existen ciertos aspectos que provocan una considerable reducción de la eficacia de los procesos de búsqueda. Además de las conocidas limitaciones provocadas por el uso de palabras clave, se suman la inexperiencia de los usuarios en el uso de los motores de búsqueda y sus herramientas adicionales.

Así aparecen los Metabuscadores, como una alternativa para tratar de paliar la baja precisión lograda hasta ahora por los buscadores. Además del uso de Metabuscadores, también se proponen otras alternativas para tratar de mejorar el grado de relevancia de los resultados obtenidos por los buscadores, tales como las técnicas de expansión de la consulta basada en el uso de términos relacionados semánticamente con los términos de la consulta original del usuario.

Existen muchas estrategias para llevar a cabo la técnica de expansión de la consulta, cada una con sus características propias. Básicamente, estos mecanismos pueden ser clasificados en: **automáticos**, **manuales** e **interactivos**. Las técnicas **automáticas** tratan de añadir nuevos términos de forma automática relacionados semánticamente con aquellos que son introducidos manualmente por el usuario, para así conseguir que el sistema devuelva una colección de documentos más acorde a la idea del usuario. Esta aproximación tiene algunos inconvenientes principalmente léxicos, como por ejemplo la polisemia (distintos significados para una misma palabra). Distintas implementaciones han sido desarrolladas para tratar de identificar el significado adecuado de los términos de la consulta de forma automática. Estos algoritmos reciben el nombre de **Word Sense Disambiguation (WSD)**. El método de expansión de la consulta de forma **interactiva** requiere de la colaboración del usuario. El sistema propone una serie de términos al usuario que pueden estar relacionados con su consulta para que este elija aquellos que más se aproximen a la idea de lo que busca. Generalmente este tipo de sistemas utilizan una estructura en árbol donde ordenan de los términos desde los más generales hasta los más específicos. Este tipo de sistemas suelen ser lentos e incómodos para el usuario porque requieren de un alto nivel de participación del mismo para responder a todas las preguntas a las que le somete el sistema. Por otro lado, la expansión de la consulta en muchas ocasiones se realiza utilizando estructuras de conocimiento como por ejemplo **Wordnet**, aunque pueden ser utilizados muchos otros tesauros y ontologías.

Como ya se ha dicho, se han diseñado muchas arquitecturas sobre Metabuscadores en las cuales se pueden identificar varios componentes comunes. Un ejemplo pueden ser aquellos componentes encargados de lanzar las consultas a los diferentes motores de búsqueda o bien los componentes encargados de calcular el grado de relevancia de los documentos recuperados.

La mayoría de las arquitecturas propuestas están basadas en el uso de **agentes** específicos, cada uno de ellos con diferentes funciones asignadas e intercomunicados a través de la red. Cada agente tiene designada una función específica, pero trabajan de forma colaborativa con otros agentes para conseguir una reducción de la complejidad del sistema. Para ello es necesario un lenguaje para la comunicación entre agentes, conocido por todos ellos y que permitirá la colaboración entre los mismo. Este lenguaje común puede ser por ejemplo ACL (**Agent Communication Language**).

La lógica borrosa podría jugar un papel fundamental en esta arquitectura basada en agentes, principalmente en la tarea de unir la información procedente de diferentes fuentes (agentes) y gestionar los resultados de forma eficiente y satisfactoria.

1.4.3. Minería de Opiniones: Análisis de Sentimientos

El Análisis de Sentimientos, también llamado **Opinion Mining**, es uno de los temas de investigación más recientes dentro del análisis de datos en el ámbito de las ciencias de la Computación. Hoy en día es uno de los campos más importantes, difíciles y demandados por la repercusión que tiene tanto para las empresas como para la sociedad. En las investigaciones desarrolladas por el grupo de investigación del autor de este manual, se han propuesto diversas aplicaciones y métodos. En el artículo "Sentiment analysis: A review and comparative analysis of web services" (Serrano-Guerrero, Olivas, Romero, Herrera-Viedma, 2015) se dispone de una descripción del estado actual de la investigación y aplicaciones en Análisis de Sentimientos y Opiniones, de la que se presenta un breve extracto a continuación.

Las técnicas de recuperación de información textual se centran en el procesamiento, la búsqueda o la extracción de información objetiva. Los hechos tienen un componente objetivo; sin embargo, hay otros elementos textuales que expresan características subjetivas. Estos elementos son principalmente opiniones, sentimientos, valoraciones, actitudes y emociones, que son el foco del Análisis de Sentimientos. Todos ellos están estrechamente relacionados, sin embargo, presentan ligeras diferencias. Este hecho implica el nacimiento de muchas tareas relacionadas en este nuevo campo de investigación, como la minería de opiniones, el análisis de subjetividad, la detección de emociones o detección del **spam** de opinión, entre otros.

El Análisis de Sentimientos ofrece muchas oportunidades para desarrollar nuevas aplicaciones, especialmente debido al gran crecimiento de las herramientas disponibles. Por ejemplo, las recomendaciones sobre los temas propuestos en fuentes como blogs y redes sociales. El sistema de recomendaciones puede funcionar teniendo en cuenta aspectos tales como las opiniones positivas o negativas sobre esos temas o productos. Los sitios web de opiniones podrían recopilar información de diferentes fuentes con el fin de resumir o componer una opinión global sobre un candidato, producto, etc., sustituyendo así a los sistemas que requieren explícitamente opiniones o resúmenes.

Los sistemas de pregunta-respuesta representan otro campo en el que las opiniones desempeñan un papel importante. La detección de preguntas sobre opiniones y sus posibles respuestas y su tratamiento son esenciales para generar buenas respuestas. La detección de información subjetiva es realmente importante en campos relacionados con la argumentación en los que las frases objetivas suelen ser más valiosas. Pero ciertamente, uno de los campos más importantes en los que el análisis de sentimientos tiene un mayor impacto es en la industria. Pequeñas y grandes empresas, al igual que otras organizaciones como los gobiernos, desean saber lo que la gente dice sobre sus marcas, productos o miembros.

Como se ha dicho, el Análisis de Sentimientos es un concepto que abarca muchas tareas como la extracción de sentimientos, la clasificación de sentimientos, clasificación de subjetividad, resumen de opiniones o detección de **spam** de opinión, entre otros. Para llevar a cabo cualquiera de estas actividades, el Análisis de Sentimientos tiene que lidiar con muchos desafíos.

El primero es la definición de los elementos que intervienen. Por lo tanto, es necesario definir claramente conceptos como opinión, subjetividad o emoción, sin embargo, esta tarea no es fácil. Por ejemplo, de una manera sencilla, la opinión de un usuario puede ser considerada como un sentimiento positivo

o negativo acerca de una entidad o de un elemento de esa entidad. Por otro lado, la subjetividad no implica necesariamente un sentimiento, sino que permite expresar sentimientos o creencias, y específicamente, nuestros propios sentimientos o creencias y nuestras emociones.

Estas definiciones tienen que ser formalizadas mediante expresiones matemáticas que pueden ser calculadas y utilizadas como entradas para los sistemas. Por lo tanto, el éxito del Análisis de Sentimientos depende principalmente de la capacidad de extraer la información necesaria de esas definiciones a partir de los textos para llevar a cabo esas tareas. Así, las técnicas ya comentadas de Procesamiento del Lenguaje Natural (PNL, Minería de Textos) son imprescindibles para conseguir buenos resultados en función de la tarea a realizar. Este es otro de los principales retos de este campo de investigación, junto con todos los problemas relacionados con la adaptación de técnicas típicas para clasificar o resumir, así como la creación de nuevas técnicas y algoritmos especializados en opiniones.

A pesar de la complejidad y la dificultad de este problema, muchas empresas y universidades están desarrollando nuevas herramientas e instrumentos y servicios web que tratan varios de los temas mencionados. Estos servicios podrían incluirse, especialmente para la investigación, en otras aplicaciones o plataformas sin necesidad de ser experto en Análisis de Sentimientos.

Los conceptos de **Opinion Mining**, **Sentiment Analysis** y **Subjectivity Analysis** se usan frecuentemente como sinónimos; sin embargo, sus orígenes no son exactamente los mismos y algunos autores consideran que cada concepto presenta connotaciones diferentes, al igual que otros estrechamente relacionados, como por ejemplo el Análisis Afectivo.

- **Revisión de los principales conceptos**

Una opinión podría definirse simplemente como un sentimiento, una visión, una actitud, una emoción o una valoración positiva o negativa acerca de algo (producto, persona, evento, organización o tema) con respecto a un usuario o grupo de usuarios.

Siguiendo esa definición, una opinión puede ser matemáticamente definida como una 5-tupla $(e_j; a_{jk}; so_{ijkl}; h_i; t_l)$ donde:

- e_j representa una entidad
- a_{jk} es el aspecto/característica k-ésima de la entidad e_j

so_{ijkl} es el valor sentimental de la opinión del propietario (*holder*) h_i en el aspecto ajk de la entidad e_j en el tiempo t_l . h_i es el que tiene la opinión y t_l es el momento en que la opinión fue expresada.

Ese valor puede ser **positivo**, **negativo** o **neutro**, o incluso una clasificación más granular.

Las opiniones se pueden clasificar en diferentes grupos, por ejemplo, pueden ser opiniones **regulares** y **comparativas**. La mayoría de los dictámenes son periódicos y pueden subdividirse en dictámenes **directos** o **indirectos**. Las opiniones directas expresan una idea sobre una entidad o un aspecto de una entidad, mientras que las opiniones indirectas expresan

una opinión sobre una entidad o un aspecto de una entidad basada en los efectos en otras entidades.

Por otra parte, las frases comparativas expresan la semejanza entre entidades que consideran aspectos o características comunes. Además, las opiniones pueden clasificarse en **explícitas** o **implícitas**, dependiendo de si expresan ideas **subjetivas** u **objetivas**.

Aparte del **sentimiento** y la **opinión**, hay dos conceptos importantes cercanos a ellos, la **subjetividad** y la **emoción**. Una oración subjetiva puede expresar algunos sentimientos, puntos de vista o creencias personales; sin embargo, no implica necesariamente ningún sentimiento. Así, la diferencia entre oraciones **objetivas** y **subjetivas** es que una oración objetiva expresa algunos hechos o información sobre el mundo, mientras que una frase subjetiva expresa algunos sentimientos, puntos de vista o creencias personales. Un ejemplo podría ser la frase “creo que se han ido”. Sin embargo, la subjetividad a veces implica hasta cierto punto sentimientos cuando se trata de **afecto, juicio, apreciación, especulación, acuerdo**, etc.

Por otro lado, una emoción puede ser vista como una expresión de nuestros propios sentimientos y pensamientos subjetivos. Las emociones están muy cerca de los sentimientos, de hecho, la forma de medir la fuerza de una opinión está ligada a la intensidad de ciertas emociones, como el **amor, la alegría, la sorpresa, la ira, la tristeza** o el **miedo**. Un ejemplo podría ser la frase “amo este coche”, en la que el orador expresa su amor objetivo por su coche.

También es necesario comentar el concepto de **estado de ánimo**, que podría considerarse como una mezcla de sentimientos, emociones, sentimientos, que mueven al autor de un determinado texto a escribir ese comentario, observación, crítica, etc.

• Tareas

Surgen muchas tareas vinculadas al Análisis de Sentimientos. Algunas de ellas están estrechamente relacionadas y es difícil separarlas claramente porque comparten muchos aspectos. Las más importantes son:

1. **Clasificación de los sentimientos.** También llamada orientación de sentimientos, orientación de opinión, orientación semántica o polaridad de sentimientos. Se basa en la idea de que un documento/texto puede expresar una opinión de un titular sobre una entidad y trata de medir el sentimiento de ese titular hacia la entidad. Por lo tanto, consiste básicamente en clasificar las opiniones en tres categorías principales: **positivo, negativo** o **neutro**. Parece una tarea simple, sin embargo, es una tarea realmente compleja, especialmente cuando las opiniones provienen de múltiples dominios o idiomas. Esta tarea está estrechamente relacionada con la predicción de la valoración de los sentimientos, que consiste en medir la intensidad de cada sentimiento. Por ejemplo, se pueden utilizar diferentes escalas para medir una opinión, el rango $[-1, 1]$ donde -1 indica el grado negativo máximo y 1 el grado positivo máximo, o una escala de cinco estrellas en las que el usuario puede seleccionar cero estrellas para expresar la negatividad máxima o cinco estrellas en caso contrario.

2. **Clasificación de subjetividad.** Consiste principalmente en detectar si una frase dada es subjetiva o no. Una frase objetiva expresa información factual mientras que una frase subjetiva puede expresar otro tipo de información personal, como por ejemplo **opiniones, evaluaciones, emociones y creencias**. Además, las frases subjetivas pueden expresar lo positivo o lo negativo pero no todas lo hacen. Esta tarea puede ser vista como un paso previo a la clasificación de los sentimientos. Una buena clasificación de subjetividad puede asegurar una mejor clasificación de sentimientos. Incluso se considera como un proceso más difícil que el de distinguir entre sentimientos positivos, neutros o negativos.
3. **Resumen de opiniones.** Se centra especialmente en la extracción de las características principales de una entidad compartida en uno o varios documentos y los sentimientos sobre ella. Por lo tanto, se pueden distinguir dos perspectivas en esta tarea: resumen de un solo documento o de varios documentos. La integración de documentos individuales consiste en analizar los hechos presentes en el documento analizado, por ejemplo, cambios en la orientación de los sentimientos a lo largo del documento o vínculos entre las diferentes entidades o características encontradas, y mostrando principalmente aquellos textos que mejor las describen. Por otro lado, en los resúmenes multidocumento una vez detectadas las características y entidades, el sistema tiene que agrupar y/o ordenar las diferentes frases que expresan sentimientos relacionados con esas entidades o rasgos. Al final puede ser presentado como un gráfico o un texto que muestre las principales características/entidades y cuantifique el sentimiento de alguna manera en cada uno de ellos, por ejemplo, agregando intensidades de sentimientos o contando el número de sentimientos positivos o negativos.
4. **Recuperación de opiniones.** Intenta recuperar documentos que expresan una opinión sobre una consulta determinada. En este tipo de sistemas, se necesitan dos puntuaciones para cada documento, la puntuación de relevancia frente a la consulta y la puntuación de opinión sobre la consulta, y ambos se utilizan normalmente para clasificar los documentos.
5. **Sarcasmo e ironía.** Se centra en detectar afirmaciones con contenido irónico y sarcástico. Este es una de las tareas más complicadas en este campo, especialmente, debido a la falta de acuerdo entre los investigadores sobre cómo la ironía o sarcasmo pueden ser formalmente representadas y definidas.
6. **Otros.** Además de las actividades mencionadas anteriormente, se pueden destacar otras tareas relacionadas con el Análisis de Sentimientos, como la **detección de género o autoría**, que intenta determinar el género o la persona que ha escrito un texto/opinión o la **detección de spam de opinión** que trata de detectar opiniones o reseñas que contienen contenidos no confiables publicados para distorsionar la opinión pública hacia las personas, empresas o productos.

Técnicas

Se suelen agrupar desde el punto de vista de las diferentes aplicaciones/retos que se pueden encontrar en SA o en los principales tópicos de SA. Alguna clasificación los agrupa bajo cinco grupos principales: análisis de sentimientos a nivel de documento, sentimientos a nivel de frase, análisis de sentimientos basado en aspectos, análisis comparativo de sentimientos y adquisición de léxico de sentimientos. Otras se centran principalmente en la agregación de opiniones, el spam de opinión y el análisis de contradicciones, especialmente aplicado a servicios Web, por ejemplo, *microblogs* o *streaming* de datos, entre otros.

Cuatro tipos para clasificar las técnicas de Análisis de Sentimientos: aprendizaje automático, basado en diccionarios, estadística y semántica.

- **Enfoques de aprendizaje automático**

Como ya hemos visto, se pueden agrupar en dos categorías principales: técnicas supervisadas y no supervisadas. El éxito de ambos se basa principalmente en la selección y extracción del conjunto apropiado de características utilizadas para detectar sentimientos. En esta tarea, las técnicas de procesamiento del lenguaje natural (PLN) juegan un papel muy importante porque algunas de las características más importantes utilizadas son, por ejemplo:

1. *Términos* (palabras o n-gramas) y su frecuencia.
2. *Información de la ‘part of speech’*, los adjetivos juegan un papel importante pero los sustantivos pueden ser significativos.
3. Las *negaciones* pueden cambiar el significado de cualquier oración.
4. Las *dependencias sintácticas* (análisis de árbol) puede determinar el significado de la oración.

Con respecto a las técnicas supervisadas, las *Máquinas de Soporte Vectorial* (SVM), *Naive Bayes*, *Máxima Entropía* son algunas de ellas, mientras que las técnicas semisupervisadas y no supervisadas se proponen cuando no es posible tener un conjunto inicial de documentos/opiniones etiquetados para clasificar el resto de ítems.

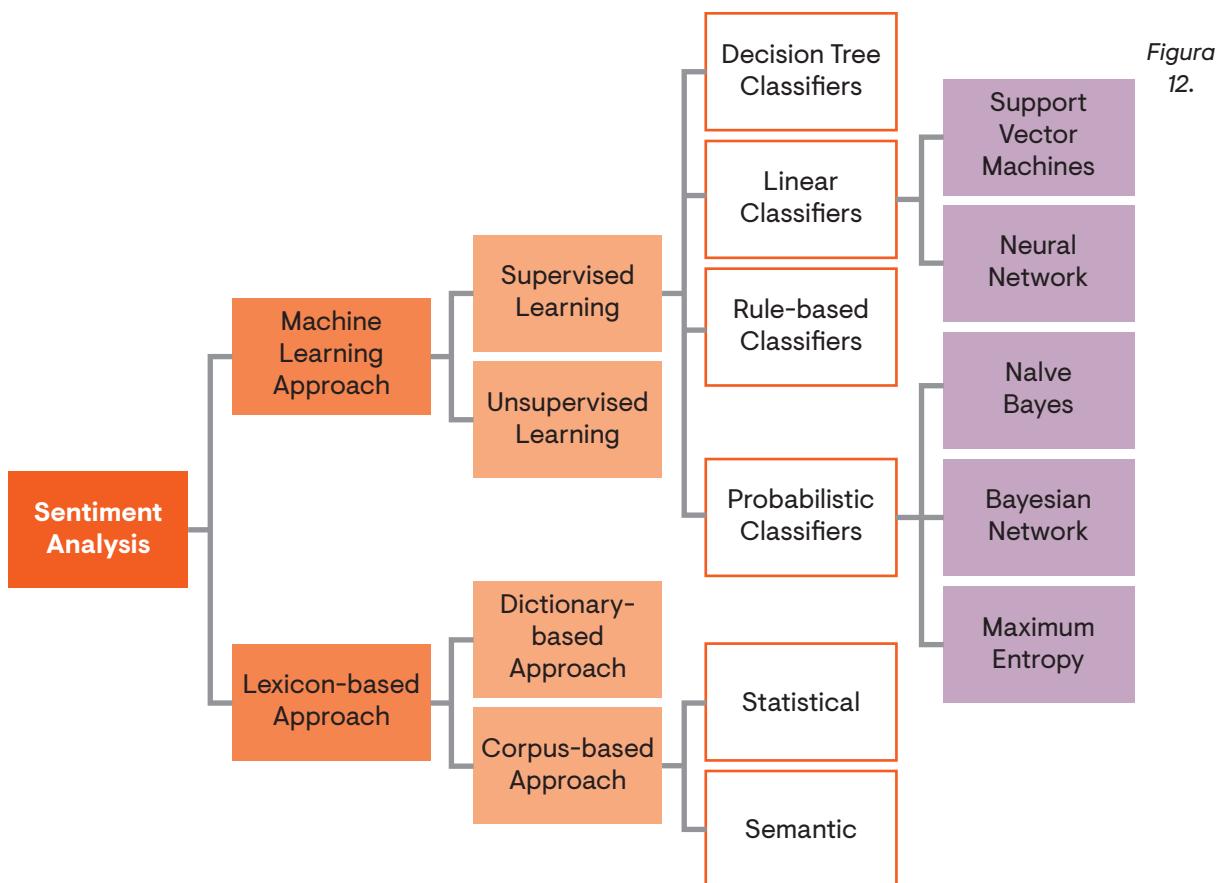
Además, los enfoques híbridos, que combinan técnicas supervisadas y no supervisadas, o incluso técnicas semisupervisadas, pueden ser útiles para clasificar los sentimientos.

- **Enfoques basados en el léxico**

Se basan principalmente en un léxico de sentimientos, es decir, una colección de sentimientos conocidos y precompilados, términos, frases e incluso modismos, desarrollados para los géneros tradicionales de comunicación, como el *Opinion Finder lexicón*. Estructuras aún más complejas como *ontologías* o *diccionarios* que miden la orientación semántica de las palabras o frases pueden ser usadas para este propósito. Aquí se pueden encontrar dos subclasiﬁcaciones: enfoques basados en diccionarios y en Corpus.

La primera se basa generalmente en el uso de un conjunto inicial de términos (semillas) que normalmente se recogen y anotan de forma manual. Este conjunto crece buscando los sinónimos y antónimos en un diccionario. Un ejemplo de ese diccionario podría ser *WordNet*, que se utilizó para desarrollar un tesauro llamado *SentiWordNet*. El principal inconveniente de este tipo de enfoques es la incapacidad de hacer frente a orientaciones específicas de dominio y contexto. Aun así, podría ser una solución interesante dependiendo del problema.

Las técnicas basadas en corpus surgen con el objetivo de proporcionar diccionarios relacionados con un dominio específico. Estos diccionarios se generan a partir de un conjunto de términos de opinión de semillas que crecen a través de la búsqueda de palabras relacionadas mediante el uso de técnicas estadísticas o semánticas. Métodos basados en estadística como el *Análisis Semántico Latente* (LSA) o simplemente se puede utilizar la frecuencia de aparición de las palabras dentro de una colección de documentos. Y por otro lado, métodos semánticos como el uso de sinónimos y antónimos o relaciones de tesauro como *WordNet* también pueden representar una solución interesante.



Técnicas usadas habitualmente para el Análisis de Sentimientos. Fuente: Serrano-Guerrero et al. (2015).

- **Procesamiento del lenguaje natural y recuperación de información en el análisis de sentimientos.**

El Análisis de Sentimientos puede ser considerado como un problema de PLN muy restringido, donde solo es necesario comprender los sentimientos positivos o negativos respecto a cada frase y/o las entidades o temas a tratar. Sin embargo, a pesar de ser un problema restringido, todos los trabajos en este campo, así como todos los trabajos en Recuperación de Información, siempre luchan contra los problemas no resueltos del PLN (manejo de la negación, reconocimiento de entidades con nombre, desambiguación del sentido de la palabra, etc.) que son esenciales para detectar recursos literarios como la ironía o el sarcasmo y, en consecuencia, para encontrar y valorar los sentimientos.

Uno de los aspectos principales del PLN son los diferentes niveles de análisis. Dependiendo de si el objeto del estudio es un texto o documento completo, una o varias frases vinculadas, una o varias entidades o aspectos de esas entidades, se pueden realizar diferentes tareas de PNL y Análisis de Sentimientos. Por lo tanto, es necesario distinguir tres niveles de análisis que determinarán claramente las diferentes tareas del Análisis de Sentimientos: nivel de documento, nivel de frase y nivel de entidad/aspecto.

El nivel de documento considera que un documento es una opinión sobre una entidad o aspecto de la misma. Este nivel está asociado a la tarea llamada clasificación de sentimientos a nivel de documento. Sin embargo, si un documento presenta varias frases tratando diferentes aspectos o entidades, entonces el nivel de oración es más adecuado.

El nivel de la oración está estrechamente relacionado con la tarea llamada clasificación de subjetividad que distingue las frases que expresan información fáctica de las frases que expresan opiniones y puntos de vista subjetivos.

Y finalmente, cuando se necesita información más precisa, entonces surge el nivel de entidad/aspecto. Es el nivel de grano más fino, considera un objetivo sobre el que el ponente de opinión expresa una opinión positiva o una opinión negativa. Este último nivel es posiblemente el más complejo porque es necesario extraer con gran precisión muchas características tales como fechas o períodos de tiempo, las diferentes características/aspectos y entidades a tener en cuenta, así como las relaciones entre ellos, los opinadores y sus características.

Muchas propuestas siguen las mismas estrategias generales que otros trabajos de Recuperación de Información anteriores como lo que hemos visto anteriormente, pero reemplazando varias variables estadísticas o semánticas por aspectos relacionados con los sentimientos. Por ejemplo, se propone el uso de la cohesión léxica, es decir, la distancia “física” entre las ubicaciones de los términos significativos o subjetivos para clasificar los documentos.

En otras propuestas se aplican métodos supervisados bien conocidos como las Redes Neuronales o las SVM a la Clasificación de Sentimientos, las cuales han sido utilizadas profusamente en Recuperación de Información. En este caso también, la diferencia con otros trabajos sobre Recuperación de Información es la selección de características.

1.5. Conclusiones

En este documento se ha presentado una introducción al aprendizaje estadístico y la Minería de Datos. Se ha comenzado por el origen, los datos, estableciendo inicialmente las diferencias entre datos, información y conocimiento, qué tipos de datos se suelen manejar, cómo se consiguen y dónde se almacenan habitualmente, introduciendo el concepto de “lago de datos”. Se ha proseguido con una clasificación de los diferentes tipos de “minerías”: datos, textos, opiniones, gráficos... y se han enmarcado dentro de lo que hoy se denomina “Inteligencia de Negocio”. Se ha presentado una crítica sobre cómo se suele afrontar habitualmente el análisis de datos, que nos permita reflexionar sobre en qué grado se suelen hacer las cosas de una forma adecuada.

A continuación se han descrito los diferentes tipos de análisis, distinguiendo por ejemplo entre conceptos como “predicción” y “pronóstico”. Se ha introducido el papel del “científico de datos” con taxonomías exhaustivas tanto de los métodos basados en la estadística como los basados en Inteligencia artificial (aprendizaje automático) para el análisis de datos, y se ha hecho una descripción de qué métodos resultan más adecuados para afrontar cada uno de los diferentes tipos de retos que suelen presentarse.

Esta parte se ha concluido con una descripción detallada de la “Ingeniería de Conocimiento”, por su importancia a la hora de establecer los criterios que deben guiar el proceso de análisis de datos o la relación con los expertos en los temas a analizar, que nos pueden guiar en el análisis.

La segunda parte se ha dedicado a describir en profundidad la metodología genérica comúnmente usada para el proceso de descubrimiento de conocimiento en bases de datos y la minería de datos (*KDD: Knowledge Discovery in Databases*). También se ha descrito más superficialmente otra metodología muy usada, la CRISP-DM. Esta parte se ha finalizado con una introducción a las técnicas y herramientas que se suelen usar actualmente para estos propósitos en entornos Big Data.

Por último, se han descrito con mucho detalle tres ejemplos de aplicaciones sofisticadas de los conceptos presentados. El primero tiene que ver con un sistema para la prevención de incendios forestales (Minería de datos), el segundo con diferentes aplicaciones en acceso y recuperación de información (Minería de textos) y el tercero con la importancia actual del análisis de sentimientos (Minería de opiniones).

Esta visión panorámica pretende ser un mapa completo de los objetivos, orientación y técnicas (tanto estadísticas como provenientes de la Inteligencia Artificial) para afrontar el análisis de datos en su estado actual desde el punto de vista de las Ciencias de la Computación. No se han presentado herramientas concretas debido a la imposibilidad de profundizar mínimamente en el gran número de las disponibles actualmente y no haber un criterio robusto para elegir una u otra para estudiar en detalle. Es por ello que queda fuera del alcance de esta asignatura. Se han detallado tres ejemplos de minería de datos, textos y opiniones, que han permitido profundizar en diversos temas relevantes en este campo, como por ejemplo el uso de técnicas de Soft-computing para la minería de textos o el papel cada vez más importante del análisis de datos no estructurados.



Glosario

Aprendizaje Automático (Machine Learning)

Rama de la Inteligencia Artificial en la que se diseñan mecanismos para dotar a los sistemas computacionales de capacidad de aprendizaje, en el sentido de la capacidad de descubrir regularidades (patrones) en datos o situaciones anteriores y aplicarlos a nuevos problemas o situaciones análogas. Se pueden considerar diversos paradigmas y grupos de técnicas, como el aprendizaje supervisado (clasificación) y el no supervisado (clustering).

Científico de Datos (Data Scientist)

Profesional que debe poseer conocimientos de computación, bases de datos, Inteligencia Artificial, Aprendizaje Automático, estadística, visualización, reconocimiento de patrones, sociología, psicología, KDD y Minería de Datos... y que debe ser capaz de seleccionar y guiar las herramientas y técnicas más adecuadas para cada problema y objetivos concretos en un proceso de análisis de datos.

Descubrimiento de Conocimiento en Bases de Datos (KDD, Knowledge Discovery in Databases)

Proceso (metodología) para, a partir de una base de datos (habitualmente estructurada), tratar de encontrar regularidades, 'patrones' en los datos que puedan ser representados formalmente y aplicados a situaciones futuras, con fines habitualmente de segmentación, predicción o pronóstico.

Ingeniería de Conocimiento

Parte de la Inteligencia Artificial encargada del desarrollo de Sistemas Basados en el Conocimiento (SBC/KBS), como pueden ser los Sistemas de Ayuda a la Decisión (DSS Decision Support Systems). La tradición de los SBC comenzó con los denominados "Sistemas Expertos", sistemas computacionales que tratan de emular las capacidades de un experto en un tema basándose en la extracción del conocimiento del propio experto o grupo de expertos y 'transmitiéndoselo' al sistema. Con la proliferación del almacenamiento y uso de datos de forma masiva, los SBC actuales suelen apoyarse en ambos pilares: expertos y datos.

Inlier

Son observaciones detectadas como atípicas (outliers) pero que no tienen el comportamiento de un verdadero outlier, se comportan de forma similar al resto de los datos 'normales'. Una vez que se detectan los valores atípicos, puede ser necesario descartar los inliers. Los valores inestables pueden estar a veces erróneamente relacionados con el concepto de "ruido". Muchos autores definen un inlier como aquellos ejemplos que se encuentran entre clusters.

Inteligencia Artificial

La Inteligencia Artificial (IA/AI -siglas en inglés-) se puede ver como la disciplina del ámbito de la computación y los sistemas de información que pretende simular computacionalmente comportamientos humanos que pueden ser considerados como inteligentes. Hay diversas ramas dentro de la IA, como el Aprendizaje Automático, la Ingeniería del Conocimiento, la Visión Artificial o la Robótica.

Inteligencia de Negocio (Business Intelligence)

Se suele definir como la capacidad de transformar datos en información para ayudar a gestionar una empresa, que consiste en los procesos, aplicaciones y prácticas que apoyen la toma de decisiones ejecutivas.

Lago de Datos (Data Lake)

Un repositorio (físico o conceptual) para grandes cantidades y variedades de datos, tanto estructurados como no estructurados. El lago acepta entradas desde diversas fuentes y puede preservar tanto la fidelidad de los datos originales como las diversas transformaciones que se les van haciendo, incluso simultáneamente desde diferentes departamentos, lo que puede generar diferentes evoluciones en paralelo de los mismos datos originales.

Lógica Borrosa o Difusa (Fuzzy Logic)

La Teoría de Conjuntos Borrosos fue introducida por Lotfi A. Zadeh (Azerbaiyán, 1921-2017). Bajo el concepto de Conjunto Borroso (Fuzzy Set) reside la idea de que los elementos clave en el pensamiento humano no son números, sino etiquetas lingüísticas. Estas etiquetas permiten que los objetos pasen de pertenecer de una clase a otra de forma suave y flexible. Uno de los objetivos de la Lógica Borrosa es proporcionar las bases del razonamiento aproximado que utiliza premisas imprecisas como instrumento para formular el conocimiento.

Minería de Datos

Análisis de datos en el que se parte de datos estructurados y casi siempre numéricos. El objetivo es encontrar regularidades (“patrones”) que permitan establecer modelos normalmente de predicción o de clasificación para situaciones futuras.

Minería de Opiniones

Análisis de datos en el que se parte de colecciones de documentos de texto, habitualmente pequeños, como los típicos mensajes en redes sociales. El objetivo es manifestarse sobre la “polaridad” (bueno o malo) de un mensaje con respecto a un determinado tema, encontrando regularidades (“patrones”) semánticas (aunque normalmente sólo se manejan desde el punto de vista lexicográfico) que permitan ayudar en tareas como la percepción de un producto o un político a través de las opiniones de los usuarios de las redes sociales. También se suele denominar “Análisis de Sentimientos” (Sentiment Analysis).

Minería de Textos

Análisis de datos en el que se parte de colecciones de documentos de texto. El objetivo es encontrar regularidades (“patrones”) semánticos (aunque normalmente solo se manejan desde el punto de vista lexicográfico), que permitan ayudar en tareas como el acceso, la búsqueda y la recuperación de información o la elaboración automática de resúmenes de dichos textos.

Outlier

Un valor atípico (outlier) es un punto en los datos tan diferente de los otros que se sospecha que ha sido creado por diferentes mecanismos. La detección de valores atípicos debe entenderse como un concepto multimodal: detección de valores atípicos relacionada con la combinación de valores de todas las tuplas y columnas de una base de datos relacional o con todos los documentos de una base de datos NoSQL. Existen muchos métodos para detectar los vectores o tuplas considerados valores atípicos, por ejemplo, basados en Estadística, Agrupamiento, Redes neuronales, Aprendizaje automático, Lógica borrosa, etc.

Soft-Computing

La computación suave se diferencia de la computación convencional (dura) en que, a diferencia de ella, es tolerante a la imprecisión, la incertidumbre y la verdad parcial para lograr la trazabilidad, la robustez y el bajo coste de las soluciones. El modelo a seguir para el soft computing es la mente humana. Los principales componentes de la computación suave (SC) son la lógica borrosa (FL), la teoría de redes neuronales (NN) y el razonamiento probabilístico (PR), con este último subsumiendo las redes de creencias, los algoritmos genéticos, la teoría del caos y partes del aprendizaje automático.



Enlaces de interés

Búsqueda eficaz de información en la Web

Libro de J. A. Olivas en el que se describe someramente lo que es un Sistema de Recuperación de Información, para posteriormente poder profundizar en algunos aspectos específicos.

<http://hdl.handle.net/10915/18401>

La lógica borrosa y sus aplicaciones.

Artículo divulgativo de J. A. Olivas con una introducción completa a la lógica borrosa y el razonamiento aproximado que permite introducirse en este campo de una forma sencilla y sin necesitar conocimientos previos.

<http://arantxa.ii.uam.es/~dcamacho/logica/recursos/fuzzy-into-esp.pdf>

Tesis doctoral de José A. Olivas

Contribución al estudio experimental de la predicción basada en Categorías Deformables Borrosas (Tesis Doctoral). Desarrollo completo y detallado de un proceso de KDD.

<http://hdl.handle.net/10578/18399>

Wordnet

Possiblemente el tesauro más usado en la actualidad es WordNet, basado en las relaciones semánticas entre diferentes palabras. Imprescindible para hacer minería de texto y procesamiento de lenguaje natural.

<http://www.wordnet.com>

Zadeh (D.E.P), la Lógica Borrosa y el Análisis de datos masivos.

Comentario interesante disponible en el blog de la VIU.

<https://www.universidadviu.es/zadeh-d-e-p-la-logica-borrosa-analisis-datos-masivos/>

Bibliografía



Referencias bibliográficas

Baeza-Yates, R., Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley. Essex, UK.

Berry, M., Linoff, G. (1996). *Data Mining Techniques*. Wiley Computer Publishing. New York.

Calatrava, C., Oruezabal, M. J., Olivas, J. A., Romero, F. P., Serrano-Guerrero, J. (2015). A Decision Support System for Risk Analysis and Diagnosis of Hereditary Cancer, Proc. of the 2015 *International Conference on Artificial Intelligence* IC-AI'2015, CSREA Press, USA.

Davenport, T. H., Harris, J. G. (2007). *Competing on Analytics: The New Science of Winning*. Harvard Business Press.

Fayyad, U., Piatetsky-Shapiro, G., Smyth, P. (1996). The KDD Process for Extracting Useful Knowledge from Volumes of Data. *Communications of the ACM*, November 1996/ Vol 39, N° 11, 27–34.

Metodología CRISP-DM. (27/12/2009). [Mensaje en blog]. Recuperado de <https://exde.wordpress.com/2009/03/13/a-visual-guide-to-crisp-dm-methodology/>

Olivas, J. A. (2000). Contribución al estudio experimental de la predicción basada en Categorías Deformables Borrosas (Tesis Doctoral), Universidad de Castilla-La Mancha. Disponible en RUIdeRA <http://hdl.handle.net/10578/18399>

Olivas, J. A. (2002). La Lógica Borrosa y sus aplicaciones. BOLETIC 24 (Revista de la Asociación profesional del cuerpo de sistemas y tecnologías de la Administración del Estado, Monográfico sobre Inteligencia Artificial).

Olivas, J. A. (2011). *Búsqueda eficaz de información* en la Web, Edulp, La Plata, Argentina. <http://hdl.handle.net/10915/18401>

Romero-Cordoba, R., Olivas, J. A., Romero, F. P., Alonso-Gonzalez, F., Serrano-Guerrero, J. (2017). An Application of Fuzzy Prototypes to the Diagnosis and Treatment of Fuzzy Diseases. Int. *Journal of Intelligent Systems* 32(2), 194-210.

Serrano-Guerrero, J., Olivas, J. A., Romero, F. P., Herrera-Viedma, E. (2015). Sentiment analysis: A review and comparative analysis of web services. *Information Sciences* 311, 18–38.

Sobrino, A., Puente, C., Olivas, J. A. (2014). Extracting Answers from causal mechanisms in a medical document. *Neurocomputing* 135, 53–60.

Zadeh, L. A. (1982). A note on prototype set theory and fuzzy sets. *Cognition* 12, 291-297.

Bibliografía recomendada:

Adriaans, P. W., Zantinge, D. (1996). *Data Mining*. Addison-Wesley.

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A. (1996). *Fast Discovery of Association Rules*. AAAI/MIT Press, Cambridge MA.

Agrawal, D., Das, S., Abbadi, A. E. (2010). Big Data and Cloud Computing: New Wine or Just New Bottles?. *Proc. of the VLDB 2010*, Vol. 3, No. 2.

Agrawal, D., Das, S., Abbadi, A. E. (2011). Big Data and Cloud Computing: Current State and Future Opportunities. *Proc. of the ETDB 2011*, Uppsala, Sweden.

Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth, Belmont, CA.

Quinlan, J. R. (1979). Discovering Rules by Induction from a Large Collection of Examples. En D. Michie (ed.) *Expert Systems in the Microelectronic Age*, Edinburgh University Press.

Quinlan, J. R. (1983). Induction of Decision Trees. *Machine Learning* 1, 81-106.

Quinlan, J. R. (1988). C4.5: *Programs for Machine Learning*. Morgan Kaufmann, San Mateo CA.

Mayer-Schönberger, V., Cukier, K. (2013). *Big data. La revolución de los datos masivos*. Turner.

Piatetsky-Shapiro, G., Frawley, W. (1991). *Knowledge Discovery in Databases*. AAAI/MIT Press, Cambridge MA.

Siegel, E. (2013). *Analítica predictiva. Predecir el futuro utilizando Big Data*. Anaya Multimedia-Anaya Interactiva.



Autor
Dr. José A. Olivas Varela