Trabajos en el área de Recuperación de la Información del grupo IXA de la Universidad del País Vasco

I. Alegria, M. J. Aranzabe, O. Arregi, A. Casillas, A. Ezeiza, N. Ezeiza, R. Urizar
IXA taldea. Euskal Herriko Unibertsitatea.
i.alegria@si.ehu.es

Resumen En este artículo se presentan los trabajos realizados por el grupo IXA de la Universidad del País Vasco en el área de la Recuperación de la Información (IR). El objetivo principal del grupo en este área es la introducción de herramientas de ingeniería lingüística para el euskera en sistemas IR/IE, ya que además de permitir la aplicación de herramientas previamente desarrolladas por el grupo, se pueden obtener nuevas fuentes de conocimiento para la mejora de estas y otras herramientas. En concreto se describen los trabajos realizados en tres proyectos concretos: un buscador Internet/Intranet avanzado para textos en euskera, que está actualmente en funcionamiento, un extractor de terminología en estado bastante avanzado y un proyecto para integración de un conjunto de herramientas para la recuperación y extracción multilingüe de información en bases de datos documentales (*Hermes*)[1].

1 Introducción

En este artículo se presentan los trabajos realizados por el grupo IXA (ixa.si.ehu.es) de la Universidad del País Vasco en el área de la Recuperación de la Información (IR). Se describen los trabajos realizados en tres proyectos concretos:

- un buscador Internet/Intranet avanzado para textos en euskera llamado *GaIn*
- un extractor de terminología en estado bastante avanzado
- un categorizador de entidades dentro de un proyecto para integración de un conjunto de herramientas para la recuperación y extracción multilingüe de información en bases de datos documentales (*Hermes*)

[1] Los trabajos descritos a continuación han sido parcialmente subvencionados por el Gobierno Vasco dentro del programa Universidad-Empresa (Código UE-1999/02) y por el MCyT (Proyecto *Hermes*, 8/DG00141.226- 14247/200).

2 El buscador *GaIn*

GaIn es un buscador avanzado de textos en euskera para Internet/Intranet [1]. Su objetivo es tener indexados los textos en euskera de una Intranet o de toda la red y permitir una búsqueda basada en lemas. Esto se consigue básicamente con la introducción, en un buscador convencional, de un reconocedor de idioma y un lematizador robusto.

La necesidad de la lematización es evidente en lenguajes de gran flexión, sobre todo si es nominal, como el euskera. Si tenemos en cuenta que de un lema pueden generarse cientos de formas distintas, sin una herramienta de este tipo la búsqueda de todas las apariciones de cierto lema (que es lo que suele quererse encontrar) sería una labor tediosa.

Un método para obviar la lematización es el metacarácter *, pero no es suficiente debido a los cambios ortográficos que se producen al juntar lema y sufijo y además a la interferencia de formas no correspondientes a ese lema (sobre todo en lemas cortos). P. ej. si queremos buscar las flexiones de *egia* (verdad) tendríamos que usar *egi** (para cubrir las flexiones *egien, egietako, ...*) pero así también obtendremos todos los derivados del verbo hacer (*egin*) y de más lemas como *egitarau* (programa) *egitate* (hecho), *egitura* (estructura) y tantos otros.

Además una lematización estándar no es suficiente ya que en los documentos pueden aparecer tanto variantes dialectales como, sobre todo, referencias a nuevas personas, lugares, términos etc. que no están incluidos en el léxico. Por lo tanto el segundo paso ha sido conseguir un lematizador robusto e integrarlo en el buscador.

El proyecto sigue adelante y el objetivo futuro es no indexar todos los lemas sino solo aquellos términos, entidades, etc. mono o multipalabra que sean significativos en el texto.

2.1 La adaptación del programa

Los componentes básicos de un buscador (*search engine*) son tres: robot, indexador y buscador. En lugar de programarlo hemos recurrido al software disponible en el mercado (sobre todo al software libre), y una vez seleccionado uno de ellos lo hemos adaptado a nuestras necesidades.

Una vez estudiadas las características nos decidimos por el Swish-E de la Universidad de Berkeley, ya que se adaptaba perfectamente a las características requeridas:

- completitud (tiene los 3 componentes)
- código fuente disponible
- multiplataforma
- multiformato
- modularidad
- libre distribución

A continuacion se describen los componentes fundamentales: robot, indexador y buscador.

2.2 El robot

Es el módulo que va buscando textos por la red y seleccionando los que pueden ser interesantes. Si no lo dejamos salir de un dominio será para Intranet y si le ponemos dominio libre será para Internet. En este último caso la necesidad de disponer de un canal de un ancho de banda elevado es esencial, porque sino además de poder colapsar el trabajo sobre Internet de los usuarios la tarea se vuelve muy lenta. Este es un problema que hemos tenido y la alternativa que hemos tomado provisionalmente para la solución Internet es apoyarnos en un buscador Internet para conseguir los precandidatos a indexar.

Sin embargo en la solución Intranet el robot trabaja muy adecuadamente aunque ha sido necesario hacer unos pequeños retoques en orden a:

- distinguir varios tipos de documento, entre ellos los *frames*
- ofrecer la posibilidad de salir del dominio
- saltar los *proxies*

El robot adaptado estaba programado en *Perl* y se han utilizado las clases *HTTP* y *HTML-Parser*.

Además de los cambios generales mencionados le hemos añadido un identificador de idioma, que actúa como filtro para solo indexar textos en euskera, cuando en el documento no aparece la etiqueta *LANG*.

Como nuestra primera necesidad era distinguir si el texto estaba en euskera o no, hemos seguido las técnicas de las palabras más frecuentes y los trigramas [13] y el algoritmo ha sido muy simple y basado en umbrales de frecuencia en función de unos primeros resultados experimentales. El sistema se ha evaluado probándolo con textos obtenidos de Internet en función del idioma, y los resultados son muy buenos. En ningún caso un texto en otro idioma se identificó como en euskera y solamente en un caso de 30 se ha tomado por otro idioma el de un texto en euskera (página con nombres de grupos de música).

Un problema pendiente es el de las páginas con texto en varios idiomas (por ahora si tiene parte en euskera la indexamos completa).

2.3 El indexador

El indexador es el módulo encargado de generar unos índices para que la posterior búsqueda sea lo más eficiente posible.

El indexador de la herramienta escogida (Swish-E) está escrito en C y es sencillo y de fácil adaptación. Los índices se almacenan en formato texto, guardando por cada palabra un registro compuesto por tantos vectores como apariciones de la palabra, y en cada vector la información necesaria para identificar el documento y la posición de la palabra en el mismo. Para que los índices no sean demasiado grandes se eliminan del índice, por medio de una lista de parada o *stop-list*, las palabras funcionales de apariencia frecuente. Una de las labores realizadas ha sido generar una lista de este tipo para el euskera.

Sin embargo, el cambio más importante ha sido la indexación por lemas con vistas a cumplir el objetivo enunciado. Hemos integrado un lematizador [3][10] del que disponíamos previamente y conseguido, además de un buscador más avanzado, un índice más compacto.

Para conseguir que el lematizador sea robusto el proceso de análisis se hace incrementalmente en tres fases:

- lematización estándar
- lematización de variantes dialectales y errores frecuentes
- lematización sin léxico

Los resultados son totalmente satisfactorios ya que más del 99% de las palabras se lematizan correctamente.

En el futuro queremos aumentar la inteligencia del buscador indexando no la mayoría de los lemas (salvo los de la *stop-list*) sino que lematizando solo los lemas o sintagmas más significativos, siguiendo el trabajo que estamos realizando en extracción de terminología y entidades.

2.4 El buscador

Es el módulo que por medio de una interfaz captura las palabras clave en que el usuario basa su pregunta y consultando los índices genera una página con enlaces a las páginas que contienen la información correspondiente.

En el módulo de búsqueda se ha adaptado la interfaz y se ha incluido también el módulo de lematización en previsión de que se pregunte por la forma y no por el lema.

2.5 Evaluación

Aunque, debido a la dificultad de medir el *recall* en este tipo de sistemas, no hay medidas cuantitativas, el resultado obtenido es totalmente satisfactorio para un dominio Intranet según *Plazagunea*, la empresa que está comercializando esta herramienta.

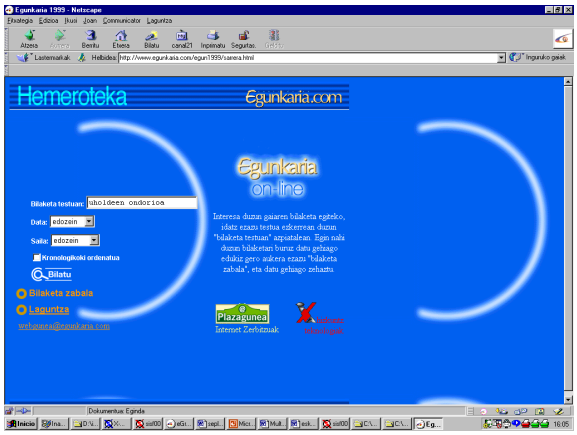

Fig.1: Página de búsqueda de la hemeroteca

Cualitativamente se ha detectado un incremento del *recall* con una disminución mínima de la precisión.

Esta herramienta se esta utilizando con éxito en varias aplicaciones, destacando las siguientes:

- el portal *Jalgi* www.jalgi.com que incorpora este buscador tanto a contenidos propios (tienda virtual, enciclopedia, etc.) como a otros indexados a partir de información recuperada por el robot.
- La hemeroteca del diario *Egunkaria* www.egunkaria.com/hemeroteka que no usa el robot pero si el resto de módulos.
- El portal dedicado a ciencia y tecnología www.zientzia.net

En las figuras 1 y 2 vemos un ejemplo de la búsqueda y la respuesta en este segundo caso.

Fig.2: Resultado de la búsqueda

Se ha buscado *uholdeen ondorioa* (la consecuencia de las inundaciones) y en las dos primeras opciones de la respuesta en el titular del artículo aparecen *...uholde baten ondorioz...* (...como consecuencia de una riada...) y *...uholdeen ondorioz* (...como consecuencia de las inundaciones).

La metodología y técnicas aplicadas pueden servir, sobre todo para idiomas con pocos recursos en Internet, para crear un buscador Internet/Intranet para un idioma específico siempre que se disponga de unos recursos lingüísticos básicos (identificador de idioma, lematizador).

3 Extracción de terminología

El trabajo que venimos realizando [2][23] en este campo está dando los primeros resultados.

Los campos de aplicación de la extracción (semi)automática son variados pero se pueden dividir en dos grandes grupos:

- trabajo terminológico propiamente dicho, como puede ser la construcción de listas o diccionarios de términos
- indexación de información, que esta estrechamente ligada con todo el área de recuperación de la información (*Information Retrieval*).

Respecto al primer campo cabe decir que la obtención (semi)automática de terminología es de gran interés en los campos de traducción técnica y en el mundo editorial. Además, en

áreas en donde el desarrollo terminológico es muy dinámico, como puede ser el mundo de la informática, sin herramientas de este tipo es casi imposible un trabajo terminológico efectivo.

El campo de la indexación de la información presenta un interés incluso superior, debido, en gran parte, a la gran cantidad de información disponible en línea pero que no va acompañada con herramientas de búsqueda/selección adecuadas.

En nuestro caso estamos desarrollando una herramienta de este tipo para el euskera, lengua de flexión rica y aglutinante, lo que conlleva una necesidad mayor de análisis morfosintáctico. Además al ser el proceso de unificación incipiente, investigaciones a nivel de terminología son escasas. Los corpus a tratar serán de dominio de especialidad y en nuestros experimentos hemos elegido dos áreas: administración pública e informática.

3.1 Estado del arte

Se detecta en los últimos años un creciente interés por este tema. Como ejemplo de ello podemos nombrar una serie de herramientas: *LEXTER* [5], *Termight* de AT&T [8], *TERMS* de IBM [15] y ACABIT [9].

La primera tarea a afrontar es la delimitación de las características de los términos. La mayoría de los trabajos se suelen limitar a buscar sintagmas nominales o algunas de las formas morfosintácticas más frecuentes de los mismos. El trabajo de Justeson & Katz [15] estudia para el inglés estas estructuras y llega a la conclusión anterior, aunque no está claro que esto sea cierto para otros idiomas.

Una vez estudiada la estructura de los términos objetivo de la búsqueda se suele proceder a construir la herramienta para extracción automática de los hipotéticos términos, ya que posteriormente habrá, en la mayoría de los casos, un postproceso manual para la selección definitiva. Para ello se suelen combinar técnicas de NLP (basadas en conocimiento lingüístico) con técnicas estadísticas más o menos sofisticadas.

3.1.1 Herramientas lingüísticas

Las técnicas lingüísticas se utilizan preferencialmente para una preselección de los posibles términos a tratar. Estos precandidatos suelen ser posteriormente sometidos a algún filtro de carácter cuantitativo para obtener la lista definitiva de candidatos. Para la primera preselección conviene tener el texto lo más analizado posible, siendo normalmente un requerimiento mínimo el etiquetado morfosintáctico. Estas herramientas conocidas como *taggers* han tenido un gran desarrollo durante los últimos años y están disponibles para un número importante de idiomas.

Unido al proceso de etiquetado morfosintáctico, está la lematización, que aunque en las lenguas de nuestro entorno (español, francés, ingles, ...) a veces no se le da gran importancia, en lenguas de flexión nominal rica, y más en las aglutinantes, es de gran importancia para conseguir unos buenos resultados. De todas formas los resultados no son sencillos de interpretar ya que p. ej. en euskera *sistema eragileak* o *sistema eragilearen* son flexiones en nominativo plural y genitivo del término *sistema eragile* (sistema operativo), sin embargo *sistemaren eragile* (accionador del sistema) no corresponde al mismo término.

Aunque debido a razones de eficiencia un análisis sintáctico profundo no suele aplicarse, un análisis sintáctico superficial si es necesario cuando los términos se quieren seleccionar en función de patrones sintácticos.

De todas formas en ciertos proyectos se evitan tanto el tratamiento morfológico como sintáctico y se recurre sencillamente a conjuntos de dos y tres palabras [21]. En algunos sistemas también se utiliza una lista de delimitadores de términos (artículos, preposiciones, etc.) para evitar el tratamiento sintáctico.

Además de para la preselección, el tratamiento lingüístico es importante también tanto para la normalización lingüística de los términos como para un tratamiento posterior, intentándose detectar, para ello, casos de flexión (esto se puede tener en cuenta en la preselección), variación gráfica o sintáctica del mismo término, sinonimia, hiperonimia, etc. [9][14][17].

Aunque muchas de las herramientas se quedan en la mera obtención de una lista, cada vez son más numerosas las propuestas que intentan procesar esta lista y el corpus original y obtener resultados más elaborados como pueden ser clasificar la lista de términos, obtener redes terminológicas, desambiguar términos, etc. En estos casos el tratamiento lingüístico se suele complicar y el tratamiento semántico adquiere una especial relevancia.

3.1.2 Técnicas estadísticas

En la mayoría de los proyectos se utilizan técnicas estadísticas para elegir entre los precandidatos que ya cumplen los requisitos lingüísticos.

Los métodos aplicados varían mucho según los proyectos: lo más simple es la exigencia de una frecuencia mínima [15], pero lo más habitual es la combinación de diferentes fórmulas estadísticas entre la que destaca por su utilización la llamada *mutual information* [7].

A partir de los resultados de esta fórmula se pueden hacer otros cálculos como sucede en [21], donde se comparan los resultados sobre el corpus de estudio con resultados generales obtenidos a partir de un gran corpus equilibrado.

También se pueden aplicar técnicas estadísticas para la normalización de términos, como sucede en el proyecto de la Universidad de Manchester [12] donde se propone el uso de una fórmula de nombre *C-value*, que es capaz de distinguir que *soft contact lenses, hard contact lenses,* y *contact lenses* son términos relacionados y que sin embargo *soft contact* no.

Como ya se ha apuntado el uso de herramientas semánticas deben mejorar los resultados, aunque, según el grupo de la Universidad de Manchester [17], usando características semánticas de WordNet no se mejora significativamente el reconocimiento de términos, aunque puede ser debido a razones técnicas y no de principio.

3.1.3 Resultados

Los resultados que se obtienen no son suficientes para una extracción totalmente automática pero si permiten ver la viabilidad de la realización de buenas herramientas de ayuda en la extracción de terminología. Los resultados de los diferentes sistemas estudiados parecen converger hacia una cobertura de aproximadamente el 95% con una precisión cercana al 50%.

De todas formas esta forma de evaluación de las herramientas, consistente en comparar los resultados con una lista previa extraída manualmente, es puesta en cuestión por varios autores [6][9], remarcándose que para una evaluación conveniente es importante tener en cuenta la aplicación para la que está destinada la herramienta.

3.2 Sistema propuesto

Basándonos en las herramientas mencionadas hemos venido desarrollando varios trabajos en paralelo de cara a la construcción del extractor de terminología. Las tres líneas de trabajo han sido:

- Modelización de la composición de los términos.
- Realización de un analizador sintáctico superficial que nos permite buscar los modelos de los términos detectados en la fase anterior.
- Realización del módulo estadístico que maneje tanto las estadísticas a nivel local como a nivel comparativo.

3.2.1 Estructura de los términos

La modelización de la composición de los términos [23] se ha llevado a cabo cuantificando la estructura morfosintáctica de los términos, a partir de 3 diccionarios técnicos. Disponemos de los primeros resultados y los estamos contrastando con su aparición en los corpus. Aunque los sintagmas nominales no cubren todo el espectro, ya que los verbos tienen una cierta frecuencia, son la mayoría de los mismos y el objetivo actual del sistema.

Tipo	Patrón	Frecuencia(%)
Sintagma Nominal	N_{nc} N	30.6
	A_{prep} N_{nc}? N	23.7
	N_{nc} N_{nc}? A_{pos}	17.3
	A_{prep} A_{prep} N	3.0
	N_{nc} A_{prep} N	2.3
Sintagma Verbal	N_{nc}? N_{abs} V	11.0
	ADV V	4.3

Tabla 1- Patrones morfosintácticos[2]

La expresión regular que aparece a continuación expresa la estructura de aproximadamente el 97% de los términos multipalabra:

$((N_{nc} \mid A_{prep})^{+} (N \mid A_{pos}^{+}) \mid ((ADV \mid (N_{nc}^{*} N_{abs})) V)$

En la tabla 1 se muestra la estructura del 92.2% de los términos estudiados y los que estamos buscando en una primera aproximación.

[2] N es nombre, A_{pos} adjetivo tras nombre, A_{prep} adjetivo antes de nombre, V verbo y ADV adverbio. N_{abs} nombre en caso absolutivo y N_{nc} nombre sin marca de caso.

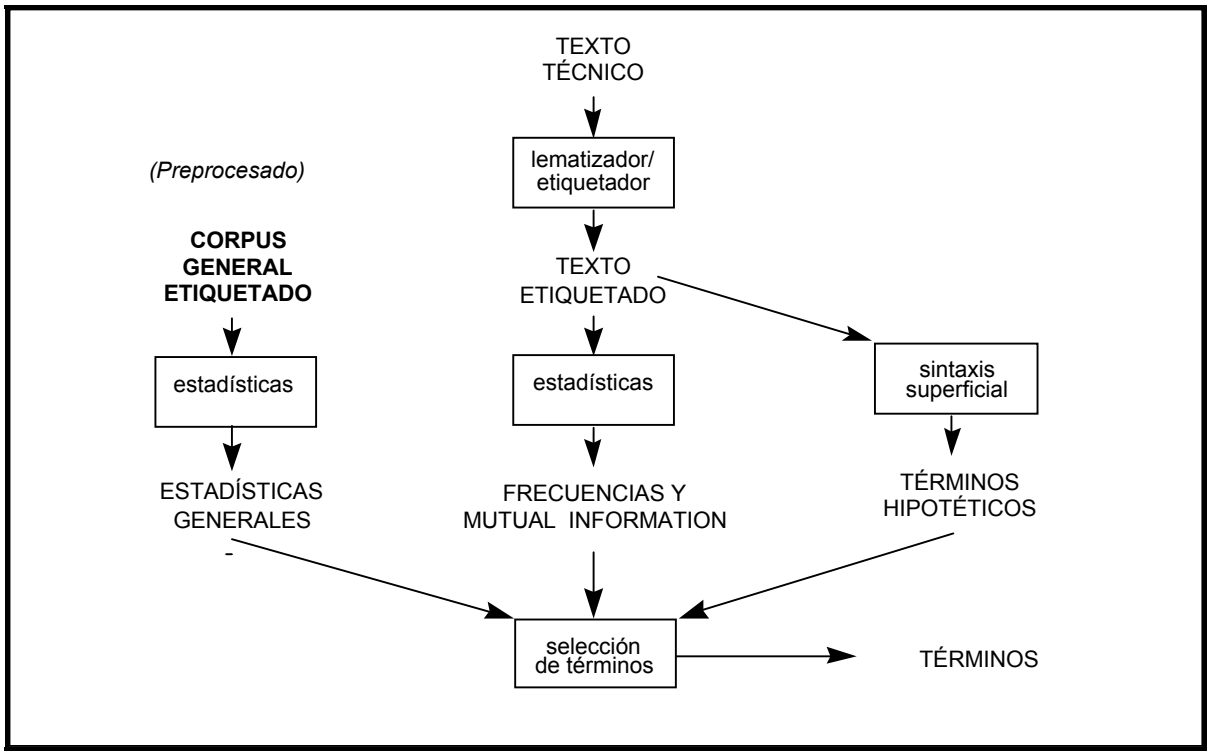

Fig. 3.- Arquitectura del extractor de terminología

Para el análisis sintáctico de superficie que localiza estas estructuras morfosintácticas se utiliza un modelo de estados finitos tras usar el lematizador/etiquetador descrito en el buscador de Internet.

Los términos seleccionados estarán lematizados, pero no en cada uno de los componentes sino que, debido a la estructura morfosintáctica del euskera, se debe mantener la flexión de todos los componentes salvo el último.

3.2.2 Arquitectura del sistema

La arquitectura del sistema se refleja en la figura 3, y por el momento prescinde del tratamiento semántico.

Según se puede ver en dicha figura, a partir de un gran corpus equilibrado se han obtenido las estadísticas de referencia de palabras y lemas que son utilizadas para relativizar las frecuencias que se obtienen de los textos procesados.

Teniendo realizado el preproceso anterior, cuando a partir de un texto (libro, artículo, manual, etc.) se quiere obtener la terminología, se procede al análisis morfológico y posterior lematización/etiquetado del texto, seguido del análisis sintáctico superficial. A partir de ello se obtienen los precandidatos que siguen los patrones sintácticos determinados anteriormente, y se pasa a su tratamiento estadístico.

La versión que estamos evaluando actualmente usa básicamente la fórmula de *mutual information*, pero teniendo en cuenta también la frecuencia de los componentes en el corpus general equilibrado. Además, debido a que la lematización solo afecta al último componente del término, se deben combinar en el denominador frecuencias de palabras y de lemas.

Para realizar la evaluación del sistema y experimentar con distintas medidas estadísticas se ha extraído manualmente la terminología de 20 textos de las áreas de administración pública e informática.

A falta de la evaluación cuantitativa los primeros resultados son prometedores.

4 Categorización de entidades

Hermes es el nombre de un proyecto en curso cuyo objetivo es el desarrollo de herramientas para la recuperación y extracción multilingüe de información en bases de datos documentales.

Dentro de los amplios objetivos del proyecto, que esta siendo desarrollado en colaboración con los grupos de la UPC y de la UNED, nuestro grupo está desarrollando para el euskera dos herramientas básicas de gran interés en el campo de recuperación y extracción de la información: un clasificador semántico y un extractor de entidades.

El primero de ellos asigna automáticamente a un texto una categoría y está en un estado avanzado de desarrollo. Una comunicación de esta conferencia lo describe ampliamente.

El extractor de entidades intenta etiquetar dentro del texto las entidades MUC (www.muc.saic.com): fechas, cantidades numéricas, porcentajes, nombres de persona, nombres de lugar, nombres de instituciones y empresas y, finalmente artefactos, herramientas, etc. Esta herramienta es un paso previo fundamental para la extracción de correferencias y la interrelación de documentos, tanto dentro del mismo idioma como entre los idiomas previstos en el proyecto (catalán, español e inglés, además del euskera)

Como la obtención de fechas y números estaba resuelta anteriormente, y la de artefactos está relacionada con la extracción de terminología, nos hemos centrado en la identificación y categorización de nombres propios: nombres de persona, lugar, instituciones y empresas.

La bibliografía sobre este tema es muy amplia [16][24] y últimamente destaca la creciente aplicación de técnicas de aprendizaje automático, pero para ello hace falta contar con un corpus de aprendizaje.

4.1 Sistema propuesto

El sistema que estamos construyendo se compone de tres elementos básicos:

- un identificador de entidades basado en la estructura morfosintáctica.
- un primer categorizador basado en un heurístico
- un segundo categorizador basado en aprendizaje automático.

El identificador utiliza el mismo etiquetador expuesto en el sistema de extracción automática [10] y posteriormente se le hace un análisis sintáctico superficial basado en estados finitos, que explotando las etiquetas morfosintácticas y la aparición o no de mayúsculas, limita los sintagmas nominales que pueden ser entidades.

Debido a que la declinación del término se refleja en su último componente, el tratamiento de la flexión es fundamental para intentar separar entidades independientes que aparecen en el texto una a continuación de otra.

El primer clasificador, que estamos construyendo actualmente, utiliza varias fuentes de información:

- etiquetas de los elementos del sintagma ya que se distingue entre nombres de lugar y otros nombres propios, aunque el proceso de desambiguación a veces falla.
- el caso de flexión del término: los nombres de lugar suelen aparecer en inesivo, los de persona en ergativo, ...
- listas de nombres de personas y lugares
- lista de nombres que pueden seguir al término (palabras en euskera correspondientes a *presidente*, *director*, *ciudad*, *rio*, *sociedad*, *S.L.*, ...)

Combinando estas informaciones un algoritmo debe asignar en primera instancia una categoría a la entidad.

El proceso anteriormente descrito va a ser aplicado a un corpus de unas 20.000 entidades, y el resultado será posteriormente revisado manualmente.

Con el corpus de aprendizaje obtenido en el paso anterior será la base del sistema de categorización que queremos construir con técnicas de aprendizaje automático, aunque si el primer clasificador da buenos resultados podrían combinarse ambos.

5 Referencias

[1] Aizpurua I., Alegria I., Ezeiza N. 2000. GaIn: un buscador Internet/Intranet avanzado para textos en euskera. *Actas del XVI Congreso de la SEPLN*.

[2] Alegria I., Ezeiza N., Oronoz M., Urizar R. 1999. Extracción Automática de Terminología a partir de Etiquetado y Lematización. *VI Simposio Internacional de Comunicación Social*. Santiago de Cuba, 1999.

[3] Alegria I., Aranzabe M., Ezeiza A., Ezeiza N, Urizar R. 2001. Using Finite State Technology in Natural Language Processing of Basque. *ICAI'2001*.

[4] Ananiadou S. 1994. A Methodology for Automatic Term Recognition. *Proc. of the Conference on Computational Linguistics (Coling-94)*, 1034-1038, Kyoto, Japan.

[5] Bourigault D. 1992. Surface grammatical analysis for the extraction of terminological noun phrases. *Proc. of the Conference on Computational Linguistics (Coling-92)*, 977-981, Nantes, France.

[6] Bourigault D., Habert B. 1998. Evaluation of Terminology Extractors: Principles and Experiments. *Proc. Of the First Conference on Language Resources & Evaluation (LREC'98),299-305.* Granada.

[7] Church K., Hanks P. 1990. Word Association Norms, Mutual Information and Lexicography. *Computational Linguistics* **16**:22-29.

[8] Church K., Dagan I. 1994. Termight: Identifying and Traslating Technical Terminology. *Proc. of the 4th Conference on Applied Language Processing*, Stuttgart, Germany.

[9] Daille B., Jacquemin C. 1998. Lexical Database and Information Access: A Fruitful Association?. *Proc. Of the First Conference on Language Resources & Evaluation (LREC'98),669-673.* Granada.

[10] Ezeiza N., Aduriz I., Alegria I., Arriola J.M., Urizar R. 1998. Combining Stochastic and Rule-Based Methods for Disambiguation in Agglutinative Languages. *COLING-ACL'9*8, Montreal (Canada). August 10-14, 1998.

[11] Ferrari K.T., Prince S. 1996. Création et Extension Automatiques de Dictionnaries Terminologiques Multilingues Specialisés a partir de Corpus Monolingues. *Proc. of the Conference on Natural Language Processing and Industrial Applications,* 79-86. Moncton, N.B. Canada.

[12] Frantzi K.T., Ananiadou S. 1996. Extracting Nested Collocations. *Proc. of the Conference on Computational Linguistics (Coling-96)*, 41-46.

[13] Grefenstette, G. 1995. *Comparing Two Language Identification Schemes*. Proceedings of 3rd International. *Conference on Statistical Analysis of Textual Data (JADT'95),* Rome, Italy, Dec. 1995

[14] Hamon T. & Nazarenko A. 1998. Using a general semantic information to help terminology structuration. *Proc. Of the First Conference on Language Resources & Evaluation (LREC'98),675-680.* Granada.

[15] Justeson J.S., Katz S.M. 1995. Technical Terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering* **1** (1): 9-27. Cambridge University Press.

[16] Martínez, R., Abaitua, J., Casillas, A.1998. Bitext Correspondences through Rich Mark-up. *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL'98)*, 812-818.

[17] Maynard D. & Ananiadou S. 1998. Term Sense Disambiguation using Domain-Specific Thesaurus. *Proc. Of the First Conference on Language Resources & Evaluation (LREC'98),681-685.* Granada.

[18] Oueslati R., Frath P., Rousselot F. 1996. Term Identification and Knowledge Extraction. *Proc. of the Natural Language Processing and Industrial Applications (NLP+IA)*, 191-196, Moncton, Canada.

[19] Pazienza M.T. ed. 1997. Information Extraction. *Lecture Notes in Artificial Intelligence 1299. Springer.*

[20] Strzalkowski, T. ed. 1999. *Natural Language Information Retrieval*. Kluver Academic Publishers.

[21] Su K., Wu M., Chang J. 1996. A Corpus-based Aproach to Automatic Compound Extraction. *Proc. of the Conference on Computational Linguistics (Coling-96)*, 243-247.

[22] Swish-E sunsite.berkeley.edu/SWISH-E

[23] Urizar R., Ezeiza N., Alegria I. 2000. Morphosyntactic structure of terms in Basque for automatic terminology extraction. *Proc. of the Euralex'2000*.

[24] Wakao, T., Gaizauskas, R., Wilks, Y. 1996. Evaluation of an Algorithm for the Recognition and Classification of Proper Names. *Proceedings of the 16th International Conference on Computational Linguistics (COLING'96),* 418-423.