
Traductor de Pictogramas a Texto



Trabajo de Fin de Grado
Curso 2017–2018

Autor

Salvador González Álvarez
José María López Pulido

Director

Virginia Francisco Gilmartín
Susana Bautista Blasco

Trabajo de Fin de Grado en Ingeniería del Software
Facultad de Informática
Universidad Complutense de Madrid

Traductor de Pictogramas a Texto

Trabajo de Fin de Grado en Ingeniería del Software

Autor

Salvador González Álvarez
José María López Pulido

Director

Virginia Francisco Gilmartín
Susana Bautista Blasco

Convocatoria: *Febrero/Junio/Septiembre 2018*

Calificación: *Nota*

Trabajo de Fin de Grado en Ingeniería del Software

Facultad de Informática

Universidad Complutense de Madrid

20 de noviembre de 2018

Autorización de difusión

El abajo firmante, matriculado en el Máster en Ingeniería en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: “TITULO DEL TRABAJO”, realizado durante el curso académico CURSO bajo la dirección de DIRECTORES en el Departamento de XXXXXXXXXXXXXXXXXXXXXXXXXX, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

Nombre Del Alumno

20 de noviembre de 2018

Dedicatoria

Texto de la dedicatoria...

Agradecimientos

Texto de los agradecimientos

Resumen

Desde las primeras civilizaciones, el ser humano ha sentido la necesidad de comunicarse, en la actualidad esta necesidad es plasmada día a día en Internet: redes sociales, blogs... el hombre del siglo XXI, es más social y por ese motivo la necesidad de comunicarse es mas importante que nunca. No todas las personas tienen la misma capacidad de expresarse y o comunicarse, y por esta razón nacen los sistemas alternativos de comunicación.

El uso de pictogramas es uno de estos sistemas, en los últimos años hemos visto una evolución en la utilización de pictogramas en el ámbito educativo, en especial en niños con TEA (Trastorno del Espectro Autista) y trastornos del lenguaje en general como puedan ser Asperger, Síndrome de Down, parálisis cerebral, etc.

Por estos motivos hemos decidido crear un servicio web que permita al usuario traducir pictogramas a lenguaje natural. Una aplicación que facilite y ayude a la comunicación entre personas con diferentes capacidades comunicativas.

Palabras clave

Máximo 10 palabras clave separadas por comas

Abstract

Abstract in English.

Keywords

10 keywords max., separated by commas.

Índice

1. Introduction	1
1. Introducción	3
1.1. Motivación	3
1.1.1. Objetivos	4
1.1.2. Estructura del trabajo	4
2. Estado de la Cuestión	5
2.1. Sistemas Aumentativos y Alternativos de Comunicación	5
2.2. Pictogramas	7
2.2.1. Sistemas pictograficos más extendidos	8
2.3. Generación de lenguaje natural	12
2.4. Servicios Web	15
2.4.1. Ventajas de los servicios Web	17
2.4.2. Inconvenientes de los servicios Web	17
2.5. Sistema de traducción texto-picto	18
2.5.1. Araword	18
2.5.2. PictoTraductor	19
2.5.3. Pictar	19
3. Herramientas	21
3.1. SimpleNLG	21
3.1.1. Objetivos de la comunicación SimpleNLG	22
3.1.2. Generación de textos SimpleNLG	23
3.1.3. Generación de frases sencillas	23
3.1.4. Generación de frases complejas	24
3.2. Django	26
3.2.1. ¿Qué es Django?	26
3.2.2. ¿Por qué elegimos Django?	26

4. Conclusiones y Trabajo Futuro	29
4. Conclusions and Future Work	31
A. Título	33
B. Título	35
Bibliografía	37

Índice de figuras

2.1. Pictograma que representa un smarthwatch.	7
2.2. Pictograma que representa pintarse los labio.	7
2.3. Pictograma que representa una espada.	8
2.4. Figura Bliss(1).	9
2.5. Figura Bliss(2).	9
2.6. Sistema MinSpeak.	10
2.7. Clasificación de los pictogramas SPC.	10
2.8. Tablero ARASAAC para actividades escolares.	12
2.9. Traducción usando www.pictotraductor.com	12
2.10. Diferentes pictogramas con el mismo significado	13
2.11. Ejemplo de verbos conjugados	13
2.12. Ejemplo de diferente genero y numero	14
2.13. Fases de generación del lenguaje.	15
2.14. Estructura servicios Web.	16
2.15. Ejemplo Araword.	18
2.16. Frase en pictoTraductor	19
3.1. Sistema GLN en el que se basa simpleNLG.	22
3.2. Tabla ejemplo composición frase.	25

Índice de tablas

Chapter 1

Introduction

Introduction to the subject area.

Capítulo 1

Introducción

“Frase célebre dicha por alguien inteligente”

— Autor

1.1. Motivación

La comunicación es una parte fundamental del correcto desarrollo social y humano, pero no es igual de accesible para todos, para personas con diferentes diversidades funcionales existen muchos obstáculos para una comunicación correcta.

La aparición de diferentes vías de comunicación, como por ejemplo los pictogramas, ha supuesto un gran avance para lograr una comunicación efectiva, apoyados gracias a la aparición de diferentes recursos software, como traductores de lenguaje natural a pictogramas, han ayudado a estas personas a poder comunicarse en su día a día, pero la realidad es que la comunicación sigue estando muy limitada y en ocasiones se reduce a personas con el mismo tipo de diversidad funcional. Este es el caso de usuarios cuya vía de comunicación son los pictogramas, ya que el resto de usuarios desconoce su significado lo que produce que la comunicación quede notablemente reducida o en muchos casos sea completamente imposible. Esta es la razón, por la cual existe la necesidad de crear una herramienta que ayude a la comunicación de los usuarios de pictogramas con su entorno, una software que sea capaz de traducir pictogramas a lenguaje natural, esta herramienta software proporcionará una vía de comunicación derribando barreras y ayudando a la inclusión social de personas que por diversidad funcional están en riesgo de exclusión.

El objetivo de este TFG es el de crear un traductor de pictogramas a texto, que permita a usuarios cuya principal vía de comunicación son los pictogramas comunicarse de manera fluida, para ello crearemos un servicio web que permita la interpretación (no la traducción literal) de pictogramas

a texto. Los resultados de este trabajo ayudarán a las personas que presentan problemas de inclusión debido a la necesidad de usar pictogramas para comunicarse.

1.1.1. Objetivos

El objetivo principal de este trabajo, es la creación de un traductor de pictogramas a lenguaje natural, buscando la interpretación del mensaje y no una traducción literal. Un producto que sea accesible para todos, buscando la facilidad de uso a través de una interfaz cómoda, accesible e intuitiva, para de esta manera llegar al máximo público posible.

Buscamos también afianzar los conocimientos adquiridos durante el grado, y aplicar esos mismos conocimientos utilizando un diseño y visión, sobre el proceso y el producto orientada a la mantenibilidad del mismo, una gran experiencia de uso y durante la realización del producto adquirir nuevos conocimientos, gracias al desarrollo de un trabajo con impacto real. Para ello desarrollaremos un servicio web, que permita introducir un conjunto de pictogramas y devuelva al usuario, una interpretación en lenguaje natural del significado del mensaje introducido. Integraremos este servicio dentro de una aplicación web accesible para todo el mundo.

1.1.2. Estructura del trabajo

Capítulo 2

Estado de la Cuestión

RESUMEN: A lo largo de este capítulo trataremos los aspectos más importantes que han sido necesarios o forman parte de la base de este trabajo como los Sistemas Aumentativos y Alternativos de Comunicación, pictogramas y tecnologías que hemos utilizado para la realización del mismo.

2.1. Sistemas Aumentativos y Alternativos de Comunicación

La comunicación es una necesidad, y en casos de autismo o parálisis cerebral donde el lenguaje oral está gravemente limitado, la utilización de sistemas de comunicación de lenguaje no verbales, sustituyen o sirven de apoyo al lenguaje verbal.

Los Sistemas Aumentativos y Alternativos de Comunicación ¹ (SAAC) son distintas formas de expresar el lenguaje oral, que tienen como objetivo aumentar y/o compensar los problemas de comunicación de muchas personas con dificultades para conseguir una expresión verbal funcional.

El término comunicación aumentativa describe las formas en las que se comunica el individuo cuando el uso del lenguaje natural no es suficiente, por el contrario la comunicación alternativa hace referencia a diferentes métodos o vías de comunicación que sustituyen completamente el lenguaje oral, como por ejemplo los pictogramas o la lengua de señas.

¹<http://www.arasaac.org/aac.php>

Lo que se pretende con los SAAC es conseguir una comunicación funcional, adecuada y generalizable, que permita al individuo expresarse y alcanzar una mayor integración en su entorno social. Los SAAC ayudan a mejorar la calidad de vida de las personas con problemas para comunicarse, al otorgar al individuo una vía para mejorar su comunicación. Cabe destacar que los SAAC no son de uso exclusivo de las personas con alteraciones en la comunicación, sino que son utilizados por todos a diario, por ejemplo cuando empleamos gestos, señales de tráfico, etc.

Según la American SpeechLanguage-Hearing Association (A.S.H.A. 1981), Lloyd y Karlan (1984) y Carmen Basil(1988) los SAAC se pueden clasificar en dos grandes grupos:

1. SAAC sin ayuda: sistemas que carecen de soporte físico, por ejemplo el lenguaje de signos. Abarca desde el uso de mímica y gestos hasta el uso de signos manuales.
2. SAAC con ayuda: sistemas que necesitan un soporte físico independiente del emisor. Este tipo abarca desde sistemas sencillos basados en dibujos o imágenes hasta objetos reales en miniatura o los sistemas de comunicación escritos como el Braille. Están orientados a personas con problemas en el habla o con dificultades cognitivas o de aprendizaje.

Se ha demostrado que la utilización de los SAAC tiene múltiples ventajas (Álvarez, 2013) y son indispensables para aquellas personas que en mayor o menor medida, tienen impedido el uso del lenguaje por ejemplo por patologías como el autismo (Warrick, 2010), la deficiencia mental, la parálisis cerebral o personas que por enfermedad o lesión cerebral, están impedidos de manera temporal para expresarse de manera adecuada o correcta. Además de ser un recurso necesario para estas personas se ha demostrado que tienen múltiples beneficios:

1. Ayudan al desarrollo de la comunicación, proporcionando estrategias para la misma al usuario.
2. Posibilitan el desarrollo personal y social además de fomentar las relaciones interpersonales.
3. Evitan el aislamiento, fomentando habilidades afectivo-sociales.

Cabe destacar que la complejidad de los SAAC depende del nivel cognitivo del usuario (Dolores Abril Abadín, 2010), cada persona es diferente, y por ejemplo en el uso de imágenes, el nivel de abstracción y complejidad deben adecuarse a las necesidades de cada individuo. Dentro de los SAAC con ayuda se encuentran los pictogramas de los cuales hablaremos en la siguiente sección.



Figura 2.1: Pictograma que representa un smartwatch.



Figura 2.2: Pictograma que representa pintarse los labios.

2.2. Pictogramas

Se define pictograma como un signo icónico dibujado y no lingüístico, que representa un objeto real o un significado. Se engloban dentro de los SAACS con ayuda o asistido que requieren un apoyo físico, y se aplican a personas que no están alfabetizadas ya sea por causa de la edad o de alguna discapacidad. (Figura 2.14). La utilización de pictogramas para la escritura o pictografía se remonta al neolítico, donde se utilizaban piedras talladas para la representación de signos de cierta semejanza con el objeto que representan.

El origen de los sistemas pictográficos actuales se sitúa en 1981 con Mayer y Johnson y se caracteriza por una facilidad de interpretación.

Las características principales de los pictogramas son:

1. Guardan relación con aquello que representan, a lo que hace referencia el pictograma (Figura 2.14).
2. Son elementos gráficos que combinados representan el objeto tomado como referente (Figura 2.2).
3. La imagen debe de ser comprensible por el mayor número de personas, independientemente de la formación, idioma o discapacidad.
4. A la hora de construir un pictograma, este proceso de construcción

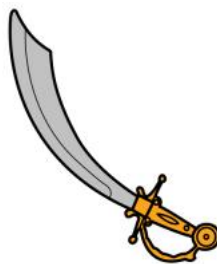


Figura 2.3: Pictograma que representa una espada.

debe de seguir unas reglas que le permitan mantener una coherencia visual, es decir la legibilidad de un pictograma debe de ser inmediata.

5. Deben de ser sencillos y representar únicamente los elementos más importantes, evitando posibles estímulos distractores o información irrelevante. (Figura 2.3)

A continuación analizaremos varios de los sistemas pictográficos más extendidos.

2.2.1. Sistemas pictograficos más extendidos

2.2.1.1. Bliss

Sistema Bliss desarrollado por Charles K. Bliss en 1949, esta formado por más de 2000 símbolos ², esta catalogado como SAAC con ayuda, tiene la dificultad de que el usuario debe de conocer el sistema ya que no cumple la relación con el objeto representado(Figura 2.4).

El sistema pictografico Bliss fue creado con la idea de crear un lenguaje global (McDonald, 1985), donde según el nivel de representación de los símbolos los pictogramas se agrupan en tres grandes grupos: pictográficos, ideográficos, arbitrarios.

1. Símbolos pictográficos: la forma del símbolo recuerda las palabras o conceptos que representan. Como podemos observar en símbolo que representa la casa en la figura 2.5
2. Sistemas ideográficos: expresan una idea, no describen el objeto, esto se puede observar en la figura aunque si guardan cierta relación con lo que representan 2.5 en el símbolo que representa el agua.
3. Sistemas arbitrarios: expresan un concepto abstracto y el símbolo no guarda ningún relación con el objeto representado.

²<https://www.uv.es/bellochc/logopedia/NRTLogo8.wiki?6>



Figura 2.4: Figura Bliss(1).



Figura 2.5: Figura Bliss(2).

2.2.1.2. Minspeak

Es un sistema pictográfico, creado por Bruce Baker en 1982 motivado por las carencias que observó en personas sin habla, este sistema no estipula cual es el significado de los símbolos, ya que este significado se establece entre el logopeda y el usuario, de ahí que en un sistema MinSpeak el pictograma que representa un arco iris puede significar felicidad pero en otro puede significar lluvia. (Martín Guerrero, 2018). Es una técnica que busca a través de la combinación de diferentes símbolos la representación de una palabra o frase. Permite desarrollar un lenguaje a través de la construcción de frases o secuencias que son fáciles de entender, pero tiene la desventaja de que el usuario necesita recordar la secuencia de símbolos necesarios para la generación de la respuesta correcta.

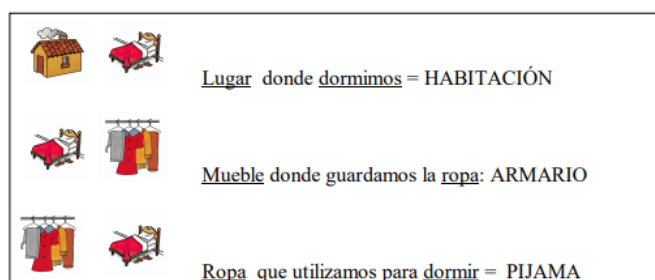


Figura 2.6: Sistema MinSpeak.



Figura 2.7: Clasificación de los pictogramas SPC.

2.2.1.3. SPC

El Sistema Pictografico de Comuniocación (SPC) ³ fue desarrollado por Roxana Mayer-Jhonson en 1981, busca a través de su diseño simple la similitud con aquello que representan. Los pictogramas están catalogados en seis grupos como podemos ver en la Figura 2.7, además utiliza un sistema de colores para dicha organización.

Los usuarios pueden comunicarse utilizando SPC señalando imágenes los pictogramas en tableros, o un libro, pueden coger los pictogramas pegarlos en un panel o utilizar tablets o pantallas a modo de tablero.

Esto supone un problema dado que la comunicación quedad limitada a los pictogramas seleccionables, además carece de sintaxis lo que simplifica la construcción de frases.

³<http://masquemayores.com/magazine/psicologia/tipos-de-sistemas-alternativos-y-aumentativos-de-la-comunicacion-sistemas-pictograficos-de-comunicacion/>

2.2.1.4. ARASAAC

ARASAAC (El Portal Aragonés de la Comunicación Aumentativa y Alternativa)⁴, es un sitio web que ofrece recursos gráficos ,material y herramienta para la comunicación de aquellas personas con algún tipo de discapacidad. El proyecto ARASAAC nace en el año 2007, financiado por el Departamento de Educación Cultura y Deporte del Gobierno de Aragón y coordinado por la Dirección General de Innovación, Equidad y Participación de dicho departamento, este proyecto nace con el objetivo de "de romper las barreras en la comunicación". ARASAAC cuenta con más de 33000 pictogramas únicos, repartidos entre pictogramas a color, en blanco y negros, con traducciones en más de quince idiomas.

Actualmente, ARASAAC se ha convertido en un sistema pictograma reconocido internacionalmente, entre otros factores porque sus recursos están bajo la licencia Creative Commons. Dentro de estos recursos cabe destacar sus bibliotecas de pictogramas, el acceso a diferentes herramientas software ⁵ como un generador de tableros, y software específico que utilizan recursos de ARASAAC como AraWord ⁶ y más herramientas que facilitan no solo la inclusión social de personas con discapacidades en el habla, si no también el desarrollo de herramientas para este propósito. Por todas las facilidades mencionadas anteriormente hemos decidido utilizar el sistema ARASAAC, a continuación pasaremos a explicar la formación de lenguaje a través del sistema pictográfico ARASAAC.

2.2.1.5. Lenguaje a través de pictogramas

Como se hemos explicado al principio los pictogramas forman parte de los Sistemas Alternativos de Comunicación y están englobados dentro del subgrupo sistemas con ayuda. Comúnmente la comunicación se realiza a través de tableros, de manera que el usuario señala uno a uno los pictogramas en dicho tablero hasta completar la frase, en la figura 2.8, podemos observar un tablero de ejemplo con relación a actividades desarrolladas en el colegio. Una palabra puede tener varios pictogramas como podemos observar en la figura 2.10 la palabra vamos tiene diferentes pictogramas, ya que la elección del pictograma debe de ser adecuado al usuario, teniendo en cuenta la complejidad del pictograma, la frase, además los verbos carecen de conjugación como podemos observar en la figura 2.11, no pasa lo mismo para la determinar genero y numero donde si existen diferentes variaciones del pictograma, como podemos observar en la figura 2.12. Las frases con pictogramas suelen tener una complejidad reducida, limitándose casi siempre a sujeto verbo y objeto, utilizando solo palabras significativas, por ejemplo en la figura 2.9,

⁴<http://www.arasaac.org>

⁵<http://www.arasaac.org/herramientas.php>

⁶http://www.arasaac.org/software.php?id_software=2

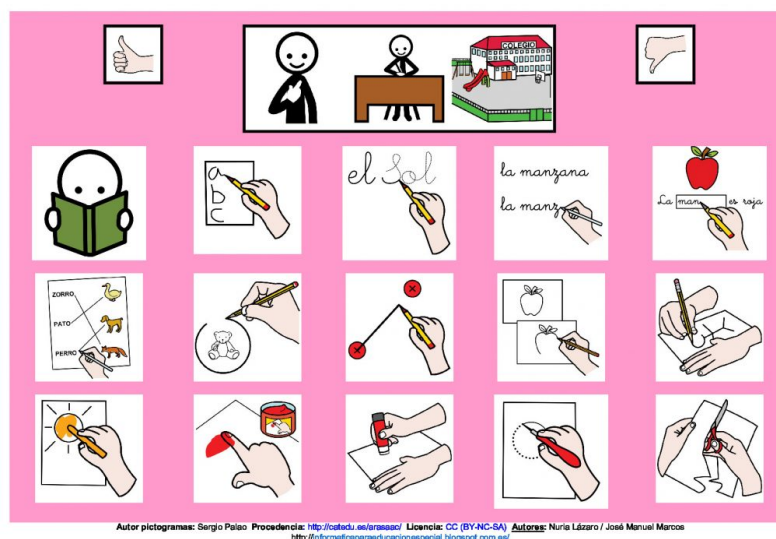


Figura 2.8: Tablero ARASAAC para actividades escolares.



Figura 2.9: Traducción usando www.pictotraductor.com

podemos observar que la preposición *a* y el determinante *la*, no tienen un pictograma como tal si no una imagen con la palabra ⁷.

2.3. Generación de lenguaje natural

La generación de lenguaje Natural (NGL Natural Language Generation) es la utilización de técnicas de inteligencia artificial para crear programas informáticos que generan lenguaje natural ya sea hablado o escrito ⁸, la NGL se engloba dentro de la lingüística computacional aunque engloba muchas otras áreas de estudio fuera de la informática como la lingüística o la psicología. Se busca que el programa generador se comunique igual que

⁷www.pictotraductor.com

⁸<https://searchdatacenter.techtarget.com/es/definicion/Generacion-de-lenguaje-natural-o-NLG>



Figura 2.10: Diferentes pictogramas con el mismo significado

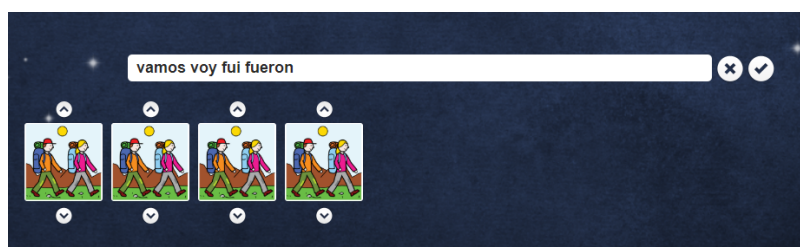


Figura 2.11: Ejemplo de verbos conjugados

si de una persona se tratare.(Robert Dale, 2000) (Vicente et al., 2015)

Para que la generación del lenguaje sea efectiva la inteligencia artificial ademas necesita poder, procesar el lenguaje natural (Natural Language Processing NLP) es decir poder entender y analizar el significado de aquello que se dice y también necesita entender el lenguaje natural es decir comprender el propósito de aquello que se le comunica.

Falsamente podemos pensar que lo más importante de la generación del lenguaje es la generación de un texto gramaticalmente correcto, pero lo más importante de la comunicación es que el texto generado explique aquello que se desea transmitir, por eso podemos concluir que el proceso de generación natural se podría resumir en la construcción de un mensaje en lenguaje natural que transmite un mensaje de manera clara.

Existe dos maneras de que un programa informático se comunice con un usuario, la primera forma es a través de mensajes estáticos definidos en el código, este acercamiento si bien es valido, es un sistema cerrado y no flexible en el cual no cabe lugar a la interpretación del lenguaje, la segunda vía es generación de lenguaje basada en conocimiento o *deep generation* donde es el propio sistema el encargado de generar el lenguaje a través de su conocimiento basándose en el conocimiento lingüístico del que dispone (Carlos García Ibáñez, 2004) .



Figura 2.12: Ejemplo de diferente genero y numero

Actualmente pese a que la rama de la generación de lenguaje natural⁹ es una de las ramas más antiguas de la inteligencia artificial, su estado actual es aun muy prematuro.

Los sistemas generadores de lenguaje natural se pueden clasificar siguiendo diferentes criterios para la siguiente organización se han tenido en cuenta dos factores clave como son la entrada del sistema.

1. Datos a texto (o D2T), este tipo puede ser muy variado dado que parten de datos que pueden desde numéricos como pueden ser aquellos dato provenientes de sensores, hasta corpus de datos como podría ser la respuesta de una base de datos,
2. Texto a texto, este tipo puede englobar desde aplicaciones que tienen como entrada desde textos hasta oraciones, ejemplos de este tipo pueden ser aplicaciones de resúmenes como por ejemplo "<https://resoomer.com/es/>".

Durante la generación del lenguaje se puede abstraer el proceso y encapsularlo en diferentes fases que describiremos brevemente a continuación (Vicente et al., 2015):

1. Macro planificación es la primera etapa donde se debe seleccionar el contenido, esto puede definirse como un subproceso dentro del primer proceso por el cual el sistema define aquello le es útil. Después de esto el sistema debe estructurar el contenido, para que el texto generado sea coherente es preciso estructurar el contenido del texto en el orden correcto.
2. Micro planificación , partiendo de los resultados del proceso anterior se determinan que partes formaran el texto resultado, agregando las estructuras que formaran parte de este, el léxico que se utilizara para expresar los conceptos y los hechos recogidos en la primera etapa y la generación de expresiones que aparecerán en el texto.

⁹ <https://www.beeva.com/beeva-view/tecnologia/introduccion-a-la-generacion-del-lenguaje-natural>

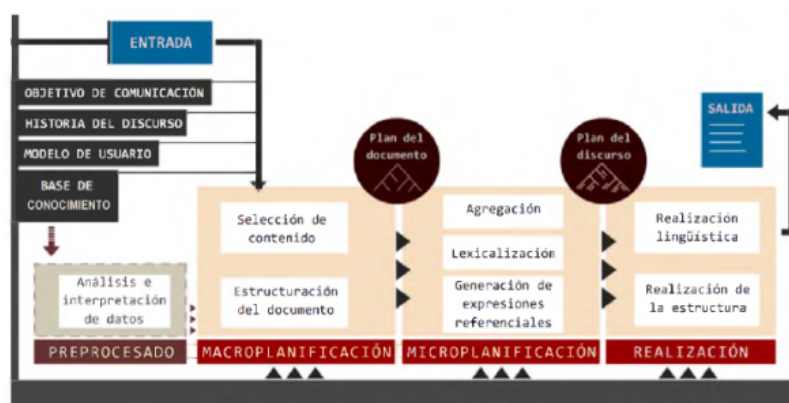


Figura 2.13: Fases de generación del lenguaje.

3. Realización, la generación o realización del texto se puede dividir en dos partes: realización lingüística, en la que se determina la representación en palabras del texto de salida, y la realización de la estructura, en la que se genera el texto en el formato correcto para su visualización en un medio determinado, como puede ser HTML, o un documento LaTeX.

2.4. Servicios Web

El WC3 (World Wide Web Consortium) define un servicio Web como ¹⁰:

Un sistema software diseñado para soportar la interacción máquina-a-máquina, a través de una red, de forma interoperable. Cuenta con una interfaz descrita en un formato procesable por un equipo informático (específicamente en WSDL), a través de la que es posible interactuar con el mismo mediante el intercambio de mensajes SOAP, típicamente transmitidos usando serialización XML sobre HTTP conjuntamente con otros estándares web.

Un servicio web es un componente accesible mediante peticiones Web estándar tales como GET o POST. Normalmente se denomina servicio web a una colección de procedimientos accesibles desde Internet. La gran ventaja de los servicios web es que al acceder desde el navegador a un servicio web este nos devuelve una página web con un formato estándar como puede ser XML.

Los servicios Web fueron creados para solucionar el problema de la interoperabilidad entre las aplicaciones. Al principio de los 90, apareció el gran problema de integrar aplicaciones diferentes. Una aplicación podía ha-

¹⁰<http://www.jtech.ua.es/j2ee/publico/servc-web-2012-13/sesion01-apuntes.html>

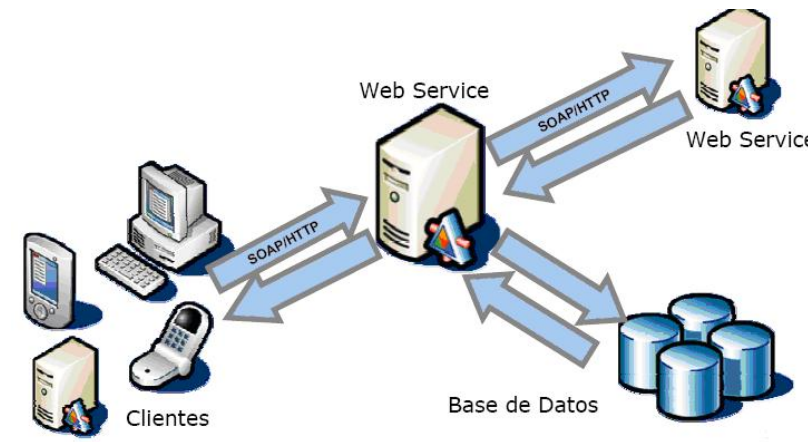


Figura 2.14: Estructura servicios Web.

ber sido desarrollada en C++ o Java, y ejecutarse bajo Unix, un PC, o un computador mainframe. No había una forma fácil de intercomunicar dichas aplicaciones. Fué el desarrollo de XML el que hizo posible compartir datos entre aplicaciones con diferentes plataformas hardware a través de la red, o incluso a través de Internet.

A nivel técnico, los servicios Web pueden implementarse de varias formas. En este sentido se pueden distinguir dos tipos.

- Servicios Web SOAP. Estos servicios utilizan el lenguaje XML para poder comunicarse utilizando el protocolo SOAP (Simple Object Access Protocol), este lenguaje es usado tanto para definir la arquitectura como el formato de los mensajes. Estos tipos de servicios deben contener un lenguaje legible para la máquina con la descripción de operaciones ofrecidas por el servicio escrito en WSDL (Web Services Description Language), el cuál está basado también en XML con el fin de definir interfaces sintácticamente.
- Servicios Web Restfull (Representational State Transfer Web Services). Los servicios Restfull son más adecuados para integraciones básicas ad-hoc. Estos servicios suelen mezclarse mejor con HTTP que los basados en SOAP, ya que no necesitan mensajes XML ni definiciones WSDL. Los servicios Web Restfull utilizan muy conocidos como SML, URI, MIME, y sobre todo HTML. Además posee una infraestructura la cuál podríamos llamar "ligera", ya que consiguen utilizar herramientas mínimas para formar estos servicios. Por ello, el desarrollo de este tipo de servicios son baratos y su adaptación es poco costosa.

En una arquitectura de Servicios Web se pueden distinguir tres partes

fundamentales para el establecimiento de la comunicación ¹¹:

- Proveedor de servicios web. Es el encargado de enviar al **Publicador del servicio** un fichero WSDL con la información básica de dicho servicio.
- Publicador del servicio. Una vez la información llega al publicador este descubre quien es el **Proveedor** (protocolo WSDL), y se pone en contacto con él (protocolo SOAP).
- Proveedor del servicio. Acepta la petición del servicio y envía los datos estructurados en formato XML usando SOAP.

2.4.1. Ventajas de los servicios Web

Las tres principales ventajas para usar estos servicios son ¹²:

- Ayudan a mejorar la conexión y operabilidad entre dos aplicaciones software sin importar sus propiedades o sino comparten plataforma de instalación.
- Favorecen los protocolos y estándares basados en texto, ya que estos facilitan acceder y comprender su contenido.
- Permiten y facilitan que software y servicios de diferentes compañías se puedan combinar formando un servicio funcional.

2.4.2. Inconvenientes de los servicios Web

Las tres principales desventajas de los servicios serían ¹³:

- Las transacciones que llevan consigo estos servicios están mucho menos desarrolladas que otros estándares abiertos.
- Su rendimiento es bajo comparado con otros modelos de computación distribuida, debido sobre todo a adoptar un formato basado en texto.
- Al estar muy ligados con HTTP, las medidas de seguridad que tratan de bloquear la comunicación entre programas se pueden burlar fácilmente.

¹¹<http://www.jtech.ua.es/j2ee/publico/servc-web-2012-13/sesion01-apuntes.html>

¹²<https://sites.google.com/site/preyctodetics/home/servicios-web>

¹³<http://fabioalfarocc.blogspot.com/>

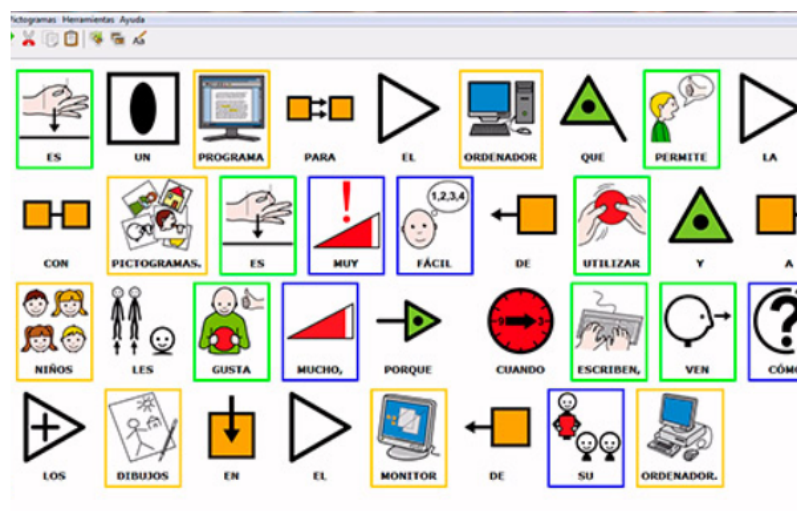


Figura 2.15: Ejemplo Araword.

2.5. Sistema de traducción texto-picto

Nos centraremos en los sistemas de traducción texto a picto, ya que de los sistemas contrarios, de picto a texto no hemos logrado encontrar ninguna información. Por lo que entendemos que actualmente no se dispone de una tecnología capaz de procesar la realización de frases sintácticas partiendo de pictogramas.

A continuación trataremos los principales sistemas de este tipo de traducción (Agustín Hernández-Gil Alonso, 2013):

2.5.1. Araword

Araword es una aplicación gratuita realizada por Arasacc (Portal Aragonés de la Comunicación Aumentativa y Alternativa), que permite la escritura simultánea de texto a pictograma ¹⁴.

Esta herramienta tiene como objetivo principal crear un procesador de textos, para facilitar la elaboración de materiales y adaptación de los mismos a aquellas personas con dificultad para comunicarse con otras. Se podría parecer a un word, el cuál presenta un documento en blanco donde se van escribiendo frases y traduciendo a pictogramas (Figura 2.14).

¹⁴<http://www.arasaac.org/software.php>



Figura 2.16: Frase en pictoTraductor

2.5.2. PictoTraductor

PictoTraductor es una herramienta pensada y desarrollada para facilitar la comunicación con personas que poseen dificultades para expresarse de forma oral y que se comunican mejor mediante el uso de imágenes. Principalmente está pensado y dirigido para padres y profesionales, y así poder comunicarse de una manera rápida y fácil.

La interfaz que presenta la página web consta de una barra de búsqueda en la que se introduce el texto que se quiere traducir a imagen. A medida que se va escribiendo una palabra se va mostrando el pictograma que representa esa frase.

En cuanto a la funcionalidad esta herramienta es una de las más desarrolladas, destacando la posibilidad de crear para usuario registrarse en la aplicación, pudiendo de esta manera personalizar la aplicación como por ejemplo: guardar pictogramas favoritos del usuario, fotografías o imágenes favoritas y compartir algunas de sus traducciones en las redes sociales. Pero, hay algunas funcionalidades de esta herramienta que todavía se podrían mejorar, como son: La conjunción de tiempos verbales y la búsqueda de algunas palabras las cuales no encuentran una traducción precisa (Figura 2.14).

2.5.3. Pictar

Pictar (Martín Guerrero, 2018) es una aplicación web en la que se puede diferenciar dos funcionalidades diferentes para ayudarnos en la formación de frases mediante pictogramas.

- Un primer apartado funcional en el que la aplicación se parece mucho a las mencionadas anteriormente, en la que el usuario introduce el texto que desea traducir y al pulsar el botón de buscar se le genera los pictogramas adecuados para la frase introducida.
- El segundo apartado funcional es el que hace más diferente a esta

herramienta de las otras, esta parte se compone por un buscador de pictogramas, donde el usuario introduce una palabra para encontrar su traducción en imagen. Y una segunda parte compuesta por un tablero en el cual se van poniendo los pictogramas encontrados en el buscador mencionado anteriormente. Con el fin de conseguir mediante la sucesión de pictogramas traducirlo a lenguaje de texto.

Capítulo 3

Herramientas

RESUMEN: A lo largo de este capítulo analizaremos las diferentes herramientas utilizadas para el desarrollo de este TFG.

3.1. SimpleNLG

La herramienta SimpleNLG es una biblioteca Java que proporciona el control directo sobre la construcción y formación de frases. Define un conjunto léxico cubriendo la mayoría de las categorías gramaticales y sus formas de combinarlas (Jozef Trzpis, 2015).

Para la construcción de una estructura sintáctica se ha de tener en cuenta los siguientes aspectos:

- Inicializar los componentes necesarios y sus elementos léxicos adecuados.
- Utilizar las propiedades de la API.
- Combinar los componentes con las propiedades de la API para fabricar estructuras sintácticas más complejas.
- Pasar la estructura realizada aplicando las inflexiones necesarias, y los ordena, función de las características, para después devolverlo como un texto ya formado.

Una característica muy importante es la robustez de la herramienta, formas sintácticas no completas o mal formadas no darán fallo, aunque el resultado que obtenga no será muy correcto.

Otra propiedad fue conseguir una separación entre la forma de constituir

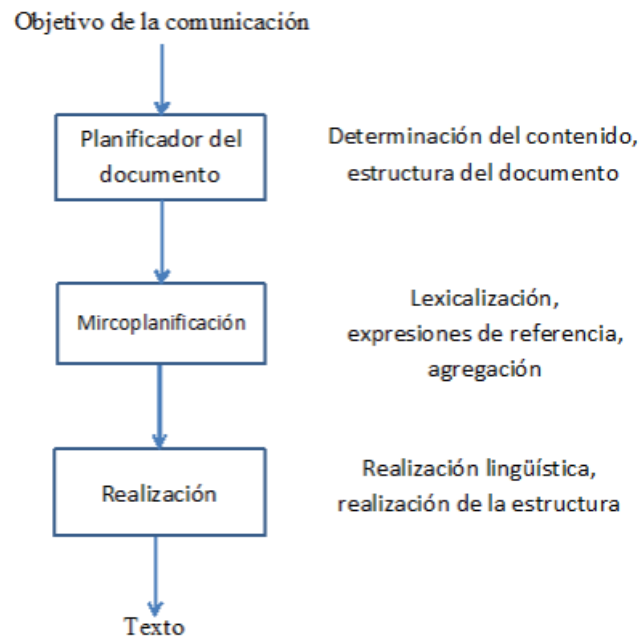


Figura 3.1: Sistema GLN en el que se basa simpleNLG.

operaciones morfológicas y las operaciones sintácticas. La biblioteca léxica integrada ofrece un generador con amplia cobertura morfológica y diferente a los componentes sintácticos. Esto lo hace muy eficaz para operaciones que no requieran de una gran dificultad sintáctica.

3.1.1. Objetivos de la comunicación SimpleNLG

La herramienta NLG tiene como fin producir un texto comprensible para el usuario. La mayoría de estos sistemas siguen un proceso para la formación de frases similar, compuesto por tres componentes conectadas entre sí. Estas tres fases son (Figura 2.14):

Planificador del documento

- Determinación del contenido: Elige que tipo de información saldrá como resultado, dependiendo del tipo de de público objetivo, la información que se le propondrá como entrada y de otras restricciones posibles.
- Estructura del documento: Toma la decisión de como deben agruparse los trozos que se han introducido en la entrada, y el orden en el que saldrán.

Microplanificación

- Lexicalización: Deciden las palabras específicas para expresar el contenido. En este apartado se encargan de elegir los nombres, verbos, adjetivos... que aparecerán en el texto.
- Expresiones de referencia: Decide las expresiones que se deben utilizar para hacer referencia a determinadas palabras, es decir, existen algunas expresiones que pueden ser denominadas de diferentes maneras, por ejemplo:
 - 11/2018
 - Noviembre
 - Noviembre del año 2018.
- Agregación: Decide cómo las estructuras obtenidas del planificador del documento deben ser clasificadas dentro de un párrafo u oración.

Realización

- lingüística: Utiliza las reglas gramaticales para convertir las representaciones abstractas en un texto real.
- estructural: Convierte frases y párrafos en elementos marcados cuando se mostrará el texto.

3.1.2. Generación de textos SimpleNLG

Como hemos mencionado anteriormente SimpleNLG es una librería de java por tanto es un fichero ".jar" que tiene que ser importado dentro del proyecto donde se va a utilizar.

Una vez importado dentro del proyecto se puede empezar a escribir código de la aplicación utilizando la API que proporciona SimpleNLG. Ofrece diferentes opciones de crear frases desde las más simples hasta las más elaboradas.

3.1.3. Generación de frases sencillas

La forma más sencilla en la que la herramienta SimpleNLG nos muestra un resultado es una cadena de salida igual que la cadena que le entra de entrada. Por ejemplo ¹: El código SimpleNLG para conseguirlo será el siguiente:

- NLGElement s1 = nlgFactory.createSentence("mi perro es feliz");

Ahora necesitamos usar el "Realizador" para generar la salida de la cadena:

¹<https://github.com/citiususc/SimpleNLG-ES/wiki/Secci%C3%B3n-III-E28093-Primeros-pasos>

- `String output = realiser.realiseSentence(s1);`
- `System.out.println(output);`

El resultado es:

- mi perro es feliz

3.1.4. Generación de frases complejas

En el apartado anterior hemos visto la manera de fabricar frases sencillas en la que es necesaria una mayor carga de trabajo por parte del usuario. Pero se puede conseguir que sea la propia librería quien se ocupe de ello. SimpleNLG es una herramienta muy flexible, y por esta razón existen diferentes maneras para generar la misma frase. Además, también es capaz de entender lo que es una frase utilizando la clase llamada `SPhraseSpec`. Ésta es accesible a través de `NLGFactory`, utilizando el método `createClause`. Los conceptos de 'frase verbal', 'frase nominal', 'frase preposicional', 'frase adjetiva' y 'frase adverbial' también se pueden acceder a través de `NLGFactory`, usando `createVerbPhrase`, `createNounPhrase`, `createPrepositionPhrase`, etc ².

Para construir las oraciones utilizando SimpleNLG, se podrían seguir estos pasos:

- Crear una instancia de `NLGFactory`.
- Crear la oración utilizando el método `createClause` de `NLGFactory`.
- Crear un verbo, sujeto y complemento directo usando los métodos `createVerbPhrase` y `createNounPhrase` de `NLGFactory`.
- Si se necesita, se pueden crear frases preposicionales, frases adjetivas y adverbios utilizando los métodos `createPrepositionPhrase`, `createAdjectivePhrase` y `createAdverbPhrase` de `NLGFactory`.
- Indicar el papel que las partes del discurso desempeñarán en la frase. Por ejemplo, especificar que una frase sustantiva es el sujeto de la oración y otra frase nominal es el complemento directo, utilizando `setSubject` y `setObject`.
- Crear un objeto SimpleNLG llamado Realizador.
- Pedir al "realizador" que transforme la instancia de `SPhraseSpec` en una cadena sintácticamente correcta.

²<https://github.com/citiususc/SimpleNLG-ES/wiki/SecciC3B3n-III-E28093-Primeros-pasos>

Parte del discurso	Tipo de frase	Ejemplos	Método
Sujeto	Frase nominal	"el niño"	setSubject()
Verbo	Frase verbal	"dió"	setVerb()
Complemento directo	Frase nominal	"un regalo"	setObject()
Complemento indirecto	Frase preposicional	"al perro"	setIndirectObject()
Complementos	Frase preposicional	"en el parque"	addComplement()
	Cláusula "que"	"que Sara ve a Juan"	
	Frase adjetiva	"encantado de conocerte"	
	Frase adverbial	"muy rápido"	
Modificadores	Cláusula "que"	"la chica que conocía"	addModifier()
	Frase adjetiva	"bonita"	
	Frase adverbial	"rápidamente"	

Figura 3.2: Tabla ejemplo composición frase.

Una vez seguidos todos estos pasos, se obtiene como resultado una cadena de texto compuesta por una frase u oración, además de poder ser tratada como cualquier otra cadena de texto de Java. Cabe mencionar que a no ser de que se quieran construir frases más elaboradas, no es necesario detallar que un sustantivo es un `NPPhraseSpec` o que un verbo es una `VPPhraseSpe`. En la siguiente tabla se muestra las partes principales con las que simpleNGL puede generar diferentes tipos de frases sintácticas (Figura 2.14).

A continuación utilizaremos la frase del ejemplo anterior "mi perro es feliz" para observar como se combina las diferentes partes de la oración. Como se ha indicado, se hace uso de la construcción `SPhraseSpec` que permite definir una frase, ya que SimpleNLG reunirá las partes en una estructura gramatical adecuada. Por lo tanto, una frase se podría construir de la siguiente manera:

- `SPhraseSpec p = nlgFactory.createClause();`
- `p.setSubject ("mi perro");`
- `p.setVerb ("es");`

- `p.setObject ("feliz");`

Este conjunto de llamadas a `simpleNLG` define los componentes de la frase: un sujeto, un verbo, y un objeto de la oración. Por último, se llama al "realizador", que se encargará de juntar estos diferentes componentes y devolverá un texto sintáctico y morfológicamente correcto:

- `Cadena salida = realiser.realiseSentence (p);`
- `System.out.println (salida);`

Por tanto, la salida resultante será:

- `mi perro es feliz.`

3.2. Django

3.2.1. ¿Qué es Django?

Es un framework de desarrollo web ³ de código abierto escrito en python en su totalidad, además respetando el patrón de diseño conocido como modelo-vista-controlador. La meta principal de esta aplicación es facilitar la creación de sitios web, para Django es muy importante el re-uso de código, la conectividad y el desarrollo rápido.

Django da soporte de base de datos permitiendo crear los modelos de datos necesarios y a través de su API administrar y gestionar dicha base de datos. Además concede al usuario la posibilidad de poder ejecutar sus propias consultas SQL.

En la parte de servicios Web, Django incluye un servidor ligero que ofrece la posibilidad de realizar pruebas y trabajar en una etapa de desarrollo. Para una etapa más de producción sería más conveniente contar con otra aplicación como puede ser Apache.

3.2.2. ¿Por qué elegimos Django?

Debido a las características enumeradas en el apartado anterior y al elegir python como nuestro lenguaje de programación toda la implementación del trabajo se realizará con Django.

El framework ofrece un servidor web en el que se podrá crear y gestionar una base de datos. En dicha base de datos estarán contenidas las imágenes de cada pictograma y su correspondiente expresión, es decir permitirá asociar cada palabra con el pictograma que represente su mismo significado.

Para llevar a cabo todas las consultas necesarias de la base de datos se implementarán los tres métodos de servicio web de tipo Restful: GET, POST,

³conjunto de componentes que ayudan a desarrollar sitios web más rápido y cómodo

DELETE. El método GET será el más utilizado ya que con él se podrá obtener la mayor información sobre las imágenes y las palabras, dependiendo de la información que se requiera obtener se llamarán a los diferentes métodos. Todos estos resultados serán volcados en el servicio web creado a partir de Django, donde mediante un pictograma se conseguirá obtener su significado sintáctico.

Capítulo 4

Conclusiones y Trabajo Futuro

Conclusiones del trabajo y líneas de trabajo futuro.

Chapter 4

Conclusions and Future Work

Conclusions and future lines of work.

Apéndice	A
----------	----------

Título

Contenido del apéndice

Apéndice	B
----------	----------

Título

Bibliografía

*Y así, del mucho leer y del poco dormir,
se le secó el cerebro de manera que vino
a perder el juicio.*

Miguel de Cervantes Saavedra

AGUSTÍN HERNÁNDEZ-GIL ALONSO, S. P. G., CARLOS MARTÍNEZ DÍAZ. *Conversor de texto a pictogramas*. Phd, Universidad Complutense de Madrid, 2013.

ÁLVAREZ, L. L. *Los sistemas Alternativos y/o Aumentativos de Comunicación: la Comunicación Bimodal como recurso en el aula de Audición y Lenguaje*. Tfg, Universidad de Valladolid, 2013.

CARLOS GARCÍA IBÁÑEZ, P. G., RAQUEL HERVÁS BALLESTEROS. Una arquitectura software para el desarrollo de aplicaciones de generación de lenguaje natural. 2004.

DOLORES ABRIL ABADÍN, A. V. C., CLARA I. DELGADO SANTOS. *Comunicación Aumentativa y Alternativa*. CEAPAT, 3 edición, 2010.

JOZEF TRZPIS, D. *Adaptación de una herramienta de Generación de Lenguaje Natural al idioma Español*. Phd, Universidad Politécnica de Madrid, 2015.

MARTÍN GUERRERO, A. *PICTAR: una herramienta de elaboración de contenido para personas con TEA basada en la traducción de texto a pictogramas*. Phd, Universidad Complutense de Madrid, 2018.

MCDONALD, E. T. *Sistema Bliss enseñanza y uso*. Ministerio de educación y ciencia, 3 edición, 1985. ISBN 84-369-1176-8.

ROBERT DALE, H. S., HERMAN MOISL. *Handbook of Natual Language Processing*. Ministerio de educación y ciencia, 2000. ISBN 0-8247-9000-6.

VICENTE, M., BARROS, C., PEREGRINO, F. S., AGULLÁ, F. y LLORET, E. La generación de lenguaje natural: análisis del estado actual. *Computación y Sistemas*, vol. 19, páginas 721 – 756, 2015. ISSN 1405-5546.

WARRICK, A. *Comunicación sin habla*. CEAPAT, 3 edición, 2010. ISBN 0-9684186-0-0.

*—¿Qué te parece desto, Sancho? — Dijo Don Quijote —
Bien podrán los encantadores quitarme la ventura,
pero el esfuerzo y el ánimo, será imposible.*

*Segunda parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes*

*—Buena está — dijo Sancho —; fírmela vuestra merced.
—No es menester firmarla — dijo Don Quijote—,
sino solamente poner mi rúbrica.*

*Primera parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes*

