

SDJ EXTRA

by *Software Developer's*
JOURNAL

¡Especialmente para los lectores de SDJ Extra!:

Visual Prolog 6.3 Personal Edition

la versión completa junto con los paquetes adicionales GUI

WIN-PROLOG 4.600

versión de desarrollo

¡6 LIBROS GRATIS!

Fernando C. N. Pereira and Stuart M. Schieber **Prolog and Natural Language Analysis**

Jonathan Bartlett **Programming from the Ground Up**

W. N. Venables, D. M. Smith, and the R Development Core Team **An Introduction to R**

Mark Watson **Practical Artificial Intelligence Programming in Java**

Tim Hendtlass **Real Time Forth**

Anthony A. Aaby **Compiler Construction using Flex and Bison**



Inteligencia artificial

Redes de neuronas en los juegos

Implementación de redes de neuronas que controlan los objetos en los juegos

Chatbot inteligente

Ehab El-agizy y Moustafa Zamzam nos enseñan a crear nosotros mismos un chatbot que reconozca los comportamientos del usuario

IA de Elbot

Fred Roberts presenta la Inteligencia Artificial de Elbot

Búsqueda de fuentes de datos en páginas web por palabras clave

Matthew Michelson y Craig Knoblock se centran en la extracción de información

Echo Bots – Acercamiento minimalista a la Inteligencia Artificial

Jeremy Gardiner analiza si los chatbots escritos son de verdad IA

Acercamiento « NLP » al TalkToMyPalm

WabyanKo presenta lo original de uno de los sistemas más pequeños, « NLP » - TalkToMyPalm

¿Cuándo nos superarán?

William Wynn analiza si la IA se desarrollará hasta tal punto que controle a la Humanidad

Uniendo la investigación en IA con los Servicios Web y la Web Semántica

David Burden explica cómo construir un robot útil



+ Entrevista con Hugh Loebner



Matthew Michelson
Craig A. Knoblock

Mas allá de la búsqueda de fuentes de datos a través de palabras clave en la World Wide Web

Una de las funciones más importantes de la World Wide Web es la capacidad de proporcionar al usuario acceso a gran cantidad de información. Sin embargo, mucha de esta información aún está desorganizada y es inaccesible más allá de una simple búsqueda de palabras clave. Por ejemplo, ten en cuenta el sitio web de subastas EBay (<http://www.ebay.com>).

Si un usuario quiere determinar el precio medio de demanda de un artículo o contar los sucesos, tendrá que hacer una búsqueda de palabras clave, recuperar todas las listas, filtrar aquellos que no sean pertinentes y determinar artículo por artículo la respuesta a las preguntas. También ten en cuenta que una búsqueda de palabras clave no encontraría todos los artículos que tengan algún error de ortografía. Está claro que este tipo de búsqueda no es lo suficientemente potente para conceder respuestas interesantes acerca de los datos, especialmente si preferimos que los programas en lugar de los usuarios determinen estas respuestas.

Es preferible insertar un mecanismo en las listas de EBay que permita que se les haga solicitudes de una manera estructurada. De esta manera, determinar el precio promedio o el cálculo de un artículo sería sencillo: una solicitud de una sola línea que incluso puede realizarse a través de un programa. Este artículo presenta uno de estos métodos que permite que las bases de datos, tales como EBay, admitan solicitudes estructuradas. Le llamamos a cada una de las listas un "envío" y el objetivo es extraer de cada envío los "atributos" incrustados en el mismo que describen la identidad. Esta extracción es más conocida formalmente como Extracción de Información (IE).

Utilizando nuestro ejemplo de EBay, los envíos pueden tratar sobre coches. En este caso, los atributos serían fragmentos descriptivos sobre el coche, tales como el fabricante, el modelo o el año, y realizando la IE en un envío nos permitiría seleccionar todos los atributos importantes, incluso si están mal escritos y sin importar su ubicación en el texto del envío. Cuando hayamos extraído los atributos, podemos entonces añadir etique-

Matthew Michelson estudia en la Universidad de Southern California. **Craig A. Knoblock** es el jefe principal de proyectos en el Instituto de Ciencias de la Información y profesor adjunto de investigación de la USC. Contactos: {michelson,knoblock}@isi.edu

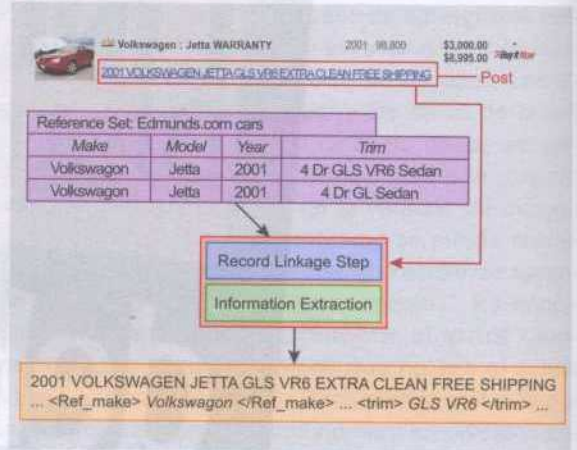


Figura 1. Los puntos de nivel de similitud de registro

tas alrededor de ellos, y consultar la fuente de datos utilizando estas etiquetas como esquema de la consulta.

La adición de estas etiquetas se conoce como "comentario." El enfoque general al comentario se muestra en la Figura 1, utilizando un ejemplo de envío de EBay. Las partes diferentes de esta figura se describen luego en el artículo, pero es práctico señalar los ejemplos de comentarios en el fondo de la figura en la casilla en negro.

Las fuentes de datos no estructuradas y no gramaticales

En este artículo nos centramos en las fuentes de datos de comentarios que "no están estructuradas" y "no son gramaticales." Cuando hablamos de fuentes de datos no estructuradas queremos decir que varían en el orden e inclusión de los atributos, de envío a envío. Un envío de EBay de un coche podría incluir el modelo y año, en ese orden, mientras que otro tendría el año y el fabricante. Ya que el orden y la inclusión de los atributos varían aleatoriamente dentro del envío, no podemos explotar esta estructura, lo que simplificaría el problema.

Lo "no gramatical" se refiere al hecho de que la mayor parte del tiempo los envíos no concuerdan con las reglas del lenguaje. Si así fuera, podríamos explotar las técnicas de Procesamiento de Lenguaje Natural (Natural Language Processing) para descubrir más fácilmente los atributos. Por ejemplo, podría ayudar a identificar los sustantivos en el correo enviado. Sin embargo, y ya que el envío no es gramatical, no puede ser analizado como una oración.

Nos centraremos específicamente en las bases de datos no estructuradas y no gramaticales porque ya ha habido mucha investigación sobre la extracción de atributos desde datos semi-estructurados y estructurados, tales como páginas web, así como sobre los datos que concuerdan con las reglas gramaticales, tales como la extracción de identidades de los artículos de noticias. La IE en las fuentes de datos no gramaticales y sin estructura plantea una dificultad porque sin estructura ni gramática hay pocas pruebas para identificar qué artículos en un envío son atributos y cuáles son símbolos que pueden ser ignorados, a los que llamamos "chatarra" de noticias.

Los Sets de Referencia

Ya que no podemos fiarnos de la estructura o la gramática de las fuentes de datos, debemos infundirle a la IE conocimientos externos para darle pistas sobre qué fragmentos de un envío son atributos y cuáles son "chatarra".

Estos conocimientos externos vienen en forma de "sets de referencia". Un set de referencia es un grupo de referencias que viene con los atributos asociados. Pueden ser:

- una base de datos online u offline,
- un conjunto de documentos online u offline.

Utilizando nuestro ejemplo de los coches de EBay, un set de referencia podría ser la página web de coches de Edmunds (<http://www.edmunds.com>) que proporciona los atributos para las entidades de coches.

En este caso los atributos son datos tales como el modelo, el año y la tapicería.

La información en este sitio web puede ser organizada para que luzca similar a una base de datos, en la que cada entidad tiene sus atributos asociados, y esta base de datos de coches constituye un set de referencia. La Figura 1 incluye una parte del set de referencia de ejemplo del sitio web de Edmunds.

Este set de referencia se utiliza alineando en cada envío para que se corresponda mejor con el miembro del set de referencia.

Este proceso de alineación se conoce como Enlace del Registro (Record Linkage), y se representa por la casilla azul de la Figura 1 marcado como Paso de Enlace de Registro. Este paso toma como entrada al set de referencia y

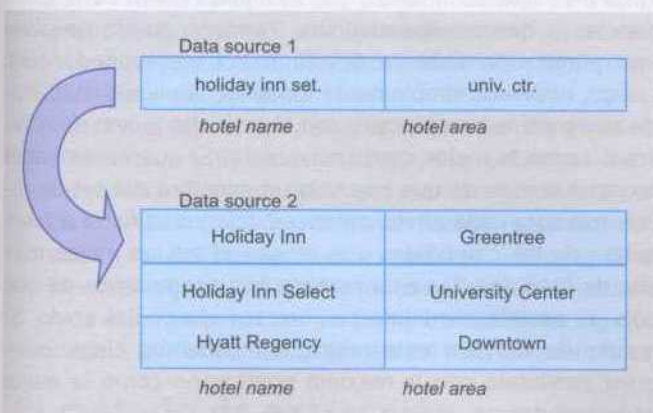


Figura 2. Ejemplo de enlace del registro tradicional sobre dos fuentes de datos de hoteles



Figura 3. Enlace del registro entre un envío sobre hoteles y un set de referencia de hoteles

al envío, y como salida a la mejor correspondencia posible con el primero, si existe alguna. Entonces el envío que mejor se corresponde proporciona las pistas necesarias para realizar la IE, pues podemos buscar los valores del atributo del miembro del conjunto de referencia en el envío. La siguiente sección se centra precisamente en cómo se realiza exactamente el enlace del registro.

El paso de enlace del registro

Hasta este punto hemos descrito qué son los envíos y qué los hace únicos, que es específicamente su naturaleza no estructurada y no gramatical. Sin embargo, debido a su falta de gramática y estructura, nos damos cuenta de que necesitamos algún tipo de conocimiento externo para realizar la IE. Este conocimiento viene en forma de sets de referencia, y afirmamos que utilizamos estos sets descubriendo el registro del set de referencia que mejor se corresponde con el envío que estemos comentando.

El descubrimiento de esta mejor correspondencia se conoce como enlace del registro. El Enlace del Registro (Record Linkage) ha sido estudiado mucho tiempo en las comunidades de inteligencia artificial y de base de datos. El enlace tradicional toma un registro de una fuente de datos y encuentra el que le corresponde en otra. Esto se hace examinando los atributos de cada fuente de datos y decidiendo que se correspondan probablemente los registros que contengan los atributos más parecidos. La Figura 2 muestra un buen ejemplo de enlace del registro tradicional en dos fuentes de datos de hoteles.

Nuestra diferencia en el enlace del registro

Sin embargo, nuestro enlace del registro es diferente en lo mínimo y requiere un nuevo enfoque. Un envío aún no se descompone en atributos, así que los enfoques tradicionales del enlace de registros no se corresponden. Es decir, los atributos para examinar la similitud están incrustados en el envío, de manera que no es posible la comparación a través de un atributo.

En cambio, nuestro método implica la creación de un vector de calificación de similitud entre el envío y todos los atributos del conjunto de referencia concatenados. De esta forma podemos aproximarnos al atributo por la similitud de atributo para todos ellos de una vez, lo que concede una similitud entre todo el registro y el envío. A esto se llama "similitud a nivel de registro" (record level similari-

ty). De modo que, por ejemplo, utilizando el primer miembro del set de referencia de la Figura 1, la calificación de la similitud a nivel de registro serían las calificaciones parecidas entre los envíos, "VOLKSWAGEN JETTA 2001 GLS VR6 TRANSPORTE GRATIS SIN RECARGOS EXTRA" y los atributos concatenados "Volkswagon Jetta 2001 4 Dr GLS VR6 Sedan."

De cualquier manera, no hay suficiente información en la similitud a nivel de registro para distinguir una correspondencia. Específicamente, es posible que dos registros compartan la misma similitud a nivel de registro que un envío, mientras que difieren en cuanto a los atributos de los registros que causaron esta similaridad.

Por ejemplo, tengamos en cuenta la Figura 3, que muestra el enlace de registros entre un envío sobre hoteles y un set de referencia de hoteles. En la figura, cada miembro del conjunto se corresponde con el envío de 2 atributos, con el nombre del hotel en común. Sin embargo, el primero se corresponde con el área del hotel mientras el segundo lo hace con las estrellas. La zona del hotel es más discriminativa que la clasificación por estrellas, así que necesitamos alguna manera de reflejar las similitudes entre el envío y cada atributo individualmente. Esto se hace agregando calificaciones de similitud entre el envío y cada atributo del set de referencia de acuerdo con la calificación del vector de similitud. Estas comparaciones del atributo individual le aproximan al atributo a través de la comparación, y se llaman "similitud a nivel de campo" (field level similarity). Con nuestro ejemplo en ejecución podríamos generar calificaciones de similitud entre el envío y "Volkswagon", entre el envío y "Jetta" y de ahí en adelante.

De modo que nuestro vector total de similitudes incluye ambas nociones de similaridad a nivel de registro, con la concatenación de atributos, y la similitud a nivel de campo, con cada atributo individualmente. Debería tenerse en cuenta que este vector de similitudes no se crea para ca-

da registro del set de referencia. De hecho, en el enlace del registro en general, los miembros de un conjunto de datos nunca se comparan con todos los miembros de otro conjunto. En su lugar, sólo se elige un subconjunto de registros llamado "candidatos", y estos candidatos se utilizan en el proceso de enlace de los registros.

La elección de candidatos para el enlace del registro se llama "blocking" (o agrupamiento en bloques). El objetivo del blocking es limitar el tamaño de los grupos de candidatos, sin eliminar ninguna correspondencia verdadera, tan rápido como sea posible. Nuestra técnica de comentario es independiente del algoritmo de blocking escogido. Por ejemplo, podemos decirle sencillamente a cualquier miembro del set de referencia que comparte una misma ficha con el envío que es un candidato.

Una vez todos los registros de candidatos del set de referencia tengan un vector de calificación de similitud, estos son recalificados de una manera binaria. En cada índice del vector de similitud especialmente, los candidatos con el máximo valor en ese índice cambian su valor a 1, y el resto de los candidatos cambian su valor a 0. Dado el ejemplo de dos vectores que corresponden a candidatos:

$$V1 = (0.1, 0.2, 0.5, \dots, 2.1)$$

$$V2 = (0.2, 0.1, 0.5, \dots, 0.8)$$

Después de la recalificación binaria, se convertirían en:

$$V1 = (0, 1, 1, \dots, 1)$$

$$V2 = (1, 0, 1, \dots, 0)$$

La recalificación binaria se hace para enfatizar la diferencia de las calificaciones entre los vectores. Ciertos vectores tendrán índices con calificaciones de similitud cercanas, pero estas estarán exageradas por la recalificación binaria, de modo que la mejor correspondencia debe ser detectada más fácilmente.

La construcción y calificación de los vectores de similitud son en sí un paso de preprocesamiento para la porción de aprendizaje de la máquina del paso de enlace de registro que nos dirá cual de los candidatos es, de hecho, el que mejor se corresponde con el envío.

La porción de aprendizaje del enlace del registro de la máquina toma el set de candidatos (que ahora es un conjunto de vectores binarios) y los etiqueta como correspondencias o no-correspondencias. También puede devolver una puntuación fiable asociada a las correspondencias. Luego, podemos simplemente tomar al candidato etiquetado como correspondencia y con el más alto índice de fiabilidad, como la mejor correspondencia. Si queremos hacer prevalecer la idea de que hay sólo un miembro del set de referencia para cada envío, entonces dejaremos fuera a cualquiera de los candidatos que tengan el mismo índice más alto de fiabilidad. De esta manera nos aseguramos de que sólo un posible candidato se corresponda con el envío. Si queremos suavizar esta restricción podemos elegir cualquier candidato con la máxima puntuación como la mejor correspondencia.

La técnica real de aprendizaje de la máquina utilizada se conoce como Máquina de Soporte de Vectores (Support

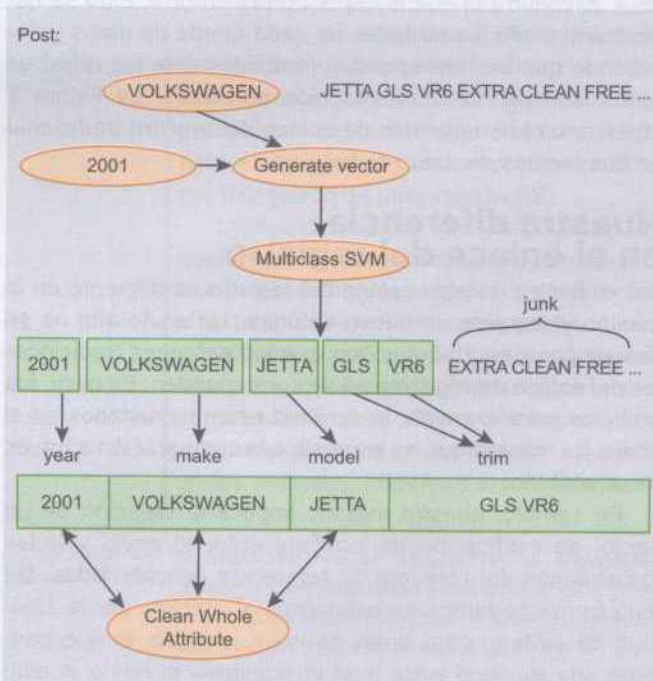


Figura 4. El proceso de extracción al completo

Vector Machine o SVM). Las SVMs se utilizan ampliamente en la investigación sobre la inteligencia artificial como una técnica efectiva de aprendizaje de la máquina. La idea esencial es tener en cuenta a los vectores en el espacio dimensional n . Si los problemas de aprendizaje fueran fáciles, podríamos acoplarlos en un plano dimensional n , llamado hiperplano, entre los vectores que son correspondencias y aquellos que no lo son. Dependiendo entonces de dónde se encuentre un vector, podríamos determinar su clase. Ya que no podemos hacer esto, la SVM asocia cada uno de los vectores en un nuevo espacio dimensional n , en el que son capaces de acoplarse a un hiperplano que distingue los conjuntos. Entonces este determina si un vector es una correspondencia o no, dependiendo de dónde se encuentre con relación a este nuevo hiperplano en el nuevo espacio de funciones.

Las SVMs son técnicas supervisadas de aprendizaje de la máquina, lo que significa que necesitan datos de entrenamiento etiquetados. Estos datos de entrenamiento etiquetados consisten en pares de envíos y en miembros de los sets de referencia que previamente han sido marcados como correspondencias y no-correspondencias. Esto permite a la SVM aprender cómo ajustar el hiperplano a los datos, de modo que pueda clasificar a los vectores que antes no se veían.

Cuando la SVM ha determinado cuál es la mejor correspondencia, con el envío se efectuará un procedimiento final en el paso de enlace del registro. Agregamos un comentario al envío que incluye los atributos del miembro del set de referencias que mejor se corresponde. En la Figura 1, uno de estos atributos se muestra con las etiquetas <Ref_make>. Esto se hace por varias razones.

- Primero, proporciona valores comunes sobre los cuales se solicitan los datos. Si utilizamos los valores reales extraídos para consultar los datos, incurrimos en el mismo problema de la búsqueda de palabras clave, es decir, que diferentes errores ortográficos excluirán de la consulta a ciertos envíos.
- Segundo, este incluye atributos que podrían no haber sido introducidos por el usuario. En nuestro ejemplo de coches de EBay en curso, un envío podría incluir un modelo, la tapicería y el año. Si consultamos los envíos sobre el fabricante, este envío, sin embargo, quedará excluido pues no incluye el fabricante de manera explícita. En cambio, con los atributos del set de referencia comentados en él, ahora tendría un fabricante y sería devuelto por la consulta.
- Por último, hay ciertos atributos que son extremadamente difíciles para la IE. La extracción podría realizarse escasamente en estos atributos que quedarían inutilizados para las consultas, pues devolverían o muchos resultados erróneos o insuficientes envíos. Sin embargo, si el paso de enlace del registro ha funcionado bien, entonces utilizaríamos estos atributos del set de referencia en su lugar, de manera que ese atributo aún sea útil para consultar datos.

El método de extracción

Una vez hayamos encontrado al miembro que mejor se corresponda con el set de referencias, podemos aprovechar

al mismo para la IE. La intención es coger cada símbolo del envío y observar si se corresponde con alguno de los atributos del set de referencia. A esto se le llama Paso de Extracción de la Información y se muestra en la Figura 1 como una casilla en verde.

Tomamos de manera específica cada símbolo del envío y creamos un vector de calificación de similitud entre ese símbolo y cada atributo del miembro que le corresponde del set de referencia. Esto es similar a la creación de un vector de puntuación de similitud en el paso de enlace del registro, excepto cuando no realizamos una recalificación binaria en este vector, pues no necesitamos escoger como antes un ganador entre el grupo de candidatos. También, este vector contiene un único conjunto de puntuaciones para comparar la similitud de este símbolo con los atributos especiales llamados "comunes".

Estos atributos comunes son en general datos que no son fácilmente representados por los sets de referencia, aunque muestren suficientes características identificativas para ser explotados. Los precios y las fechas serían ejemplos de atributos comunes. Estas características más fiables permiten la extracción de estos tipos de datos empleando técnicas más tradicionales, como las expresiones regulares. Así que incluimos las puntuaciones que se correspondan con una expresión regular en cuanto al precio, por ejemplo. Esta puntuación le daría una calificación positiva a una correspondencia y mostraría 0 si fuese lo contrario. Esto permite la extracción de atributos prácticos dentro del envío, que son fácilmente extraídos mediante algunas reglas expertas.

Cuando hayamos creado el vector de calificación de similitud, intentamos identificar el tipo de atributo del vector de símbolo, pasando este vector a una SVM multi-clase. Una SVM multi-clase es capaz de identificar un símbolo como miembro de las clases n . En nuestro caso, el $n-1$ de las clases son los tipos de atributo, tales como el fabricante del coche o el modelo, y la n -ésima clase es la "chatarra", lo que significa que el símbolo puede ser ignorado.

Una vez han sido identificados todos los símbolos en un envío, como pertenecientes a un atributo o a la chatarra, limpiamos cada uno del total de atributos extraídos del envío. El total de atributos extraídos es sencillamente la concatenación de cada uno de los símbolos en el envío que tengan la misma etiqueta. En el ejemplo antes mencionado, los símbolos "GLS" y "VR6", sería cada uno identificado como la tapicería de un coche, de modo que el total extraído de la tapicería sería "GLS VR6". Necesitamos etiquetar los símbolos de manera intuitiva e individual-

Otras Lecturas:

- Matthew Michelson y Craig A. Knoblock, "Semantic Annotation of Unstructured and Ungrammatical Text", 19a Conferencia Internacional sobre Inteligencia Artificial (IJCAI), Edinburgo, Escocia, 2005
- Matthew Michelson, "Building Queryable Datasets from Ungrammatical and Unstructured Sources", Tesis Doctoral, Universidad de Southern California, 2005
- Estos artículos pueden conseguirse en www.isi.edu/~michelson

mente pues los datos pueden no tener estructura alguna, así que no podemos explotar la estructura del envío. En cambio, esto introduce símbolos ruidosos, es decir, símbolos que debían haber sido etiquetados como chatarra y no lo fueron.

Para rectificar este problema tomamos cada total del atributo extraído y lo comparamos con su atributo correspondiente del miembro del set de referencias. Eliminamos un símbolo a la vez en el atributo extraído, y si este nuevo atributo se corresponde mejor que el antiguo con el atributo del set de referencia, este símbolo se convierte en candidato para la eliminación. Después de procesar cada símbolo de esta manera, o quitamos el candidato a la eliminación que proporcionó la mejor correspondencia con el atributo del set de referencias, o finalizamos el proceso, pues no hay símbolos que proporcionen una mejor correspondencia. Si no finalizamos, entonces volvemos a comenzar el ciclo.

Este proceso no sólo elimina los símbolos ruidosos del atributo extraído, sino que le ha añadido una beneficiosa interpretación gramática. Si un símbolo podría pertenecer fácilmente a más de un tipo de atributo podríamos etiquetarlo de acuerdo con ambos tipos y luego dejar que el proceso de limpieza elimine el símbolo del atributo al que no debe pertenecer.

Esta es la técnica completa de extracción de los atributos de un envío para construir los comentarios que permitan hacer consultas estructuradas. La Figura 4 muestra el proceso de extracción al completo.

Deben tenerse en cuenta unas cuantas observaciones sobre este proceso de extracción. Primero, ya que habitualmente estos atributos se superponen en un set de referencia, este enfoque de la extracción es resistente a los errores cometidos en el paso de enlace del registro. Si el paso de enlace del registro falla en identificar correctamente un miembro correspondiente del set de referencia, la correspondencia que identifique por lo general poseerá suficiente información similar como para darle utilidad. Por ejemplo, con los coches de EBay, si el paso de enlace del registro devuelve un Volkswagen Jetta 2001, pero con la tapicería incorrecta, habría suficiente información útil como para extraer el fabricante, modelo y año del coche del envío. Esta es una de las razones por las que el algoritmo no debe detenerse después del paso de enlace del registro, aunque podría parecer tentador, pues en ese punto la mayoría de los datos son comentados con valores estándar.

Otra razón para realizar la extracción, más allá del deseo obvio de ver los valores reales que los usuarios introducen para los atributos, es la de entrenar al sistema para extraer todos los atributos que le ayudarán a extraer atributos "comunes" de dos maneras. Primero, hace de la chatarra una clasificación aún más rara, lo que mejora la exactitud para clasificarla. Segundo, es práctico en la clasificación de aquello que no lo es. Ten en cuenta el siguiente ejemplo. Quizás un modelo de coche se llame "\$35". De hecho, este es un caso extraño, pero si el sistema no estuviese entrenado para extraer los modelos de coche, este símbolo seguramente sería clasificado como precio. Sin embargo, el sistema puede saber que es un modelo de coche y no un precio.

Observaciones del Algoritmo

Este algoritmo ha sido probado empíricamente para emplear otros métodos de enlace del registro y de extracción en la tarea de anotación semántica. Para ser más específicos, el ejemplo en ejecución que se utiliza a lo largo de este artículo es bastante fácil, y el algoritmo ha sido probado para que funcione bien con datos mucho más difíciles, por ejemplo en envíos a los que les falten muchos atributos y tengan varios errores ortográficos en el valor del atributo.

Los lectores interesados pueden leer las publicaciones científicas que se mencionan en este artículo, en la sección Otras Lecturas, para comprobar las verdaderas cifras de la ejecución. Además, las publicaciones de investigación ofrecen un tratamiento más detallado y avanzado de los algoritmos.

Otro aspecto del funcionamiento a considerar con un método de aprendizaje supervisado es la cantidad de datos de entrenamiento que el sistema necesita para funcionar bien. Nuevamente, las publicaciones de investigación muestran que ese entrenamiento, en sólo el 10% de los datos (menos de 80 ejemplos en algunos casos) no causa la degradación del funcionamiento del algoritmo. Esto es beneficioso, pues los datos de entrenamiento de la etiquetación son tediosos y caros.

Una última observación con respecto a la ejecución tiene que ver con los sets de referencia. Puede parecer que si utilizamos dos sets de referencia diferentes (pues el algoritmo no está sujeto a uno sólo), tendríamos que utilizar el producto cruzado de los registros en estos dos sets de referencia como uno solo, como un amplio set de referencias. Esto empeoraría el funcionamiento general debido a la gran envergadura del nuevo set de referencia. Este no es el caso.

El algoritmo admite fácilmente el uso de sets de referencia de una manera iterativa, así que se pueden usar tantos sets de referencia como sean necesarios. De hecho, esto no es una optimización práctica en la que los sets de referencias más pequeños de los atributos únicos de referencia pueden formarse y ser utilizados en lugar de los sets de referencia individuales más grandes.

Conclusión

Este artículo ha presentado un método a través del cual se les puede dar utilidad a los datos no gramaticales y no estructurados de la World Wide Web, comentando los datos para los que admitan consultas estructuradas. Estas consultas se desplazan más allá de la simple búsqueda de palabras clave para convertir los datos en información. En el futuro, cuando los agentes de la web actúen en nuestro nombre, reservando viajes y canjeando bienes, este tipo de consultas serán cruciales en el proceso de toma de decisiones.

La investigación que aquí se expone es un primer paso, y esperamos que el progreso futuro contribuya a estrechar el agujero entre el tiempo en el que los usuarios tienen que realizar tareas tediosas en la World Wide Web y el tiempo en el que sus agentes las realicen por ellos. ■