

InTiMe

Plataforma de Integración de Recursos de PLN

InTiMe

Integration Platform of NLP Resources

José Manuel Gómez

Departamento de Lenguajes y Sistemas Informáticos
Universidad de Alicante
Carretera Sant Vicent del Raspeig s/n
03690 Sant Vicent del Raspeig (Alicante)
jmgomez@dlsi.ua.es

Resumen: La plataforma InTiMe (INtegration of Tools and corpora In the text-MEss project) es un ambicioso proyecto del Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Alicante. Nace con la idea de integrar, en una misma plataforma, gran parte de los recursos utilizados actualmente en Procesamiento del Lenguaje Natural (PLN). De esta forma, cualquier investigador incluido en la plataforma tendrá acceso inmediato, independientemente del sistema operativo que use o de su ubicación, a todas las herramientas y corpus integrados en el sistema. También será capaz de dar a conocer, si así lo desea, a toda la comunidad científica los nuevos recursos desarrollados en sus investigaciones. Evitando así que los investigadores tengan que desarrollar herramientas ya existentes, ahorrando tiempo y recursos y centrandos los esfuerzos en actividades más novedosas. Como veremos en el presente artículo, InTiMe agilizará la compartición del conocimiento y el uso de los recursos generados en PLN aumentando la productividad sin tener que cambiar la metodología de trabajo.

Palabras clave: Herramientas PLN, Integración recursos, InTiMe, PLN

Abstract: The InTiMe platform (INtegration of Tools and corpora In the text-MEss project) is an ambitious project of the Department of Languages and Computer Systems at the University of Alicante. Born with the idea of integrating, in a single platform, almost of the resources currently used in Natural Language Processing (NLP). Thus, any researcher included in the platform will have immediately access, regardless of the operating system he use or his location, all the tools and corpora integrated in the system. It will also be able to disclose, if he so wish, to the entire scientific community developed new resources in his investigations. Avoiding so that researchers need to develop tools that already exist, saving time and resources and focusing efforts on newer activities. As we will see in this article, InTiMe expedite the sharing of knowledge and the use of resources generated in PLN increasing productivity without changing the methodology of work.

Keywords: NLP tools, resource integration, InTiMe, NLP

1. *Introducción*

La investigación se basa, principalmente, en la idea de compartir conocimientos, herramientas y corpus que permitan a los investigadores aunar sus esfuerzos para lograr metas mayores. En áreas de investigación como el Procesamiento del Lenguaje Natural (PLN) esto adquiere una mayor importancia pues las soluciones a los problemas que se plantean hoy en día se basan en la combinación

de diversos recursos. Por lo tanto, un investigador debe ser capaz de conocer los recursos disponibles, saber utilizarlos correctamente y, a su vez, dar a conocer su propio trabajo.

Es muy común que cada recurso lo desarrolle diferentes personas que tienen intereses muy concretos y es inusual que piensen en una futura integración de su trabajo con el resto de recursos (Graça, Mamede, y Pereira, 2006). Es más, cuando se intenta integrar

todos los recursos y herramientas es cuando surgen los problemas de cómo se van a comunicar las aplicaciones entre sí y cómo van a procesar los distintos corpus.

El problema crece cuando en un grupo de investigación existe personal investigador temporal y no se ha planteado ninguna política de integración de estos recursos. Esta situación se resume en que los grupos de investigación disponen de gran cantidad de herramientas, aplicaciones y corpus (tanto propios como ajenos), en diferentes lenguajes de programación y sistemas operativos, con formatos de salida y entrada particulares, de localización muchas veces difícil puesto que dependen de la persona o personas que los han generado, y de reutilización compleja ya que requiere un esfuerzo adicional para integrarlos en otros desarrollos (Monteagudo y Cuetto, 2005).

Para resolver estos problemas, muchos grupos de investigación han decidido, a lo largo de su vida, aplicar alguna metodología de integración de recursos de PLN que diera a conocer a sus propios miembros los recursos disponibles. Algunos de estos proyectos de integración son parciales pues sólo tienen en cuenta algún aspecto concreto: o bien se centran en un tipo de recurso o en un dominio específico. Entre estos proyectos se pueden destacar el BancTrad (Badia et al., 2002), que proponen un formato estándar para la integración de corpus etiquetados paralelos junto con herramientas para acceder a él; el Emdros text database system (Petersen, 2004), el cual es un motor de base de datos para el análisis y la recuperación del texto analizado o anotado; el Natural Language Toolkit (Bird y Loper, 2004) que es un conjunto de bibliotecas y programas para el procesamiento simbólico y estadístico del lenguaje natural; y el Festival speech synthesis system (Taylor, Black, y Caley, 1998), el cual es un framework para construir sistemas de síntesis del habla.

También están los proyectos que únicamente definen protocolos o formatos para la comunicación entre distintos procesos de PLN, como el Annotation Graphs Toolkit (Maeda et al., 2001) que es una implementación del formalismo de Grafos Anotados de (Bird y Liberman, 2001), y el más influyente trabajo en éste área: la arquitectura Atlas (Bird et al., 2000), que generaliza el trabajo de (Bird y Liberman, 2001) para permitir el uso de señales multidimensionales. El pro-

blema de estos formatos es que no permiten la separación de la información en capas teniendo que cargar, en cada proceso, todas las anotaciones previas. También podemos encontrar el sistema EMU (Cassidy y Harrington, 2001) que está enfocado, específicamente, en tratamiento del habla.

Existen otros proyectos de integración que intentan abarcar tanto la especificación de los corpus y datos como de las herramientas en una única plataforma. El ejemplo más destacado lo podemos encontrar en el proyecto GATE (Cunningham, Wilks, y Gaizauskas, 1996; Bontcheva et al., 2004), que permite añadir módulos en Java de forma muy sencilla y rápida aunque requiere de más trabajo en caso de otros lenguajes de programación. GATE, además, define un formato de datos basado en la arquitectura TIPSER (Grishman, 1996) y en el Annotation Graphs Toolkit. Un sistema muy similar al GATE es el UIMA (Ferrucci y Lally, 2004), que está basado en el proyecto TEXTTRACT (Neff, Byrd, y Boguraev, 2004) de IBM. Pero, al igual que GATE, exige un cambio en la metodología de los grupos de investigación que pretendan usar la plataforma UIMA. Otros trabajos menos conocidos son los realizados por (Graça, Mamede, y Pereira, 2006) con una propuesta cliente/servidor que unifica las herramientas de PLN utilizando repositorios etiquetados con información multicapa que elimina la necesidad de cargar toda la información en cada proceso; y el trabajo de (Monteagudo y Cuetto, 2005), otra herramienta cliente/servidor que, a través de un middleware, unifica las herramientas y establece su propio formato de datos para comunicar los distintos procesos.

Aunque hay bastantes herramientas, protocolos y formatos que te permiten integrar herramientas y recursos, todos fallan en algún aspecto. Algunos de ellos son muy específicos y únicamente abarcan un conjunto de recursos de un área concreta del PLN (Badia et al., 2002; Petersen, 2004; Bird y Loper, 2004; Taylor, Black, y Caley, 1998). Otros no permiten muchos tipos de datos, por ejemplo, se centran en datos de texto o de habla únicamente (Maeda et al., 2001; Bird y Liberman, 2001; Bird et al., 2000; Cassidy y Harrington, 2001). También están los que obligan a trabajar en algún lenguaje informático concreto, un sistema operativo, plataforma, o que obligan a cambiar la metodología de trabajo

ajustándose a un API, protocolo o formato (Bontcheva et al., 2004; Bird y Loper, 2004; Monteagudo y Cueto, 2005; Grishman, 1996; Graça, Mamede, y Pereira, 2006). Ninguno tiene en cuenta los proyectos de terceros o, simplemente, es muy complicado la integración de éstos, como es el caso de GATE y UIMA. Esto lleva a eliminar de la integración la gran mayoría de herramientas y recursos desarrollados actualmente que no se ajustan a la metodología escogida. Es decir, olvidan la naturaleza heterogénea de las herramientas de PLN. Esto conlleva a un aumento de trabajo de los investigadores cuando quieren adaptar dichas herramientas. Pero, en definitiva, la principal razón por la que ninguno de estos proyectos de integración no se han implantando y generalizado entre la comunidad científica es que requiere, para su adopción, un esfuerzo extra por parte de los investigadores para adaptarse a nuevas metodología, protocolos, lenguajes, etc.

En este trabajo presentamos una plataforma de integración que respecta la metodologías de trabajo de los investigadores pero que, a su vez, permite integrar cualquier tipo de herramienta o corpus, independientemente del lenguaje de programación, sistema operativo o formatos de entrada y salida. La única condición que debe cumplir el recurso integrado es que no requiera una interfaz gráfica para su funcionamiento. Además, permite añadirle una capa más para aplicar cualquier protocolo o metodología de integración vistos anteriormente. También podremos integrar los recursos ya desarrollados o de terceros así como conocer la existencia y el funcionamiento de los recursos que ya estén integrados en el sistema desde cualquier parte del mundo y utilizando una única aplicación cliente. Por último, destacar que el uso de esta plataforma por parte de los investigadores sólo requerirá un esfuerzo muy pequeño o nulo.

2. Descripción de la plataforma InTiMe

InTiMe es una plataforma que permite conocer, acceder, usar y compartir herramientas y corpus. Tiene una arquitectura cliente/servidor y distribuida realizada con servicios web que permite, por una parte, integrar en los servidores cualquier recurso y, por otra, tener acceso a las herramientas que trabajan con esos corpus remotamente y eje-

cutarla como si dicha herramienta estuviera ejecutándose localmente. Al ser distribuida, un investigador puede subir un recurso a cualquier servidor y que éste sea conocido por el resto de usuarios. Además, gracias a las características del cliente InTiMe, se puede ejecutar una herramienta usando InTiMe exactamente igual a cómo se utilizaría la herramienta original, con las mismas entradas y salidas, y los mismos parámetros y ficheros de configuración. Esto permite que el investigador pueda seguir trabajando sin modificar un ápice su metodología de trabajo. El proyecto InTiMe ha sido desarrollado íntegramente en JAVA. Ésto permite que InTiMe pueda funcionar en cualquier sistema operativo con soporte para JAVA y que se pueda integrar las herramientas compiladas para dichas plataformas. Por ejemplo, podremos tener acceso a herramientas específicas de Microsoft Windows, GNU Linux o Mac OS siempre y cuando dispusiéramos un servidor InTiMe en cada una de esos sistemas operativos con sus respectivas herramientas.

2.1. Arquitectura del sistema

La arquitectura de la plataforma InTiMe puede variar y adaptarse a las necesidades del grupo de investigación, permitiendo que uno o varios clientes puedan acceder simultáneamente a uno o varios servidores de la plataforma. La arquitectura que se presenta en la figura 1 es tan sólo un ejemplo de una de ellas.

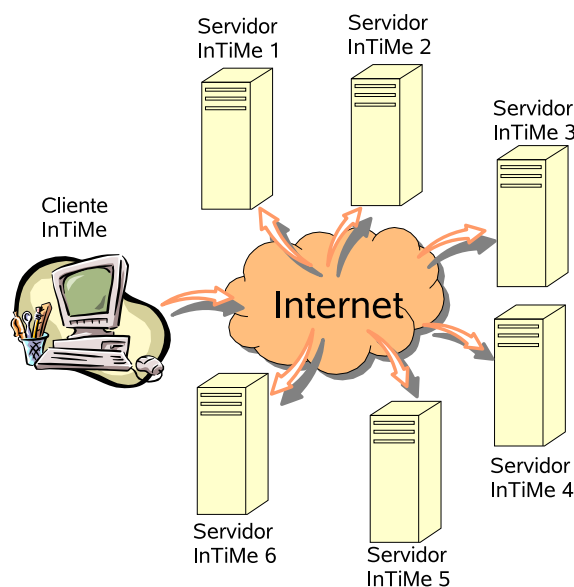


Figura 1: Arquitectura general de la plataforma InTiMe

2.2. Los servidores InTiMe

Cada servidor InTiMe almacenará un subconjunto de las herramientas y los corpus que se pretendan integrar en la plataforma pero, al mismo tiempo, conocerá qué recursos hay instalados en los otros servidores. Esto permitirá al cliente conectarse a cualquiera de estos servidores y acceder a todos los recursos. Por lo tanto, cuando se sube un recurso a un servidor, la información sobre dicho recurso, así como el servidor en que se encuentra, es esparcida entre el resto de servidores de la red. Lo mismo ocurre cuando un servidor es dado de alta. Un servidor InTiMe, a su vez, puede contener redes locales de servidores que contengan parte de los recursos que éste oferte, actuando de manera muy similar a un proxy. En la figura 2 se puede apreciar un ejemplo de esta arquitectura.

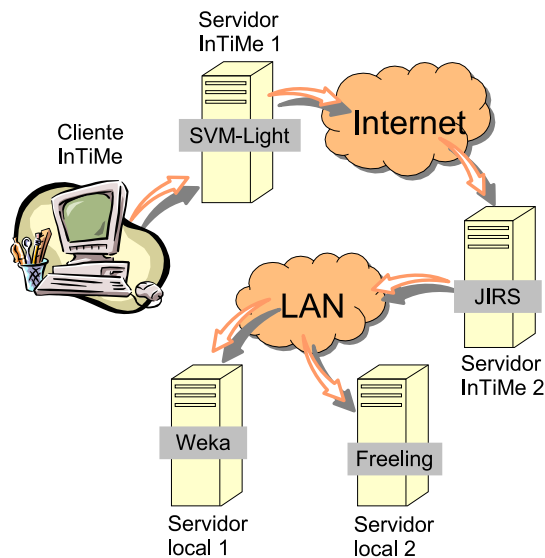


Figura 2: Servidor InTiMe funcionando como proxy

En este ejemplo hemos instalado cuatro servidores que tienen diversas herramientas. Dos de estos servidores están en una red local sin acceso directo a Internet, pero están subyugados al servidor 2. Con esta arquitectura, si el cliente quiere ejecutar el clasificador *SVM-Light*¹ únicamente debe pedir al servidor 1 que lo ejecute. Sin embargo, si lo que quiere ejecutar es el sistema de recuperación de pasajes *JIRS*² y no conoce en qué servidor se encuentra instalado, el cliente realizará la

petición a un servidor que conozca. Éste reenviará la petición al servidor correcto, el cual ejecutará la aplicación y devolverá los resultados de nuevo al cliente. Además, el cliente actualizará su información con la dirección del nuevo servidor para futuras peticiones. En el caso de que el cliente del ejemplo quiera utilizar las herramientas *Weka*³ o *Freeling*⁴, ambos en servidores InTiMe de una red local, él únicamente tendrá que pedirlo a un servidor conocido (en el ejemplo es el servidor 1), este servidor sabrá que dichas herramientas las tiene el servidor 2 y le lanzará la petición a dicho servidor. Aunque éste servidor no contiene esos recursos, sabe que se encuentran en dos servidores locales de los cuales él tiene acceso. De esta forma, la petición del cliente se redirigirá al servidor local que contiene la herramienta y el resultado de la ejecución se le devolverá al usuario. Después, la aplicación cliente, se actualizará añadiendo al servidor 2 como la fuente de las herramientas *Weka* y *Freeling* puesto que él no tiene acceso directo a los servidores locales 1 y 2.

Un servidor InTiMe puede integrar tantos recursos como se quiera o sea capaz de contener. Además, un mismo recurso puede estar en varios servidores para, en caso necesario, repartir la carga de trabajo. Es más, esta plataforma permite construir aplicaciones más potentes a partir de otras y que cada parte de esas aplicaciones se encuentren en un mismo servidor o en distintos, permitiendo que se ejecuten de forma local o remota, serializada o distribuida o, en definitiva, en cualquier combinación de éstas.

2.3. Los clientes InTiMe

Cada usuario tendrá a su disposición un cliente InTiMe que le permitirá listar, consultar, buscar y ejecutar las herramientas de cualquier servidor conectado a la red así como consultar en qué servidores están disponibles los corpus. La función del cliente InTiMe será recoger la entrada, analizar los parámetros de la aplicación, subir esta información más todos aquellos archivos que requiera la herramienta, obtener los resultados de las salidas estándar y de error de la aplicación y descargar todos aquellos archivos que se hayan generado por la ejecución de la herramienta en el servidor.

El cliente, finalmente, presentará por la

¹<http://svmlight.joachims.org/>

²<http://jirs.dsic.upv.es/>

³<http://www.cs.waikato.ac.nz/ml/weka/>

⁴<http://www.lsi.upc.es/~nlp/freeling/>

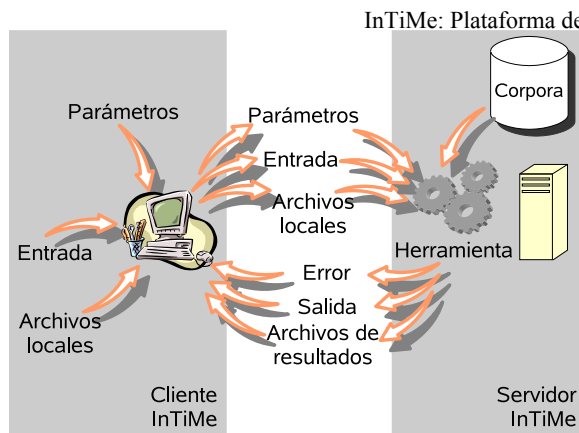


Figura 3: El cliente InTiMe

salida estándar y de error lo que devolvió la herramienta en ambas salidas cuando se ejecutó en el servidor. También descargará los archivos de resultados que se hayan podido generar durante la ejecución.

Para cada herramienta integrada en la plataforma se puede definir qué argumentos tiene, cuáles de ellos serán archivos que deberán subirse al servidor y cuáles de ellos serán archivos a descargar como resultado de la ejecución de la herramienta. Por ejemplo, imaginemos que queremos utilizar el sistema JIRS (Gómez et al., 2005) de recuperación de pasajes. Para realizar una búsqueda con este sistema y que nos devuelva una lista de pasajes relacionados tendríamos que lanzar siguiente comando:

```
jirs SearchPassages -language spanish
                    -collection clefqa
```

Con el cliente InTiMe únicamente tendríamos que escribir:

```
intime jirs SearchPassages -language spanish
                          -collection clefqa
```

Como vemos, lanzar el buscador de pasajes de JIRS con la plataforma InTiMe supondría una diferencia mínima, es decir, el investigador utilizaría JIRS prácticamente igual que si la tuviera instalada en su máquina, con los mismos argumentos y las mismas entradas y salidas.

La plataforma InTiMe permite, además, simplificar los argumentos de una aplicación definiendo argumentos con valores por defecto o *argumentos condicionados*, es decir, argumentos cuyo valor dependerá de los valores de otros argumentos. De esta forma, si

tuviéramos que utilizar una herramienta que cuando se utiliza un corpus en español tiene una serie de argumentos y cuando se utiliza con un corpus en inglés otros, con InTiMe podríamos definir un argumento llamado, por ejemplo, *lang* que nos permitiera encapsular el resto de parámetros. Así, si quisiéramos lanzar dicha herramienta con el corpus en español podríamos escribir:

```
intime tool -lang es
```

Ahorrándonos tener que escribir o conocer el resto de parámetros. Esto permite a un investigador simplificar el uso de las herramientas que use, definiendo los argumentos que mejor resultados den, para futuros usos o para el resto de la comunidad científica.

Pero la plataforma InTiMe no sólo permite ejecutar herramientas, sino que, además, nos permite conocer qué recursos ya han sido instalados en la red de servidores InTiMe. De esta forma, podemos listar o buscar todos los recursos mediante el siguiente comando:

```
intime list [-cat category_name]
            [-show all|cat|services]
            [-search key_words]
```

Los recursos integrados en la plataforma están clasificados por categorías. Así podemos listar los recursos categorizados o sin categorizar, visualizar sólo las categorías o los recursos de una categoría dada, o buscar cualquier recurso a partir de unas palabras claves.

Pero no sólo podemos listar los recursos sino que podemos obtener ayuda de uso para cada herramienta o corpus con el siguiente comando:

```
intime help tool_name [-language lang]
                    [-format format]
```

Esto nos devolvería la ayuda de una herramienta dada, preferiblemente en el lenguaje y formato elegidos.

3. Metodología de trabajo con la plataforma InTiMe

Nuestra metodología de trabajo no pretende modificar la forma de trabajar de los grupos de investigación a la hora de integrar las herramientas y, por ello, no hemos especificado ningún formato para el intercambio de datos. Así, si un usuario utiliza el analizador morfosintáctico del *freeling* o el clasificador *SVM-Light* a través de InTiMe, el

usuario deberá dar al cliente InTiMe las entradas, parámetros y ficheros de configuración que necesitan esos recursos y, a cambio, recibirá exactamente la misma salida que recibiría si los ejecutara de forma local. Pero no creemos que ésto sea una limitación por dos razones: (i) el usuario siempre puede adaptar las herramientas para que trabajen con algún protocolo de integración existente y seguir trabajando con InTiMe; y (ii) permite a los usuarios adaptar las entradas y salidas a sus necesidades y compartir estas adaptaciones al resto de la comunidad científica.

4. Conclusiones

En este trabajo hemos presentado una plataforma de integración para cualquier tipo de aplicación, herramienta o corpus con la única condición de que no requiera un entorno gráfico. Esto incluye a prácticamente la totalidad de las herramientas utilizadas actualmente en el PLN. Es más, permite integrar recursos tanto nuevos, como ya existentes o de terceros sin mucho esfuerzo. InTiMe tiene la ventaja, con respecto a otras plataformas de integración, de que no modifica la metodología de trabajo de los científicos adaptándose perfectamente a cualquiera de ellas. Esto hace de InTiMe una herramienta idónea para la compartición de conocimientos y recursos entre miembros de un mismo grupo de investigación o, en general, entre toda comunidad científica.

En estos momentos se está implantando paulatinamente en el marco del proyecto Text-Mess en el cual participan 5 universidades españolas. Debido a la heterogeneidad del proyecto y de los intereses de los participantes, nos hemos visto en la necesidad de evitar cualquier intento de implantación de un estándar para la comunicación de datos, limitándonos a la compartición de los recursos sin establecer ningún cambio en la metodología de cada grupo. Además, por experiencias anteriores, hemos podido comprobar que estos complejos proyectos de integración se suelen abandonar en poco tiempo quedándose en un cúmulo de buenas intenciones. Así, la plataforma InTiMe permite que todos los grupos aprovechen del trabajo del resto pero, a su vez, no exige ningún esfuerzo extra por parte de sus investigadores aportando innumerables ventajas como, por ejemplo, el acceso y uso inmediato de cualquier herramientas incluida en la plataforma. Permite listar,

buscar o pedir información sobre alguna herramienta o recurso incluido en el sistema. También nos permite integrar aplicaciones y herramientas que sólo funcionan en un sistema operativo concreto y crear aplicaciones a partir de otras aplicaciones o herramientas, concatenando procesos de forma serie, paralela, cliente/servidor, distribuida o de cualquier combinación de éstos.

A medida que InTiMe crezca en número de recursos, se convertirá, cada vez más, en una herramienta muy útil para los investigadores, permitiendo conocer instantáneamente qué herramientas están disponibles entre la comunidad científica y usarlas inmediatamente. Con esto ahorraremos tiempo y recursos, sabiendo exactamente qué recursos tenemos disponibles en todo momento evitando esfuerzo innecesario y centrando la labor del investigador en desarrollar recursos novedosos. Dando, así, una ventaja competitiva con respecto a otros grupos que no la usen. Además, aporta un plataforma de integración que los grupos de investigación pueden adoptar, incluso si ya disponen de alguna metodología.

5. Trabajos futuros

Aunque la herramienta está operativa en estos momentos y está siendo utilizada en varios proyectos propios, queda mucho trabajo por realizar. Uno de las tareas pendientes más importantes es evaluar el grado de satisfacción de los usuarios. Tampoco hemos desarrollado completamente el analizador de argumentos para que el cliente se adapte a los argumentos de las herramientas integradas, pero sí que permitimos que se puedan pasar dichos argumentos a la aplicación aunque sea cambiando la sintaxis original. Tampoco hemos terminado el proceso de sincronización entre servidores InTiMe y estamos trabajando en un administrador web que permita instalar y configurar el servidor de forma sencilla. Uno de los trabajos futuros más interesantes es la creación de un banco de pruebas que permita probar todas las aplicaciones y herramientas integradas en la plataforma a través de un portal web. Esto nos permitirá tener inmediatamente una demostración de nuestras herramientas cara a posibles inversores. Y esto lo realizará de forma automática en el momento en que se incluya dicho recurso en un servidor InTiMe.

También pensamos incluir en la platafor-

ma una comprobación automática y rutinaria de las herramientas para comprobar su correcto funcionamiento, evitando así que, debido a actualizaciones del sistema operativo o cambios en la propia herramienta, éstas queden inoperativas durante largos periodos de tiempo sin que nadie se percate de ello. La plataforma, automáticamente, enviará un correo electrónico a los responsables del recurso cuando detecte algún fallo en alguna de sus herramientas. Esta comprobación también se realizará a nivel de servidores InTiMe. Así, cada servidor InTiMe podrá ser controlado por otros servidores InTiMe para que, periódicamente, se revisen mutuamente.

Por último, y no menos importante, la plataforma InTiMe gestionará, mediante cifrado de claves públicas y privadas, el acceso a los recursos por parte de los usuarios, permitiendo, de esta manera, que los recursos de un mismo servidor puedan ser accedidos por ciertos grupos de usuarios únicamente. Con este mecanismo podremos reservar recursos que tengan restricciones de copyright a usuarios con licencias válidas o con aquellas instituciones en las que hayamos concertado un acuerdo o convenio para el uso de nuestras herramientas.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el proyecto QALL-ME (FP6-IST-033860), el cual es el 6th Framenwork Research Programme de la Comunidad Europea, y el proyecto TEXT-MESS (TIN-2006-15265-C06-01) del gobierno español.

Bibliografía

Badia, Toni, Gemma Boleda, Jenny Brumme, Carme Colominas, Mireia Garmendia, y Martí Quixal. 2002. Banctrad: un banco de corpus anotados con interfaz web. *Procesamiento del Lenguaje Natural*, 29:293–294.

Bird, Steven, David Day, John Garofolo, John Henderson, Christophe Laprun, y Mark Liberman. 2000. Atlas: A flexible and extensible architecture for linguistic annotation. En *Proceedings of the Second International Conference on Language Resources and Evaluation*, páginas 1699–1706, Paris, France. European Language Resources Association.

Bird, Steven y Mark Liberman. 2001. A formal framework for linguistic annotation. *Speech Communication*, 33:23–60.

Bird, Steven y Edward Loper. 2004. Nltk: The natural language toolkit. En *Proceedings 42nd Meeting of the Association for Computational Linguistics*, páginas 214–217, Barcelona, Spain. Association for Computational Linguistics.

Bontcheva, Kalina, Valentin Tablan, Diana Maynard, y Hamish Cunningham. 2004. Evolving gate to meet new challenges in language engineering. *Natural Language Engineering*, 10(3/4):349–373.

Cassidy, Steve y Jonathan Harrington. 2001. Multi-level annotation in the emu speech database management system. *Speech Communication*, 33:61–77.

Cunningham, Hamish, Yorick Wilks, y Robert J. Gaizauskas. 1996. Gate: a general architecture for text engineering. En *Proceedings of the 16th conference on Computational linguistics*, páginas 1057–1060, Morristown, NJ, USA. Association for Computational Linguistics.

Ferrucci, David y Adam Lally. 2004. Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348.

Gómez, José Manuel, Manuel Montes y Gómez, Emilio Sanchis, y Paolo Rosso. 2005. A passage retrieval system for multilingual question answering. En *Text, Speech and Dialogue: 8th International Conference, TSD 2005*, volumen 3658 de *Lecture Notes in Computer Science*, páginas 443–450, Karlovy Vary, Czech Republic. Springer Berlin / Heidelberg.

Graça, João, Nuno J. Mamede, y João D. Pereira. 2006. Nlp tools integration using a multi-layered repository. En *Proceedings of the Merging and Layering Linguistic Information Workshop*, páginas 24–32, Genoa, Italy.

Grishman, Ralph. 1996. Tipster text phase ii architecture design. En *Proceedings of a workshop on held at Vienna, Virginia*, páginas 249–305, Morristown, NJ, USA. Association for Computational Linguistics.

- Maeda, Kazuaki, Steven Bird, Xiaoyi Ma, y Haejoong Lee. 2001. The annotation graph toolkit: Software components for building linguistic annotation tools. En *Proceedings of the First International Conference on Human Language Technology Research*, Human Language Technology Conference, páginas 1–6, San Diego, USA. Association for Computational Linguistics.
- Monteagudo, Lorenza Moreno y Armando Suárez Cueto. 2005. Una propuesta de infraestructura para el procesamiento del lenguaje natural. *Procesamiento del Lenguaje Natural*, 35:303–310.
- Neff, Mary S., Roy J. Byrd, y Branimir K. Boguraev. 2004. The talent system: Text-tract architecture and data model. *Natural Language Engineering*, 10(3-4):307–326.
- Petersen, Ulrik. 2004. Emdros: a text database engine for analyzed or annotated text. En *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, International Conference On Computational Linguistics, páginas 1190–1193, Geneva, Switzerland. Association for Computational Linguistics.
- Taylor, Paul A., Alan W. Black, y Richard J. Caley. 1998. The architecture of the the festival speech synthesis system. En *Proceedings Third ESCA/COCOSDA Workshop on Speech Synthesis*, páginas 147–152, Sydney, Australia. International Speech Communication Association.