

# Identificar entidades para un modelo de acción en Planificación Automática

Carlos Henriquez Miranda, Jaime Guzmán Luna

chenriquez@uac.edu.co, jguzman@unal.edu.co

Universidad Autónoma del Caribe, Colombia  
Barranquilla - Colombia

**Resumen:** La Planificación Automática (PA) es la disciplina que busca la producción de una secuencia de acciones que permiten alcanzar una meta específica, y que requiere la definición de un modelo de acción como flujo de entrada. Sin embargo, comenzar la elaboración de este modelo es una tarea difícil incluso para expertos planificadores. Este paper propone identificar los elementos que corresponden a un modelo de acción para tareas de PA a partir del análisis de un conjunto de páginas Web donde se encuentra información de planes ya producidos y que resuelven un problema particular. El proceso consta de varias fases, la extracción de información de la Web, un preprocesamiento usando herramientas de Procesamiento de Lenguaje Natural (PLN), la identificación de entidades para un modelo de acción y finalmente, el almacenamiento de las acciones extraídas en una ontología para formar una gran Base de Conocimiento (BC) que permita más tarde utilizarlas para otros dominios en PA. En este artículo, se presenta el resultado de investigación parcial del uso de las herramientas de extracción, preprocesamiento, identificación de componentes y almacenamiento en la ontología.

**Palabras clave:** Extracción de información, Planificación automática, Ontología, Web, PLN, PDDL.

**Abstract:** Automated Planning (AP) is the discipline of Artificial Intelligence that aims to produce a sequence of actions to achieve a specific goal, which requires the definition of an action model as an input stream. However, start the construction of such model is a difficult task even for experts. This paper proposes to extract information from the Web in order to identify the elements that correspond to the model of action for AP tasks, by analyzing a set of Web pages to find information related to some produced plans to solve a particular problem and extract from them a set of steps that allow the solution to any given problem. After the data were obtained, they will be processed using Natural Language Processing (NLP) tools, in order to recognize a set of actions that is part of a domain model in AP. All the extracted actions are stored in an ontology to structure a Knowledge Base (KB) allowing later, use them to other domains in AP. In this paper we present the partial researching overcomes about the uses of the extraction tools, pre-processing, identification and storage components in the ontology.

**Keywords:** Information Extraction, Automatic Planning, Ontology, Web, PLN, PDDL.

## 1. Introducción

Del área Inteligencia Artificial (IA) se desprende el concepto de planificación, que según [1] es el proceso de búsqueda y articulación de una secuencia de acciones que permiten alcanzar un objetivo. La planificación busca producir planes para que sean usados por humanos o agentes inteligentes. Para lograr que esta producción se realice automáticamente (mediante un planificador), hay que definir una serie de elementos, como: (i) estado inicial, que es la situación de partida, (ii) la meta que describe las condiciones que se tienen que dar para considerar por terminado el proceso, (iii) las acciones que transforman un estado a otro, (iv) el plan mismo, que es el conjunto de acciones que permite pasar del estado inicial al estado final y (v) las heurísticas, que es el conocimiento que permite obtener de forma eficiente un plan. Algunos elementos, como las acciones, hacen parte del modelo de acción conocido como modelo de dominio. Otros elementos, como el estado inicial, hacen parte del problema a resolver por un planificador. Ambas especificaciones son necesarias para lograr el proceso de planificación, sin embargo, la construcción de estos modelos, a partir de cero es una tarea difícil aun para un conjunto de expertos.

En este trabajo, se propone identificar inicialmente un conjunto de acciones y otros elementos a partir de recursos encontrados en la Web. En ésta, se encuentra una enorme cantidad de recursos representados en diferentes

fuentes y formatos. Gran parte de estos recursos se presentan en lenguaje natural en forma de blogs, wikis o redes sociales [2], otras en formas semiestructuradas como los que ofrecen ventas de productos o servicios. Por esa variedad de presentaciones y estructuras, recuperar información desde estos recursos se vuelve una tarea difícil y a veces imposible. El área conocida como Recuperación de Información (RI) aborda este problema encontrando documentos relevantes desde un gran repositorio en respuesta a un criterio definido [3]. Como existe mucha información, se han creado buscadores que se encargan de recuperarla por intermedio de consultas clave [4]. Pero no sólo es traer documentos relevantes de una consulta específica, ya que en la Web cualquier individuo puede brindar información valiosa en diferentes áreas, como la economía, industria, medicina, robótica entre otras, sino que deben existir otro tipo de sistemas apoyados en técnicas de Inteligencia artificial (IA) que se encarguen de buscar dentro de los documentos, explorar su contenido y extraer información pertinente de un tema en particular. Estas nuevas herramientas se enmarcan en el área conocida como Extracción de información (EI) que se ocupa de estructurar los contenidos dentro de los textos que son relevantes para el estudio de un dominio particular [5]. En otras palabras, el objetivo de un sistema de EI es encontrar y enlazar la información relevante mientras ignora la extraña e irrelevante [6].

Más concretamente, una tarea de EI es definida por el documento de entrada y el objetivo de extracción. La entrada pueden ser documentos sin estructura escritas en lenguaje natural o documentos semiestructurados que se presentan como tablas o listas detalladas y enumeradas y lo que se extrae de allí puede ser: nombres de personas, países, años, autos, ocupaciones, entre otros; siempre dependiendo del criterio específico que represente las necesidades de búsqueda.

El resto del artículo se organiza de la siguiente manera: en la próxima sección, se hace una revisión de los trabajos más significativos, luego, se propone un modelo para identificar las acciones; en la siguiente parte del texto se muestran los resultados de un prototipo y, finalmente, se presentan las conclusiones.

## 2. Trabajos previos

En los trabajos más representativos de la literatura se han abordado diferentes dominios de información, usando múltiples formas para llevar a cabo el proceso de EI. Por ejemplo: en [7], se describe un enfoque para análisis de la comunicación empresarial específicamente extrayendo información de los correos electrónicos, por su parte en [8], se ubican entidades y sus respectivos conceptos a partir de la exploración de las tablas en un documento HTML. En [9] se presenta un paradigma de extracción que facilita el descubrimiento de relaciones extraídas de texto, independiente del dominio y del tamaño del recurso Web. Por otro lado en [10], se explota la apariencia visual de la información impulsado por relaciones espaciales que se producen entre los elementos de una página HTML. En [11], se construyen herramientas robustas para la extracción de información Web ante cambios en la estructura de la página basadas en un modelo de costo mínimo y en [12] se propone un enfoque de descubrimiento de patrones para la rápida generación de extractores de información. Finalmente en [13] y [14] se muestra como desde un sitio Web se obtienen un conjunto de acciones referentes a planes que luego se convierten en elementos de dominio de PA específicamente para el lenguaje de definición PDDL.

PDDL es un lenguaje centrado en la acción, inspirado en el modelo STRIPS para formular problemas de planificación. Se ha convertido en el lenguaje estándar desde 1998 utilizado en la competición internacional de planificación (ICAPS). Para especificar una tarea en PDDL se identifican dos archivos, el primero define el dominio que describe los predicados y las posibles acciones, el segundo detalla el problema donde se describen los objetos, el punto inicial y la meta. [15]

A partir de [13] y [14] se realizó una primera aproximación en [16] y específicamente en este trabajo usamos otras técnicas de PLN y almacenamos lo extraído en una ontología. El uso de ontologías como BC mejora lo propuesto en [13], [14] y [16] ya que se consigue un gran avance en los procesos de adquisición y gestión del conocimiento asociado a los dominios de planificación, problemática que es tratada por [17] “*la comunidad de planificación debe estudiar las técnicas y herramientas relacionadas con la gestión de conocimiento desarrolladas en otras áreas, y estudiar cómo adaptarlas*

*e integrarlas en los sistemas de planificación.*”. Además se pueden generar nuevos planes por recombinación aprovechando el conocimiento almacenado, así como también, poseer un vocabulario expresivo con el que se puedan definir nuevos problemas de planificación planteando nuevos estados y acciones [18]. Almacenar las acciones en una ontología permite contar con una potente herramienta para realizar interesantes inferencias durante el proceso de búsqueda de planes, lograr mayor expresividad para modelar dominios complejos y usar planificadores potentes previa traducción a un lenguaje de planificación específico.

Para las tareas de extracción se usa un *Wrapper*, programa especializado que identifican los datos de interés con base a una serie de reglas gramaticales, que luego de extraerlos, los transforma a una estructura de datos apropiada para su posterior manipulación [19]. Particularmente, el proceso consiste en una exploración al documento Web escogido y el *Wrapper* revisa toda su estructura HTML en busca de información relevante dependiendo de patrones inicialmente definidos. Con el *Wrapper* se combinan servicios Web basados en RestFul que permiten tener diseños más simples, bajo consumo de recursos, URI por recurso y generalmente por ser servicios fáciles de construir y adoptar [20-22]. Adicionalmente, se exploran herramientas para PLN para tareas de *stopwords*, *stemming* y análisis morfológico [23-24].

## 3. Modelo para identificación de acciones

Para el proceso de extracción, se toma como referencia el recurso Web conocido como *WikiHow*, un repositorio que contiene manuales con más de 150 mil artículos, donde se encuentran textos escritos por un grupo de colaboradores que explican paso a paso como llevar a cabo ciertas tareas. Una página de *WikiHow* consta de título, descripción, categoría, ingredientes y un conjunto de pasos. En la Figura 1 se puede ver un extracto de una página que describe cómo hacer galletas.



Figura 1. Página de WikiHow

Para el proceso de extracción, se usa la Tabla 1, que describe de dónde se va a extraer cada componente.

Recurso Web	Planificación
Page	Plan
Title	Objetivo final
Ingredients	Objetos
Steps	Acciones
Category	Categoría

Tabla 1. Comparación entre la Web y Planificación

Cada página analizada representa un posible plan compuesto por una secuencia de pasos que permiten alcanzar un objetivo deseado. Cada plan pertenece a una categoría del recurso *WikiHow*.

Para todo el proceso, se propone un modelo presentado en la Figura 2, dividido en cuatro fases: La primera, fase de Extracción, recibe una URL (dirección donde está la página) y entrega un conjunto de datos útiles para su procesamiento posterior. La segunda, fase de preprocesamiento, toma la trama recibida con la información extraída y realiza actividades correspondientes a un análisis léxico. La tercera fase identifica cada uno de los componentes y la última fase, toma los componentes identificados y los guarda en una ontología.

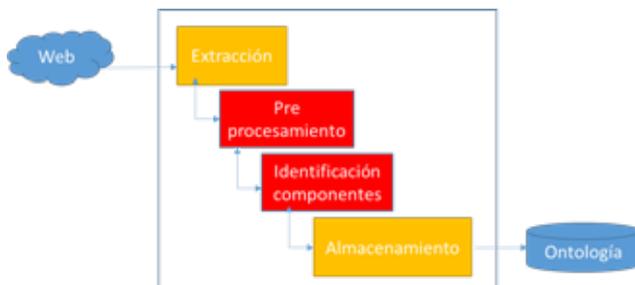


Figura 2. Modelo para identificar acciones

A continuación, se explicará detalladamente lo que realiza cada fase:

En la fase 1, se realiza la extracción de los pasos de un plan. Esta recibe la página a procesar y obtiene a partir de ella las entidades representadas en título, categoría, ingredientes y pasos. Para este componente, se hace un análisis de la estructura de la página y se implementa un algoritmo (*Wrapper*) basado en la Tabla 1. Para realizar el proceso de extracción de información asociada a lo que se requiere, se identificaron, en cada recurso, patrones en donde se localizaban el plan y sus pasos. Se revisaron muchas páginas similares y se obtuvo el patrón para la extracción.

En la Figura 3, se puede observar el resultado de la ejecución del *Wrapper* cuando se le pasa la página de la figura 1 representada en la url: "<http://www.wikihow.com/Make-Bear-Paw-Cookies>".

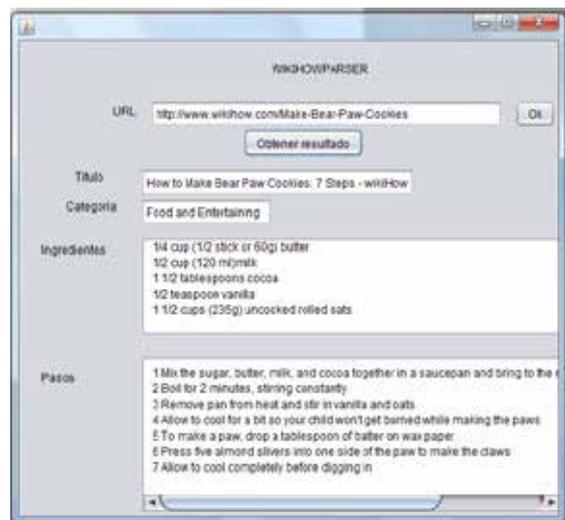


Figura 3. Extracción del Wrapper

En la fase 2, se toma el resultado de la extracción (ver figura 3) y se realiza actividades correspondientes a un análisis léxico. Dentro de estas actividades, se realiza primero un análisis morfológico en los pasos extraídos para clasificar cada entidad en alguna de las siguientes categorías: verbos (*VERB*), sustantivos (*NOUN*), adjetivos (*ADJECTIVE*) y adverbios (*ADVERB*). Para la implementación de este componente del modelo se usa la herramienta *Stanford POS Tagger* creada por investigadores del grupo de procesamiento del lenguaje natural en la Universidad de Stanford [25]. Luego de esta etapa, se limpian los datos borrando algunas palabras sin significado como artículos, pronombres, preposiciones, etc. Estas palabras en el idioma inglés se conocen como *stopwords* [26]. Adicionalmente, para algunas entidades, se utilizan herramientas de *stemming* para reducir una palabra a su raíz o lema.

En la fase 3, se identifican los componentes que harán parte del plan. Para esto, con base en la tabla 1, se van identificando cada componente. Por ejemplo el título y categoría serán la meta final y la categoría respectivamente, en el modelo de acción del plan. Las acciones serán sacadas de los pasos y los objetos de los ingredientes.

En la fase 4, se toma todo lo identificado y se almacena en una ontología. En la Figura 4, se muestra una ontología básica que permite almacenar las entidades como: plan, categoría, objetos y acciones. Para lograr el proceso de almacenamiento, cada entidad tiene una correspondiente clase en la ontología: *Plan*, *Categoría*, *Objeto* y *AccionExtraida*. Éstas son usadas para crear los respectivos individuos que permiten hacer el poblamiento de la ontología. Además, la ontología tiene las propiedades de cada clase, así como la relación entre cada una de ellas.

Con la información ya almacenada se podrán crear modelos de acción para un dominio de planificación en lenguaje PDDL de la mano de un experto o se podrán realizar aplicaciones inteligentes que puedan llevar la ontología, que guarda el modelo de dominio, directamente a un planificador.



Figura 4. Extracción del Wrapper.

#### 4. Experimentos y Resultados

Para validar parte del modelo, se hizo un prototipo que utilizó diferentes herramientas de programación. Este prototipo se hizo en Java 7.0 con uso de librerías especiales HttpClient 3.1 usando Netbeans IDE 7.11 bajo la arquitectura Rest usando Web services y como servidor GlassFish 3.1.2. Para el proceso de extracción desde recursos Web, se usaron las bibliotecas de Jericho HTML Parser. Para la clasificación de entidades se usó la herramienta Stanford POS Tagger y tareas de stemming snowball[27]. Para el diseño de la ontología, se utilizó protege 3.48 y para la manipulación, consultas y protección las librerías de Apache Jena 2.10. Para el experimento, se tomó WikiHow y se pasaron al modelo las direcciones (URL) de diferentes categorías (5) que organizan un conjunto de planes. En la Figura 5, se muestra el resultado específico de la categoría: Computers and electronics, el cual arrojó una totalidad de 201 planes.

Se tomaron varias páginas Web del sitio *WikiHow* y con su dirección URL se exploraron y se extrajeron un conjunto de pasos. En la Figura 5, se muestra el recurso Web “*How to Prepare Your Boat for Transport*” al cual se aplicó el proceso de extracción.



Figura 5. Recurso WikiHow

Para realizar el proceso de extracción de información asociada a lo que se requiere, se identificaron, en cada recurso, patrones en donde se localizaban el plan y sus pasos. Se revisaron muchas páginas similares y se obtuvo el patrón para la extracción. Por ejemplo, para sacar el título, en la página se encuentra que éste aparece rodeado de los tags <TITLE></TITLE>. Para los pasos se empleó un procedimiento similar. En la Figura 6, se muestra el resultado del proceso que obtiene el título y los pasos. Note

que sólo se extrae lo útil e importante según el fin específico.

Con todo lo extraído, se identifican, de la traza las entidades de cada paso. Por ejemplo del paso 1 “Take the time to plan for your boat move” se dividen en entidades individuales así:

<<Take>><<the>><<time>><<to>><<plan>><<for>><<y our>><<boat>><<move>>. Estas entidades son clasificadas en una categoría ya definida con anterioridad. Por ejemplo, se obtienen las posibles acciones del plan, como el verbo presente en la traza de cada paso (ver Tabla 1).

```

Plan es How to Prepare Your Boat for Transport: 20 steps
1 Take the time to plan for your boat move
2 Choose a Professional Boat Transport Company experienced in moving y
3 Communicate with your Boat Transportation Company
4 Measuring Your Boat for Transport
5 Understand how payment is to be made for both the boat transport ser
6 Remove and properly store the following items
7 Items to be removed inside your boat
8 Items to be removed outside your boat
9 Check for Zebra Mussels
10 Decide whether to shrink-wrap your boat or not
  
```

Figura 6. Extracción de pasos

```

Entidades clasificadas por paso
Loading default properties from tagger models/ws-j-0-18-left3words.tagger
Loading POS tagger model from models/ws-j-0-18-left3words.tagger ... done
Paso 1
Take VERB the N/A time NOUN to N/A plan NOUN for N/A your N/A
Paso 2
Choose VERB a N/A Professional NOUN Boat NOUN Transport NOUN Com
Paso 3
Communicate VERB with N/A your N/A Boat NOUN Transportation NOUN
Paso 4
Measuring VERB Your N/A Boat NOUN for N/A Transport NOUN
Paso 5
Understand NOUN how ADVERB payment NOUN is VERB to N/A be VERB
  
```

Figura 7. Clasificación Entidades

En la Figura 7, se muestra el resultado de la clasificación morfológica de cada entidad. Las que no tienen ningún significado para el estudio aparecen como *N/A*. Después del proceso de clasificación encontramos que existen varios verbos identificados en algunos pasos, lo que acarrea problemas para la futura representación en un lenguaje de planificación (PDDL). Así que, en el siguiente paso se eliminan las palabras que causan ruido (*stopWords*) y las que aparecen con *N/A* para que quede una traza de entidades más limpia.

Para la prueba de las herramientas creadas, se examinaron un conjunto de categorías de *WikiHow* y los resultados se muestran en la Tabla 2.

Tabla 2. Resultados de extracción por categoría

Categoría	Planes Hallados	Pasos Por plan
Pasatiempos	201	1803
Vida Familiar	201	1681
En el trabajo	54	465
Arte y entretenimiento	201	1622
Viajes	93	892

Luego de todo el proceso de extracción, se finaliza guardando las entidades restantes en la ontología como se muestra en la Figura 14. Esto se hace creando, por cada entidad, los individuos correspondientes a las clases en la ontología. Para determinar, por ejemplo, cuál es la acción se toma por cada paso la entidad que sea clasificada como verbo en la frase, esto se hace con la ayuda de su categoría gramatical. También, para esta identificación, se utiliza la clase operador de la ontología que contiene una infinidad de lista de verbos en inglés.



Figura 8. Clasificación Entidades

Adicionalmente, se tomaron varios recursos de planes al azar, cuyo resultado se muestra en la Tabla 3.

Tabla 3. Resultados de extracción acciones por plan

Plan	Entidades Procesadas	Entidades Borradas	% Eliminación	Acciones Extraídas
How to Make Egg 'n' Chicken Chunks	93	42	45,16	9
How to Make Shot Glass Candles with Step-by-Step	104	37	35,57	15
How to Take the Paris Metro	168	69	41,07	22
How to Make Pempek	76	27	35,52	12
How to Take the Paris Metro	77	33	42,85	12
How to Add Medicinal Plants to Your Garden	113	36	31,85	16
How to Clean a House	58	15	25,86	12
How to Make Caribbean Carob Cake	174	72	41,37	9
How to Ride a Personal Watercraft (PWC)	287	99	34,49	38
How to Prepare Your Boat for Transport	203	63	31,03	39

A partir de los resultados obtenidos, podemos decir que todas las categorías analizadas en el experimento arrojaron el conjunto de planes y acciones esperadas según el recurso brindado. El máximo de planes hallados fue 201 que es el máximo número de planes por categoría y los números de pasos más altos encontrados fueron de 1803. Para todos los planes, hubo una extracción de la lista de pasos sin ningún inconveniente en todas las páginas procesadas. Al hacer un análisis ad-hoc de forma manual de algunos planes de salida, se muestra que el sistema funcionó muy bien en la extracción del conjunto de planes por categoría y el conjunto de pasos por cada plan.

Después de la extracción, se nota que las herramientas de clasificación y limpieza funcionan adecuadamente y que el porcentaje de entidades sin significado en los planes es alta (mayor que 26%). Cabe anotar que este es un trabajo

inicial antes de pasar a las siguientes etapas necesarias que son el análisis sintáctico y análisis semántico específicamente para la desambiguación del sentido de las palabras. (*Build-Plan*).

Para finalizar, consideramos que se ha logrado el objetivo en la extracción de acciones, la identificación de entidades y el poblamiento automático de la ontología. Sin embargo, cabe anotar que se encontraron ciertos problemas con la parte de la identificación de las entidades en un 100% como estaba previsto y esto se debe a las mismas características que posee el lenguaje natural. De todas formas quedan por implementar etapas de análisis sintáctico y semántico que complementarían y mejorarían el proceso propuesto. De todas formas, las tecnologías que hacen parte de las herramientas usadas están sujetas a mejoras en el futuro. En cuanto al *wrapper* implementados, cumplieron sus objetivos, pero no están preparados para posibles cambios en la estructura HTML. Para eso se explorará el concepto de generación de *wrapper* automáticamente con técnicas inductivas (*wrapper induction*).

## 5. Conclusiones y trabajos futuros

En este artículo se muestran los resultados iniciales de las herramientas utilizadas para cada uno de los procesos de un modelo propuesto para la extracción de información y la identificación propia de entidades útiles. Éstas pretenden unirse al grupo de herramientas que permitirán la construcción de forma automática de planes a partir de la existencia del conocimiento disponible en la Web. En los resultados obtenidos, se puede observar que el proceso de extracción y análisis de información se hace de manera adecuada a partir del modelo propuesto. No obstante, se presentan problemas en la identificación de la totalidad de las entidades.

Sin embargo, de todo lo extraído, se logró almacenar las entidades identificadas (acciones) en una ontología permitiendo contar con una potente herramienta para realizar interesantes inferencias durante el proceso de búsqueda de planes, lograr mayor expresividad para modelar dominios complejos y usar planificadores potentes previa traducción a un lenguaje de planificación específico.

En este trabajo también se puede percibir la importancia actual del área de la extracción de información en cuanto a la gran cantidad de conocimiento encontrado en la Web. Esto debería ser aprovechado mucho más para recuperar y centralizar conocimiento valioso en áreas donde se participa activamente bajo un paradigma de colaboración entre expertos de cada línea de conocimiento.

Para trabajos futuros, se propone mejorar el proceso de extracción de información que se hace con *wrapper*, haciendo uso de técnicas inductivas que permitan la recuperación de entidades independientemente del formato del recurso.

## Referencias bibliográficas

- [1] S. Russell y P. Norvig. "Inteligencia artificial un enfoque moderno". Prentice Hall. Segunda edición Madrid, España. 2004

- [2] L. Perez. "Redes Sociales, Blogs y Wikis: Tendencias y realidades" .2010.Fecha de consulta: 07 septiembre de 2012.  
URL: [www.slideshare.net/gentedeinternet/blogs-redes-sociales-y-wikis](http://www.slideshare.net/gentedeinternet/blogs-redes-sociales-y-wikis)
- [3] F. Martinez. "Recuperacion de informacion : Modelos, sistemas y evaluacion". EL KIOSKO JMC.2004
- [4] O. Lobo, M. Dolores."Metodos y tecnicas para la indización y la recuperación de los recursos de la Worl Wide Web". Boletín de la Asociación Andaluza de Bibliotecarios. n.57. 1999
- [5] A. Tellez. "Extracción de Información con Algoritmos de Clasificación" Tesis para optar al grado maestro de ciencias. Instituto nacional de astrofísica, optica y electronica. Tonantzintla.Puebla. Mexico. 2005
- [6] J. Cowie, W.Lehnert. "Information Extraction". Magazine Communications of the ACM .Vol. 39 Issue 1, pp 80-91 . Jan. 1996
- [7] M.Laclav'ík, S. Dlugolinsky', M. Seleng,M. Kvassay, E. Zolt', L. Hluchy'. "Email Analysis and Information Extraction For Enterprise Benefit".Computing and Informatics, vol. 30, pp 57–87. 2011
- [8] B. Dalvi,W. Cohen,J.Callan."WebSets: Extracting Sets of Entities from the Web Using Unsupervised Information Extraction" WSDM '12 Proceedings of the fifth ACM international conference on Web search and data mining ,pp 243-252. 2012
- [9] O. Etzioni,M. Banko,S. Soderland,D. Weld ."Open Information Extraction from the Web" . Magazine Communications of the ACM - Surviving the data deluge vol. 51 Issue 12, pp 68-74. December 2008.
- [10] G. Penna,D. Magazzeni, S. Orefice. (2010)."Visualextraction of information from webpages". Journal of Visual Languages & Computing .Vol. 21, Issue 1, , pp 23–32. February 2010.
- [11] D. Liu, X. Wang,L.Li,Z. Yan. "Robust Web Extraction Based on Minimum Cost Script Edit Model".Web Information Systems and Mining Lecture Notes in Computer Science vol. 7529, pp 497-509. 2012
- [12] C.H. Chang, C.N Hsu, S. C. Lui. "Automatic information extraction from semi-structured Web pages by pattern discovery". Journal Decision Support Systems - Web retrieval and mining. Vol. 35 Issue 1,pp 129 - 147 . April 1 2003.
- [13] A. Addis, G.Armano, D. Borrajo."Recovering Plans from the Web" . Proceedings of SPARK, Scheduling and Planning Applications woRKshop, ICAPS'09. 2009
- [14] A. Addis, D. Borrajo. "From Unstructured Web Knowledge to Plan Descriptions". Information Retrieval and Mining in Distributed Environments Studies in Computational Intelligence vol. 324, pp 41-59. 2011
- [15] M. Fox, D. Long. "pddl2.1 : An Extension to pddl for Expressing Temporal Planning Domains". Journal of Artificial Intelligence Research 20 pp 61-124. 2003
- [16] C. Henriquez, J Guzman . "Modelo de extracción de información desde recursos web para aplicaciones de la planificación automática" .Prospectiva . Vol. 10, Nº. 2 (Julio - Diciembre), 2012 , pp. 74-80 Diciembre 2012.
- [17] S. Biundo, R. Aylett, M. Beetz, D. Borrajo, A. Cesta, , T. Grant, ,T. McCluskey ,A. Milani y G. Verfaille." Technological Roadmap on AI Planning and Scheduling" . Informe Técnico IST-2000-29656, PLANET, the European Network of Excellence in AI Planning, 2003.
- [18] A. Ruiz. "Una aproximación ontológica al modelado de conocimiento en los dominios de planificación" . Tesis Doctoral Universidad Complutense de Madrid.España (2010).
- [19] A. Fernández . "Extracción de Información de la Web Basado en Ontologías" . Tesis Maestría del Instituto de Computación – Facultad de Ingeniería Universidad de la República. Montevideo, Uruguay. 2004
- [20] S. Tyagi.. Oracle. "RESTful Web Services". 2006. Fecha de consulta: 6 junio 2012. URL :<http://www.oracle.com/technetwork/articles/javase/index-137171.html>.
- [21] R. NAvarro. "Rest vs Web Service " pp. 5-10. 06/2006. Fecha de consulta: 15 Junio 2013. URL: <http://users.dsic.upv.es/~rnavarro/NewWeb/docs/RestVsWebServices.pdf>
- [22] A. Rodriguez. "RESTful Web services: The basics:" 2008. Feha de consulta: 10 Junio 2013. URL: <http://www.ibm.com/developerworks/webservices/library/ws-restful/>
- [23] L. Alonso .” Herramientas libres para PLN”. 11/2005. Fecha de consulta: 10/08/2013. URL : <http://www.cs.famaf.unc.edu.ar/~laura/freeNLP>
- [24] E. Méndez, J. Moreira.“Lenguaje natural e indización automatizada”. Ciencias de la Información, vol. 30, n. 3, pp. 11-24. 1999.
- [25] K. Toutanova, C. Manning. “Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. .Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora” , 63-70. 2000.
- [26] R. Baeza-Yates and B. Rebiero-Neto. “Modern Information Retrieval” .Addison Wesley, London, England. Chapter 1, pp. 9-15. 2003
- [27] M. Porter. “Snowball: A language for stemming algorithms” 2012. Fecha de consulta :11/12/2012. URL: <http://snowball.tartarus.org/texts/introduction.html>