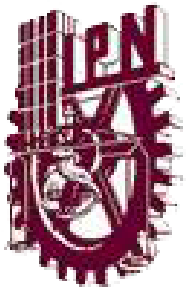


Instituto Politécnico Nacional
Centro de Investigación en Computación

Alineación automática de textos paralelos a nivel de palabras usando información lingüística diversa

Que para obtener el grado de:

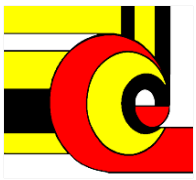
Doctorado en Ciencias de la Computación

Presenta:

Eduardo Antonio Cendejas Castro

Director:

Dr. Grigori Sidorov



México, D.F. Mayo 2013

SIP-14

INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las 16:00 horas del día 3 del mes de Diciembre de 2012 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis titulada:

"ALINEACIÓN AUTOMÁTICA DE TEXTOS PARALELOS A NIVEL DE PALABRAS USANDO INFORMACIÓN LINGÜÍSTICA DIVERSA"

Presentada por el alumno:

CENDEJAS	**CASTRO**	**EDUARDO ANTONIO**
Apellido paterno	Apellido materno	Nombre(s)

Con registro:

A	0	7	0	2	5	2

aspirante de: **DOCTORADO EN CIENCIAS DE LA COMPUTACIÓN**

Después de intercambiar opiniones los miembros de la Comisión manifestaron ***APROBAR LA TESIS***, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Director de tesis

Dr. Grigori Sidorov

Dr. Sergio Suárez Guerra

Dr. Alexander Gelbukh

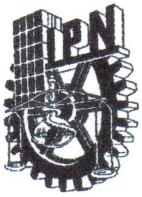
Dr. Héctor Jiménez Salazar

Dr. Raúl Morales Carrasco

PRESIDENTE DEL COLEGIO DE PROFESORES

Dr. Luis Alfonso Villa Vargas

CENTRO DE INVESTIGACION EN COMPUTACION DIRECCION



INSTITUTO POLITECNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESION DE DERECHOS

En la Ciudad de México, el día 16 del mes de Enero del año 2013, el que suscribe Eduardo Antonio Cendejas Castro alumno del Programa de Doctorado en Ciencias de la Computación con número de registro A070252 adscrita al Centro de Investigación en Computación, manifiesta que es autora intelectual del presente trabajo de Tesis bajo la dirección del Dr. Grigori Sidorov y cede los derechos del trabajo intitulado ALINEACIÓN AUTOMÁTICA DE TEXTOS PARALELOS A NIVEL DE PALABRAS USANDO INFORMACIÓN LINGÜÍSTICA DIVERSA, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección eacc2000@hotmail.com. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Eduardo Antonio Cendejas Castro

Resumen

La alineación de texto es una importante área de investigación en el campo de la lingüística computacional, especialmente para las tareas relacionadas con la traducción automática. También constituye un apoyo en otras áreas relacionadas, tales como la traducción asistida, lexicografía asistida, lingüística contrastiva y terminología.

El objetivo de los algoritmos de alineación consiste en establecer una correspondencia entre las unidades estructurales de textos paralelos: textos que están escritos en diferentes idiomas, pero son traducciones entre sí. Estas correspondencias se pueden establecer en varios niveles: textos, párrafos, oraciones y palabras.

Existen diversos enfoques y recursos que pueden ser empleados para obtener la alineación. Las dos aproximaciones principales siguen un enfoque lingüístico o un enfoque estadístico. A partir de éstas se han desarrollado varias técnicas, cada una de ellas con sus propias ventajas y desventajas.

Durante la alineación de textos paralelos a nivel de palabras surgen varios problemas originados por la desigualdad de las reglas gramaticales y la diversa cobertura de unidades léxicas en cada idioma. Es por ello, que a menudo los algoritmos de alineación se crean orientados a pares de lenguajes específicos, por ejemplo: español - inglés, inglés - francés, inglés - rumano, etc. Además, los algoritmos deben considerar los recursos disponibles para llevar a cabo la tarea de alineación y su complejidad computacional.

Esta tesis presenta una metodologia que es programada mediante un algoritmo que utiliza tanto técnicas estadísticas, como lingüísticas, es flexible y se puede configurar fácilmente. El algoritmo se basa en recursos lingüísticos, tales como: información morfológica, equivalencias léxicas de traducción, cognados y dominios semánticos. El algoritmo propuesto muestra mejores resultados en la alineación que los métodos del estado del arte.

Abstract

Text alignment is an important research area in the field of computational linguistics, especially for the tasks related to statistical machine translation. It is also important in other related areas such as computer assisted translation, computer assisted lexicography, contrastive linguistic and terminology.

The purpose of the alignment algorithms is to establish a correspondence between structural units in parallel texts: texts that are written in different languages but are translations of each other. These correspondences can be established on several levels: texts, paragraphs, sentences and words.

Various approaches and resources can be employed to achieve the alignment. Two main approximations are: linguistic approach and statistic approach. There are various techniques in each approach with their own advantages and disadvantages.

During the alignment of parallel texts at the word level, several problems arise originated by the dissimilarity of grammar rules and different coverage of lexical units of in each language. Often the alignment algorithms are created with orientation to specific language pair; for example, Spanish - English, English - French, English - Romanian, etc. Besides, the algorithms should consider the available resources to perform the alignment task and its computational complexity.

This thesis presents a methodology that is programmed with an algorithm that uses both statistic and linguistic techniques, is flexible, can be simply configured, and is adaptable to the computer environment. The algorithm is based on linguistic resources such as: morphological information, lexical translation equivalents, cognates, and semantic domains. The proposed algorithm shows better results in alignment than state of the art methods.

Agradecimientos

A Dios, por darme una familia maravillosa y darme las oportunidades para salir adelante.

A mi hija Angela, que es lo más valioso que tengo en la vida.

A mi esposa Grettel, por impulsarme para alcanzar mis metas.

A mis padres, por el apoyo que me brindan en todos los aspectos. A mi madre por su comprensión y sabios consejos. A mi padre por ser mi ejemplo de vida, espero un día ser como él.

A mi hermana y su familia por su alegría y buenos consejos.

A mi asesor, el Dr. Grigori Sidorov, por ser mi guía durante esta etapa. Agradezco su tiempo, confianza y comprensión a mis circunstancias.

A los miembros del tribunal, por sus valiosos y atinados comentarios, consejos, disposición e inestimable ayuda.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) y al Centro de Investigación en Computación del Instituto Politécnico Nacional (CIC - IPN) por el apoyo otorgado en la realización de este trabajo que contribuye a la educación e investigación en México.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

1.1. Actualidad en el alineamiento a nivel de palabras

La lingüística computacional trata de la construcción de modelos de lenguaje natural, de tal manera que sean entendibles por las computadoras. Tiene diversos campos especializados, dentro de los cuales se encuentra el procesamiento de lenguaje natural (PLN), cuyos objetivos fundamentales son el estudio, análisis y resolución de problemas relacionados con el entendimiento y la generación automática del lenguaje natural.

El campo del procesamiento de lenguaje natural ha presentado avances significativos en los últimos años, sobre todo en lo que respecta a la alineación bilingüe para apoyar la traducción, traducción asistida, lexicografía asistida, lingüística contrastiva y terminología.

En esta área se encuentran distintas tareas de gran utilidad y vital importancia alrededor de todo el mundo, como la traducción de un lenguaje a otro. La traducción involucra más elementos de los que pudiera pensar una persona cuando se realiza de manera manual, pero la computadora necesita una serie de procesos complejos y repetitivos para poder llevarla a cabo.

En la traducción computacional existen diversos enfoques, como los basados en diccionarios, en estadística y en ejemplos. Cada enfoque posee sus propias ventajas y desventajas, pero también comparten métodos y es aquí donde se puede mencionar a la *alineación de palabras*.

La alineación de palabras es una tarea del PLN que consiste en encontrar correspondencias entre palabras de textos paralelos. En otras palabras, si se asume que se cuenta con un corpus[1] bilingüe con los lenguajes L_1 y L_2, la alineación de palabras

[1]Colecciones muy grandes de textos, normalmente con alguna información lingüística adicional,

consistiría en indicar qué palabra del lenguaje L_1, se corresponde con una palabra en el lenguaje L_2, hasta encontrar todas las correspondencias. El software que se encarga de analizar los textos paralelos y establece las correspondencias entre párrafos, oraciones o palabras se le conoce como "alineador".

Existen diferentes métodos de alineación de palabras, que pueden ser clasificados como lingüísticos o estadísticos/probabilísticos. Dentro de los primeros se utiliza información lingüística para encontrar las correspondencias, son lentos y dependen de los recursos disponibles de cada uno de los lenguajes. Para los segundos, los modelos generativos probabilísticos son complejos para implementar y lentos para entrenar, los estadísticos son más simples y se basan en asociaciones estadísticas de las palabras.

Todos los modelos desarrollados presentan características positivas y negativas y es imposible establecer uno cuyo desempeño sea el óptimo, debido principalmente a las diferencias entre los lenguajes, los recursos con los que cuenta el método o algoritmo de alineación y por las consideraciones que se deben hacer por el dominio, ya sea general o específico de un corpus.

Entre las aplicaciones para la alineación de palabras, además de la traducción computacional, se encuentran: la enseñanza de lenguas, el completar memorias de traducción, la identificación de expresiones idiomáticas, la lexicografía multilingüe y en los últimos años, la desambiguación de los sentidos de las palabras.

Cuando se habla de alineación paralela, se involucra de manera directa el uso de corpus o textos paralelos, pues estos constituyen la entrada de los algoritmos de alineación. Afortunadamente, se cuenta con suficiente material para utilizar, dado que en Internet hay gran disponibilidad de textos multilingües. Sin embargo, estos textos no siempre se encuentran organizados o estructurados de la manera más conveniente para el algoritmo o método a utilizar.

1.2. Planteamiento del problema

La identificación de correspondencias entre palabras de diferentes lenguajes en textos paralelos, tiene muchas complicaciones y restricciones, según el entorno donde se aplica el método de alineación o sistema.

La mayoría de los sistemas de alineación se han centrado en idiomas con un grado de flexión bajo, por lo que la metodología suele tomar palabras flexionadas como unidad básica. En caso de idiomas altamente flexionados esto provoca alta dispersión de los datos, resultando en un escollo insalvable para la traducción estadística. En el caso específico del español, aunque es un idioma medianamente flexionado, el problema

como las marcas morfológicas, sintácticas, referenciales, etc.

radica en sus verbos, donde la concordancia compleja hace que la misma raíz verbal tenga muchas formas [1].

El problema de alineación de sentencias u oraciones es un problema bastante bien comprendido, pero la alineación de palabras es mucho menos que eso. Mientras que los programas de alineación de oraciones independientes del lenguaje, típicamente consiguen un rango aproximando del 90 %, los sistemas a nivel de palabras no lo consiguen, de hecho se encuentran muy por debajo en los casos de independencia del lenguaje [2].

Existen muchas razones concebibles por las cuales la alineación de palabras es menos efectiva que la alineación de oraciones. Las diferentes estructuras entre los lenguajes aseguran que las palabras, comparativamente, rara vez permanezcan en una relación uno a uno entre lenguajes de un texto paralelo. Esto puede ser causado por [2]:

- Las características estructurales gramaticales entre las palabras de los lenguajes pueden corresponder morfológicamente, sintácticamente o en ninguna de ellas.
- Las convenciones ortográficas pueden discrepar en donde deben ser hechas las divisiones de las palabras.
- Los alineadores de palabras deben trabajar necesariamente con tipos de palabras en vez de con "tokens" de palabras.

Otro de los problemas a considerar, es la forma en que el algoritmo de alineación acepta las entradas al sistema: podrían ser textos desalineados, alineados a nivel de párrafos u oraciones. Esto tiene un gran impacto en los resultados de la alineación palabra a palabra presentados al finalizar el proceso. Asimismo, los resultados deben mostrarse de manera que puedan ser fácilmente verificados, interpretados y corregidos por un usuario final.

La mayoría de los métodos de alineación están basados en modelos generativos. Aunque los modelos estándar pueden teóricamente ser entrenados sin supervisión, en la práctica varios parámetros presentados deberían ser optimizados usando datos comentados o anotados, esto aunado a la dificultad de agregar características a los modelos generativos estándares [3].

Una consideración que puede volverse un problema más en la alineación de palabras para los textos paralelos, es la utilización de recursos. Ésta se puede presentar de dos maneras distintas a la hora de desarrollar los sistemas de alineación:

- *Recursos limitados*: Donde se permite a los sistemas utilizar sólo los recursos proporcionados [4].
- *Recursos ilimitados*: Donde se permite a los sistemas utilizar cualquier recurso además de los proporcionados. Tales recursos pueden ser explícitamente mencionados en la descripción del sistema [4].

Por estas razones son necesarios métodos que incrementen los porcentajes de éxito en el alineado a nivel de palabras y en las memorias desarrolladas.

1.3. Soluciones conocidas

Actualmente existen diferentes proyectos que tratan de resolver la tarea de alineación de la manera más eficiente. La mayoría de ellos están inmersos en procesos de estudio y son, por lo tanto, desarrollados por universidades o centros de investigación alrededor del mundo.

Consecuencia de lo anterior, se tiene gran cantidad de proyectos que plasman diversas ideas para realizar la alineación automática de bitextos a nivel de palabras. El desarrollo de los esquemas conlleva arduos años de análisis, en los que se intenta incrementar beneficios y reducir inconvenientes con el surgimiento de nuevas versiones de los proyectos o variantes de los métodos.

Cabe mencionar que muchos de los proyectos actuales, con fines de investigación, se basan en proyectos más maduros y probados. Por ejemplo, el proyecto GIZA++, es uno de los más conocidos. Éste es un conjunto de herramientas con diversos fines, entre ellos la alineación de textos paralelos.

Otra característica importante de dichos proyectos es que al ser de investigación se encuentran en constante desarrollo y mejora, además de que los hace más alcanzables para otros investigadores, con lo cual se pueden retomar ideas, mejorar métodos o incluso crearlos. En el Capítulo 3 (Estado del Arte) se profundizará en dichos métodos y alineadores.

1.4. Solución propuesta

Para poder obtener una buena correspondencia a nivel de palabras a partir de textos paralelos, se plantea la creación de un algoritmo de alineación que comprenda los enfoques estadístico y lingüístico, con la finalidad de obtener un sistema altamente flexible en el que se puedan incorporar recursos de diversa naturaleza.

A continuación se presentan algunas características que serán explicadas a detalle en capítulos posteriores.

Entorno general de trabajo:

- Se trabaja con los idiomas español - inglés

- Los textos paralelos son tomados de los compendios de información gratuitos que existen en Internet para poder alimentar al sistema
- El desarrollo computacional es programado en C++

Las consideraciones para los pasos a desarrollar sobre la metodología son:

- *Pre-procesamiento*: Permite ajustar las entradas del sistema para procesarlas de manera adecuada. Dentro de esta fase se consideran las siguientes tareas en caso de ser necesarias:
  - Análisis de los archivos de entrada
  - Optimización
  - Tokenización
  - Análisis morfológico
  - Etiquetado
- *Resolución*: Involucra la aplicación del algoritmo de alineación sobre las estructuras de trabajo, calculando las probabilidades y reduciendo los ruidos (errores) que se pudieran ir presentando. Además en esta etapa se aplican los procesos concernientes al enfoque lingüístico.
- *Resultados*: En esta fase se desarrolla el diccionario y se optimiza la matriz de trabajo con los resultados finales para poder trasladarlos a una estructura que permita presentar los resultados de manera sencilla, entendible y en diferentes formatos.

Consideraciones sobre el algoritmo:

- Es un algoritmo de alineación a nivel de palabras
- Está enfocado en los idiomas español - inglés
- Utiliza textos paralelos
- Utiliza información léxica
- No tiene límites en cuanto a recursos lingüísticos. Si algún elemento lingüístico es importante para obtener una buena alineación se utilizará sin restricciones, siempre y cuando esté disponible
- Se utilizan algunas métricas para su evaluación
- Utiliza varias iteraciones con distintos enfoques para después mediante su combinación y análisis obtener una mejor alineación

- Tiene como base una variante del método estadístico K-Vec

Sobre las entradas del sistema:

- Utiliza como entrada textos paralelos bilingües alineados a nivel de oraciones, sin etiquetado previo
- Puede apoyarse de cualquier recurso lingüístico disponible

Sobre las salidas del sistema:

- Las alineaciones a nivel de palabras se almacenan en una matriz de trabajo
- Genera como consecuencia un diccionario bilingüe entre las palabras utilizadas en los textos paralelos
- Tiene una interfaz para presentar los resultados de manera sencilla e intuitiva para los usuarios finales. Se manejan tres formatos de presentación:
  - XML
  - Matriz (oración vs. oración)
  - Texto (párrafo vs. párrafo)

1.5. Justificación

Debido al gran avance en las comunicaciones y los medios de transporte, la globalización es una palabra muy común hoy en día. Esto trae como consecuencia que personas con diferentes lenguas convivan o se comuniquen para diversas actividades, y es por ello que los sistemas de traducción se vuelven vitales para llevar a cabo esta interacción.

En este punto es donde toma gran importancia la alineación, ya que es la primera etapa en la extracción de información estructural y parámetros estadísticos de corpus bilingües, y específicamente, la alineación de palabras bilingües es el primer paso de la mayoría de las aproximaciones actuales para la traducción computacional estadística [3].

La variedad de métodos de alineación de palabras en corpus bilingües, tiene que ver con el enfoque con que éstos se construyen en función de pares de lenguajes con características específicas. Es por ello, que aún queda mucho campo por investigar y desarrollos por implementar, todo con el propósito de mejorar esta tarea tan importante y útil y conseguir mejoras sustanciales en los resultados obtenidos de la alineación de palabras para lograr un acercamiento a los porcentajes que maneja la alineación de oraciones.

A pesar de que existen múltiples proyectos encargados de realizar estudios y desarrollos que involucran la alineación de palabras sobre corpus bilingües, la especificidad del lenguaje limita el espacio de búsqueda. Si se realiza una investigación para el idioma español, por ejemplo, los proyectos son limitados y es por ello importante reforzar este campo para brindar más resultados que permitan realizar comparaciones, variantes y combinaciones en busca de mejoras y de un algoritmo óptimo que obtenga resultados confiables y fáciles de utilizar para otras aplicaciones y para los usuarios finales de los sistemas de alineación.

La selección de los lenguajes que deberán conformar el corpus bilingüe, estuvo relacionada con la disponibilidad de los recursos electrónicos del par de idiomas español - inglés. El inglés continúa siendo la lengua más difundida en nuestros días. De hecho, pueden encontrarse múltiples contenidos que han sido transcritos del inglés al español y viceversa. Esto está relacionado con la cantidad de parlantes que hay en la actualidad de cada uno estos idiomas (véase la Tabla 1.1) y trae como consecuencia que en Internet se encuentren recursos suficientes para trabajar y poder realizar pruebas pertinentes y conseguir resultados favorables y concisos.

Tabla 1.1: Parlantes actuales para el par de idiomas español - inglés

Idioma	Parlantes (Lengua materna)	Parlantes (1ra y 2da lengua)
Español	358,000,000	417,000,000
Inglés	341,000,000	508,000,000

Además es de vital importancia encontrar variantes de los métodos existentes, investigando y modificando sus parámetros de entrada, los recursos que utiliza, las interfaces que se plantean, la forma de presentar sus resultados, etc. Para lo cual se debe trabajar arduamente en el estudio de los métodos actuales y la búsqueda de nuevas soluciones.

1.6. Preguntas de investigación

- ¿Existen grandes diferencias entre utilizar un algoritmo que usa información lingüística de los que no la usan?
- ¿Cuál es el la relación costo / beneficio del utilizar algoritmos más complejos que utilizan información lingüística comparado con los que no los utilizan?
- ¿Cómo deben ser las entradas para que un algoritmo de alineación a nivel de palabras funcione de manera óptima y por consecuencia se obtengan mejores resultados?

- ¿Qué tan funcionales y sencillas son las interfaces que existen actualmente para mostrar los resultados de la alineación?
- ¿Cuáles serían los recursos lingüísticos más importantes sobre los cuales podemos apoyar al algoritmo de alineación?
- ¿Sería conveniente que un alineador utilizara diferentes métodos para después combinar los resultados y obtener una mejor alineación?
- ¿Cuál sería un buen método de alineación sobre el cual basarse para comenzar a trabajar?

1.7. Herramientas utilizadas en el desarrollo de la solución propuesta

El enfoque utilizado para crear el software propuesto, fue el orientado a objetos debido a que es más flexible, permite la reutilización de código y la extensión de aplicaciones, además que facilita el mantenimiento de la aplicación.

La programación de las aplicaciones del sistema se realizó mediante el lenguaje Visual C++ (MFC y Windows Forms), que es un lenguaje de programación de propósito general y permite crear aplicaciones visuales con un enfoque orientado a objetos.

Las bases de datos fueron generadas y gestionadas mediante el manejador de MySQL. Se emplean en la gestión de los diccionarios y para procesos de optimización en el proceso de alineación.

Es importante destacar que no se hace uso de librerías externas que realizan el proceso de alineado, las funciones importantes como las de E/S de los archivos, el alineado, el pre-procesado, entre otras, fueron desarrolladas como parte del presente trabajo.

1.8. Hipótesis

Una variante de un método de alineación de textos paralelos a nivel de palabras que combine técnicas estadísticas e información lingüística diversa, permitirá obtener mejores resultados que los algoritmos que utilizan estas técnicas de manera independiente. Además, la precisión de las correlaciones establecidas a través de los recursos lingüísticos debe ser elevada por la distinción de sentidos que proveen las equivalencias léxicas de traducción, los dominios semánticos y el método propuesto de identificación de cognados válidos con la incorporación de significados.

1.9. Objetivos

Para dar solución a las necesidades planteadas y poder cumplir satisfactoriamente con cada etapa de investigación y desarrollo de la propuesta descrita anteriormente, se definió un conjunto de objetivos que se mencionan a continuación.

1.9.1. Objetivo general

Desarrollar un novedoso método de alineación de textos paralelos a nivel de palabras que permita alinear pares de palabras de los lenguajes español - inglés, utilizando recursos lingüísticos diversos.

1.9.2. Objetivos específicos

Como objetivos específicos se mencionan los siguientes:

- Diseñar y desarrollar el algoritmo de alineación de textos bilingües a nivel de palabras que pueda utilizar información lingüística diversa.
- Diseñar y desarrollar el módulo encargado de las entradas de información, ya sean los archivos a alinear o los recursos lingüísticos a utilizar por el algoritmo.
- Diseñar y desarrollar el módulo de pre-procesamiento de los archivos de entrada.
- Diseñar y crear los módulos de procesamiento lingüístico incorporando información morfológica, equivalencias léxicas de traducción, dominios semánticos y cognados.
- Diseñar y desarrollar el módulo de configuración del algoritmo para proporcionar flexibilidad a su utilización.
- Diseñar y desarrollar el módulo encargado de mostrar los resultados en diferentes formas o formatos.
- Diseñar y desarrollar una interfaz para manipular el algoritmo planteado y sus módulos, que sea eficiente, sencilla y que facilite la interpretación de los resultados a los usuarios finales.
- Realizar pruebas para obtener resultados y compararlos con resultados de otros alineadores.

1.10. Limitaciones

Algunas limitaciones que están presentes en el trabajo son:

- La eficacia del algoritmo depende también de la calidad de los recursos lingüísticos que se proporcionan como entrada.
- No se tomarán como entrada archivos etiquetados.
- Algunos módulos lingüísticos, como el de árboles sintácticos no se optimizarán ni afinarán, ya que sólo se desarrollarán con el propósito de prueba para el algoritmo de alineación.

1.11. Estructura del documento

El presente trabajo está organizado en 6 capítulos, los cuales se describen a continuación:

- Capítulo 2. Marco Teórico: Posee las bases necesarias para desarrollar la presente propuesta, tomando en cuenta los aspectos importantes de los corpus paralelos, la alineación, las bases de datos, el pre-procesamiento, la evaluación y las estrategias para la alineación de palabras basadas en matrices de pistas.
- Capítulo 3. Estado de Arte: Presenta algunos de los desarrollos representativos en cuanto a los alineadores y sus los métodos de alineación, ya sean estadísticos o lingüísticos.
- Capítulo 4. Algoritmo HWA: Muestra el diseño y desarrollo del algoritmo HWA, el funcionamiento de sus módulos y la utilización de su interfaz.
- Capítulo 5. Resultados: Presenta datos sobre los resultados obtenidos con el desarrollo propuesto y su comparación con otros resultados obtenidos con diferentes algoritmos.
- Capítulo 6. Conclusiones: Presenta las aportaciones y conclusiones del trabajo propuesto, con base en los resultados obtenidos. Además menciona los trabajos futuros con los cuales se podría complementar el trabajo actual.

Capítulo 2

Marco Teórico

2.1. Corpus

Un corpus es una gran colección de textos electrónicos (de un millón de palabras como mínimo) que se usa para la investigación, y sobre todo para el desarrollo de software de traducción o procesamiento de lenguaje natural. Generalmente se conservan y distribuyen en CD-ROM. Algunos se pueden consultar por Internet.

La principal ventaja de los corpus es que, al estar digitalizados, se pueden consultar fácilmente por ordenador. Existe una gama de programas que permiten realizar todo tipo de búsquedas, de forma precisa y consistente y a gran velocidad como:

- Buscar palabras y frases.
- Calcular el número de apariciones del texto buscado.
- Presentar los datos en su contexto.

2.1.1. Tipos

Algunos de los tipos de corpus que podemos encontrar son:

- *Bases de datos de árboles*: Son textos etiquetados sintácticamente. En general, los análisis sintácticos tienen una estructura en forma de árbol, lo que explica su nombre. Sin embargo, existen también bases de datos de árboles que tienen una estructura de gráfica con conexiones adicionales entre las palabras, en los que la construcción sintáctica no corresponde a un árbol simple, por ejemplo: NEGRA/TIGER, PDT: Prague Dependency Treebank, Corpus Le Monde, TUT: Turin University Treebank, Spanish Treebank (UAM), ISST: Italian Syntactic-Semantic Treebank, Penn Treebank, Susanne Corpus.
- *Corpus orales*: Están constituidos por señales de voz, eventualmente con sus transcripciones de anotación fonética. Un corpus oral contiene grabaciones de llamadas telefónicas, entrevistas o programas de radio, por ejemplo.

- *Corpus multimodales*: Están constituidos por otros datos orales como prosodia, gestos, movimientos de la boca, inclusive grabaciones sonoros y fílmicas (noticias, documentales).

- *Corpus textuales*: Están constituidos por lengua escrita o por lengua oral transcrita. Predominan, por lo general corpus textuales que se originan en su totalidad de textos ya que se pueden elaborar con bastante menos esfuerzo que otros corpus. Comúnmente, tienen varios cientos de millones de palabras. Otros tipos de corpus cuentan apenas con poco más de un millón de palabras.

Se podría efectuar también otra subdivisión de los corpus:

- *Corpus sincrónicos vs. diacrónicos*: Para el corpus sincrónico se recopila material que se compone de la lengua actual (por ejemplo de 1945 hasta el presente). Para el corpus diacrónico se recogen textos de varias etapas históricas de la lengua a fin de poder observar los cambios en la misma(por ejemplo, los años 1200 hasta 1900).

- *Corpus monolingües vs. multilingües*: Los corpus monolingües contienen, en comparación con los corpus multilingües, textos únicamente en una lengua. Los corpus multilingües (y los corpus bilingües) son muy escasos en comparación con los corpus monolingües porque los textos tienen que existir en versiones traducidas. Los corpus monolingües son de gran utilidad para hacer comparaciones entre las lenguas, elaborar diccionarios y elaborar las memorias de traducción.

- *Corpus históricos vs. textuales modernos*: Mientras los corpus textuales modernos pueden recurrir al material ya en forma digital, los textos para corpus históricos tienen que ser digitalizados por OCR (reconocimiento óptico de caracteres) a través de un escáner. Por ello, deben tomarse en cuenta algunos problemas especiales: ¿Se emplea el manuscrito o una edición? ¿Cómo se manejan las correcciones, las pasadas, las glosas, etc.? Otro problema es la codificación de las letras y otros signos de escritura porque algunos caracteres no existen ni siquiera en el Unicode.

- *Corpus de referencia vs. monitor*: Un corpus de referencia tiene, en comparación con un corpus monitor, un tamaño establecido. Generalmente es de libre acceso y está estandarizado. Por lo contrario, un corpus monitor aumenta de manera constante su tamaño. El corpus incluye, por ejemplo, cada día datos nuevos según criterios fijos, tal como los cumple Birmingham Bank of English. El corpus Wortschatz der Universität Leipzig presenta cada día las palabras nuevas más frecuentes.

- *Corpus dialectales*: Los corpus dialectales están disponibles normalmente sólo en forma oral. Una razón es que los dialectos por lo general no tienen una norma de escritura correcta y que en muchos dialectos no existe en absoluto la tradición de una escritura.

2.1.2. Corpus anotados

Para ser útiles, los corpus tienen que ser representativos. Esto significa que deben contener muestras significativas y proporcionales de todas las variedades de una lengua, en cuanto a sus géneros por ejemplo: periodístico, jurídico, administrativo, técnico-científico, ensayo, literario (poesía, novela, teatro), formas dialectales, lenguaje coloquial, zonas geográficas, etc.

Los corpus crudos, es decir, los corpus que contienen exclusivamente las palabras y signos de puntuación de los textos originales, tienen una utilidad bastante limitada para la lingüística computacional. Para que los corpus empiecen a ser verdaderamente útiles en el desarrollo de aplicaciones, se debe enriquecer el texto de los documentos originales con diversos tipos de información lingüística, a ser posible, de manera automática o semiautomática.

Estos corpus enriquecidos con información lingüística reciben la denominación de corpus etiquetados, ya que la información añadida al texto original se incorpora contenida dentro de etiquetas, en general, empleando alguna especificación del lenguaje XML (eXtensible Markup Language).

Hay casos en los que la búsqueda de datos en un corpus basándose exclusivamente en la forma, es difícil o imposible:

- Palabras irregulares (*go, gone, went*)
- Formas gramaticales (frases con orden verbo-sujeto)

Para solucionar este tipo de problemas, la tendencia actual es la de extender los corpus con todo tipo de información sobre las palabras y las frases que contiene. Este tipo de corpus se denominan "corpus anotados" (*annotated corpora*) y pueden contener:

- Atributos de formato: saltos de página, párrafos, cursiva, sangría
- Información sobre el texto: fecha de edición, autor, género, registro
- Información gramatical

2.1.3. Corpus paralelos

En la década de los ochentas, se introdujo la idea de almacenar electrónicamente los corpus en un formato bilingüe. El concepto consistía en la construcción de una concordancia bilingüe teniendo un operador de datos que manualmente introdujera el texto y su traducción [5], creando así una herramienta de referencia valiosa para los traductores. Por esta razón, es preferible que los corpus multilingües contengan textos que sean equivalentes o traducciones el uno del otro (corpus paralelo).

Los *corpus o textos paralelos* son textos que tienen el mismo contenido semántico, pero expresado en lenguajes diferentes [6], también son conocidos como corpus bilingües o bitextos. El término "paralelo" no implica que los textos tengan una correspondencia exacta entre palabras, oraciones y/o párrafos; es decir, dos textos pueden estar completamente desalineados sin dejar de ser textos paralelos.

2.1.4. Aplicaciones

La mayor utilidad de los corpus paralelos es para los estudios de lingüística contrastiva y de traducción. Dentro de las aplicaciones se pueden mencionar [7]:

- *Dialectología / Sociolingüística*: Investiga en el corpus cómo se distinguen los dialectos / sociolectos entre sí y con la lengua estándar (en todos los niveles lingüísticos). Estudia, por ejemplo, cómo cambian los dialectos o si las mujeres hablan de otra forma que los hombres.

- *Lingüística histórica*: Investiga en el corpus, cuándo una palabra determinada fue reemplazada por otra, cuándo una palabra cambió su modo de uso, cuándo aparece la primera vez una construcción sintáctica, etc.

- *Psicolingüística*: Los psicolingüistas investigan si, por ejemplo, la frecuencia de las palabras influye en la velocidad de reacción y qué papel desempeña la frecuencia de distintas lecturas de las palabras ambiguas. Si el corpus está etiquetado sintácticamente, se puede estudiar qué tamaño tienen las unidades sintácticas del procesamiento y si coinciden con las unidades sintácticas.

- *Lexicografía*: Para crear un diccionario, el lexicógrafo analiza en el corpus en qué contexto aparece una palabra determinada, qué lecturas de una palabra existen, qué palabras aparecen con frecuencia en una combinación establecida (colocaciones), qué palabras no tienen uso, qué palabras se emplean en un lenguaje profesional o qué sustantivos tienen al mismo tiempo más de un género.

- *Sintaxis*: Con la ayuda de un corpus (sintácticamente etiquetado) es posible verificar si existe una construcción sintáctica. Se puede investigar qué adjetivos aparecen con el verbo «*estar*» y cuáles con el verbo «*ser*», cuáles adjetivos aparecen delante y cuáles detrás de un substantivo o en qué contexto un verbo está en modo subjuntivo y en cuál está en modo indicativo.

- *Semántica*: Estudia en un corpus (etiquetado de lecturas de palabras) cómo se utiliza una palabra determinada y qué sentido tiene. Además en qué contexto se presentan las lecturas de una palabra o si la palabra aparece como metáfora.

- *Fonología*: Estudia cómo se pronuncian los extranjerismos, si mediante la prosodia es posible distinguir las lecturas de una palabra, cómo se pueden clasificar los acentos y por qué las personas que no son originalmente hispanohablantes cuando hablan español tienen un acento.

- *Lingüística computacional*: Los lingüistas computacionales utilizan los corpus como recurso para la elaboración automática de un diccionario. Además, los lingüistas computacionales usan los corpus como recurso para la extracción de frases de los corpus bilingües para una memoria de traducción, para la extracción automática de colocaciones, para la extracción automática de las diferencias del lenguaje en todos los niveles (sintaxis, semántica, etc.)

Los corpus paralelos bilingües también se usan en la traducción automática en sistemas de traducción estadística basada en alineamiento léxico y la posición de palabras. Suelen tomarse en consideración no frases enteras, sino secuencias de tres palabras, para las que se busca la equivalencia en otra lengua. Los corpus son una fuente de datos directa para las máquinas. A base del principio de la analogía sacan del corpus ejemplos típicos de frases o partes de ella para llegar a realizar la traducción de textos no traducidos todavía. Las últimas tendencias en los sistemas de traducción automática se alejan cada vez más de análisis sintácticos y semánticos completos utilizando gramáticas de reglas formales para ir basándose en datos de uso de la lengua viva [8].

Otras de las principales aplicaciones de los corpus paralelos son la enseñanza de idiomas, la estilística, el desarrollo y prueba de gramáticas computacionales, concordancias bilingües y la extracción léxica. La estilística investiga sobre el estilo de autores literarios para determinar la autoría de textos controvertidos. Las aplicaciones de concordancia bilingüe son sistemas que permiten realizar búsquedas de traducciones en los textos alineados.

Las búsquedas deben poderse hacer tanto en el texto original como en el texto de la traducción. Así, si un corpus paralelo alineado contiene textos traducidos entre la lengua A y la lengua B, el programa de concordancia debe permitir la búsqueda de una expresión en lengua A y ver las distintas maneras en que fue traducida en la lengua B en cada contexto. Por otra parte, el programa de concordancia también debe permitir buscar una expresión en lengua B y ofrecer todos los contextos en que aparece B junto a sus contextos originales en la lengua A [9].

Finalmente, un buen programa de concordancia bilingüe debe permitir hacer búsquedas realmente bilingües, es decir, búsquedas del tipo: "deseo consultar las frases del original y su traducción en aquellos casos en que en el original aparezca X y en la traducción aparezca Y". Los programas de concordancia bilingüe son extremadamente útiles como herramienta de consulta durante una traducción, superando en funcionalidad, realismo y eficacia a los clásicos diccionarios bilingües de palabras.

Los programas de concordancia no sólo permiten buscar la traducción de palabras y expresiones de una lista cerrada, como suelen permitir los diccionarios, sino que permiten buscar en los textos la traducción de cualquier tipo de fragmento textual que se pueda expresar en forma de expresiones regulares, al tiempo que facilitan no sólo su traducción, sino su contexto de uso, en traducciones reales y documentadas [9].

También es destacable el uso de los corpus paralelos como memorias de traducción en aplicaciones de traducción asistida por ordenador, especialmente cuando se trata de corpus de consulta libre adaptados para su uso como memorias de traducción distribuidas a través de Internet [9].

Para facilitar su examen, los corpus paralelos deben estar alineados de alguna forma (frase con frase o párrafo con párrafo).

2.2. Alineación

La alineación de un corpus paralelo consiste en la reestructuración de los textos de forma tal que se establezca una correspondencia entre los párrafos, las oraciones y/o las palabras de los textos involucrados [10].

Algunos autores, entre ellos Mikhailov[10], distinguen cuatro diferentes niveles de alineación, a saber:

- *Alineación de 1er orden*: Alineación utilizada cuando los textos que conforman el corpus son muy cortos. Se denomina alineación a nivel de texto (completo).
- *Alineación de 2do orden*: La alineación de 2do grado se refiere a la alineación donde los segmentos por alinear son los párrafos.
- *Alineación de 3er orden*: La alineación de 3er grado se refiere a la alineación donde los segmentos por alinear son las oraciones.
- *Alineación de 4to orden*: La alineación de 4to grado se refiere a la alineación donde los segmentos por alinear son las palabras.

2.2.1. Niveles de resolución

El trabajo de detección de párrafos es mucho más simple que el trabajo de detección de oraciones, pues casi siempre están en una relación uno a uno, lo cual resulta fortuito ya que la alineación a nivel de oraciones mejora mucho si se realiza primero la alineación a nivel de párrafos [11].

El nivel de resolución de oraciones presentó un desafío mayor a la alineación automática de unidades textuales más bastas (como párrafo o texto), descubriendo que en él, las correspondencias uno a muchos y muchos a uno, no son raras.

Aunque los resultados obtenidos en este nivel han sido absolutamente exactos cuando están probados en recopilaciones relativamente limpias y extensas, siguen siendo alineaciones parciales, pues ocultan el grado más fino de resolución: el de palabra, donde

la relación uno a uno entre los elementos que son alineados llega a ser cada vez más rara.

La alineación a nivel de palabras, puede ser definida como la indicación de la correspondencia entre las palabras en un texto paralelo, como se muestra en el siguiente ejemplo.

Dados los textos paralelos en los lenguajes L_1 y L_2:

- L_1 - *Mi nombre es Eduardo.*
- L_2 - My *name is Eduardo.*

La Tabla 2.1 muestra los resultados obtenidos, después de ejecutar un algoritmo de alineación.

Tabla 2.1: Ejemplo de una matriz de alineación

	Mi	*nombre*	*es*	*Eduardo*	.
My	×				
name		×			
is			×		
Eduardo				×	
.					×

El ejemplo anterior presenta el caso más simple de relación (uno a uno) Sin embargo, en este nivel se pueden encontrar alineaciones:

- uno a muchos
- muchos a uno
- muchos a muchos
- uno a ninguno
- ninguno a uno

2.2.2. Problemas en la alineación de palabras

La suposición inicial para la alineación de palabras es que tenemos un corpus paralelo alineado a nivel de oraciones. Dado un par de oraciones paralelas, debemos enlazar (alinear) palabras que son traducciones entre ellas mismas. Puede haber un gran número de alineaciones posibles, pero se necesita encontrar el mejor alineamiento. Los principales problemas presentados en el alineamiento son [12]:

1. Encontrar las traducciones más probables de una palabra SL (palabra del lenguaje fuente u origen), sin importar su posición. Esta parte la cuida el modelo de traducción, que en sí tiene muchas aplicaciones. Por ejemplo, como este modelo proporciona probables traducciones de palabras, se puede utilizar para realizar la tarea de construir un diccionario bilingüe más fácilmente.

2. Alinear las posiciones en la oración SL con posiciones en la oración TL (palabra del lenguaje destino o meta). Este problema está dirigido por el modelo de distorsión, que tiene cuidado de las diferencias de órdenes de palabras de los dos lenguajes.

3. Encontrar cómo muchas de las palabras de TL son generadas por una palabra de SL. Además de que una palabra de SL puede no generar ninguna palabra TL, o una palabra TL puede ser generada por ninguna palabra SL (inserción de nulos - NULL insertion).

2.3. Métodos de alineación

Las investigaciones sobre alineación de textos han seguido dos corrientes, a saber, la corriente estadística y la corriente lingüística [13].

Las *técnicas estadísticas* ofrecen una gran velocidad en el proceso de alineación y cierta independencia entre la técnica y los lenguajes que se están tratando. Su punto débil es que este método puede fallar si la traducción no cumple con la filosofía sobre la cual se apoya dicha técnica. El cumplir con la filosofía implica que la traducción debe tener alta semejanza estructural con el texto original; oraciones grandes (en caracteres) en el original tienden a ser traducidas a oraciones grandes y oraciones cortas tienden a ser traducidas a oraciones cortas [14].

Las *técnicas lingüísticas* tienden a ser mucho más lentas en el proceso de alineado, además de que resultan dependientes de los lenguajes tratados. No obstante, se espera que éstas generen mejores resultados al considerar el sentido de los textos [11].

Los recursos lingüísticos son por ejemplo pistas declarativas, las cuales son pistas a asociaciones entre pares de palabras de clases de palabras relacionadas. Otros enfoques para alineación de palabras están siendo investigados. Los sistemas como el alineador de pistas requieren un cierto grado de conocimiento contextual y son consumidores de tiempo, y los corpus involucrados a menudo son muy grandes [15].

2.3.1. Técnicas estadísticas

Los dos tipos principales de alineación de palabras son el enfoque de longitud y el de asociación. Ambos enfoques hacen uso de la estadística.

Enfoque de longitud

El enfoque de longitud está basado en modelos de traducción probabilística calculados de corpus paralelos [15]. En estos métodos se desecha la información léxica y la única información utilizada es la longitud de las unidades de texto.

Se busca el mejor alineamiento A, dados los textos S y T:

$$argA\ max\ P(A \mid S,T) = argA\ max\ P(A,S,T)$$

Un alineamiento es una secuencia de cuentas $(B_1, ..., B_k)$, asumiendo que son independientes nos queda: $P(A,S,T)_k = P(B_k)_{1..k}$

La cuestión es calcular $P(B_k)$ dadas las frases de los textos.

Enfoque de asociación

El enfoque de asociación está construido en medidas de correspondencia y se origina de estudios anteriores del análisis léxico de corpus paralelos [16].

Los pasos básicos en la alineación con el enfoque de asociación son [15]:

- *Segmentación léxica*: Identificación de fronteras de elementos léxicos en lenguajes fuente y destino.
- *Correspondencia*: Posibles relaciones de traducción entre elementos léxicos son identificados de acuerdo a criterios de correspondencia, resultando en un diccionario de asociación con puntajes de asociación para cada entrada.
- *Alineación y extracción*: Las traducciones más confiables de los diccionarios de asociación son marcados en la alineación, a menudo usando un algoritmo de búsqueda de "los primeros mejores" combinado con restricciones heurísticas y lingüísticas. Las palabras alineadas son extraídas a un diccionario de traducción bilingüe [16].

La información estadística es aplicada en enfoques de asociación en forma de medidas de co-ocurrencia. Medidas de similitud de cadenas son también aplicadas, resultando en la localización de cognados.

2.3.2. Cognados

Los algoritmos estadísticos (basados en longitud y/o asociación) para alinear palabras no hacen uso de ningún recurso lingüístico adicional durante el proceso de alineación, lo cual implica una independencia considerable del lenguaje.

Sin embargo, estos algoritmos tienen la desventaja de que una vez que asocian accidentalmente un par de entidades desalineadas, tienden a ser incapaces de corregirse. Es por ello que comienzan a introducirse elementos lingüísticos con el fin de reforzar las alineaciones. De esta forma, se toman en cuenta las características propias de los idiomas durante el procesamiento. Se propone entonces, entre otros, un método de alineamiento basado en cognados, que junta los criterios de la longitud / asociación con la noción de cognación [17] para mejorar la eficiencia de los algoritmos antes propuestos.

Los *cognados* son pares de palabras con etimología común y con fonología, ortografía y características semánticas similares. La similitud se debe generalmente o bien a una relación genética o préstamos de un idioma a otro [18]. Así, se facilita la alineación de palabras que coinciden en su totalidad o de manera parcial. En este sentido, los cognados no sólo incluirían pares de palabras genéticamente relacionadas o préstamos, sino también nombres propios, números y signos de puntuación [18].

En el caso de los idiomas español e inglés, muchas de sus palabras poseen el mismo origen. Se dice que más del 65 % de todas las palabras inglesas proviene del latín y que éste es también la base del 75 % al 80 % de las palabras españolas. Por ejemplo, las palabras «*arquitectura*» y «*architecture*» provienen del latín «*architectura*».

La identificación de cognados ha sido empleada, tanto en la lingüística histórica o comparativa, como en la lingüística de corpus. En lingüística histórica tiene como objetivo el establecimiento de las relaciones entre las lenguas y la reconstrucción de las historias de familias lingüísticas. En la lingüística de corpus, los cognados han sido empleados en numerosas tareas relacionadas con textos paralelos; incluyendo alineación [19], traducción de lexicones [20] y mejora de modelos existentes de traducción automática estadística [21].

En general, el decidir si dos palabras están genéticamente relacionadas requiere de conocimiento experto de la historia de las lenguas en cuestión, pues con el pasar del tiempo, las palabras cambian de forma y significado. Después de varios siglos, los cognados adquieren a menudo formas fonéticas muy diferentes. Por otra parte, la similitud fonética de algunas palabras semánticamente equivalentes puede ser por mera casualidad [22].

Los enfoques para medir la similitud de las palabras, se han dividido en dos grupos [22]: (1) los ortográficos, que realizan la comparación por caracteres o símbolos de codificación y (2) los fonéticos, que se basan en las características de los sonidos individuales, lo que presupone una trascripción inicial de las palabras en su representación fonética.

Enfoques ortográficos

Los enfoques ortográficos consideran que los símbolos alfabéticos expresan sonidos concretos, empleando una función de identidad binaria en el nivel de comparación de los caracteres. Este tipo de aproximaciones, son comúnmente usadas en lingüística de corpus.

Enfoques fonéticos

En los enfoques fonéticos la alineación supone la transcripción de los sonidos en segmentos fonéticos discretos. Los algoritmos de alineación suelen contener dos componentes principales: una métrica para medir la distancia entre los fonemas y un procedimiento para encontrar la mejor alineación. El primero suele ser calculado sobre la base de características o rasgos fonológicos que codifican ciertas propiedades de los fonemas. El valor numérico asignado por la función a un par de segmentos se conoce como el costo, o penalización, de la sustitución.

2.3.3. Técnicas lingüísticas

Por otro lado, los métodos lingüísticos se apoyan en recursos léxicos existentes, como diccionarios bilingües de gran escala [23], lexicones con información morfológica [24] y árboles sintácticos [25], para establecer la correspondencia entre las unidades estructurales.

Los diccionarios permiten la extracción de información léxica. De esta forma, se cuenta con la palabra en el texto origen y las posibles traducciones en el texto meta. Estos datos pueden ser empleados entonces en la creación o variación de las probabilidades de los enlaces obtenidos en la fase estadística.

Del mismo modo, la información morfológica y la sintáctica, son fuentes de conocimiento para incrementar o disminuir la certidumbre de cada par de alineación. Con la utilización de la primera es posible comparar lemas y verificar las categorías gramaticales. Por otro lado, el conocer la sintaxis de las oraciones permite identificar sus partes (sujeto, verbo, predicado, etc.) y efectuar comparaciones con sus contrapartes en el segundo lenguaje.

Otros elementos lingüísticos, que se han comenzado a incorporar para reforzar las alineaciones producidas por los algoritmos estadísticos, incorporan información semántica.

2.3.4. Información semántica

Por la disponibilidad cada vez mayor de recursos bilingües, se invierte más esfuerzo en la investigación de la efectividad de los acercamientos basados en léxicos. Aunque

en este sentido es importante destacar que la mayoría de éstos se limitan al uso de diccionarios bilingües y que no se han explotado otros recursos como lexicones computacionales alineados y corpus que proporcionen una colección de ejemplos.

Lexicones computacionales: PWN

Uno de los recursos más importantes empleados en la actualidad para aplicaciones que requieren procesamiento de lenguaje natural son los lexicones computacionales. Éstos describen las relaciones léxicas y semánticas existentes entre las palabras y recopilan sus sentidos al igual que un diccionario monolingüe.

El recurso de mayor cobertura de palabras es *Princeton WordNet* (PWN), un lexicón de concordancia semántica para el idioma inglés, que agrupa los sinónimos en conjuntos denominados synsets y proporciona las definiciones para estos conjuntos. PWN se ha venido desarrollando desde los años 80 bajo la dirección del psicolingüista George Miller en la Universidad de Princeton [26]. El propósito de éste es producir una combinación de diccionario y tesauro que sea intuitivamente utilizable y respalde el análisis de texto automático. Sin embargo, su metodología de construcción obliga a los intérpretes a familiarizarse con los inventarios de sentidos de cada palabra.

Lexicones alineados: MWN

Para la construcción de lexicones computacionales multilingües alineados, se han empleado dos metodologías. En la primera, se construyen las redes de palabras de cada lenguaje de manera independiente. Cada red posee su propia estructura y sistema de identificación de significados. Posteriormente, se realiza la alineación haciendo uso de un índice interlingua (ILI) que es el encargado de establecer las relaciones entre las lenguas implicadas. Este acercamiento es el empleado por *EuroWordNet* [27], una base de datos multilingüe con lexicones para varios idiomas europeos (holandés, italiano, español, alemán, francés, checo y estonio).

En la segunda metodología las redes utilizan la estructura implementada en PWN, es decir poseen la misma fundamentación teórica en la idea de la matriz de vocabulario. Los synsets para cada uno de los idiomas alineados son creados con correspondencia exacta con los synsets de PWN, incluso respetando la identificación. Las relaciones semánticas también fueron importadas de los synsets ingleses correspondientes.

Este enfoque fue empleado para el desarrollo de *MultiWordNet* (MWN) [28], un lexicón multilingüe conformado por redes de palabras para el español, el italiano y el hebreo, entre otros lenguajes. La precisión de este proyecto es tal, que a partir de MWN pueden identificarse las brechas léxicas entre los idiomas, representadas con la leyenda "gap".

2.4. Alineación óptima de unidades multi - palabras

Independientemente de la técnica empleada (estadística y/o lingüística) para identificar relaciones de traducción, hay que considerar las estrategias para determinar el alineamiento óptimo en unidades multi-palabra. Para ello, se han utilizado diversos modelos basados en matrices de pistas y tablas de contingencia con pruebas de asociación.

2.4.1. Matrices de pistas

Una matriz de pistas resume información de varias fuentes que pueden ser usadas para identificar relaciones de traducción. Sin embargo, no hay una forma obvia de utilizar esta información para alineación de palabras.

Existen muchas formas de agrupar ("clustering") palabras y no hay un procedimiento de maximización obvio para encontrar el alineamiento óptimo cuando las unidades multi-palabra (MWUs) están involucrados. El procedimiento de alineación depende en gran medida de la definición de una alineación óptima.

Un procedimiento típico para alineación automática de palabras es comenzar con enlaces de palabras uno a uno. Los enlaces que tienen una fuente común o palabras del lenguaje destino son llamados enlaces traslapados. Grupos de enlaces traslapados, los cuales no se traslapan con ningún otro enlace fuera del conjunto son llamados grupos de enlace (*link clusters* - LC). Alinear palabras una a una a menudo produce traslapes y en esta forma implícitamente se crean unidades multi-palabras alineadas como parte de los grupos de enlace [16].

Modelos de alineación direccionales

El enfoque de alineación de palabras es asumir un modelo de alineación de palabras direccional similar a los modelos estadísticos de traducción. El modelo de alineación direccional asume que hay a lo más un enlace para cada palabra del lenguaje fuente.

En otras palabras, la alineación de palabras es la búsqueda del mayor enlace para cada palabra del lenguaje fuente. Los modelos direccionales no permiten enlaces múltiples de un elemento a varios elementos destino. Sin embargo, los elementos destino pueden ser enlazados a múltiples palabras del lenguaje fuente para que puedan ser alineados a la misma palabra del lenguaje destino.

Alineación direccional combinada

Los grupos de enlaces direccionales pueden ser combinados de diferentes maneras. La unión de los conjuntos de enlace usualmente causa traslapes y, por lo tanto, grupos de enlaces muy grandes. Por otro lado, una intersección de grupos de enlace elimina

todos los traslapes y deja solo los enlaces de palabras uno a uno altamente confiables.

La unión e intersección de enlaces no produce resultados satisfactorios. Otra estrategia de alineación es una combinación refinada de grupos de enlaces como los sugeridos por Och y Ney [29] . En este enfoque, la intersección de enlaces es iterativamente extendida por enlaces adicionales, los cuales aprueban una de las siguientes dos restricciones.

Un enlace nuevo es aceptado si ambos elementos en el enlace no están alineados todavía. Explorando en un espacio de dos dimensiones de un bitexto, el nuevo enlace es adyacente de manera horizontal o vertical a un enlace existente y el nuevo enlace no causa enlaces adyacentes a otros enlaces en ambas dimensiones.

Enlace competitivo

Otro enfoque de alineación es el enfoque de enlace competitivo propuesto por Melamed [30]. En este enfoque se asume que existen sólo enlaces de palabras uno a uno. La alineación se realiza en un avaro comportamiento de búsqueda del "primero mejor", donde los enlaces con los más altos puntajes de asociación se alinean primero, y los elementos alineados son removidos del espacio de búsqueda. Este proceso se repite hasta que ya no haya enlaces.

Alineamiento forzado al mejor primero

Otro enfoque iterativo de alineación propuesto por Tiedemann [16]. En este enfoque, el enlace con el puntaje más alto en la matriz de pistas se agrega al conjunto de grupos de enlace si cumplen ciertas restricciones. El puntaje más alto se elimina de la matriz y la búsqueda de enlaces se repite hasta que no haya más enlaces. Esto es básicamente una búsqueda del "primero mejor" donde varias restricciones pueden ser posibles.

Los enlaces sin traslape son siempre aceptados (por ejemplo, un enlace sin traslape crea un nuevo grupo de enlace). Otras posibles restricciones son umbrales de valores de pistas, los umbrales para puntajes de pistas diferencian entre enlaces adyacentes o restricciones sintácticas.

2.4.2. Tablas de contingencia

Las tablas de contingencia son una forma simple de representar datos. Si consideramos un ejemplo donde se quiere clasificar a 10 estudiantes dependiendo si se han matriculado o no en cursos de física y química, entonces podemos asignar valores a diferentes categorías: para los que matricularon ambos cursos el número 3, para los que sólo matricularon física el número 2, 1 para los de química y 4 para aquellos que no matricularon ningún curso. Estos datos pueden ser representados en una matriz

conocida como tabla de contingencia [31].

Tabla 2.2: Tabla de contingencia para clasificar 10 alumnos

	Física	Sin Física	Total
Química	$a = 3$	$c = 1$	$a + c = 4$
Sin Química	$b = 2$	$d = 4$	$b + d = 6$
Total	$a + b = 5$	$c + d = 5$	$T = a + b + c + d = 10$

En la Tabla 2.2 podemos observar que la celda $a = 3$ representa el número total de estudiantes que registraron ambos cursos, las celdas $b = 2$ y $c = 1$ representan el número total de estudiantes que registraron sólo el curso de física y química respectivamente.

La representación de los datos de esta manera ayuda en la obtención de información adicional o en la fácil identificación de datos que no se expresan directamente en la información inicial. Por ejemplo, podemos saber cuántos alumnos no han registrado ningún curso, el número total de alumnos que están en Física sin importar si están registrados en Química, etc.

Tabla 2.3: Tabla de contingencia para representar ocurrencia de palabras

	$Palabra2$	$!Palabra2$	
$Palabra1$	$n11$	$n12$	$n1p$
$!Palabra1$	$n21$	$n22$	$n2p$
	$np1$	$np2$	npp

Las entradas de la Tabla se interpretan de la siguiente manera:

- $n11$ = frecuencia conjunta de que $Palabra1$ y $Palabra2$ ocurran
- $n12$ = frecuencia de que $Palabra1$ ocurra y que $Palabra2$ no
- $n21$ = frecuencia de que $Palabra2$ ocurra y que $Palabra1$ no
- $n22$ = frecuencia de que ni $Palabra1$ ni $Palabra2$ ocurran
- npp = total de piezas en que se divide el texto
- $n1p, np1, np2, n2p$ son los conteos marginales

2.4.3. Pruebas de asociación

Las pruebas de asociación son pruebas estadísticas para encontrar la similitud entre dos entidades. Una vez que una tabla de contingencia es creada, varias pruebas de asociación pueden ser utilizadas para encontrar si dos variables en la tabla están relacionadas en dependencia de los puntajes producidos por dichas pruebas. Algunas de las pruebas son:

- Información común en puntos (*Pointwise Mutual Information*)
- Puntaje-T (*T-score*)
- Proporción de probabilidad logarítmica (*Log-likelihood ratio*)
- Coeficiente de Dice (*Dice coefficient*)
- Proporción de probabilidades (*Odds ratio*)
- Coeficiente Phi(*Phi coefficient*)

Información común en puntos (PMI)

Es la proporción de la probabilidad de que dos eventos ocurran al mismo tiempo y la probabilidad combinada de que los dos eventos ocurran independientemente. Esta es una comparación directa de lo que es observado a lo que debería ser esperado si los dos eventos fueran independientes.

Los valores esperados se determinan como sigue:

- $m11 = \frac{np1 * n1p}{npp}$
- $m12 = \frac{np2 * n1p}{npp}$
- $m21 = \frac{np1 * n2p}{npp}$
- $m22 = \frac{np2 * n2p}{npp}$

Entonces, el valor de la información común en puntos se define como:

$$PMI = \log \frac{n11}{m11}$$

Puntaje T

El puntaje - T determina si existe alguna asociación no aleatoria entre dos palabras. El puntaje - T es definido como la proporción de la diferencia entre lo observado y lo esperado, dividida por la raíz cuadrada de lo observado.

$$T - Score = \frac{n11 - m11}{\sqrt{n11}}$$

Proporción de probabilidad logarítmica

La proporción de probabilidad logarítmica mide la diferencia entre los valores observados y los valores esperados. Es la suma de la relación de los valores observados y esperados.

$$Log - likelihood\ ratio = 2 * \sum (n_{ij} * \log \frac{n_{ij}}{m_{ij}})$$

Coeficiente de Dice

El coeficiente de Dice depende de la frecuencia de los eventos que ocurren juntos y sus frecuencias individuales.

$$Dice\ coefficient = 2 * \frac{n11}{np1 + n1p}$$

Proporción de probabilidades

La proporción de probabilidades es el cociente entre el número total de veces que un evento se produce y el número total de veces que no se produce. Es el cociente de productos cruzados de la tabla de contingencia de 2×2 y mide la magnitud de la asociación entre dos palabras.

$$Odds\ ratio = \frac{n11 * n22}{n21 * n12}$$

Coeficiente Phi

Las palabras se consideran positivamente asociadas si la mayoría de los datos pertenecen a la diagonal (es decir, si $n11$ y $n22$ son más grandes que $n12$ y $n21$) y negativamente asociadas si la mayoría de los datos caen fuera de la diagonal.

$$Phi\ coefficient = \frac{n11 * n22 - n21 * n12}{\sqrt{np1 * n1p * n2p * np2}}$$

Capítulo 3

Estado del Arte

Como se especificó en el capítulo anterior, los métodos que existen para la tarea de alineación automática de textos paralelos a nivel de palabras pueden ser clasificados de la siguiente manera:

- Métodos estadísticos de alineación (no utilizan información léxica)
- Métodos lingüísticos de alineación (que utilizan información léxica)

3.1. Métodos que no utilizan información léxica

3.1.1. Alineamiento basado en longitud: *Gale - Church*

Este método calcula $P(B_k)$ en base a la longitud (en caracteres) de las frases. El método requiere que los textos estén alineados a nivel de párrafos. Sólo contempla los alineamientos siguientes:

1:1, 1:0, 0:1, 2:1, 1:2, 2:2

Dados los textos s_1,...,s_i y t_1,...,t_j, se define el coste de su alineamiento con la siguiente distancia: $D(i, j)$.

Calculando el mínimo de esta distancia se obtendrá el mejor alineamiento.

3.1.2. Alineamiento basado en longitud: *Brown y Wu*

Brown: Una adaptación del método de Gale y Church midiendo la distancia en palabras en lugar de en caracteres.

Wu: Demuestra que la asunciones de Gale y Church no son válidas para lenguajes dispares (p.e. inglés y chino). Amplía dicho método con pistas léxicas.

3.1.3. Alineamiento basado en distancias: *Fung - McKeon*

Vector de distancias: Distancias entre las apariciones de una misma palabra.

Por ejemplo si la palabra «*calidad*» aparece en las posiciones 10, 58, 113, 181 y 214 del texto, su vector de distancias será:

(48, 55, 68, 33)

Se comparan los vectores de los dos textos. Si dos vectores son similares, se considera que las palabras pueden coincidir. Tanto este método, como el anterior, son apropiados para textos en los que los límites de las frases no están bien identificados.

3.1.4. Alineamiento basado en asociación: *Log-Likelihood-Ratio statistic*

Es una serie de variantes de métodos de alineación de palabras, basados en estadística de asociación. Todas las variantes están basadas en "log-likelihood-ratio (LLR) statistic" presentado por Dunning en 1993 [32]. Entre las diferentes variantes de los métodos se puede mencionar [33]:

- *Métodos de alineación de tipos de palabras uno a uno*: como su nombre lo indica estos métodos solo permiten relaciones entre palabras uno a uno.
- *Método 1*: Utiliza los puntajes del algoritmo LLR para ligar palabras de acuerdo al algoritmo de enlace competitivo propuesto por Melamed [34] para alinear palabras en un par de oraciones.
- *Método 2*: La desventaja del método anterior es que se hace la decisión de alineación para cada par de oraciones independientemente de las decisiones para las mismas palabras en otras oraciones. Se puede mejorar la exactitud al predisponer el método hacia enlazar palabras en una oración dada que también están ligadas en muchas otras oraciones. Una forma simple de realizarlo es llevar a cabo una segunda alineación basada en probabilidad condicional de un par de palabras siendo ligadas de acuerdo al método 1, dado que ambos ocurren en un par de oraciones.
- *Método 3*: El método anterior falla al desplegar relaciones monótonas entre llamadas y precisión, así como el puntaje del umbral aislado se ve afectado. Esto parece ser debido al hecho de que la medida LP, a diferencia de LLR, no descuenta estimaciones hechas en las bases de datos pequeños. Un método simple para compensar esta sobre confianza en eventos raros es aplicar descartes absolutos mediante una fórmula diferente a la del método 2.
- *Alineaciones permitiendo muchos a uno*: Se introduce la noción de grupos de palabras bilingües y se muestra como aplicaciones repetidas de variaciones del

método 3 pueden aprender mapeos múltiples al construir grupos incrementalmente.

- *Métodos de selección de alineación de "Tokens"*: Se cambia el problema de seleccionar el mejor token de palabra por una alineación de tipo de palabra y más generalmente a la incorporación de información posicional en alineación de palabra basado en asociaciones.

- El método más simple al elegir una alineación de token de palabra por una alineación de tipo de palabra es hacer una selección aleatoria (sin reemplazo) para cada tipo de palabra en la alineación de entre los tokens de ese tipo.

3.1.5. Modelos IBM

Los modelos de traducción dependen del concepto de alineación. Un modelo de traducción por alineación asume que la oración destino o final es generada de una oración fuente palabra por palabra.

Una palabra de la oración destino puede, por lo tanto, ser alineada con la palabra de la oración fuente que la produce. En una alineación cada palabra destino puede alinearse con sólo una palabra de la oración fuente. Hasta ahora la mayoría de los sistemas de traducción usan modelos de alineación basados en palabras ([13], [35], [36]. Brown et al. (1993) presentó cinco modelos de alineación basados en palabras [37]:

- *Modelo 1 y 2*: El modelo 2 es un modelo típico de alineación basada en palabras. Asume que una oración $e = e_1, ..., e_l$ es la fuente de un canal, recoge una cantidad m para la oración destino g con la distribución $Pr(m \mid e) = t$, donde t es un número pequeño y fijo, entonces para cada posición i $(0 < i <= m)$ en g, encuentra su posición correspondiente e_i en e de acuerdo a una distribución de alineación Pr y finalmente genera una palabra en su posición correspondiente en la oración destino. Los modelos siguientes tratan de incorporar el hecho de que diferentes palabras fuente pueden producir diferentes números de palabras destino, entre otras cosas.

- *Modelo 3*: En este modelo se presenta la distribución de fertilidad[1] $n(\phi i \mid e_i)$ para cada palabra fuente e_i en la oración fuente e, de esta manera puede ser utilizada estadísticamente para determinar el número de palabras destino que pueden ser generadas.

- *Modelo 4*: Es similar al modelo 3 excepto por la colocación de las traducciones de una palabra fuente.

En resumen se tiene:

[1]probabilidad de que una palabra en el lenguaje origen sea alineada con varias palabras en el lenguaje destino

- IBM1 - Sólo Probabilidades léxicas (*lexical probabilities only*)
- IBM2 - Diccionario más posición absoluta (*lexicon plus absolute position*)
- IBM3 - Fertilidad positiva (*plus fertilities*)
- IBM4 - Alineación de posición relativa invertida (*inverted relative position alignment*)
- IBM5 - Versión no deficiente del modelo 4 (*non-deficient version of model 4*)

3.1.6. Modelo oculto de Markov (HMM)

El modelo oculto de Markov (*Hidden Markov model - HMM*) es utilizado exitosamente en reconocimiento del habla para el problema de alineación de tiempo. La diferencia del modelo HMM en alineación del tiempo es que no hay restricción de monotonía para los posibles ordenamientos de palabras. El componente clave de este enfoque es hacer las probabilidades de alineación dependientes no en la posición absoluta de la alineación de la palabra, sino en su posición relativa.

Típicamente se tiene un fuerte efecto de localización en la alineación de palabras en textos paralelos: las palabras no están distribuidas arbitrariamente sobre las posiciones de la oración, pero cuidan el formar grupos. Las alineaciones tienen una fuerte tendencia a preservar los vecindarios locales cuando van de un lenguaje a otro. La formulación del problema es similar al problema de alineación del tiempo en el reconocimiento del habla, donde los modelos HMM han sido utilizados con gran éxito por largo tiempo. Utilizando los mismos principios básicos se puede reescribir la probabilidad al introducir los alineamientos ocultos para pares de oraciones.

El enfoque basado en HMM produce probabilidades de traducción comparables con modelos de alineación mixtos. Cuando se miran las alineaciones de posición generadas por el modelo HMM son, en general, más uniformes.

Existen algunas extensiones para mejorar los modelos estadísticos de alineación de palabras basados en HMM, dentro de las cuales podemos mencionar:

- Etiquetas POS para probabilidades de traducción (*POS Tags for Translation Probabilities*)

  En este modelo se introducen las probabilidades de traducción de etiquetas como un factor extra a las formulas de cálculo. Intuitivamente el papel de este factor es impulsar las probabilidades de traducción para palabras de partes del discurso que pueden a menudo ser traducidos. Así la distribución de probabilidad provee conocimiento a priori de las traducciones posibles de una palabra basada sólo en su parte del discurso.

- Secuencias de etiquetas para aumentar probabilidades (*Tag Sequences for Jump Probabilities*)

  Utiliza una secuencia de etiquetas para los lenguajes fuentes y destino como información de condicionamiento cuando se predice la alineación de las palabras del lenguaje destino.

- Modelado de fertilidad (*Modeling Fertility*)

  Se extiende el modelo HMM para determinar si para generar más palabras de la palabra previa o para moverse a una palabra diferente depende de la identidad de la palabra previa. Se introduce un factor P donde uno de sus parámetros es una variable aleatoria booleana y depende de la palabra previa a alinear.

- Modelo de traducción para nulos (*Translation Model for Null*)

  Requiere de muchos ajustes especiales para evitar que los modelos realicen alineaciones todo a nulo o nada a nulo [29].

3.1.7. Algoritmo de maximización de la entropía (EM)

El propósito del algoritmo Maximización de la entropía (*Entropy Maximization*) es iterar sobre una matriz reducida[1] para eliminar el ruido y mejorar los puntos de traducciones correctas [38].

Una vez realizado el proceso, se puede concluir que la diagonal principal es normalmente el punto de partida de la traducción. Aunque no puede ser aplicado a cualquier lenguaje por las diferencias que existen entre algunos de ellos, pero este método podría también funcionar para aquellos lenguajes, donde las oraciones son tratadas como simples bolsas de palabras en vez de secuencias de palabras [39].

3.1.8. Algoritmo de Melamed

Este algoritmo fue desarrollado por Dan Melamed (1997) específicamente para alinear palabras [40]. Utiliza oraciones alineadas como entrada en vez de generar los alineamientos.

La descripción general del algoritmo es la siguiente: Inicializar las probabilidades de cada par de palabras, utilizar un algoritmo de enlace competitivo para enlazar "tokens", para cada segmento, encontramos las palabras u y v de los dos textos que tienen el más alto puntaje de probabilidad entre pares de palabras en el segmento, se marcan

[1] En la matriz reducida los índices de las filas representan identificadores de cada palabra en el corpus fuente y los índices de las columnas representan los identificadores de cada palabra del corpus destino, cada celda incluye el número de veces que dos palabras ocurren en la misma oración alineada.

los tokens como enlazados en el segmento activo e incrementamos un contador de palabras enlazadas, continuamos ligando tokens en el segmento activo hasta que todas las palabras hayan sido ligadas o no existan más pares con puntajes de probabilidad. Se estiman algunos parámetros y se re-calculan las probabilidades, se repiten algunos procedimientos hasta la convergencia que está definida como la estabilidad relativa del modelo desde la última iteración [41].

3.1.9. Algoritmo K-Vec

Este algoritmo se basa en el hecho de que si dos palabras son traducciones una de la otra, entonces existen casi un número de veces igual y aproximadamente en la misma región del texto paralelo. No requiere un conocimiento previo de las puntuaciones o las fronteras de las oraciones. Por lo tanto, no está enfocado a un par de lenguajes en particular y puede trabajar para diferentes pares de leguajes donde las palabras han sido segmentadas.

Este algoritmo trata sólo con traducciones de palabra a palabra y no encuentra traducciones de frase a frase, que son aquellas donde un grupo de palabras (pudiendo ser una sola) en un lenguaje son traducciones de un grupo de palabras en otro lenguaje [31]. Este algoritmo utiliza las pruebas de asociación: Información común en un punto (*Pointwise Mutual Information*) y Puntaje - T para encontrar la similaridad entre dos palabras [42]. La descripción del Algoritmo K-Vec es la siguiente:

- El algoritmo K-Vec divide los dos textos en k piezas diferentes. Una pieza consiste en un cierto número de palabras del texto.
- Para cada palabra en ambos textos, se verifica si la palabra ocurre o existe en una pieza y representa su distribución en la forma de un vector binario k-dimensional (donde k es el número de piezas). La ocurrencia de una palabra en una pieza en particular se indica con un valor de 1 en el vector binario. El valor 0 indica que no hay ninguna ocurrencia de la palabra en dicha pieza. Cabe resaltar que no se considera la frecuencia de una palabra en una pieza en particular.

Considere un ejemplo donde el texto es dividido en 10 piezas, la palabra «*king*» ocurre 5 veces en la pieza 2 y 7 veces en la pieza 8, entonces el vector binario para la palabra «*king*» sería:

$V(king) = <0, 1, 0, 0, 0, 0, 0, 1, 0, 0>$

Se forma una tabla de contingencia de dos por dos para el par de palabras donde las dos palabras están en los textos de ambos lenguajes bajo consideración.

Considere un ejemplo del par de palabras «*king*» y «*roi*», palabras del inglés y francés respectivamente, entonces la tabla de contingencia para dichas palabras sería como lo muestra la Tabla 3.1.

Tabla 3.1: Tabla de contingencia para las palabras «*king*» y «*roi*»

	Y			
X		*roi*	!*roi*	Total
	king	a	c	$a+c$
	!*king*	b	d	$b+d$
		d	$b+d$	$T=a+b+c+d$

Entonces se utiliza una prueba de asociación para encontrar que tan cercanamente relacionadas están las dos palabras. K-Vec utiliza Pointwise Mutual Information (PMI) y Puntaje - T (T - Score) como pruebas de asociación [31].

Si el número de piezas en las cuales es dividido el texto es muy grande, entonces el número total de palabras en cada pieza es pequeño y una palabra y su traducción pueden no existir en las piezas correspondientes. K-Vec puede perder tales traducciones al buscar las traducciones de palabras en piezas correspondientes. Si el número de piezas es muy pequeño, entonces el número de palabras en cada pieza será muy grande y la ventaja básica de dividir el texto en piezas y buscar una palabra y su traducción en la pieza correspondiente se perderá [31].

Fung and Church sugieren que K-Vec divide el texto en un número de piezas igual a la raíz cuadrada del número total de tokens de palabras en el texto. Para datos enormes, el número de tokens en cada pieza será por lo tanto largo [42].

Fung an Church no consideran todos los pares de palabras de los dos textos como traducciones posibles o candidatas debido a que serían demasiados pares de palabras. Ellos restringen el algoritmo a pares de palabras con frecuencias entre 3 y 11, no consideran frecuencias bajas en pares de palabras, ya que la cantidad de información no es suficiente para encontrar una traducción.

Las palabras que producen pares de palabras con frecuencia alta en ambos lenguajes ocurrirán en casi cada pieza. El algoritmo busca correspondencias de palabras en las piezas correspondientes y por lo tanto, cada palabra con frecuencia alta en un lenguaje se considerara como traducción de palabras con frecuencia alta en otro lenguaje [42].

3.2. Métodos que utilizan información léxica

3.2.1. Alineamiento mejorado usando un modelo de diccionario simétrico

Se plantean una serie de mejoras a los modelos de alineación de IBM, así como también al modelo de alineación HMM usando un modelo de diccionario (léxico o vocabulario) simétrico. Esta simetrización no sólo lleva la dirección de traducción estándar de la fuente al destino en una cuenta, sino también la dirección de traducción inversa del destino a la fuente. Además de la simetrización, se presenta un modelo de diccionario alisado.

3.2.2. Alineamiento mejorado usando información morfo - sintáctica

Los sistemas de traducción existentes usualmente tratan diferentes derivaciones de la misma forma base como si fueran independientes una de otra. Este método propone explícitamente tomar en cuenta tales interdependencias durante el entrenamiento de modelos de alineación estadística. La evaluación es realizada al comparar las alineaciones obtenidas con una alineación de referencia comentada manualmente. De esta manera se incluyen dependencias morfo-sintácticas al entrenamiento de los modelos de alineación estadística [43].

3.2.3. Alineamiento mejorado usando correspondencias léxicas

Formulado por Kay-Roscheisen. Asume que las primeras y últimas frases del bitexto están alineadas (serán las primeras anclas) y construye un conjunto de posibles alineaciones según un criterio de distancia. En base a correspondencias léxicas detecta cuáles de las posibles alineaciones son más probables y las fija como nuevas anclas.

Las estructuras que utiliza este algoritmo son [41]:

- Tabla de oraciones alineables (*AST - Alignable Sentence Table*): Permite saber que oraciones en un lenguaje posiblemente corresponden a oraciones en otro lenguaje.
- Tabla de alineación de palabras (*WAT - Word Alignment Table*): Es una lista de pares de palabras y un puntaje que indican que tan similares son las distribuciones en su oración correspondiente.
- Tabla de alineación de oraciones (*SAT - Sentence Alignment Table*): Contiene pares de oraciones que están consideradas para alinear.

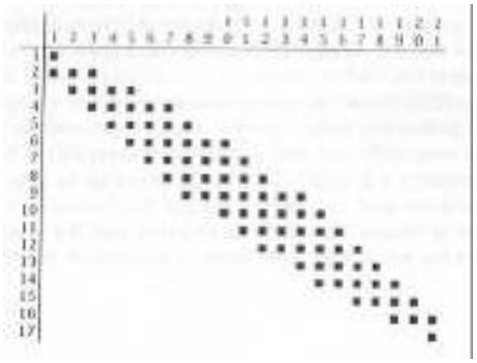


Figura 3.1: Correspondencias léxicas bilingues

El funcionamiento general del algoritmo es: Inicializar AST, calcular una WAT de AST mediante el producto cartesiano de las palabras y el cálculo de la similaridad, calcular el umbral y ordenar WAT, calcular un SAT de WAT tomando la alineación con la máxima probabilidad e iterar sobre WAT, usar las entradas de SAT como puntos reparados en la AST e interpolar entre estos puntos corregidos para producir una nueva AST y finalmente repetir algunos de los procesos hasta cubrir la SAT. Obteniendo como resultado que la SAT son las oraciones alineadas finales y la WAT son las palabras alineadas [41].

3.2.4. Alineamiento de Haruno - Yamazaki

Es una variante del método de Kay-Röcheisen pensada para lenguajes muy dispares (como inglés y japonés).

Considera que en lenguajes dispares resulta muy difícil alinear las palabras funcionales (preposiciones, artículos). Por tanto, sólo intenta alinear palabras con contenido (verbos, nombres, adjetivos).

Si los textos son pequeños, no hay contexto suficiente para aplicar las co-ocurrencias del método de Kay-Röcheisen. En ese caso, propone la utilización de un diccionario para mejorar la identificación de pares de palabras.

3.2.5. Alineamiento mejorado usando relaciones palabra - palabra

Propuesto por Chen. Maximiza la probabilidad del alineamiento, que es considerado una secuencia de beads [44]:

$$argA\ max \rightarrow A\ P(B_k)_{k=1,\ldots,m}$$

Es similar al método de los costes de Gale-Church sólo que los costes se calculan a través de un modelo de traducción basado en relaciones palabra - palabra.

3.2.6. Alineamiento con pistas

La propuesta de alineación por claves o pistas fue presentado por Tiedemann (2003). Las pistas o claves de alineación representan indicadores probabilísticos de asociaciones entre elementos léxicos acumulados de diferentes fuentes. Las pistas declarativas pueden ser tomadas de recursos lingüísticos tales como un diccionario bilingüe. Las pistas calculadas se derivan del uso de datos paralelos, por ejemplo la medida de co-ocurrencia [45].

El enfoque de alineación por pistas implementa una forma de combinar indicadores de asociación en niveles de palabras. La combinación de pistas da como resultado una matriz de pistas de dos dimensiones. Los valores en esta matriz expresan la evidencia obtenida de una asociación entre pares de palabras en segmentos de bitextos de un corpus paralelo. La alineación de palabras es entonces la tarea de identificar los mejores enlaces de acuerdo a las asociaciones indicadas en la matriz de pistas [16].

3.2.7. Alineamiento usando información morfológica

Los sistemas de traducción automática estadísticos suelen tomar como entrada un corpus bilingüe donde las oraciones correspondientes a cada idioma están alineadas. El primer paso que llevan a cabo estos sistemas suele ser la alineación palabra por palabra de los textos.

Los grupos que han desarrollado estos sistemas se han centrado en idiomas con un grado de flexión bajo o medio (por ejemplo: el inglés, el francés, el español o el chino) para desarrollarlos, por lo que la metodología suele tomar palabras flexionadas como unidad básica.

Aunque relativamente menos flexionado, el español también presenta problemas en sus verbos, donde la concordancia compleja hace que la misma raíz verbal tenga muchas formas (por ejemplo: «*vinimos*» que en inglés sería «*we come*»). Este método se centra en las diferentes posibilidades de pre-procesamiento morfológico (especialmente para el euskera, pero también para el español) con vistas a salvar la dispersión de datos y realizar un alineamiento óptimo.

El pre-procesamiento para ambos lenguajes incluye tokenización, lematización y segmentación. En la segmentación es el pre-procesamiento donde se incluye la información morfológica a utilizar. De esta manera, se preparan los textos para su posterior alineamiento a través de la utilización de un alineador como el GIZA++.

3.3. Otros métodos

Además de los algoritmos presentados en los puntos anteriores, a continuación se listan algunos otros algoritmos sobre los cuales puede ser interesante extraer características para el desarrollo de la propuesta del nuevo algoritmo:

Tabla 3.2: Otros métodos de alineación

Método o algoritmo	Autores	Referencia
Improved Discriminative Bilingual Word Alignment	Moore R., Yih W. y Bode A.	[46]
Aligning words using matrix factorization	MGoutte C., Yamada K. y Gaussier E.	[47]
Symmetric Word Alignments for Statistical Machine Translation	Matusov E., Zens R. y Ney H.	[48]
High-Performance Bilingual Text Alignment Using Statistical and Dictionary Information	Haruno M. y Yamazaki T.	[49]
Discriminative Word Alignment with Conditional Random Fields	Blunsom P. y Cohn T.	[50]
Improving Statistical Word Alignment with a Rule-Based Machine Translation System	Hua W. y Haifeng W.	[51]
A Simple Hybrid Aligner for Generating Lexical Correspondences in Parallel Texts	Ahrenberg L., Andersson M. y Merkel M.	[52]
Improving Word Alignment Quality using Morpho-syntactic Information	Popovic M. y Ney H.	[43]
Semi-Supervised Training for Statistical Word Alignment	Fraser A. y Marcu D.	[53]
Alignment by Agreement	Liang P., Taskar B. y Klein D.	[54]
Using Information about Multi-word Expressions for the Word-Alignment Task	Venkatapathy S. y Joshi A.	[55]

3.4. Métodos de cognación

Los cognados son un recurso para el reforzamiento de las alineaciones. Es por ello, que resulta importante revisar los métodos de cognación que han sido propuestos.

3.4.1. Método de truncamiento

Es también conocido como "*Condición de Simard*". El primer paso en el método de cognación propuesto por Simard et al. (1992) consiste en verificar que cada token en el par que se analiza pertenezca a una de las siguientes categorías [19]:

1. w está compuesta únicamente por letras y dígitos, pero contiene al menos un dígito.

2. w está compuesta exclusivamente por letras y tienen una longitud mínima de cuatro letras.

3. w es un signo de puntuación único.

La primera categoría se destina a la captura expresiones numéricas, que en la mayoría de los casos son independientes del lenguaje y se preservan en las traducciones. La segunda categoría se define a fin de excluir a la mayoría de las palabras funcionales, pues éstas tienden a ser de longitud cortas. Por último, se incluye la tercera categoría con la intuición de que el proceso de traducción tiene una tendencia a conservar los signos de puntuación.

Una vez verificadas las categorías, para que el par de tokens se considere cognado se debe cumplir que:

- Si ambos tokens son miembros de las categorías 1 o (3), deben ser completamente idénticos.
- Si ambos son miembros de las categorías (2), deben coincidir en los primeros cuatro caracteres.

De acuerdo a lo anterior, los pares de palabras «*capacity, capacidad*» y «*analysis, análisis*», son cognados.

3.4.2. Subsecuencia común más larga

El $LCSR$ (por sus siglas en inglés de *Longest Common Subsequence Ratio*) de dos tokens, es el cociente de la longitud de la subsecuencia común más larga (no necesariamente contigua) y la longitud del mayor token [30]. En símbolos:

$$LCSR(A, B) = \frac{longitud[LCS(A, B)]}{max[longitud(A), longitud(B)]}$$

Por ejemplo, el token «*sistema*», cuya longitud es de 7 caracteres, posee 5 caracteres que aparecen en el mismo orden en «*system*». Así, el $LCSR$ para este par de palabras es de $5/7 \approx 0{,}71$. Mientras que el $LCSR$ de «*meteorite*» y «*mete*» es sólo de $4/9 \approx 0{,}44$, siendo que este último hubiera sido considerado como cognado si el análisis de cognación se hubiera realizado empleando el método de truncamiento.

3.4.3. Coeficiente de Dice

El coeficiente de Dice ha sido empleado en muchas aplicaciones para recuperación de información clásica y como medida de solapamiento. Cuando es utilizado como medida de similitud entre dos cadenas de texto, este coeficiente puede ser calculado como:

$$s = \frac{2|bx \cap by|}{|px| + |by|}$$

Donde bx es el número de bigramas en la cadena x y by, el número de bigramas en la cadena y [56].

Por ejemplo, las cadenas «*diccionario*» y «*dictionary*» se pueden dividir en los bigramas {*di, ic, ct, ti, io, on, na, ar, ry*} y {*di, ic, cc, ci, io, on, na, ar, ri, io*} respectivamente. Por tanto, comparten los bigramas {*di, ic, io, on, na, ar*}. De esta forma, el valor del coeficiente de Dice para este par es 2(6)/19≈0.63.

En general se define una trasformación al intervalo [0,1], donde 0 indica que no hay coincidencia alguna entre los bigramas y que las cadenas son iguales.

3.4.4. Distancia de Levenshtein

La distancia de Levenshtein para el reconocimiento de cognados fue propuesta por Mann y Yarovsky (2001). Se refiere al número de operaciones que se requieren para transformar una cadena de texto inicial, en una final. Se entiende por operación, bien una inserción, eliminación o la sustitución de un carácter [57].

Por ejemplo, transformar «*edificio*» a «*edifice*», se realizan los siguientes cambios:

1. sustitución de *i* por *e* → *edificio* - *edificeo*
2. eliminación de *o* → *edificeo* - *edifice*

Por tanto, la distancia de Levenshtein es 2, pues se requirieron dos ediciones elementales para cambiar uno en el otro.

3.4.5. Métrica de penalización

Covington (1996) propone un algoritmo de búsqueda guiada para encontrar la mejor alineación de una palabra con otra, cuando se proporciona la trascripción fonética de ambas. Las entradas del alineador deben estar en su transcripción fonética amplia, usando símbolos con valores muy similares en ambos lenguajes. Las transcripciones fonéticas excesivamente particulares no ayudan. Los segmentos de dos palabras pueden estar mal alineados a causa de afijos, reduplicación y cambios de sonido, tales como la elisión o monoptongación, que alteran el número de segmentos [58].

En cada paso, el alineador puede realizar un emparejamiento (sin importar si los segmentos son o no fonológicamente similares) o un salto. Por ejemplo, la alineación:

$$\begin{array}{cccc} a & b & c & - \\ - & b & d & e \end{array}$$

es producida por saltarse la a, a continuación emparejar b con b y c con d y finalmente, saltar e. Cualquier cadena de guiones corresponden a saltar segmentos en el otro.

Para identificar la mejor alineación, el algoritmo debe asignar una penalización (coste) a cada salto o emparejamiento. La mejor alineación es la que tiene la menor penalización. La Tabla 3.3 muestra las penalizaciones otorgadas por Covington en su algoritmo.

Tabla 3.3: Métrica de evaluación utilizada en el algoritmo de Covington

Penalización	Condiciones
0	Coincidencia exacta de consonantes
5	Coincidencia exacta de vocales
10	Emparejamiento de dos vocales que difieren sólo en la longitud
30	Emparejamiento de dos vocales diferentes
60	Emparejamiento de dos consonantes diferentes
100	Emparejamiento de dos segmentos con ninguna similitud apreciable
40	Salto precedido por otro salto en la misma palabra
50	Salto no precedido por otro salto en la misma palabra

3.4.6. Distancia dialectal

El algoritmo propuesto por Nerbonne and Heeringa (1997) consiste en la sustitución de cada símbolo fonético por un vector de características [59]. Así, si se comparan lenguas, sobre la base de los símbolos fonéticos, se puede tener en cuenta la afinidad entre sonidos que no son iguales, pero que sí están relacionados por medio del conjunto de características desarrolladas por Vieregge, ACMRietveld, y Jansen (1984) [60].

Se comparan tres métodos para medir la distancia fonética: (1) distancia Manhattan o city block, (2) distancia euclidiana y (3) coeficiente de correlación de Pearson. La distancia Manhattan es simplemente la suma de todas las diferencias en los valores de las características, para cada una de las 14 definidas en el vector:

$$\delta(X, Y) = \sum_{i=1}^{n} |X_i - Y_i|$$

La distancia Euclidiana se emplea tal y como es definida, como la raíz cuadrada de la suma de los cuadrados de las diferencias de los valores de las características:

$$\delta(X,Y) = \sqrt{\sum_{i=1}^{n}(X_i - Y_i)^2}$$

Por último se utiliza $1 - r$ para interpretar la distancia, donde r es el habitual coeficiente de Pearson definido como:

$$r = \frac{1}{n-1}\sum_{i=1}^{n}(\frac{x-\overline{x}}{s_x})(\frac{y-\overline{y}}{s_y})$$

3.4.7. ALINE

Los métodos anteriores están basados en características binarias, que a pesar de ser muy utilizadas, no son óptimas para la alineación fonética. Su principal motivación es clasificar oposiciones fonológicas en lugar de reflejar las características fonéticas de los sonidos.

Kondrak (2000) propone un método que utiliza el sistema de características multivaluadas presentado por Connolly (1997). Éste contiene cerca de veinte características con valores entre 0 y 1; algunas de ellas con hasta diez valores diferentes (por ejemplo, [lugar]), mientras que otras son básicamente oposiciones binarias (por ejemplo, [nasal]). La Tabla 3.4 contiene ejemplos características multivaluadas [61].

Kondrak además incorpora pesos para expresar la relación de importancia de las características individuales [62]. Los valores de estos pesos se muestran en la Tabla 3.5.

Para encontrar un par de cognados, el algoritmo se basa en la noción de semejanza. El esquema de puntuación de similitud asigna valores positivos grandes a los pares de segmentos relacionados; valores negativos grandes a segmentos diferentes y puntuaciones negativas pequeñas a operaciones de inserción y borrado. La alineación óptima es la que maximiza la puntuación global.

Tabla 3.4: Algunas características multivaluadas

Característica	Término fonológico	Valor numérico
Lugar	bilabial	1.00
	labiodental	0.95
	dental	0.90
	alveolar	0.85
	retrofleja	0.80
	palato-alveolar	0.75
	palatal	0.70
	velar	0.60
	uvular	0.50
	faríngeo	0.30
	glotal	0.10
Manera	parada	1.00
	africada	0.90
	fricativa	0.80
	aproximada	0.60
	vocal alta	0.40
	vocal media	0.20
	vocal baja	0.00
Altura	alto	1.00
	media	0.50
	baja	0.00
Back	delantero	1.00
	central	0.50
	trasero	0.00

3.5. Alineadores

3.5.1. PLUG Word Aligner

El sistema UPLUG es un sistema modular diseñado para procesar corpus de texto con enfoque en textos paralelos y alineamiento de palabras. Incluye tres componentes principales:

- *UplugIO* - Interfaz transparente para la lectura y escritura de datos secuencial en diferentes formatos.
- *UplugSystem* - Un administrador del subsistema para combinar subtareas en procesos iterativos y secuenciales.
- *UplugGUI* - Una interfaz grafica para ejecutar y ajustar los subsistemas de Uplug.

También contiene dos administradores, uno para corpus (*Corpus Manager*) encargado de administrar el repositorio de Corpus y el otro para administrar las tareas (*Task Manager*) que permite ejecutar aplicaciones sobre los corpus.

Tabla 3.5: Características usadas en ALINE y sus pesos

Característica	Peso
Silábico	5
Voz	10
Lateral	10
Altura	5
Manera	50
Longitud	1
Lugar	40
Nasal	10
Aspiración	5
Back	5
Retroflexión	10
Vuelta	5

Este sistema tiene varias herramientas integradas como:

- *Pre-procesamiento*: Incluye separador de oraciones, un tokenizador, etiquetador externo y un parser.
- *Herramientas externas*: Etiquetador de árboles para diferentes idiomas, un etiquetador TnT, el sistema Grok (para etiquetar y separar) y un analizador morfológico.
- *Sistema de alineación para oraciones* (mediante la aproximación basada en longitud propuesta por Gale and Church) y para palabras y frases (mediante la aproximación de alineamiento por pistas - clue alignment - y el conjunto de herramientas de GIZA++)

El sistema para la alineación contiene varios módulos para procesar datos textuales. La aplicación principal es el alineador de palabras PLUG (*PLUG Word Aligner - PWA*), el cual integra al alineador de palabras Linköping (*Linköping Word Aligner - LWA*) y al alineador de palabras Uppsala (*Uppsala Word Aligner - UWA*). Ambos sistemas de alineamiento están diseñados para ser modulares y procesar bitextos paralelos en varios pasos o etapas [63].

3.5.2. PWA

PWA es una colección de herramientas para la alineación automática de correspondencias de palabras en textos paralelos bilingües. El sistema integra un conjunto de módulos para la alineación de palabras, con distintas posibilidades de cambiar la configuración y adaptar el sistema a otros pares de lenguajes y tipos de texto. El sistema requiere bitextos alineados por oraciones como entrada y produce una lista de

correspondencias de palabras y frases en el texto, además de un diccionario bilingüe de esas instancias [64].

PWA comprende dos sistemas de alineación: el *Linköping Word Aligner* (LWA) y el *Uppsala Word Aligner* (UWA). Ambos desarrollados dentro del proyecto de texto paralelo PLUG. El sistema fue implementado en el Departamento de Cómputo y Ciencias de la Información de la Universidad de Linköping en Suecia.

PWA se encuentra disponible para la comunidad investigadora mediante un acuerdo con los propietarios y tiene dos versiones para que funcione bajo los sistemas Linux y Windows. Cuenta también con un sistema para la evaluación automática de los resultados del alineamiento conocido como PLS (*PLUG Scorer*) [64]. La Figura 3.2 muestra el sistema UPLUG funcionando en Web.

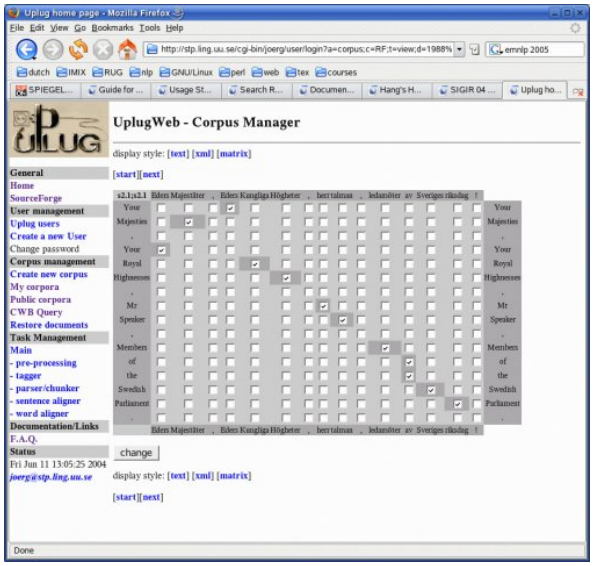


Figura 3.2: Pantalla de ejecución del sistema Uplug en Web

3.5.3. GIZA y GIZA++

GIZA es un programa de entrenamiento que enseña algunos de los modelos de traducción estadística para corpus bilingües. Está escrito en C++ con la librería STL. GIZA++ es una extensión desarrollada por el equipo de traducción máquina estadística en 1999 en el Centro de Lenguaje y Procesamienro del Habla en la Universidad Johns-Hopkins.

La extensión permite utilizar los modelos 4 y 5, modelos de alineación dependientes de clases de palabras, implementa el modelo de alineación HMM, incluye variantes del modelo 3 y 4 de IBM, entre otras cosas [65].

3.5.4. TREQ y TREQ-AL

El sistema TREQ requiere un texto paralelo alineado a nivel de oraciones, segmentado, etiquetado y lematizado. El objetivo es obtener equivalencias de traducción del texto o, expresado en otras palabras, un diccionario bilingüe basado en el texto. El algoritmo hace uso de dos suposiciones:

1. Un token léxico es traducido sólo por un token en el otro lenguaje.

2. Después de la etapa de cálculo de puntajes, las unidades de traducción son inspeccionadas de nuevo, una por una, y para cada una de ellas, los candidatos con el puntaje más alto son seleccionadas como equivalencias de traducción siempre y cuando éstos sean más altos que un umbral de confianza (empíricamente colocado en 9).

El sistema TREQ-AL toma como entrada el diccionario generado por TREQ y el texto paralelo a ser alineado a nivel de palabra. El alineamiento es expresado en términos de posición, las palabras están representadas por su posición en la unidad de traducción, por separado de cada lenguaje.

Para obtener su resultado, TREQ-AL realiza una serie de procesos reiterados con cada unidad de traducción, que son los siguientes [66]:

- Búsqueda en el diccionario generado por TREQ (*Dictionary looking-up*)
- Alineación de arriba hacia abajo (*Up-bottom alignment*)
- Alineación de abajo hacia arriba (*Bottom-up alignment*)
- Zonas de alineación(*Alignment zones*)
- Alineación de palabras final (*The final word-alignment*)

3.5.5. Alineador de la IULA

El alineador que se ha desarrollado en el Instituto Universitario de Lingüística Aplicada de la Universidad Pompeu Fabra permite, tanto alinear frases, como también llevar a cabo una alineación léxica. A diferencia de otras herramientas de alineación, basadas en criterios estadísticos, el alineador del IULA toma las decisiones basándose en la utilización de información lingüística añadida (lemas y etiquetas), es decir, es un programa dependiente de herramientas de marcaje lingüístico para las lenguas que se quieran alinear [67], [68], [69].

En este caso, funciona a partir de textos procesados con la cadena de herramientas del Corpus Técnico del IULA en inglés, catalán y castellano, ya que son las lenguas de las que se dispone actualmente de marcaje lingüístico.

Esto implica que el alineador no compara únicamente la forma de la lengua A con la forma de la lengua B, sino también, sus lemas y etiquetas morfológicas. Así pues, para establecer un par léxico, se da prioridad primero a la coincidencia de lemas, después a la coincidencia de etiquetas y finalmente, a la coincidencia ortográfica [70].

El programa trabaja en tres niveles: a) nivel de palabra, b) nivel de frase, y c) nivel de documento. En el primer nivel, se mide el grado de similitud de dos palabras; en el segundo nivel, se globalizan los resultados del nivel de palabra para cada pareja de frases; finalmente, en el tercer nivel, se establece una estrategia para decidir qué frase de la lengua A es comparada con qué frase de la lengua B, siguiendo los modelos clásicos de cálculo de longitud de frase, de número de palabras y de posición en el documento.

El resultado es una versión hipertextual de la alineación de dos textos. En el curso de desarrollo del alineador se ha demostrado que el programa mejoraba notablemente su eficiencia si era capaz de aprender de sus alineaciones, materializándose el aprendizaje en un diccionario de pares léxicos pre-alineados generado a partir de la alineación léxica.

En principio, el programa al alinear dos textos genera un diccionario automáticamente. Este diccionario se crea a partir de la indexación y acumulación de todas las alineaciones léxicas hechas por el ordenador. Una vez alineados los textos y generado el diccionario, entramos en una base de datos los datos necesarios ordenados por la "Probabilidad Biléxica Decreciente"[2] para que el traductor pueda proceder al análisis de los pares léxicos y de los textos alineados, y así poder hacer una validación, tanto de la alineación léxica, como del funcionamiento general del programa [70].

3.5.6. Clue Aligner

Este alineador combina diversas pistas de asociación. Inspirado por el enfoque de alineación de palabras "avaro" del UWA (Uppsala Word Aligner), utiliza recursos estadísticos y conocimiento lingüístico, el cual es tratado como pistas a relaciones entre palabras y frases. La información lingüística, con características contextuales, pueden señalar correspondencias de traducción entre palabras en bitextos [15].

[2]PBD - Técnica para ordenar los pares léxicos según la probabilidad de que fueran erróneos con el conjunto de informaciones generadas por el alineador.

3.5.7. Alineador Twente

El alineador Twente es un alineador de palabras desarrollado por Djoerd Hiemstra (1998) para el proyecto Twente-One[3]. Este alineador fue desarrollado para extraer terminología del corpus paralelo "Agenda 21" para ayudar en la recuperación de información en lenguajes cruzados. Es una alineador de fuente abierta, con licencia GPL, haciéndolo un buen punto de comienzo ya que es posible estudiar el código, experimentar y aplicar reingeniería sobre él [71].

3.5.8. La arquitectura NATools

El alineador Twente fue sometido a reingeniería para solventar algunos de los inconvenientes que presentaba, dando como resultado el desarrollo del alineador de palabras NATools, que es más robusto, escalable y rápido. Este alineador está basado en métodos estadísticos y conteo de co-ocurrencias de palabras de cada lenguaje. Para poder realizar su tarea involucra los siguientes pasos [39]:

- *Pre-procesamiento* (pre-process): Permite limpiar y preparar los corpus. El proceso es dependiente del lenguaje e incluye toquenizacion (segmentación) y otras formas de pre-procesado para mejorar los resultados de alineación.
- *Codificación* (encode): Codifica los corpus en formato binario, donde cada palabra es representada por un entero. También crea un diccionario con una bisección entre palabras e identificadores.
- *Creación de la matriz* (mkMatrix): Prepara una matriz reducida con co-ocurrencias de palabras, usadas en el proceso de alineación estadística.
- *Algoritmo EM*: La meta principal es transformar la matriz reducida, disminuyendo el ruido y mejorando las celdas de traducción de palabras.
- *Creación del diccionario* (mkDictionary): El paso final en el proceso de alineación es interpretar la matriz resultante, utilizando las tablas de diccionario originales y la matriz mejorada para crear un par de diccionarios.

3.5.9. Alpaco

Desarrollado en la Universidad de Minnesota por Brian Rassier y Ted Pedersen, es un programa diseñado para alinear textos paralelos. Si dos archivos son traducciones uno del otro, Alpaco (*Aligner for Parallel Corpora*) puede utilizarse para alinearlos manualmente a nivel de palabras o frases y guardar los alineamientos para futuras referencias.

[3]Proyecto para poner a las organizaciones del medio ambiente, de investigación y algunas compañías en la creación, distribución y uso de documentos de interés común sobre ecología y desarrollo sostenible.

Puede tomar como entrada archivos en modo crudo, con el formato Blinker data o con el formato de Alpaco. Alpaco fue desarrollado con Perl y TK (Kit de herramientas para interfaces de usuario gráficas) [72].

3.5.10. K-Vec++

Desarrollado en la Universidad de Minnesota por Ted Pedersen y Nittin Varma, es una implementación del algoritmo K-Vec [42] que encuentra correspondencias entre palabras, las cuales son pares de palabras o frases que son traducciones exactas de un lenguaje a otro. El algoritmo encuentra pares de palabras en textos paralelos.

También incluye un programa para combinar la salida de diferentes técnicas y producir resultados mejores que cualquier método individual, además de tener un programa para evaluar la calidad del diccionario creado [31].

3.5.11. Interactive Clue Alignment

ICA (*Interactive Clue Alignment*) es una interfaz web basada en PHP para la alineación de palabras interactiva. Utiliza para su funcionamiento el alineador de pistas (Clue Aligner), pero puede ser empleado para realizar alineaciones de manera manual, seleccionar pistas y sus pesos, inspeccionar estrategias de alineación, corregir alineamientos agregando y removiendo enlaces, desplegar el contenido de la base de datos de puntajes de pistas, entre otros. Funciona en un par de oraciones a la vez, tomado de un texto paralelo predefinido [73]. La Figura 3.3 muestra su interfaz web.

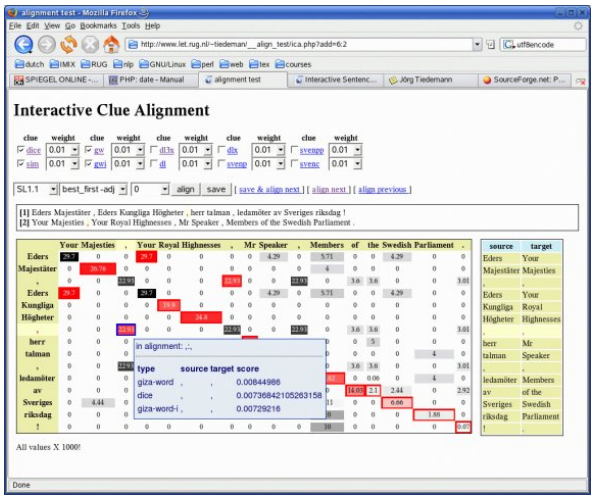


Figura 3.3: Pantalla de ejecución del sistema ICA en Web

3.5.12. Otros alineadores

Dentro de los alineadores de palabras existentes además de los presentados en puntos anteriores, podemos mencionar:

Alineador	Autores	Referencia
Knowledge intensive word alignment with KNOWA	Pianta E. y Bentivogli L.	[74]
The DuluthWord Alignment System	Thomson B. y Pedersen T.	[75]

Capítulo 4

Metodología Propuesta: Alineador de palabras HWA

Existen diferentes métodos de alineación a nivel de palabras. Algunos de ellos involucran modelos generativos probabilísticos, los cuales son complejos para implementar y lentos para entrenar. Otras aproximaciones son más simples y se basan en asociaciones estadísticas de palabras.

Todos los métodos y sus variantes presentan ventajas y desventajas según su aplicación y su entorno de trabajo. Pero todavía no se ha podido determinar cuál es el óptimo, debido principalmente a las diferencias entre los lenguajes, los recursos disponibles y por las consideraciones que se deben hacer por el dominio, ya sea general o específico de un corpus.

Esta propuesta presenta el alineador HWA (Hybrid Word Aligner). El algoritmo, para dar solución a la metodología planteada es altamente configurable y flexible, lo cual le permite adaptarse al entorno en el que se ejecuta y a los recursos disponibles. Además combina diversas técnicas, tanto estadísticas como lingüísticas, para realizar la tarea de alineación.

El algoritmo implementado para HWA trata de unificar los enfoques estadísticos y lingüísticos, con el propósito de obtener un algoritmo adaptable y parametrizable. Lo cual le permitirá ser utilizado de diferentes formas en dependencia de la exigencia de los resultados y los recursos con los que se cuente.

Una de las ideas interesantes, además de la forma de combinar los enfoques, es la identificación de los dominios semánticos de las palabras para apoyar la tarea de alineación de palabras. Este es un concepto poco utilizado, y cuando se usa es para ubicar o entrenar al alineador en un domino específico [76]. La idea propuesta analiza los dominios posibles para pares de palabras y reafirma el resultado de la alineación por coincidencia, en caso contrario reduce el nivel de confianza.

El sistema consta de cuatro módulos principales:

1. *Módulo de pre - procesamiento*: Encargado de obtener las entradas y extraer la información necesaria para el funcionamiento del algoritmo.

2. *Módulo de configuración*: Permite seleccionar las técnicas y recursos que utilizará el algoritmo para realizar la tarea de alineamiento.

3. *Módulo de resolución*: Realiza el procesamiento de la información y aplica los métodos requeridos para obtener las correspondencias entre las palabras de los textos paralelos.

4. *Módulo de resultados*: Almacena y presenta los resultados obtenidos por el algoritmo, también genera un diccionario bilingüe.

En la Figura 4.1 se puede observar la arquitectura general del alineador HWA.

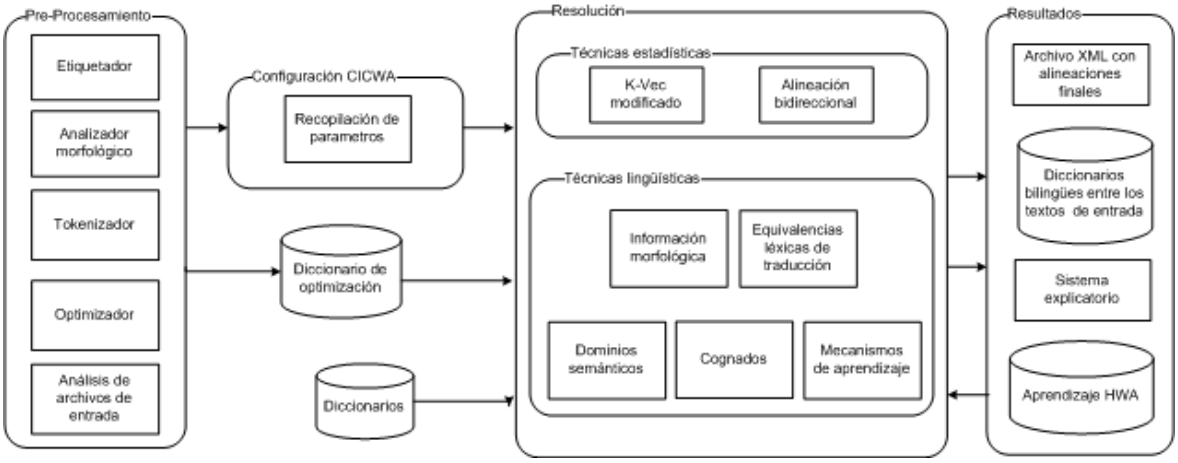


Figura 4.1: Arquitectura general del alineador HWA

4.1. Módulo de pre - procesamiento

Antes de efectuar la alineación, independientemente de la(s) técnica(s) empleada(s), es preciso pre-procesar los textos, logrando con esto un tratamiento más eficiente y complejo. Los textos crudos en HWA deben pasar a través de los siguientes procesos:

1. Análisis de los archivos de entrada

2. Optimización

3. Tokenización

4. Análisis morfológico

5. Etiquetado

Estas etapas son comúnmente conocidas como pre-procesamiento. Una vez que se concluyen se da paso a la aplicación del algoritmo, el cual identifica los pares de palabras alineadas. Los resultados arrojados por el algoritmo son almacenados en un archivo etiquetado con información de los enlaces entre las palabras del archivo en lenguaje 1 y las palabras del archivo en lenguaje 2.

4.1.1. Análisis de los archivos de entrada

Verifica la existencia, validez y formato de los archivos que el usuario define como entrada para el alineador.

El algoritmo preferentemente necesita como entrada textos paralelos alineados a nivel de sentencias, pero está en la capacidad de recibir textos alineados a nivel de párrafos o sin alineación.

Lo anterior puede tener impacto en el resultado, pero éste podrá ser minimizado según las opciones de configuración que se introduzcan en el algoritmo del HWA. Por lo tanto, la tarea importante en estos casos será la correcta configuración de los parámetros de entrada.

4.1.2. Optimización

Durante la fase de optimización se genera una lista de palabras para cada uno de los archivos, incluyendo la frecuencia de aparición. Esto es útil para optimizar el preprocesamiento, ya que en tareas como la lematización para cada palabra con múltiples apariciones, sólo se ejecutaría una búsqueda.

4.1.3. Tokenización

Una definición computacional aproximada para "palabra" es, respecto a textos escritos, cualquier elemento separado por espacios. Sin embargo, este concepto no es utilizable en lenguaje natural, ya que existen elementos que no representan una única palabra sino dos o más. Por ejemplo, en inglés, «*he's*» se utiliza para representar «*he is*». Sin duda este tipo de conjunciones existen en diferentes lenguajes y deben ser separadas (insertando espacios en blanco) para un buena interpretación.

Por otro lado, hay muchas muestras léxicas multipalabra ("tokens") en diferentes lenguajes, que poseen un solo significado léxico y que deben ser tratadas como una unidad compuesta y marcadas como tal. Por ejemplo, algunos verbos en inglés como

«*cut off*» o frases gramaticales como «*in despite of*». Para tratar estos casos, usualmente se remplazan los espacios con el guión bajo («*in despite of*» por «*in_despite_of*»).

Esta tarea es llamada segmentación (o "tokenización") y el programa que lo realiza segmentador (o "tokenizer"). Para los propósitos de traducción y alineación a nivel de palabras, es importante una buena delimitación de muestras léxicas, donde "palabra" tendrá sólo un significado de muestra léxica [66].

4.1.4. Análisis morfológico

Esta fase se encarga básicamente de la lematización de las formas de palabras, tanto para el español, como para el inglés. La lematización consiste en la reducción de las palabras de un corpus a sus correspondientes formas básicas (formas normales o lemas). Si el corpus está lematizado, el usuario puede examinar todas las variantes de una palabra sin necesidad de buscar explícitamente todas las variantes, así como extraer toda la información sobre frecuencia y distribución de una determinada palabra.

Para la lematización del español se empleó un archivo con información morfológica, en el que cada entrada posee su lema correspondiente y determinadas propiedades en función del POS ("*part of speech*"). Por ejemplo, para los verbos incluye el modo, tiempo gramatical, persona y número.

La siguiente, es la salida que se obtiene para la palabra de entrada «*río*». Como se muestra, en esta fase se pueden extraer varios lemas para una misma forma (homógrafos[1]).

río NCMS000 *río*
río VIIP1S0 *reír*

El archivo utilizado fue desarrollado en Laboratorio de Lenguaje Natural y Procesamiento de Texto del CIC y constituyó el fundamento para la construcción de una base de datos morfológica, cuya composición se resume en la Tabla 4.1.

Tabla 4.1: Composición de la base de lemas para el español

Clase gramatical	Formas	Lemas
Sustantivos	28446	15543
Adjetivos	16456	4947
Verbos	893789	3457

[1]Palabras que se escriben de forma idéntica pero tienen diferentes significados, es decir, tienen el mismo significante pero distinta etimología, por tanto, distinto significado.

Para la lematización de las palabras del idioma inglés, se utilizaron reglas de separación y listas de excepciones. Las reglas de separación indican la terminación que deberá asignarse, para cada sufijo en función del POS. Por ejemplo, se deberá eliminar la *-s-* de aquellos sustantivos que tengan esta consonante como sufijo («*cars*» y «*car*») y deberá remplazarse con una *-s-* aquellos que terminan con *-ses-* («*buses*» y «*bus*»). Éstas y un subconjunto de las reglas de separación se muestran en la Tabla 4.2.

Tabla 4.2: Reglas de separación para la lematización del inglés

Sustantivos		Adjetivos		Verbos	
Sufijos	Terminación	Sufijos	Terminación	Sufijo	Terminación
...s	...	...er	...e	...s	...
...ses	...s	...er	...	...ies	...y
...xes	...x	...est	...e	...es	...e
...zes	...z	...est	...	...es	...
...ches	...ch			...ed	...e
...shes	...sh			...ed	...
...men	...man			...ing	...e
...ies	...y			...ing	...

Si no se puede aplicar una regla de separación, la entrada a lematizar se busca cada una de las tres listas de excepción (una por categoría gramatical).

La base de datos morfológica para el idioma inglés se llenó con las entradas de los archivos de excepciones y de formas regulares y su composición es mostrada en la Tabla 4.3.

Tabla 4.3: Composición de la base de lemas para el inglés

Clase gramatical	Formas	Lemas
Sustantivos	116600	115253
Adjetivos	22910	21666
Verbos	13684	11604

Cabe mencionar que los resultados de este proceso se almacenan en estructuras dinámicas. Las cuales pueden ser consultadas en cualquier momento por los módulos correspondientes en la fase lingüística del algoritmo de alineación.

4.1.5. Etiquetado

Una vez que se tiene la información se pasa a un etiquetador que genera una cadena con todos los datos requeridos y la guarda en un archivo en formato XML, para su uso

posterior por el alineador HWA.

El etiquetado es la adición de etiquetas con información extra, como la categoría léxica de la palabra. Este es el tipo más básico de anotación. Sirve para facilitar la búsqueda, distinguiendo entre homógrafos («*coma*», «*prueba*», «*vino*», etc.).

El etiquetado de un corpus de tamaño considerable, sólo es factible si se hace automáticamente mediante programas especializados (taggers). Esto garantiza no sólo velocidad, sino también consistencia en los resultados del etiquetado.

Además de etiquetar con información léxica, también se puede encargar de la información morfológica y de resolver casos de ambigüedad si es necesario, en dependencia de la utilización que se le dará a dicha información.
El etiquetador tiene dos métodos principales:

1. *Generar una cadena*: Permite generar una cadena de etiquetas XML a partir de la información que se establezca. Por medio de esta función se crean los archivos de etiquetas XML.

2. *Interpretar una cadena*: Permite comprender una cadena con etiquetas XML, para comprender el significado de cada etiqueta y su valor asociado. Por medio de esta función se leen los archivos de etiquetas XML.

Además de trabajar en el pre-procesamiento, el etiquetador también realiza tareas en la fase de de implementación del algoritmo y la de resultados.

La Figura 4.2 muestra dos ejemplos de etiquetas, en el inciso A) se muestra la etiqueta generada para la palabra «*es*» y su información asociada: lema = *ser*, información morfológica = VIIP3S0 y posición en el texto en formato párrafo.oración.palabra = 1.1.2. En el inciso B) se presenta una etiqueta generada para el par de palabras alineadas «*es, is*» y su información relacionada: probabilidad = *0.184957890625*, posiciones en los textos en formato párrafoOrigen.oraciónOrigen.palabraOrigen ; párrafoDestino.oraciónDestino.palabraDestino = *w0.1.1;w0.1.1* y la explicación del origen del enlace = *Probabilidad*.

A) `<w lem="ser" pos="VIIP3S0" id="w1.1.2">es</w>`

B) `<wordlink certainly="0.184957890625" lexpair="es;is" xtargets="w0.1.1;w0.1.1" explanation="Probabilidad">`

Figura 4.2: Ejemplos de etiquetas

4.2. Módulo de configuración

La configuración del algoritmo es muy importante debido a que permite seleccionar los parametros y recursos con los que se realizará el proceso de alineación. La mayoría de los alineadores actuales no tienen una interfaz o módulo que permita realizar la configuración de manera sencilla. El contar con un módulo que pueda cambiar fácilmente la forma de trabajo del algoritmo permite facilmente probar distintas variantes en el proceso de obtención de alineaciones.

La amplia flexibilidad y la complejidad asociada a los aspectos que puede manejar el algoritmo genera una pregunta: ¿Cuál es la mejor configuración del algoritmo?

Para responder se necesita de un buen análisis ya que no existe respuesta obvia ni constante. El entorno computacional en el que se desenvuelve el algoritmo y los recursos con los que cuenta, son aspectos de gran impacto durante el análisis.

El entorno computacional se refiere a la rapidez requerida de respuesta, las características de la computadora en las que se ejecuta el algoritmo, etc.

En cuanto a los recursos lingüísticos, aunque el algoritmo posee algunos o puede incorporarlos de un archivo, el usuario puede prescindir de ellos en cualquier momento y de esta manera limitar los recursos durante el procesamiento. Algunas de las opciones son: uso de diccionarios, identificación de cognados, uso de synsets, dominios semánticos, etc.

El último aspecto que afecta el desempeño son los textos a alinear, su tamaño, distribución de palabras, entre otras cosas, afecta el éxito de los enlaces de palabras que obtiene el algoritmo, por lo tanto se debe tratar de reconocer ciertos patrones o entornos de funcionamiento, que identifiquen la configuración óptima del algoritmo, basados en experiencias anteriores y en las opciones configuradas para el proceso de alineación.

4.3. Módulo de resolución

El algoritmo basa su funcionamiento en técnicas estadísticas y específicamente está inspirado en el algoritmo K-Vec [42].

La simplicidad del enfoque estadístico del K-vec, permite obtener hipótesis de alineación iniciales. Sin embargo, la diversidad de las reglas gramaticales que poseen los idiomas provoca múltiples errores en las asociaciones derivadas. Es por ello que, en una fase posterior se incorpora información lingüística con el objetivo de reforzar o debilitar dichas alineaciones [77].

En resumen, se emplean dos enfoques en la fase estadística:

- *K-Vec modificado*: La técnica estadística K-Vec, variando algunas consideraciones como los tamaños de los vectores, la forma de considerar los conteos en dichos vectores, etc.

- *Alineación bidireccional*: Considera aplicar una segunda aplicación del algoritmo pero con la diferencia de que el texto origen se toma en cuenta como texto destino y el texto destino como origen.

Las técnicas lingüísticas se basan en enfoques que han sido propuestos en trabajos previos y algunos formulados específicamente para este trabajo, dentro de los cuales se encuentran:

- *Uso de información morfológica*: Es utilizada para comparar lemas, verificar categorías gramaticales, etc., y mediante eso poder aumentar la probabilidad de alineación de los pares de palabras. Este método ha sido utilizado en distintas aproximaciones [74].

- *Uso de equivalencias léxicas de traducción*: Se utiliza para reforzar los enlaces obtenidos en la fase estadística, a través del agrupamiento de sentidos y la extracción de pares de traducción.

- *Uso de dominios*: La utilización del dominio de una palabra para la tarea de alineación de textos paralelos, es un enfoque apenas utilizado, pero que permite apoyar a la técnica estadística para mejorar sus resultados.

- *Uso de cognados*: Permite analizar pares de palabras que sean iguales parcial o totalmente para encontrar alineaciones para nombres, lugares, etc. Cuando se habla de que son iguales parcialmente, en el algoritmo de entrada se establece un porcentaje que servirá como mínimo para saber a partir de él, cuales palabras se consideran parcialmente iguales a otras.

- *Uso de aprendizajes*: Permite el apoyo en resultados de alineaciones previas que hayan sido difíciles de establecer, que sirven como referencia para futuras tareas de alineado.

Es importante mencionar que estos enfoques son independientes y pueden funcionar de manera individual o en cualquier combinación. El diseño del algoritmo está optimizado para que, al configurar la utilización de varios de estos enfoques, los procesos compartan información relevante que sirve para evitar consultas de información repetidas y agilizar el tiempo de respuesta del algoritmo.

4.3.1. K-Vec modificado

El algoritmo K-vec original presentado por Fung & Church [42] comienza con la segmentación de los textos de entrada, dividiéndolos en un número establecido de piezas o segmentos. Esto hace que los segmentos obtenidos estén en función de la cantidad del texto y el número fijado. Las primera modificación que se propone es que las piezas sean párrafos u oraciones. Con esto se obtienen segmentos de diferentes tamaños, pero se aprovecha la ventaja de que los textos puedan estar alineados en un nivel superior.

El trabajo de detección de párrafos u oraciones es mucho más simple que el trabajo de detección de palabras, lo cual resulta fortuito ya que la alineación a nivel de palabras mejora mucho si se realiza primero la alineación en los niveles superiores.

El siguiente paso consiste en la generación de una lista de palabras con un vector asociado. Este vector contiene las apariciones de la palabra en cada uno de los segmentos en que se dividió el texto. En K-vec original, sólo se utilizan valores booleanos para indicar la presencia (1) o ausencia (0) de la palabra. La segunda modificación propuesta incluye la frecuencia de aparición, es decir, el número de veces que la palabra ocurre en el segmento. Así por ejemplo, si en un texto que ha sido dividido en cinco segmentos (independientemente del tamaño de los mismos), la palabra «*libertad*» aparece tres veces en el primer segmento, dos en el tercero y ninguna en el resto de las piezas, el vector asociado a «*libertad*» sería $V(libertad) = 3, 0, 2, 0, 0$

Ya que se han obtenido los vectores asociados de todas las palabras en el bitexto, se buscan vectores equivalentes entre ambos lenguajes. La comparación que se realiza para determinar equivalentes consiste de dos pasos. En el primero se descartan todos aquellos vectores que no coincidan en aparición dentro de los segmentos obtenidos. En la segunda fase se busca la distancia mínima entre los vectores no descartados, en la cual se incorpora la tercera modificación propuesta, ya que no solamente se tienen apariciones de palabras en los vectores, sino que se tiene el número de apariciones y por lo tanto es posible calcular la distancia mínima de manera diferente a la original utilizada por K-Vec.

Por ejemplo, suponga que una palabra WO_i se encuentra tres veces en los segmentos 1 y 4, de un texto origen que se ha dividido en cuatro piezas. Su vector será entonces (3, 0, 0, 3). Se ha comparado $V(WO_i)$ con todos los vectores de las palabras en el texto meta, es decir con $V(WT_1), V(WT_2), ..., V(WT_N)$. Se ha determinado que sólo dos de dichos vectores coinciden en aparición con $V(WO_i)$, digamos, $V(WT_j) = (9, 0, 0, 4)$ y $V(WT_k) = (7, 0, 0, 6)$. Los restantes se descartan por no poseer ceros en el segmento 2 y 3. Ahora hay que determinar con cuál de dichos vectores se forma un par de correspondencia. Para ello, se calculan las distancias entre vectores.

$$d(V(WO_i), V(WT_j)) = \sqrt{(9-3)^2 + (4-3)^2} = \sqrt{37}$$

$$d(V(WO_i), V(WT_k)) = \sqrt{(7-3)^2 + (6-3)^2} = \sqrt{25}$$

De los resultados obtenidos podemos observar que $V(WT_k)$ posee la mínima distancia respecto a $V(WO_i)$. Por tanto, las palabras WO_i y WT_k están relacionadas. Aquí es importante aclarar que se puede obtener más de un par de palabras relacionadas, al poseer varios vectores la distancia mínima.

Los pasos restantes del procedimiento K-Vec modificado coinciden con los reportados en el K-vec original (ver capítulo 3). Con todos los pares de correspondencia encontrados, para cada par se construye una tabla de contingencia y para cada tabla se calcula la similitud del par que se está representando. La similitud entre las palabras es determinada por medio de una prueba de asociación. El algoritmo propuesto incorpora dos pruebas de asociación: información común en un punto (PMI) y Puntaje - T.

4.3.2. Alineación bidireccional

Durante la alineación estadística, la búsqueda de pares se realiza de manera bidireccional. Esta forma, en contraste con las técnicas tradicionales, trata tanto a la "fuente" como al "objetivo", de manera simétrica [78]. El espacio de búsqueda va de la palabra en el lenguaje origen al texto meta y viceversa. Cuando los procesos de búsqueda de pares que operan en ambos sentidos alcanzan el mismo estado, se reporta el par de alineación. Esta restricción de simetría mejora el rendimiento del modelo, produciendo alineaciones más precisas.

A nivel de palabras, la alineación bidireccional mejora la calidad de los resultados si se compara con la alineación de palabra unidireccional [79], [80].

4.3.3. Uso de información morfológica

El uso de información morfológica durante la resolución, se limita al POS, que forma parte del identificador, que poseen en MWN, los conjuntos de sinónimos.

Un conjunto de sinónimos o *synset* (abreviación de "*synonym set*") representa un concepto o significado asignado a la agrupación de todas las formas de palabras que representan dicho concepto. Es por ello, que estas formas son consideradas sinónimos y pueden ser intercambiadas en función del contexto. La sinonimia es la relación léxica principal en MWN.

Las formas de palabras pueden presentar dos tipos de ambigüedad: semántica y categorial. La ambigüedad semántica se reconoce en la estructura de MWN cuando la forma de palabra está presente en dos o más synsets. La ambigüedad categorial tiene relación con la clase gramatical (POS) y puede ser detectada en fase previa, durante

la lematización.

En MWN los synsets se identifican con la clase gramatical y un offset arbitrario, como se muestra en la primera columna de la Tabla 4.4. A continuación, se presenta el conjunto de formas de palabras que puede representar a dicho synset y por último, una descripción del significado.

Tabla 4.4: Ejemplos de synsets en MWN

Id	Formas	Significado
n#04309988	foot, human_foot	the foot of a human being
n#09814306	foot, ft	a linear unit of length equal to 12 inches or a third of a yard
v#01563210	foot, pick	pay for something

En el ejemplo anterior se puede identificar que la forma «*foot*» aparece en más de un synset (ambigüedad semántica) y que éstos poseen diferente POS (ambigüedad categorial). Pero éste es sólo un fragmento. Si se hace la búsqueda en MWN por forma de palabra y no por synset, los resultados que se obtendrían para «*foot*», serían los siguientes:

$$\left.\begin{array}{l} n\#04309988 \\ n\#09814306 \\ n\#06278461 \\ n\#01661550 \\ n\#02717023 \\ n\#01824376 \\ n\#00184055 \\ n\#06208347 \\ n\#05307729 \\ n\#02709756 \\ v\#01563210 \\ v\#01300444 \end{array}\right\} \textit{Synsets asociados a } foot$$

4.3.4. Uso de equivalencias léxicas de traducción

Uno de los problemas clásicos y tradicionales de la teoría de la traducción ha sido, y lo sigue siendo, el de la equivalencia. En esta tarea, un par de traducción es considerado correcto si existe al menos un contexto en el cual éste ha sido cierto. Usualmente la extracción, que se encarga de obtener la información relevante de una palabra, sólo está orientada a las categorías principales (sustantivos, adjetivos y verbos). La extracción de equivalencias léxicas ha demostrado ser fuente inestimable de datos de traducción para los bancos de terminología y los diccionarios bilingües. La idea de obtener equivalencias léxicas en una primera fase, vino con la disponibilidad

del recurso MultiWordNet (MWN) y su concepción.

MWN, como se describió en la sección 2.3.4, es una base de datos léxica multilingüe, en la cual se ha realizado una alineación estricta entre PWN y redes de palabras para el español y el italiano, entre otros lenguajes. Estas redes describen las relaciones léxicas y semánticas existentes entre las palabras y recopilan sus sentidos al igual que un diccionario monolingüe.

La Figura 4.3 muestra la composición, por categoría gramatical, de cada una de las redes de palabras empleadas en el estudio. Con ello se tiene una perspectiva más clara de la completitud de las mismas, comparando los dos idiomas involucrados.

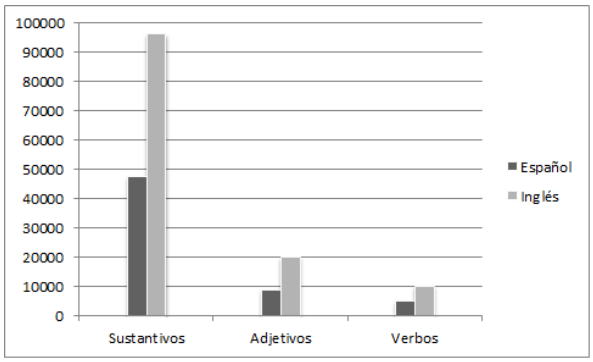


Figura 4.3: Composición de las redes de palabras en MWN

Esta estructura es la que nos permite realizar agrupamiento de sentidos para extraer pares de equivalentes de traducción, mismos que son usados en la fase de alineación, como información lingüística útil para reforzar los enlaces obtenidos.

Como el algoritmo propuesto se basa en la concordancia de synsets, descrita en la sección uso de información morfológica, en redes de palabras alineadas, una vez que se cuenta con los lemas, se extraen los synsets de las palabras no funcionales. Es importante recalcar que no se lleva a cabo ningún proceso de etiquetado durante esta fase. El hecho de que las redes de palabras estén alineadas implica que, para expresar un determinado significado, todas utilizan el mismo synset, independiente del idioma. La constitución de los synsets, permite incorporar un grupo de formas sinónimas, por tanto, todos los wordnets en dicho synset almacenan las formas de palabras que pueden representar el significado específico.

Haciendo énfasis en el proceso de extracción de equivalencias, lo que se hace a grandes rasgos, es determinar el conjunto de todos los synsets de la palabra a corresponder, es decir, todos sus posibles sentidos. Este conjunto se compara entonces con los conjuntos extraídos del contexto paralelo (texto en el idioma meta), y aquella palabra que posea mayor coincidencia o intersección de synsets, será la palabra que se relaciona con

la palabra en el origen.

El mecanismo desarrollado está enfocado en la extracción de equivalencias con limitación uno a uno [34]. Es decir, todas las palabras participan en una única equivalencia, con excepción del nulo. Esto para evitar correspondencias no deseadas, como por ejemplo, tener muchas palabras en español alineadas con la misma palabra en inglés. Así, en cada fase del algoritmo, si una traducción potencial pasa a ser parte de un par de equivalencia, la lista de traducciones potenciales se va reduciendo.

En MWN se manejan elementos léxicos simples, por lo cual las expresiones multipalabras no son encontradas, por tanto, consideraremos sólo equivalencias 1:1, durante la fase de extracción. De este modo, en el texto quedan descartadas todas aquellas frases indivisibles con un sentido específico.

4.3.5. Uso de dominios

Los dominios semánticos proveen una clasificación y un contexto a las palabras en un discurso para describir ciertas áreas concernientes a las opiniones y significados subyacentes que los grupos comparten. *Religión*, *Música* y *Deporte*, son ejemplos de dominios, pues poseen una terminología y coherencia propias [81].

Los synsets de PWN han sido etiquetados con dominios a través de una jerarquía denominada WordNet Domains (WND). Esta jerarquía, está constituida por 169 etiquetas. Cada synset puede poseer uno o más dominios asociados y cada dominio puede incluir synsets de categorías sintácticas y jerarquías diferentes.

Los dominios, al igual que el resto de la información lingüística empleada en este módulo, refuerzan las alineaciones establecidas en la fase estadística, aunque sólo son empleados cuando las equivalencias léxicas de traducción no pueden ser extraídas con base en la intersección de synsets. En síntesis, el método consiste en rechazar todos aquellos pares que posean dominios diferentes y dar mayor peso a aquellos que posean dominios coincidentes.

4.3.6. Uso de cognados

Para la selección del método de cognación a emplear, se compararon las técnicas ortográficas y fonéticas. Esta comparación se muestra en la Tabla 4.5.

Analizando las desventajas de ambas técnicas, es evidente que las correspondientes a las ortográficas, son manejables en el algoritmo de alineación propuesto. La primera desventaja, no se aplica al presente trabajo porque los lenguajes a alinear poseen el mismo alfabeto y la segunda, falsos positivos, es tratable con otra de las técnicas lingüísticas que incorpora nuestro algoritmo: la extracción de equivalencias léxicas de

Tabla 4.5: Comparativa entre los enfoques de cognación

	Ortográficos	Fonéticos
Ventajas	Pueden ser aplicados a textos estándar	Se toma en cuenta el conocimiento lingüístico
	La mayoría de los algoritmos que utilizan esta aproximación son relativamente simples	
	No requiere transcripción fonética	La información fonética es más cercana a la realidad en la lengua hablada
Desventajas	Falsos Positivos	Se requieren transcripciones fonéticas: Manuales (Consumo de tiempo) o Automáticas (Ruido)
	Alfabetos diferentes (por ejemplo: cirílico vs. latino)	Los algoritmos que utilizan esta aproximación tienden a ser mucho más complicados

las redes de palabras alineadas en MWN. Esta solución se presenta más adelante.

Por lo anterior, se eligieron dos métodos ortográficos de cognación: el coeficiente de Dice y la distancia de Levenshtein (ver sección métodos de cognación, del estado del arte).

Falsos cognados

Los falsos cognados (o falsos amigos) son definidos como como palabras en diferentes idiomas con ortografía similar que no son traducciones [82]. Parecen tener un origen común, pero tras un estudio lingüístico se puede determinar que no tienen ningún tipo de relación. La similitud entre palabras de distintas lenguas no basta para demostrar que dichos vocablos están relacionados entre sí.

Los siguientes son ejemplos de falsos cognados para el par de idiomas inglés / español, que coinciden en su totalidad ortográficamente.

Tabla 4.6: Falsos cognados para el par de idiomas inglés - español

Palabra	Significado en inglés	Traducción correcta
conductor	director de orquesta o cobrador	driver
media	medios	sock
mayor	alcalde	bigger
pan	cacerola, cazuela	bread
sensible	sensato	sensitive

Algunos otros ejemplos, con variaciones ortográficas son:

- arm (brazo) y arma
- brave (valiente) y bravo
- carpet (alfombra) y carpeta
- embarrased (avergonzado/a) y embarazada
- exit (salida) y éxito

Hay una cantidad de trabajo considerable para la identificación de cognados, pero no hay mucho en relación a falsos amigos, a pesar de ser ésta una las principales causas de error y que impacta en diversos ámbitos de procesamiento del lenguaje, tales como: traducción automática, alineación de palabras, recuperación de información y desambiguación de los sentidos, entre otros [83]. Los escasos trabajos que se han propuesto, toman en cuentan la dimensión semántica para diferenciar entre cognados verdaderos de aquellos que no lo son. Esta misma dimensión, se ha incorporado en nuestro algoritmo a través de MWN, dicho proceso se explica a continuación.

Una vez que se identifica un cognado en el método de alineación propuesto, debido a que la similitud calculada rebasó el umbral establecido en la configuración del algoritmo, se pueden presentar dos situaciones:

1. Que el par sea un cognado válido
2. Que el par sea un falso cognado

La validez o falsedad del cognado, puede verificarse a través de MWN:

- Si las palabras implicadas en el cognado no se encuentran en el lexicón → *cognado válido*.
- Si las palabras implicadas se encuentran en el lexicón hay que analizar:
  - Si son equivalencias de traducción → *cognado válido*
  - Si NO son equivalencias de traducción → *falsos cognados*

Los pares falsos de cognación son insertados en un diccionario de referencia para el aprendizaje.

4.3.7. Uso de aprendizajes

Se han creado diccionarios de aprendizaje para la mejora y optimización de las tareas de alineación, mismos que son utilizados y alimentados en todos los ejercicios ejecutados.

Específicamente, durante la resolución se hace uso de dos diccionarios. El primero de ellos almacena equivalencias léxicas de traducción y el segundo, falsos cognados. El uso de estos pares aumenta significativamente el desempeño del algoritmo, ya que incorpora conocimiento a priori al modelo de asociación de relaciones.

4.3.8. Pseudocódigo

Descripción del flujo en el proceso de alineación

1. Pre-procesado
    1.1. Para cada archivo
        1.1.1. Analizar los archivos de entrada
            a) Verificar que los archivos existan
            b) Verificar que tengan formato válido
            c) Generar la lista de palabras y frecuencias
        1.1.2. Generar la información de optimización
        1.1.3. Tokenizar
        1.1.4. Obtener la información morfológica de cada palabra
        1.1.5. Etiquetar
2. Ejecutar el algoritmo del HWA
    2.1. Parte estadística
        2.1.1. Para cada archivo
            a) Segmentar el texto
            b) Generar vectores de alineación
            c) Calcular las pruebas de asociación
            d) Establecer las mejores relaciones
            e) Generar el diccionario de alineaciones
            f) Recorrer las estructuras con el texto para asignar su correspondiente alineación
        2.1.2. Utilizar alineación bidireccional si es solicitado
        2.1.3. Generar el auto-aprendizaje
    2.2. Parte lingüística

2.2.1. Para uso de información morfológica (I. M.)
   a) Utilizar la I. M. de cada archivo y encontrar relaciones

2.2.2. Para uso de las equivalencias léxicas de traducción
   a) Buscar traducciones para cada palabra en las lista de optimización
   b) Buscar dichas traducciones en las contrapartes de cada lista, en caso de encontrarla asegurar el enlace
   c) Si no se encuentra la traducción en la contraparte, extraer nuevas equivalencias en MWN

2.2.3. Para uso de dominios
   a) Buscar intersecciones en los dominios de palabras de diferentes lenguajes para encontrar relaciones

2.2.4. Para cognados
   a) Buscar cognados en las listas de optimización para asegurar relaciones
   b) Si no se encuentran, aplicar los métodos de cognación (coeficiente de Dice o distancia de Levenshtein) para generar
   c) Aplicar método de detección de falsos cognados nuevos pares

2.2.5. Para uso del auto-aprendizaje
   a) Buscar casos especiales en la base de datos de auto-aprendizaje

4.3.9. Módulo de resultados

Una característica útil del alineador presentado es que en los resultados, además de incluir los pares alineados de palabras, se anexa una explicación del por qué se llegó a la conclusión de que dichas alineaciones eran las mejores.

Trabajos anteriores presentan sus resultados con una estructura similar a: número de oración, posición L_1, posición L_2 y confianza [4], donde la confianza puede ser segura o probable con un valor asociado. Por lo tanto, las explicaciones provistas por HWA son una fuente de ganancia, en cuanto a presentación de resultados se refiere. Para ello, cuenta con diferentes etiquetas que indican si el enlace es seguro o probable y la técnica mediante la cual fue obtenido.

Esto puede ser útil para procesos o usuarios que deseen manejar o interpretar las salidas, ya sea para generar diccionarios, ajustar manualmente el archivo de alineaciones, etc.

A diferencia de la mayoría de los modelos de alineación, en donde el entrenamiento se realiza con el algoritmo EM [29], el algoritmo HWA propone la idea de una base de aprendizajes, donde se almacenan los enlaces provistos por HWA que sean muy comunes y difíciles de establecer. Esto con la finalidad de incorporarlo a la información lingüística durante futuros alineamientos.

4.3.10. Sobre los resultados

Los resultados obtenidos con el alineador HWA, son prometedores y tienen algunas características singulares. Dentro de éstas se destaca la capacidad del alineador para brindar una explicación del por qué un par de palabras fue alineado. Esto en contraste con los formatos de salida que poseen otros sistemas de alineación. La Figura 4.4 muestra dos formas de presentar los enlaces entre palabras.

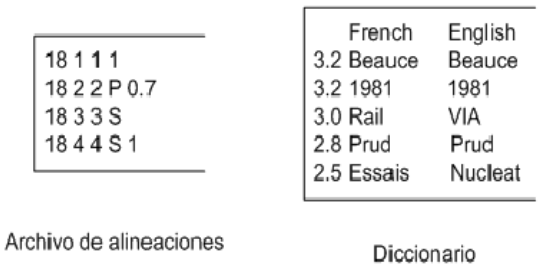


Figura 4.4: Dos formas de presentar resultados de alineaciones

Los resultados del algoritmo HWA se almacenan en un archivo con formato XML, que contiene los siguientes datos:

- Número de párrafo y sentencia donde se encuentra la palabra en el texto 1
- Número de palabra en la sentencia en el texto 1
- Palabra en el texto 1
- Número de párrafo y sentencia donde se encuentra la palabra en el texto 2
- Número de palabra en la sentencia en el texto 2
- Palabra en el texto 2
- Confianza en el enlace
- Método por el que se llegó a determinar el enlace

Además si es necesario se puede llegar a generar:

- Un diccionario entre ambos textos
- Un agregado en la base de aprendizaje

Es importante señalar que la interfaz del algoritmo HWA tendrá la capacidad de mostrar los resultados en distintos formatos, como:

- El archivo de alineamientos en formato XML

- Una matriz de alineación por oraciones
- La señalización dinámica de cada palabra en ambos textos
- Un diccionario de alineación

4.3.11. La interfaz del alineador HWA

La interfaz del alineador HWA está desarrollada en Visual C++. Fue diseñada para facilitar el trabajo del usuario en la utilización del alineador HWA. Permite incluir la información de entrada y configurar los parámetros del algoritmo y muestra los resultados de manera rápida y sencilla.

A continuación se muestran algunas pantallas de la interfaz. La Figura 4.5 muestra la pantalla inicial del alineador HWA, la cual está dividida en tres secciones:

- La sección superior permite indicar el nombre del proyecto y el directorio de trabajo donde se guardarán todos los resultados generados por el alineador. También muestra las tareas concluidas para el proyecto actual.
- El Área de Trabajo, misma que posee dos pestañas: Configuración del Algoritmo HWA y Presentación de resultados. La primera muestra información concerniente a todo el proceso involucrado con la alineación, desde la entrada de los archivos, la configuración del algoritmo y las tareas realizadas en cada fase del proceso. La segunda, tiene que ver únicamente con el formato de presentación de los resultados.
- La sección inferior muestra información concerniente a todos los pasos que se van ejecutando en cada fase del algoritmo para completar la alineación.

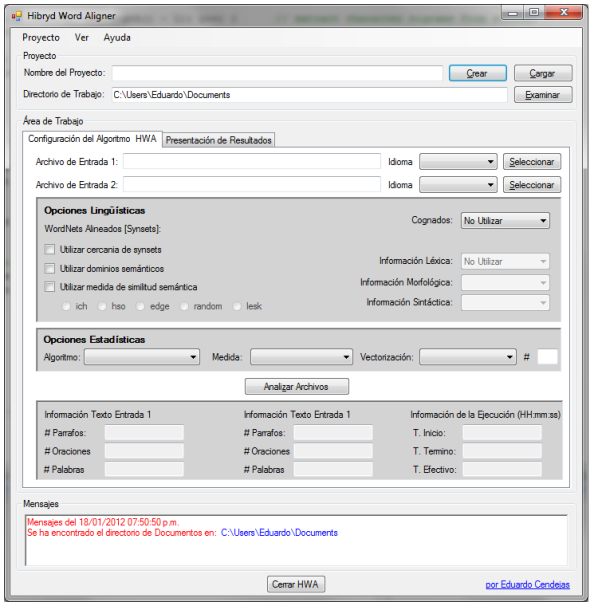


Figura 4.5: Interfaz principal de HWA

Área para la fase de entrada

Esta área permite indicarle al alineador cuáles son los archivos a tratar. Dichos archivos deben contener textos paralelos sin etiquetar. Además, permite indicar el idioma en el que están escritos los archivos, para que el alineador pueda utilizar las herramientas adecuadas de análisis y pre - procesado. El área para la fase de entrada es señalada en la Figura 4.6.

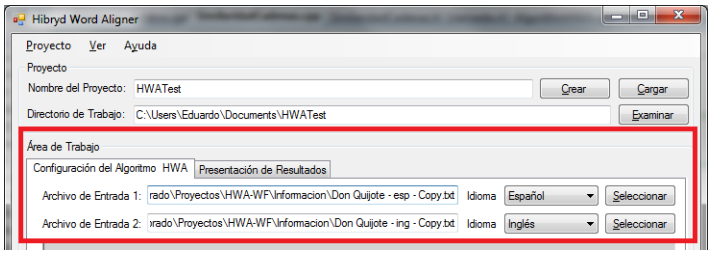


Figura 4.6: Interfaz HWA - Selección de la información de entrada

Área para la fase de configuración del algoritmo

Esta área permite configurar el algoritmo HWA de una manera rápida y sencilla. Consta de dos partes fundamentales:

1. Configuración de la parte estadística
2. Configuración de la parte lingüística

La parte estadística es la base del algoritmo HWA. En ella se puede seleccionar un algoritmo de trabajo, una prueba de asociación para encontrar relaciones entre palabras dentro de los textos (K-Vec modificado [booleano] o K-Vec modificado [Frecuencias]), la medida a utilizar en las pruebas de asociación (PMI o T-Score) y la forma de vectorización para separar los textos según convenga (Por párrafos, por sentencias o por cantidad de palabras).

La parte lingüística permite indicar los recursos que servirán de apoyo a la parte estadística para obtener mejores resultados. Dentro de estos recursos se pueden mencionar: información morfológica, equivalencias léxicas de traducción, dominios semánticos y cognados.

La tarea de configuración es fundamental para la obtención de buenos resultados, ya que el usuario tiene la libertad de seleccionar todos los aspectos que intervendrán en el proceso. El área relacionada con esta tarea es mostrada en la Figura 4.7.

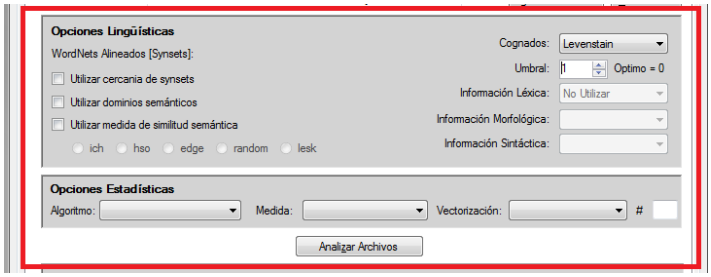


Figura 4.7: Interfaz HWA - Configuración del algoritmo HWA

Luego de seleccionar los archivos con el contenido a alinear y todos los parámetros para la ejecución del algortimo, se debe presionar el botón *Analizar Archivos*. Esta acción desencadena la ejecución de los módulos de pre - procesamiento, configuración y resolución del algoritmo.

Pestaña para la fase de salida

En esta sección se muestran los resultados de las alineaciones. HWA presenta dichos resultados en cuatro formas:

1. *Resultados XML*: Esta presentación despliega el archivo con el resultado de la alineación en formato XML, tal y como está almacenado en el archivo. Mediante las etiquetas se pueden identificar las relaciones entre párrafos, sentencias y palabras que fueron encontrados por el alineador. Además, contiene etiquetas para identificar los niveles de veracidad de las relaciones entre palabras, según el resultado de la prueba de asociación seleccionada; y una explicación de cómo se llegó a esa relación. La Figura 4.8 muestra una pantalla cuya salida usa este formato.

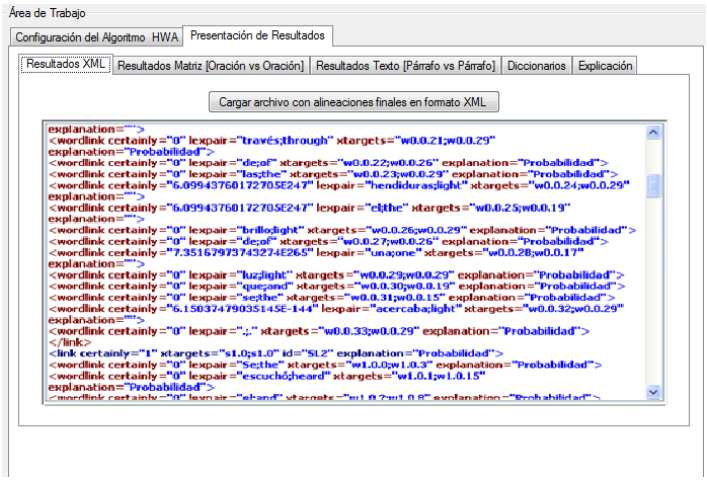


Figura 4.8: Interfaz HWA - Muestra de resultados en formato XML

2. *Resultados Matriz* (oración vs oración): Para una oración seleccionada por el usuario, se genera una matriz. En la primera fila se coloca la oración en el lenguaje 1 y en la primera columna se coloca la oración en el lenguaje 2. En las celdas intermedias se marcan las relaciones (alineaciones) entre las palabras de cada lenguaje.

3. *Resultados Texto* (párrafo vs párrafo): Según la selección del usuario, se muestra el párrafo en el lenguaje 1 y su correspondiente párrafo alineado en el lenguaje 2. Cuando se marca una palabra en uno de los párrafos se colorea dicha palabra del lenguaje 1 y su palabra alineada en el lenguaje 2.

4. *Diccionario*: Presenta el diccionario generado por el algoritmo, indicando las palabras en ambos lenguajes y la explicación de cómo se llegó a dicha relación. La Figura 4.9 muestra una pantalla empleando este formato.

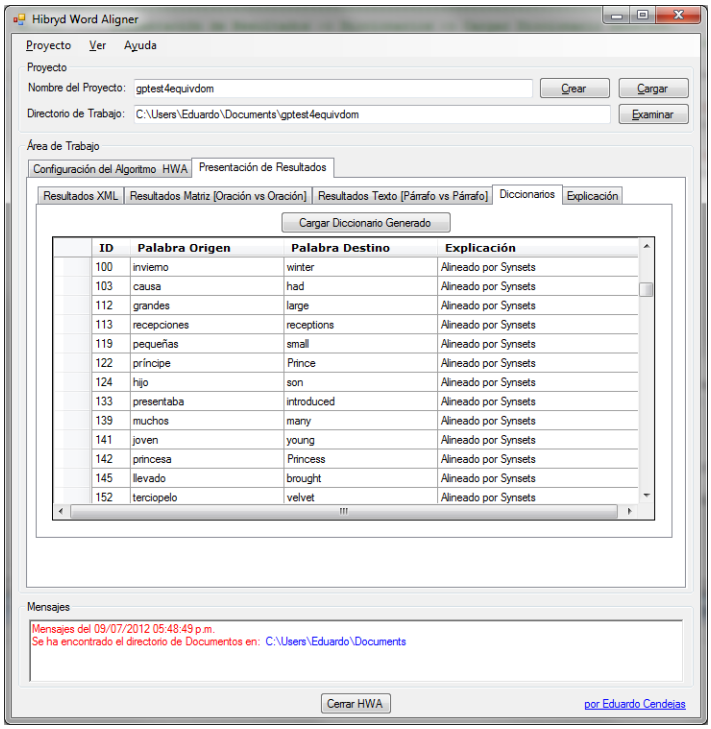


Figura 4.9: Interfaz HWA - Muestra de resultados en formato de diccionario

Capítulo 5

Resultados

5.1. Evaluación de los sistemas de alineación de palabras

Cuando se desarrollan métodos de alineación de palabras y aplicaciones de software (alineadores), se deben considerar ciertos aspectos para obtener un proyecto funcional y bajo ciertos criterios mínimos de desempeño. Para valorar esto, se utilizan medidas de evaluación que se describen de manera breve a continuación.

5.1.1. Métricas de calidad en la alineación de palabras

Un reto a tratar con los sistemas de alineación, es medir la calidad de una alineación hipotética. Sería deseable contar con una métrica intrínseca rápida de calidad en la alineación para aplicar a los sistemas de desarrollo y utilizarla como un criterio para el entrenamiento discriminativo de modelos de alineación más sofisticados [84].

Para el análisis de los resultados obtenidos se emplearon tres medidas de evaluación: precisión, recall y F - measure.

- La *precisión* es calculada como el número de equivalencias extraídas correctamente entre el número de equivalencias sugeridas por el sistema.
- El *recall* se corresponde al número de equivalencias extraídas correctamente entre el número de equivalencias sugeridas por los anotadores.

Sin embargo, ni la precisión ni el recall pueden, de manera independiente, determinar la calidad del emparejamiento. Por lo general, la maximización del recall compromete la precisión y viceversa [85]. Por tanto, se requiere una medida que combine ambos parámetros.

- La F - *measure* posee esta característica y se determina como:

$$F - measure = \frac{2 * precisión * recall}{precisión + recall}$$

En este caso, la precisión y el recall poseen el mismo peso, pero la fórmula anterior puede ajustarse si se desea otorgar mayor peso a algunas de estas dos medidas.

5.1.2. Puntos a considerar en la evaluación de los sistemas de alineación

Para poder comparar algoritmos y sistemas de alineación de palabras en corpus paralelos, es fundamental considerar varias características, para contar con un punto de referencia en la evaluación de dichos sistemas. Los puntos más importantes a tomar en cuenta son [86]:

- *El propósito del sistema de alineación*: Un programa diseñado para la extracción de un diccionario bilingüe difiere de un programa cuyo objetivo es alinear un texto completo con su traducción.
- *Unidades*: Las unidades y las multi - unidades son importantes para la traducción ya que debe decidirse en que forma deben ser contadas.
- *Recursos usados*: Cuando los sistemas son comparados, la información de cuánto tiempo toma ejecutar el sistema en un bitexto particular debe ser incluida, así como los recursos extra empleados, tales como diccionarios bilingües y listas de colocación monolingües.
- *La utilización de un gold estándar*: Cuando la salida alineada es evaluada puede compararse con un gold estándar. Éste puede ser construido antes de la alineación actual, o expertos pueden evaluar una muestra de la salida después de la alineación.
- *Las métricas y los métodos de puntuación*: ¿Qué métricas deben ser utilizadas? Cuando la salida es evaluada, hay varias preguntas sobre cómo juzgar alineaciones parciales; por ejemplo, cuando las colocaciones son borradas, insertadas, o si existen errores de segmentación o paráfrasis.
- *Análisis de error*: Es importante conocer la naturaleza de las equivocaciones que comete un sistema en particular, si produce fallas típicamente en ciertos tipos de colocaciones, en unidades con un particular rango de frecuencia, etc.

5.2. Resultados

Aun cuando el sistema tiene la capacidad de tomar diversos pares de lenguajes, solamente se hicieron pruebas para español - inglés debido a la falta de evaluadores para otros pares de lenguajes.

Para realizar experimentos, se han empleado fragmentos elegidos aleatoriamente de las novelas *Don Quijote de la Mancha* y *Guerra y paz*, en sus versiones español - inglés. En ambos casos, los corpus de prueba estaban alineados a nivel de oraciones.

La Tabla 5.1 muestra información de la composición de cada uno de los fragmentos de las novelas utilizadas y de los equivalentes alineados.

Tabla 5.1: Composición y datos de alineación de los fragmentos paralelos

	Don Quijote de la Mancha	*Guerra y paz*
# palabras españolas	828	817
# palabras inglesas	866	845
Gold estándar	822	815
Topline	728	731
# palabras NO funcionales	436	485

Para producir el gold estándar de los pares de alineación de todas las palabras, dos anotadores fueron instruidos con procedimientos específicos de cuándo asignar un equivalente nulo. No se incluyeron etiquetas de probabilidades. En caso de que hubiera un desacuerdo para un par específico, un tercer anotador definía el correcto.

El topline indica el número máximo de equivalencias que podrían ser extraídas por el sistema. Del total se eliminan las relaciones 1:0, es decir la alineación de palabras en el texto origen con un nulo en su contraparte. Esto tiene relación con el procesamiento estadístico, tomando en cuenta que cada palabra en el texto origen posee un vector de aparición en los segmentos y este vector es comparado con todos los vectores del texto meta, mediante el cálculo de la distancia vectorial. La palabra asociada al vector meta que produzca la distancia mínima con el vector origen se tomará como el par de alineación. Este procedimiento asegura que siempre se obtenga una distancia mínima mayor a cero en relaciones 1:0 y por tanto, se establece la relación entre los vectores identificados (en vez de asignar un nulo en la meta).

Existen otros inconvenientes que afectan la funcionalidad de los recursos lingüísticos. Por ejemplo, en el uso de equivalencias de traducción se manejan lemas simples, lo que implica que las expresiones compuestas nunca son reforzadas. En esta misma técnica, tampoco se consideran las palabras funcionales (artículos, pronombres, preposiciones, conjunciones), ya que éstas no están incluidas en los lexicones empleados. La incompletitud de las redes de palabras y las brechas léxicas (*gaps*) de los lenguajes

involucrados afectan también el desempeño del sistema durante el uso de equivalencias.

Para el caso de los dominios semánticos, el análisis de la red de palabras para el español arroja que casi el 13 % de los lemas no poseen ningún dominio asociado. Esto significa que hay pares cuya relación no puede establecerse con base en este elemento lingüístico.

La ambigüedad es otro aspecto perjudicial en la alineación, y está vinculada a varios de los recursos utilizados. La Tabla 5.2 resume los promedios de la cantidad de synsets asignados a cada lema y su distribución por categoría gramatical para el inglés y el español. Por ejemplo, la primera entrada de la tabla para el español indica que cada lema que tiene, en promedio, 1.56 synsets asociados. Mientras mayor es el promedio de la cantidad de synsets y dominios por lema, mayor será el grado de polisemia[1] del lenguaje y mayor dificultad se tendrá para determinar las alineaciones correctas con apoyo de los recursos ambiguos. De los valores de la tabla se puede observar que el problema principal radicaría en los verbos.

Tabla 5.2: Promedio de la cantidad de synsets y dominios por lema

	Español	Inglés
Synsets × lema	1.56	1.43
Synsets sustantivos × lema	1.32	1.24
Synsets adjetivos × lema	1.99	1.48
Synsets verbos × lema	2.34	2.14
Dominios × lema	1.48	1.48

Se incluye además la misma información para los dominios. En este sentido, es importante destacar que hay etiquetas que se repiten en una cantidad considerable de lemas, lo que haría difícil establecer las correspondencias entre estas palabras. Este es el caso del dominio *FACTOTUM*, que fue asignado si no existe otro adecuado. *FACTOTUM* está asociado al 11 % de los lemas ingleses y al 27 % de los españoles, seguido de *Biology* asociado con el 26 % y el 18 % respectivamente. Esto provoca que la información de los dominios no sea relevante para reforzar o debilitar las alineaciones de los pares de alineación en los que uno de los lemas tenga estos dominios asignados.

La ambigüedad es también un inconveniente en el uso de la notación morfológica y puede presentarse de varias formas. Por ejemplo, la palabra «*enfermo*» posee más de una categoría gramatical (sustantivo y adjetivo) y «*testigo*» puede ser clasificada con género masculino o femenino.

[1]Se denomina *polisemia* a la pluralidad de significados de cualquier signo lingüístico

Sin embargo, estos últimos factores (que sí afectan el desempeño) no se han considerado en el topline, pues independientemente de que las alineaciones no puedan ser reforzadas durante el proceso lingüístico, se parte de una relación dada por el proceso estadístico. En el topline tampoco se consideran falsos cognados, ya que el algoritmo propuesto en la sección 4.3.6 verifica la validez de estos pares con independencia del resultado numérico obtenido por el método de cognación empleado (coeficiente de *Dice* o distancia de *Levenshtein*). De este modo, cuando se determina la falsedad de la correlación, tampoco se fortalecen las alineaciones.

Tras ejecutar el algoritmo sobre los corpora, los resultados que se han obtenido son favorables. Los procesos de optimización han funcionado correctamente para poder brindar resultados en tiempos razonables. Las Tablas 5.3 y 5.4 muestran el tiempo efectivo consumido para obtener las alineaciones según la(s) técnica(s) empleada(s). El tiempo incluye, independientemente de los recursos, el pre-procesamiento de los textos, la creación del diccionario bilingüe y la base de datos con las alineaciones resultantes de la ejecución del módulo de resolución.

Tabla 5.3: Tiempos de ejecución para *Don Quijote de la Mancha*

Recurso	Técnica(s) empleada(s)							
K-Vec modificado	×	×				×	×	×
Alin. bidireccional		×				×	×	×
Equiv. léxicas			×	×	×	×	×	×
Cognados				×	×		×	×
Dominios					×			×
Tiempos	00:00:26.35	00:00:27.02	00:00:27.45	00:00:30.49	00:00:30.57	00:00:29.66	00:00:31.28	00:00:32.54

Tabla 5.4: Tiempos de ejecución para *Guerra y paz*

Recurso	Técnica(s) empleada(s)							
K-Vec modificado	×	×				×	×	×
Alin. bidireccional		×				×	×	×
Equiv. léxicas			×	×	×	×	×	×
Cognados				×	×		×	×
Dominios					×			×
Tiempo	00:00:29.46	00:00:30.05	00:00:35.94	00:00:36.15	00:00:37.77	00:00:36.25	00:00:37.64	00:00:38.27

Las pruebas realizadas, para la obtención de las tres métricas de calidad, se han dirigido en dos vertientes: (1) evaluar la aportación de los recursos lingüísticos en cuanto a la precisión del sistema y (2) comparar los resultados obtenidos con dos de los alineadores más utilizados.

5.2.1. Comparación de recursos lingüísticos

Uno de los aspectos importantes a considerar es analizar las condiciones que producen mayor efectividad del algoritmo para obtener pares de palabras exitosos, por lo cual se generaron varias estrategias de comparación entre los resultados obtenidos en dependencia de los recursos utilizados. Las tablas generadas son:

- Tabla 5.5: Medidas de evaluación para *Don Quijote de la Mancha*
- Tabla 5.6: Medidas de evaluación para *Guerra y paz*
- Tabla 5.7: Precisión para *Don Quijote de la Mancha* con base en el gold estándar y el topline
- Tabla 5.8: Precisión para *Guerra y paz* con base en el gold estándar y el topline
- Tabla 5.9: Medidas de evaluación para las palabras de contenido en *Don Quijote de la Mancha*
- Tabla 5.10: Medidas de evaluación para las palabras de contenido en *Guerra y paz*

Para el primer acercamiento se emplearon 23 oraciones alineadas de la novela *Don Quijote de la Mancha* y 37 de *Guerra y paz*. El objetivo de este análisis consiste en determinar qué tipo de recurso lingüístico (o combinación de ellos) tiene mayor impacto positivo en los resultados de la alineación. La comparación se presenta en las Tablas 5.5 y 5.6, cuyas columnas muestran los recursos empleados y los valores de las tres medidas de evaluación empleadas para cada recurso. Los mismos valores han sido graficados en las Figuras 5.1 y 5.2.

Tabla 5.5: Medidas de evaluación para *Don Quijote de la Mancha*

Recursos estadísticos empleados		Recursos lingüísticos empleados			Medidas de evaluación		
K-Vec modificado	Alin. bidireccional	Equiv. léxicas	Cognados	Dominios	Precisión	Recall	F - measure
×					12.44 %	12.53 %	12.48 %
×	×				15.22 %	15.33 %	15.27 %
		×			24.15 %	24.33 %	24.24 %
		×	×		29.35 %	29.56 %	29.45 %
		×	×	×	29.11 %	29.32 %	29.21 %
×	×	×			36.96 %	37.23 %	37.09 %
×	×	×	×		39.25 %	39.54 %	39.39 %
×	×	×	×	×	35.63 %	35.89 %	35.76 %

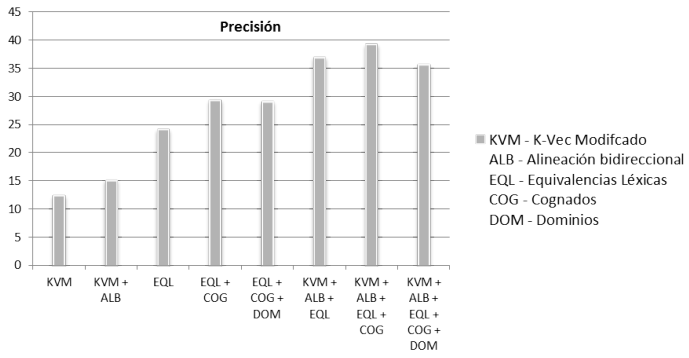


Figura 5.1: Valores de precisión por técnica(s) empleada(s) para *Don Quijote de la Mancha*

Tabla 5.6: Medidas de evaluación para *Guerra y paz*

Recursos estadísticos empleados		Recursos lingüísticos empleados			Medidas de evaluación		
K-Vec modificado	Alin. bidireccional	Equiv. léxicas	Cognados	Dominios	Precisión	Recall	F - measure
×					17.24 %	17.30 %	17.27 %
×	×				18.70 %	18.77 %	18.74 %
		×			24.08 %	24.17 %	24.13 %
		×	×		30.07 %	30.14 %	30.13 %
		×	×	×	31.91 %	32.02 %	31.97 %
×	×	×			31.54 %	31.66 %	31.60 %
×	×	×	×		35.09 %	35.21 %	35.15 %
×	×	×	×	×	34.84 %	34.97 %	34.91 %

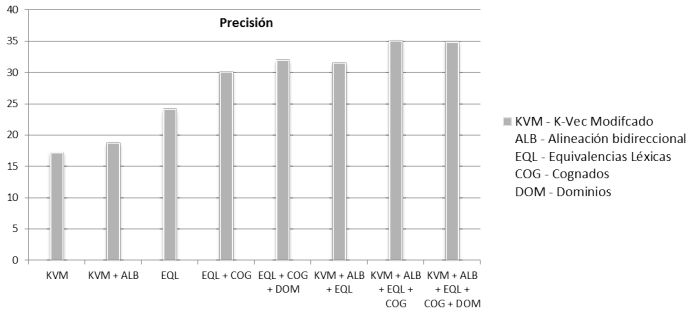


Figura 5.2: Valores de precisión por técnica(s) empleada(s) para *Guerra y paz*

Los valores anteriores de las medidas se determinaron tomando en cuenta el gold estándar. Sin embargo, si se utiliza el topline, excluyendo las alineaciones 1:0, mejoran los resultados del sistema: en promedio 3.78 % para *Don Quijote de la Mancha* y 3.27 % para *Guerra y paz*. Las Tablas 5.7 y 5.8 muestran la precisión con base en el gold estándar y el topline y las Figuras 5.3 y 5.4 representan la correlación de las dos bases para ambas novelas.

Tabla 5.7: Precisión para *Don Quijote de la Mancha* con base en el gold estándar y el topline

Recursos estadísticos empleados		Recursos lingüísticos empleados			Precisión	
K-Vec modificado	Alin. bidireccional	Equiv. léxicas	Cognados	Dominios	Gold estándar	Topline
×					12.44 %	14.15 %
×	×				15.22 %	17.31 %
		×			24.15 %	27.47 %
		×	×		29.35 %	33.38 %
		×	×	×	29.11 %	33.10 %
×	×	×			36.96 %	42.03 %
×	×	×	×		39.25 %	44.64 %
×	×	×	×	×	35.63 %	40.52 %

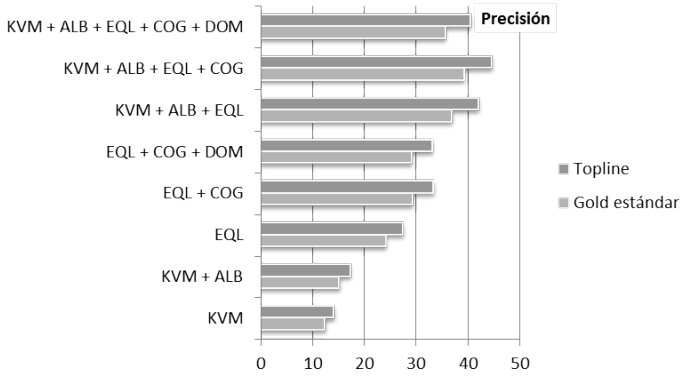


Figura 5.3: Comparación de precisión con base en el gold estándar y el topline para *Don Quijote de la Mancha*

Tabla 5.8: Precisión para *Guerra y paz* con base en el gold estándar y el topline

Recursos estadísticos empleados		Recursos lingüísticos empleados			Precisión	
K-Vec modificado	Alin. bidireccional	Equiv. léxicas	Cognados	Dominios	Gold estándar	Topline
×					17.24 %	19.29 %
×	×				18.70 %	20.93 %
		×			24.08 %	26.95 %
		×	×		30.07 %	33.65 %
		×	×	×	31.91 %	35.70 %
×	×	×			31.54 %	35.29 %
×	×	×	×		35.09 %	39.26 %
×	×	×	×	×	34.84 %	38.99 %

Independientemente del total aplicado (gold estándar o topline), la precisión del recurso de equivalencias léxicas de traducción tiene que ver con la asertividad de las alineaciones de las redes de palabras que conforman MWN. Sin embargo, el valor de recall obtenido con este recurso exclusivamente, es pobre. Esto se debe a dos razones fundamentales: (1) el hecho de que en MWN se almacenan elementos léxicos simples y por tanto, es imposible asignar un sentido específico a las expresiones multipalabra y (2) la incompletitud de las redes (véase Figura 4.3 para observar la desproporción entre el inglés y el español).

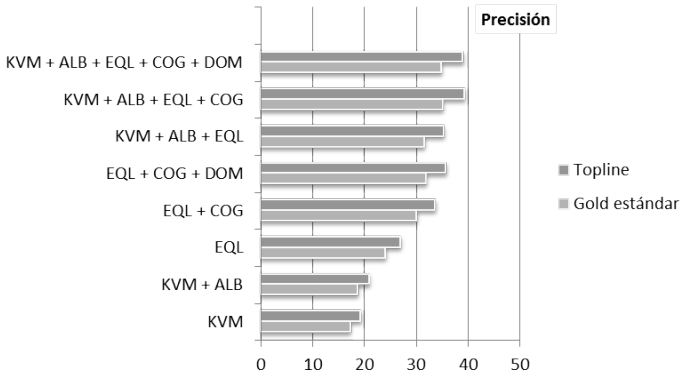


Figura 5.4: Comparación de precisión con base en el gold estándar y el topline para *Guerra y paz*

Por otra parte, cuando se incluye el análisis de cognación, la precisión es también significativa en comparación con la incorporación de dominios semánticos. Lo anterior está relacionado con la acertividad de las medidas de cognación y la utilización de la heurística de falsos cognados.

Por último, es importante notar que los mejores resultados se consiguen con la combinación de las técnicas estadísticas, las equivalencias léxicas de traducción y los cognados. Las Figuras 5.5 y 5.6 muestran la distribución del método aplicado en los pares determinados como correctos para los fragmentos de texto paralelos empleados.

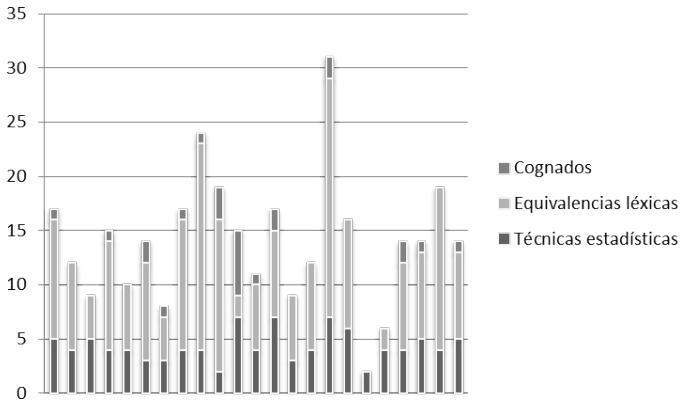


Figura 5.5: Distribución por oración del método aplicado para *Don Quijote de la Mancha*

Para un análisis más exhaustivo se representa, en la Figura 5.16 y la 5.17 (al final del capítulo), la cantidad de pares correctos propuestos por el sistema y su relación con el gold estándar y el topline para cada oración (23 de *Don Quijote de la Mancha* y 37 de *Guerra y paz*)

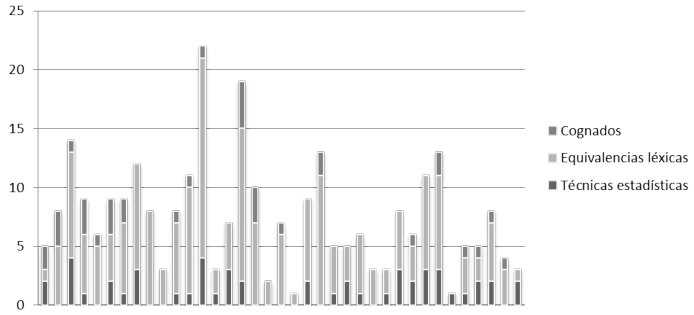
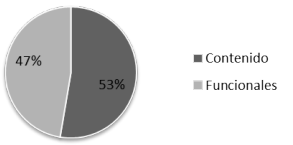
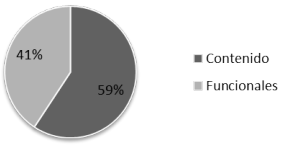


Figura 5.6: Distribución por oración del método aplicado para *Guerra y paz*

Además de los análisis anteriores, se ha realizado una comparación atendiendo a la clasificación de las palabras como funcionales o de contenido, ya que esto está directamente relacionado con el resultado de la alineación. Las palabras de contenido (sustantivos, adjetivos, verbos y adverbios) poseen un significado semántico, mientras que las funcionales por sí mismas no lo tienen. Estas últimas actúan gramaticalmente en la lengua, estableciendo relaciones entre palabras de contenido. Los artículos, pronombres, preposiciones y conjunciones son ejemplos de palabras funcionales. La Figura 5.7 muestra la composición de los fragmentos de las novelas empleadas y las Figuras 5.8 y 5.9, la proporción de los pares correctos por clasificación y método.



(a) *Don Quijote de la Mancha*



(b) *Guerra y paz*

Figura 5.7: Composición de los textos por clasificación de palabras

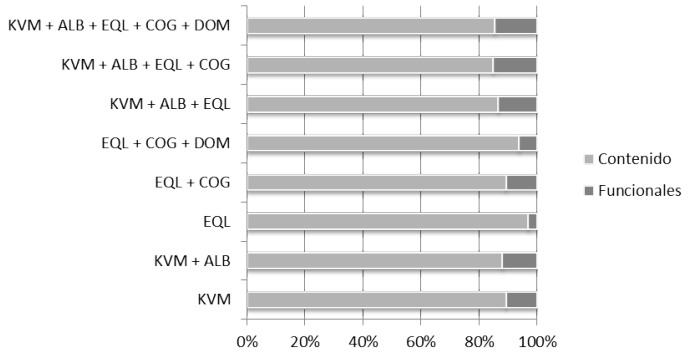


Figura 5.8: Proporción de los pares correctos por clasificación y técnica para *Don Quijote de la Mancha*

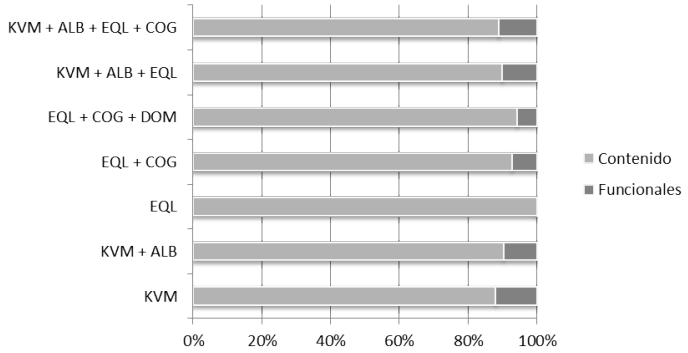


Figura 5.9: Proporción de los pares correctos por clasificación y técnica para *Guerra y paz*

Como se observa de las gráficas anteriores de columna apilada, los mejores resultados se obtienen de las palabras de contenido. Los valores de precisión y recall para éstas se presentan en las Tablas 5.9 y 5.10.

Para una comparación y resumen de las tres precisiones calculadas (con base en el gold estándar, con base en el topline y para palabras de contenido), se muestran estos valores en las Figuras 5.10 y 5.11.

Tabla 5.9: Medidas de evaluación para las palabras de contenido en *Don Quijote de la Mancha*

Recursos estadísticos empleados		Recursos lingüísticos empleados			Medidas de evaluación		
K-Vec modificado	Alin. bidireccional	Equiv. léxicas	Cognados	Dominios	Precisión	Recall	F - measure
×					21.10 %	21.30 %	21.20 %
×	×				25.46 %	25.69 %	25.58 %
		×			44.50 %	44.91 %	44.70 %
		×	×		49.77 %	50.23 %	50.00 %
		×	×	×	51.83 %	52.31 %	52.07 %
×	×	×			60.78 %	61.34 %	61.06 %
×	×	×	×		63.30 %	63.89 %	63.59 %
×	×	×	×	×	57.80 %	58.33 %	58.06 %

Tabla 5.10: Medidas de evaluación para las palabras de contenido en *Guerra y paz*

Recursos estadísticos empleados		Recursos lingüísticos empleados			Medidas de evaluación		
K-Vec modificado	Alin. bidireccional	Equiv. léxicas	Cognados	Dominios	Precisión	Recall	F - measure
×					25.57 %	25.73 %	25.65 %
×	×				28.45 %	28.63 %	28.54 %
		×			40.41 %	40.66 %	40.54 %
		×	×		47.01 %	47.30 %	47.16 %
		×	×	×	50.72 %	51.04 %	50.88 %
×	×	×			47.84 %	48.13 %	47.98 %
×	×	×	×		52.58 %	52.90 %	52.74 %
×	×	×	×	×	54.43 %	54.77 %	54.60 %

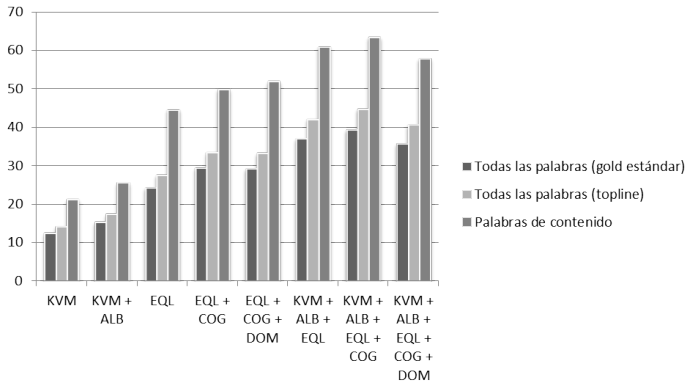


Figura 5.10: Precisiones calculadas para *Don Quijote de la Mancha*

El análisis por oración de los pares correctos para las palabras de contenido, se muestra en las Figuras 5.18 y 5.19, al término de este capítulo.

5.2.2. Comparación de algoritmos de alineación

Para el segundo acercamiento, se realizaron pruebas de comparación con los resultados obtenidos con GIZA y el algoritmo K - Vec original. La elección de estos sistemas de alineación tiene que ver con el reconocimiento de la herramienta en el campo (GIZA++) y el fundamento de la fase estadística de HWA (K - Vec original). Además, es importante hacer notar que los porcentajes obtenidos fueron calculados alimentando todos los algoritmos con parámetros similares. Estos podrían variar en dependencia de

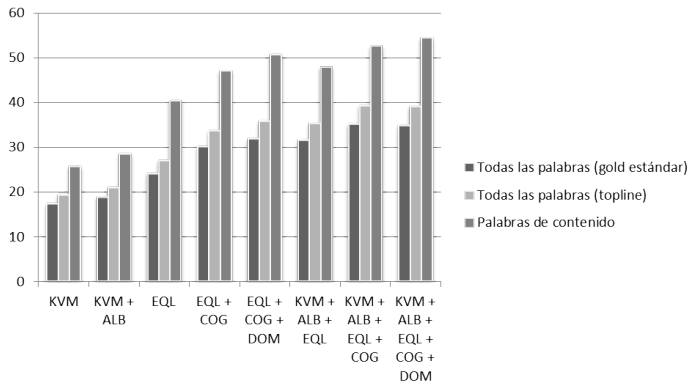


Figura 5.11: Precisiones calculadas para *Guerra y paz*

los parámetros que se configuren para la selección de relaciones válidas entre palabras para cada herramienta. Por ejemplo, si se configura GIZA++ con varias iteraciones, utilizando más información o más entrenamiento sus porcentajes de acierto mejorarán. En el caso del algoritmo K - Vec original, si se deja un límite bajo en el umbral de las pruebas de asociación para la determinación de relaciones, asocia todas las palabras pero realiza muchas alineaciones para una misma palabra (por ejemplo, tener muchas palabras en español alineadas con la misma palabra en inglés), es decir, no se evitan correspondencias no deseadas ni se presenta el resultado más óptimo.

Las Tablas 5.11 y 5.12 reúnen los resultados obtenidos de GIZA, K-Vec original y el peor y mejor desempeño de HWA, correspondientes a: la modificación de K-Vec y a la combinación de los dos recursos estadísticos + equivalencias léxicas + cognados, respectivamente. Se utilizaron, para la obtención de estos valores, los mismos fragmentos paralelos del acercamiento anterior.

Tabla 5.11: Comparación de resultados de alineación para *Don Quijote de la Mancha*

Alineador	Pares correctos	Precisión	Recall	F-measure
K - Vec original	91	10.99 %	11.07 %	11.03 %
GIZA++	94	11.35 %	11.44 %	11.39 %
HWA - peor desempeño	103	12.44 %	12.53 %	12.48 %
HWA - mejor desempeño	325	*39.25 %*	*39.54 %*	*39.39 %*

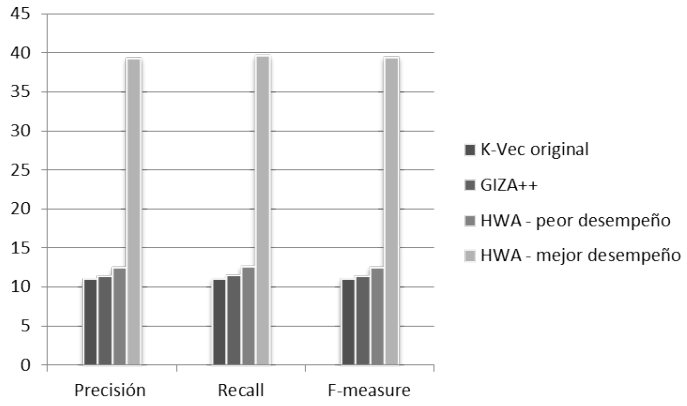


Figura 5.12: Comparación de los valores de las medidas para *Don Quijote de la Mancha*

Tabla 5.12: Comparación de resultados de alineación para *Guerra y paz*

Alineador	Pares correctos	Precisión	Recall	F-measure
K - Vec original	115	14.06 %	14.11 %	14.08 %
GIZA++	152	18.58 %	18.65 %	18.62 %
HWA - peor desempeño	141	17.24 %	17.30 %	17.27 %
HWA - mejor desempeño	287	*35.09 %*	*35.21 %*	*35.15 %*

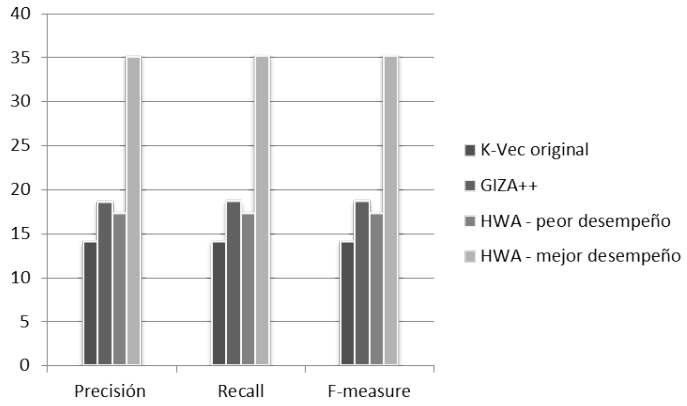


Figura 5.13: Comparación de los valores de las medidas para *Guerra y paz*

Los Apéndices A y B muestran la cantidad de pares correctos propuestos por K-Vec y por GIZA y su relación con el gold estándar y el topline para cada oración en *Don Quijote de la Mancha* y *Guerra y paz* respectivamente.

A diferencia de las técnicas propuestas, tanto el K-Vec original, como GIZA, no muestran una mejoría si sólo se toman en cuenta palabras de contenido. Esto se debe a su carácter estadístico. Mientras la probabilidad de que las palabras funcionales se

repitan en el texto es muy alta, ocurre lo contrario con las palabras de contenido. Las Figuras 5.14 y 5.15 muestran la cantidad de pares correctos tomando en cuenta la cantidad total de palabras vs. la cantidad palabras de contenido.

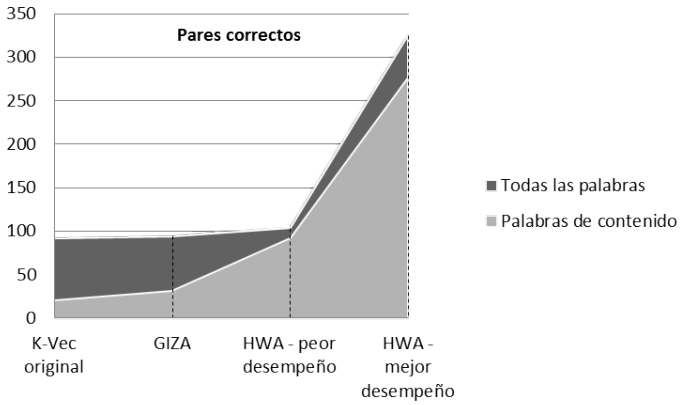


Figura 5.14: Cantidad de pares correctos por alineador para *Don Quijote de la Mancha*

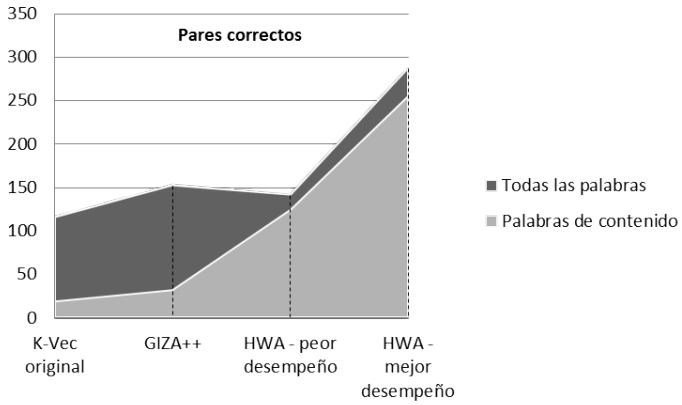


Figura 5.15: Cantidad de pares correctos por alineador para *Guerra y paz*

El análisis por oración de los pares correctos propuestos por K-Vec y GIZA para las palabras de contenido, se presenta en los Apéndices C y D.

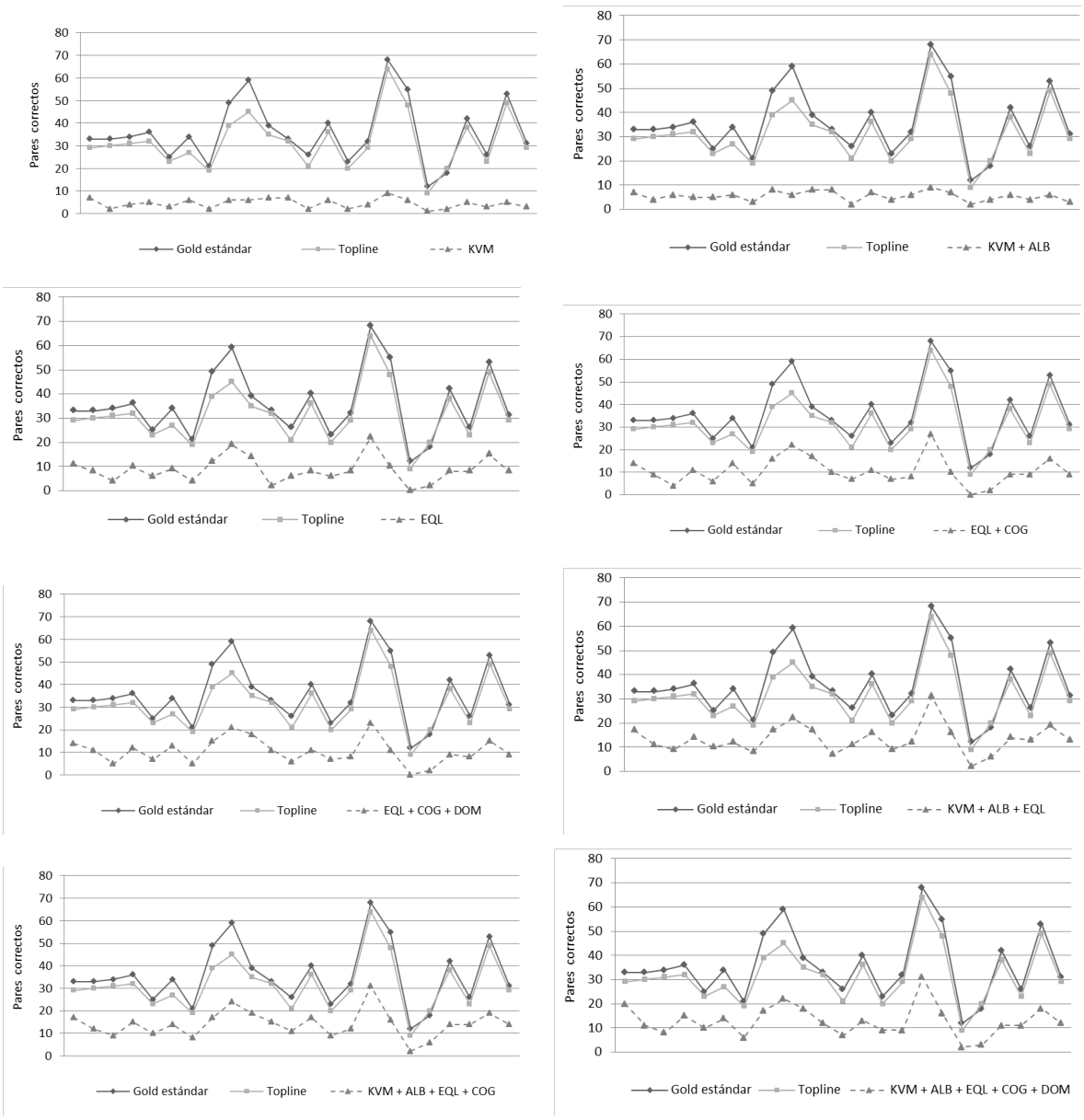


Figura 5.16: Cantidad de pares correctos, gold standard y topline para *Don Quijote de la Mancha*

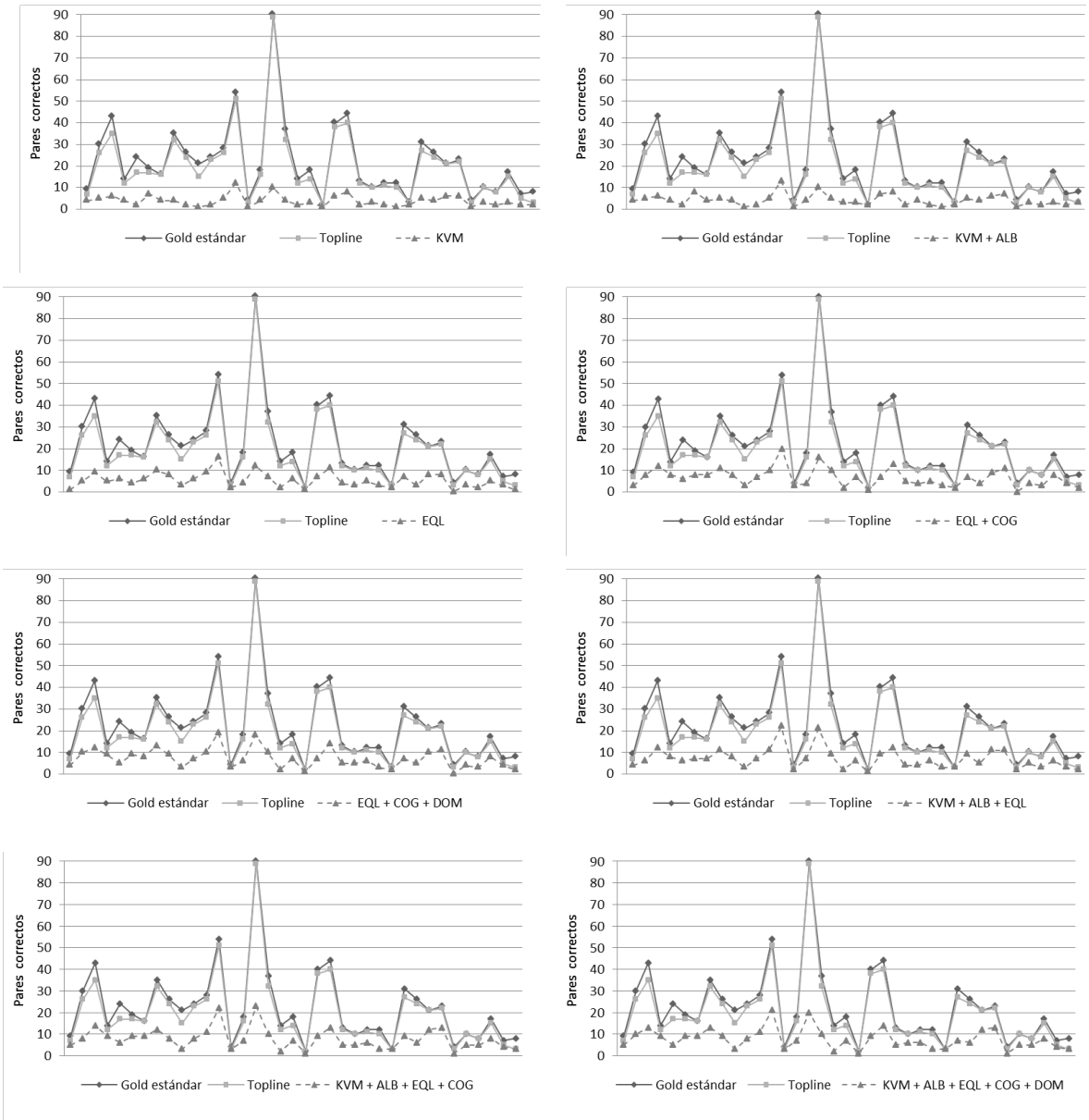


Figura 5.17: Cantidad de pares correctos, gold standard y topline para *Guerra y paz*

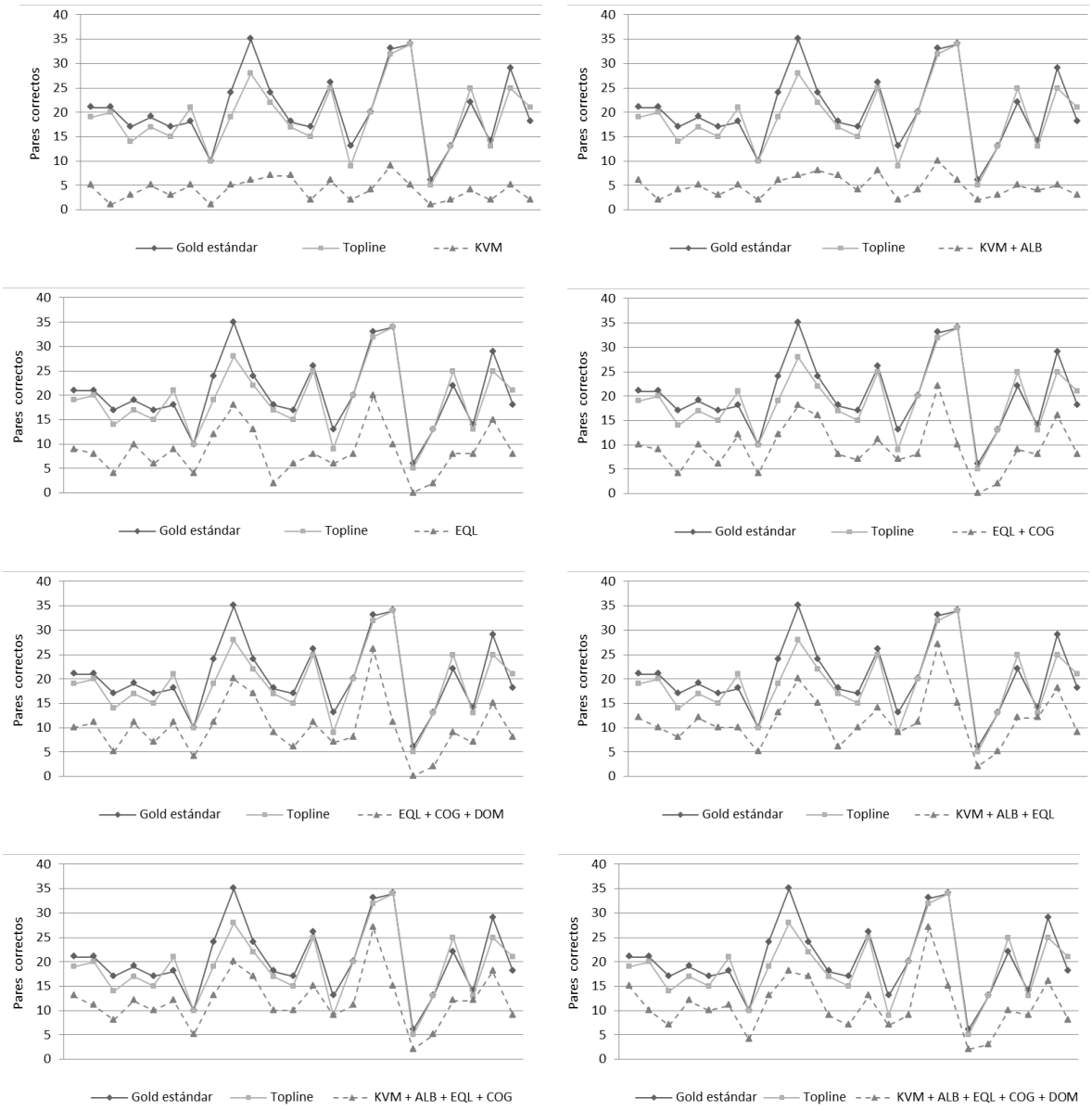


Figura 5.18: Cantidad de pares correctos, gold standard y topline para palabras de contenido en *Don Quijote de la Mancha*

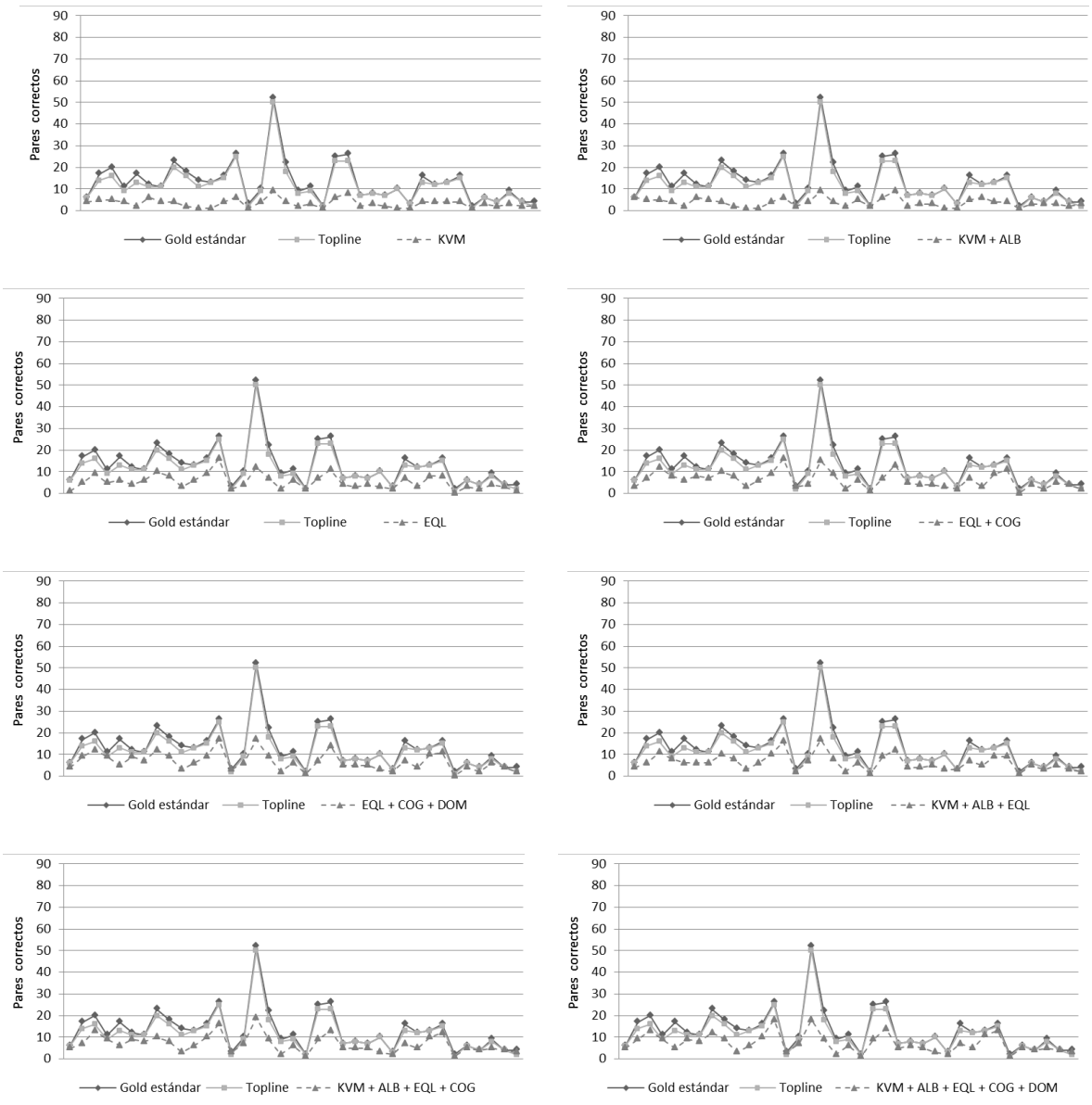


Figura 5.19: Cantidad de pares correctos, gold standard y topline para palabras de contenido en *Guerra y paz*

Capítulo 6

Conclusiones

La alineación a nivel de palabras es una importante tarea de apoyo para la mayoría de los métodos de traducción automática, aunque también ha sido empleada para la enseñanza de idiomas, la completitud de memorias de traducción, la identificación de expresiones idiomáticas, la lexicografía multilingüe y, en los últimos años, la desambiguación de los sentidos de las palabras. Su complejidad y aplicabilidad ha provocado que se considere, desde hace muchos años, objeto de estudio como tarea independiente.

En la actualidad existen múltiples aproximaciones para el establecimiento de las relaciones entre las palabras de corpus bilingües. El algoritmo propuesto combina el enfoque estadístico y el lingüístico, a través de la incorporación de recursos que refuercen o debiliten las alineaciones; a saber, se empleó información morfológica, equivalencias léxicas de traducción, dominios semánticos y cognados. Como diccionario léxico multilingüe se usó MultiWordNet, aprovechando la ventaja que su concepción ofrece: las redes de palabras que lo conforman se encuentran alineadas.

El alineador implementado está dividido en cuatro módulos principales: (1) pre - procesamiento, (2) configuración, (3) resolución y (4) resultados. En el primero se lleva a cabo el análisis de los archivos de entrada, la optimización, tokenización, el análisis morfológico y el etiquetado de ambos textos. En la etapa de configuración se establecen los parámetros de las técnicas (estadísticas y lingüísticas) que serán aplicadas en el módulo siguiente. La fase de resolución comprende la ejecución de los métodos para el establecimiento de las correspondencias entre las palabras del bitexto. Los resultados se presentan en el último módulo con cuatro formatos diferentes: XML, matrices de alineación, señalización dinámica y diccionarios de alineación.

En los experimentos realizados se usaron fragmentos en español e inglés de las novelas *Don Quijote de la Mancha* y *Guerra y paz* y se efectuaron en dos direcciones. Para los estudios en la primera dirección, se fueron incorporando recursos lingüísticos y analizando su impacto en la calidad de los resultados. La segunda estuvo enfocada a la comparación de los resultados obtenidos por el algoritmo propuesto (mejor y peor desempeño) con los arrojados por otras dos herramientas de alineación (GIZA++

y K - Vec original), configuradas a similitud. Los resultados en ambos casos fueron comparados con un gold estándar, producido por anotadores humanos.

6.1. Aportaciones

Entre las principales aportaciones de este trabajo de investigación se pueden plantear las siguientes:

- Obtención de un nuevo método de alineación de textos paralelos a nivel de palabras implementado en un alineador de textos paralelos de fácil configuración y utilización, que proporciona una visualización flexible de resultados.
- Ampliación del uso de técnicas híbridas para la alineación de textos paralelos (técnicas estadísticas en conjunto con técnicas lingüísticas).
- Extracción de equivalencias léxicas de traducción a partir de redes de palabras alineadas y la utilización del repositorio de pares léxicos obtenidos, señalando la forma en que esta información es susceptible de ser usada en el sistema de alineación a nivel de palabras.
- Inclusión de explicaciones en el ámbito de la alineación de textos paralelos, mediante etiquetas que indican la técnica con la cual fue obtenido cada par, lo que permite una interpretación más certera de los resultados y facilita el ajuste manual de los diccionarios y/o archivos generados.
- Un primer paso en algunas líneas de investigación como la utilización de dominios semánticos para apoyar a la alineación de textos, basándose en la diferención que ofrecen las categorías a la que pertenece cada elemento léxico.
- Identificación de falsos cognados por diferenciación de significados provistos por MWN y su incorporación a un diccionario de referencia para el aprendizaje.

6.2. Respuestas a las preguntas de investigación

- *¿Existen grandes diferencias entre utilizar un algoritmo que usa información lingüística de los que no la usan?* Las diferencias son muy claras al comparar los resultados obtenidos por ambos tipos de algoritmos. Mientras que los estadísticos poseen mayor cobertura, al establecer un mayor número de pares de alineación, los lingüísticos son significativamente más precisos. Esto se debe a la propia naturaleza de las técnicas. Las estadísticas basan su proceso en los conteos de apariciones de palabras y su distribución en el texto y esto hace que se comprometa la certeza de las correspondencias.

- *¿Cuál es el la relación costo / beneficio del utilizar algoritmos más complejos que utilizan información lingüística comparado con los que no los utilizan?* Actualmente la relación costo / beneficio se ha visto favorecida gracias al mejoramiento de las capacidades de cómputo de los nuevos equipos. Un algoritmo híbrido que se tardaba, hace 2 años, en realizar la alineación aproximadamente 1:30 minutos, con ajustes de optimización y con el poder de cómputo actual tarda aproximadamente 5 minutos. Esto nos permite pensar que los algoritmos híbridos serán más populares que cualquiera de las dos versiones anteriores funcionando de manera independiente. Por lo tanto, se puede concluir que el beneficio es mayor que el costo de los recursos utilizados y el tiempo requerido para obtener una respuesta.

- *¿Cómo deben ser las entradas para que un algoritmo de alineación a nivel de palabras funcione de manera óptima y por consecuencia se obtengan mejores resultados?* En la obtención de las mejores alineaciones intervienen factores que no están directamente relacionados con la configuración del algoritmo, como el tamaño del texto y su dominio de aplicación y la distribución de las palabras en los mismos. Sin embargo, sí se podrían establecer algunas consideraciones que permiten incrementar la precisión de los resultados obtenidos. Entre éstas se pueden mencionar: que los textos de entrada estén alineados a nivel de sentencias, que se cuente con un lexicón especializado para cada uno de los idiomas involucrados cuya concepción permita la concordancia de significados (como las redes alineadas en MultiWordNet y el índice interlingua en EuroWordNet), que se utilicen heurísticas de cognación y que los textos sean extensos.

- *¿Qué tan funcionales y sencillas son las interfaces que existen actualmente para mostrar los resultados de la alineación?* Una parte descuidada de la alineación de palabras es la presentación de los resultados. En los desarrollos que implementan un algoritmo de este tipo se muestran los pares de alineación en un único formato, que además, en la mayoría de las ocasiones resulta poco amigable y de difícil comprensión para los usuarios. La aplicación propuesta muestra los resultados en cuatro esquemas distintos: XML, matrices de alineación, señalización dinámica y diccionarios de alineación y en conjunto presenta información de la forma en que se establecieron las correspondencias, lo que facilita la interpretación de los resultados.

- *¿Cuáles serían los recursos lingüísticos más importantes sobre los cuales podemos apoyar al algoritmo de alineación?* Los recursos con mayor impacto son las equivalencias léxicas de traducción, cuyo punto de partida son las redes de palabras alineadas como en MultiWordNet para distintos idiomas, y el empleo de heurísticas de cognación con algoritmos que identifiquen falsos cognados.

- *¿Sería conveniente que un alineador utilizara diferentes métodos para después combinar los resultados y obtener una mejor alineación?* Sí, es importante poder tener distintas configuraciones para obtener distintas versiones de diccionarios resultantes con palabras alineadas y en fase posteriormente poder combinarlas

para mejorar o reforzar dichos pares de alineación. También es útil seleccionar la opción que permite la ejecución del algoritmo de manera bidireccional (tomando el texto origen como texto destino y viceversa)

- *¿Cuál sería un buen método de alineación sobre el cual basarse para comenzar a trabajar?* Al contrario de como se ha trabajado hasta el momento de tener un método estadístico apoyado de uno lingüístico, después de las pruebas realizadas y los resultados obtenidos, es aconsejable que un método lingüístico sea la base de un método de alineación, para después apoyarlo con técnicas estadísticas, e incluso, otras lingüísticas.

6.3. Conclusiones

Incluso cuando al algoritmo HWA podría complementarse con otros recursos, tanto estadísticos como lingüísticos, los resultados obtenidos permiten tener buenas expectativas en la obtención de relaciones correctas entre palabras. Cuando el algoritmo se ejecuta con el K - Vec modificado en forma exclusiva, como se hizo en ambas aproximaciones de los experimentos, los resultados son comparables con los obtenidos por herramientas de alineación reconocidas. Sin embargo, los mejores resultados se consiguen con la incorporación de las equivalencias léxicas de traducción y el algoritmo de identificación de cognados válidos. Si se analizan las Figuras 5.5 y 5.6, es posible percatarse de que la mayoría de los pares correctos (61.5 % del total para *Don Quijote de la Mancha* y 69.4 % para *Guerra y paz*), han sido alineados por la primera técnica lingüística.

La Figura 6.1 es un resumen de los resultados obtenidos en el capítulo anterior. Se han promediado los valores de precisión de ambas novelas, para contar con un total único de representación de cada técnica aplicada. Se han incluido también los resultados de los alineadores con los cuales se han comparado los métodos propuestos: K-Vec (KV) y GIZA.

La principal desventaja del uso de las técnicas lingüísticas es el gran consumo de recursos (lo cual provoca obviamente lentitud de respuesta del algoritmo), en contraparte con los buenos alineamientos a nivel de palabra que se consiguen. Esta relación costo - beneficio hace que se procure alto énfasis en la configuración particular del algoritmo, con el fin de obtener los mejores alineamientos.

Es importante resaltar que la mayoría de los algoritmos optan por sólo uno de los dos enfoques, ya sea el estadístico o el lingüístico, para realizar la tarea de alineación de textos paralelos en cualquiera de sus niveles. Son pocos algoritmos los que optan por un enfoque híbrido, siendo que éstos emplean únicamente dos métodos, uno de cada enfoque.

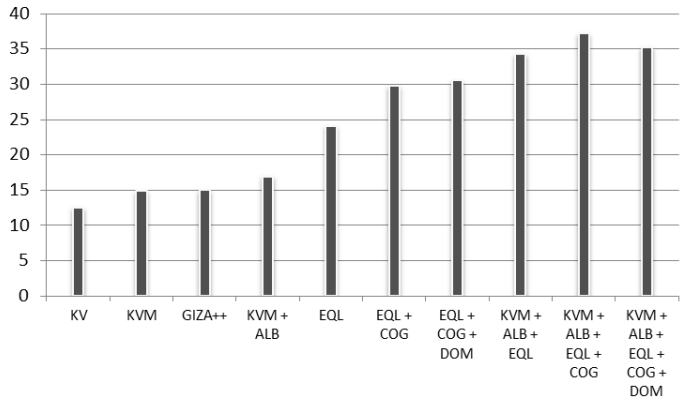


Figura 6.1: Precisión por técnica propuesta o alineador

La unificación de técnicas estadísticas y lingüísticas es una opción viable gracias a las capacidades de cómputo actuales y serán más aceptadas conforme transcurra el tiempo, aumenten las velocidades de procesamiento y disminuyan los costos.

El algoritmo HWA intenta ser uno de los primeros algoritmos que incorporan gran cantidad de métodos y permiten su combinación, en pro del perfeccionamiento de las relaciones obtenidas durante la alineación.

En cuanto a flexibilidad se refiere, los algoritmos actuales no permiten (o es limitada) la configuración de sus procedimientos y sus salidas poseen un formato único. HWA en cambio, es altamente configurable en sus procesos y proporciona la salida en diferentes formatos para el fácil entendimiento por parte de sus usuarios.

El uso de equivalencias léxicas de traducción e información de dominios semánticos puede ser muy útil para apoyar el proceso de alineación de palabras, aunque por el momento sea un enfoque poco utilizado. Dicha técnica ha sido considerada por el algoritmo HWA, marcando un primer paso en su utilización para el proceso de alineación de palabras.

El alineador HWA trata de abrir nuevos horizontes en cuanto a los sistemas de alineación híbridos, sobre los cuales se acrecentará el estudio para poder identificar las mejores configuraciones de los algoritmos y de esta forman obtener los mejores resultados, en dependencia de las entradas y los recursos disponibles.

6.4. Trabajos futuros

Como consecuencia del desarrollo y aplicación del sistema de alineación se han identificado ciertas necesidades que brindarán ventajas sobre los resultados alcanzados

y servirán para darle continuidad a los mismos:

- Agregar diversas pruebas de asociación para calcular la similaridad entre entidades.
- Mejorar y actualizar los diccionarios de manera continua.
- Mejorar el sistema explicatorio y el módulo de aprendizaje para un óptimo funcionamiento.
- Realizar un análisis para encontrar las mejores configuraciones en dependencia del tamaño de entrada de los textos paralelos y los recursos disponibles.
- Aumentar la capacidad del alineador HWA para que pueda encontrar alineaciones no sólo a nivel de palabras.
- Mejorar los módulos, información y herramientas de procesamiento para ampliar la cantidad de idiomas sobre los cuales el algoritmo pueda trabajar en la búsqueda de relaciones en los textos paralelos.

Bibliografía

[1] E. Agirre, A. Díaz, G. Labaka, and K. Sarasola. Uso de información morfológica en el alineamiento español-euskera. *Procesamiento del lenguaje natural*, (37):257–264, 2006.

[2] L. Borin. You´ll take the high road and i´ll take the low road: Using a third language to improve bilingual word alignment. In *In Proceedings of the 18th COLING*, pages 97–103, Saarbrücken, Germany, 2000.

[3] R. Moore. A discriminative framework for bilingual word alignment. In *In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 81–88, Vancouver, Canada, 2005.

[4] R. Mihalca and T.Pedersen. An evaluation exercise for word alignment. In *In Proceedings of the HLT-NAACL 2003 Workshop: Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 1–10, 2003.

[5] E. Macklovitch and M.L. Hannan. Line 'em up: Advances in alignment technology and their impact on translation support tools. *Machine Translation*, pages 41–57, 1998.

[6] C.G. Nevill and T.C. Bell. Compression of parallel texts. *Information Processing and Management: an International Journal*, 28(6):781–793, 1992.

[7] P. Procházková and T. Ramírez. Fundamentos de la lingüística de corpus: Concepción de los corpus y métodos de investigación con corpus. http://www.prochazkova.de/fundamentos_de_la_lingüística_de_corpus.pdf, 2006. Consultado 09/12/10.

[8] J. Hallebeek. El corpus paralelo. *Procesamiento del lenguaje natural*, (24):49–56, 1999.

[9] X. Gómez. Procesamiento y aplicaciones de los corpus paralelos. *Novática: Revista de la Asociación de Técnicos de Informática*, (175):50–54, 2005.

[10] M. Mikhailov. Parallel corpus aligning: Illusions and perspectives. *The Austrian Academy Corpus*, 2002.

[11] J.A. Vera and G. Sidorov. Proyecto de preparación del corpus paralelo alineado español-inglés. In *En Memorias del Encuentro Internacional de la Ciencias de la Computación*, Mexico, 2004.

[12] G. Chinnappa and A.K. Singh. A java implementation of an extended word alignment algorithm based on the ibm models. In *In Proceedings of the 3rd Indian International Conference on Artificial Intelligence*, pages 1897–1913, Pune, India, 2007.

[13] S. Vogel, H. Ney, and C. Tillman. Hmm-based word alignment in statistical translation. In *In Proceedings of the 16th conference on Computational Linguistics*, pages 836–841, Copenhagen, Denmark, 1996.

[14] W.A. Gale and K.W. Church. A program for aligning sentences in bilingüal corpora. In *In Proceedings of the Annual Meeting of the Association for Computational Linguistics*, Berkeley, California, 1991.

[15] P. Strömbäck. The impact of lemmatization in word alignment. Master thesis, Uppsala University, Uppsala, Sweden, 2005.

[16] J. Tiedemann. Word to word alignment strategies. In *In Proceedings of the 20th International Conference on Computational Linguistics*, pages 212–218, Geneva, Switzerland, 2004.

[17] M. Simard, G. Foster, and P. Isabelle. Using cognates to align sentences in parallel corpora. In *In Proceedings of the 4th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 67–81, Montreal, Canada, 1992.

[18] G. Kondrak, D. Marcu, and K. Knight. Cognates can improve statistical translation models. In *In Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 246–48, Edmonton, Canada, 2003.

[19] M. Simard, G.F. Foster, and P. Isabelle. Using cognates to align sentences in bilingual corpora. In *In Proceedings of the 4th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 67–81, Montreal, Canada, 1992.

[20] G. Mann and D. Yarowsky. Multipath translation lexicon induction via bridge languages. In *In Proceedings of NAACL 2001: 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 151–158, Pennsylvania, United States, 2001.

[21] Y. Al-Onaizan, J. Curin, M. Jahr, K. Knight, J. Lafferty, D. Melamed, F. Och, D. Purdy, N. Smith, and D. Yarowsky. Statistical machine translation. Technical report, Johns Hopkins University, 1999.

[22] G. Kondrak. Identifying cognates by phonetic and semantic similarity. In *In Proceedings of NAACL 2001: 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 103–110, Pennsylvania, United States, 2001.

[23] F. Och and H. Ney. A systematic comparison of various statistical alignment models. In *In Proceedings of the 18th Conference on Computational Linguistics - Volume 2*, pages 19–51, Saarbrücken, Germany, 2003.

[24] E. Pianta and L. Bentivogli. Knowledge intensive word alignment with knowa. In *In Proceedings of the 20th International Conference on Computational Linguistics*, pages 1086–1092, Geneva, Switzerland, 2004.

[25] Y. Ma, S. Ozdowska, Y. Sun, and A. Way. Improving word alignment using syntactic dependencies. In *In Proceedings of the ACL-08: HLT Second Workshop on Syntax and Structure in Statistical Translation*, pages 69–77, Ohio, United States, 2008.

[26] George A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[27] Eurowordnet.
http://www.illc.uva.nl/EuroWordNet/, 2001.
Consultado 23/04/11.

[28] Multiwordnet.
http://multiwordnet.itc.it/english/home.php, 2001.
Consultado 23/04/11.

[29] F. Och and H. Ney. Improved statistical alignment models. In *In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong, China, 2000.

[30] I.D. Melamed. Bitext maps and alignment via pattern recognition. *Computational Linguistics*, 1(25):107–130, 1999.

[31] T. Pedersen and N. Varma. K-vec++: Approach for finding word correspondences.
http://www.d.umn.edu/ tpederse/Code/Readme.K-vec++.v02.txt, 2002.
Consultado 09/12/10.

[32] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[33] R. Moore. Association-based bilingual word alignment. In *In Proceedings of the ACL Workshop on Building and Using Parallel Texts ParaText '05*, pages 1–8, Michigan, United States, 2005.

[34] D. Melamed. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249, 2000.

[35] P. Brown, V. Della, S. Della, and R. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[36] Y. Wang and A. Waibel. Decoding algorithm in statistical machine translation. In *In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 366–372, Madrid, Spain, 1997.

[37] Y. Wang and J. Carbonell. Grammar inference and statistical machine translation. Technical report, Carnegie Mellon University, Language Technologies Institute, School of Computer Science, 1998.

[38] D. Hiemstra. Using statistical methods to create a bilingual dictionary. Master thesis, University of Twente, Enschede, Netherlands, 1996.

[39] A. Brandao. Parallel corpora word alignment and applications. Master thesis, Universidade do Minho, Braga, Portugal, 2004.

[40] D. Melamed. A portable algorithm for mapping bitext correspondence. In *In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 305 – 312, Madrid, Spain, 1997.

[41] M. Julapalli and S. Dhond. Word alignment in bilingual parallel corpora. Technical report, 2003.

[42] P. Fung and K.W. Church. K-vec: A new approach for aligning parallel texts. In *In Proceedings from the 15th International Conference on Computational Linguistics*, pages 1096–1102, Kyoto, Japan, 1994.

[43] M. Popovic and H. Ney. Improving word alignment quality using morpho-syntactic information. In *In Proceedings of the 20th international conference on Computational Linguistics*, pages 310–314, Geneva, Switzerland, 2004.

[44] S. Chen. Aligning sentences in bilingual corpora using lexical information. In *In Proceedings of the 31st Annual Meeting of Association for Computational Linguistics*, pages 9–16, Ohio, United States, 1993.

[45] J. Tiedemann. Combining clues for word alignment. In *In Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, pages 339 – 346, Budapest, Hungary, 2003.

[46] R. Moore, W. Yih, and A. Bode. Improved discriminative bilingual word alignment. In *In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 513 – 520, Sydney, Australia, 2006.

[47] C. Goutte, K. Yamada, and E. Gaussier. Aligning words using matrix factorisation. In *In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 502 – 509, Barcelona, Spain, 2004.

[48] E. Matusov, R. Zens, and H. Ney. Symmetric word alignments for statistical machine translation. In *In Proceedings of the 20th International Conference on Computational Linguistics*, pages 219 – 225, Geneva, Switzerland, 2004.

[49] M. Haruno and T. Yamazaki. High-performance bilingual text alignment using statistical and dictionary information. In *In Proceedings of the 34th Annual Meeting of Association for Computational Linguistics*, pages 131 – 138, California, United States, 1996.

[50] P. Blunsom and T. Cohn. Discriminative word alignment with conditional random fields. In *In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 65 – 72, Sydney, Australia, 2006.

[51] W. Hua and W. Haifeng. Improving statistical word alignment with a rule-based machine translation system. In *In Proceedings of the 20th International Conference on Computational Linguistics*, pages 29 – 35, Geneva, Switzerland, 2004.

[52] L. Ahrenberg, M. Andersson, and M. Merkel. A simple hybrid aligner for generating lexical correspondences in parallel texts. In *In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 29 – 35, Quebec, Canada, 1998.

[53] A. Fraser and D. Marcu. Semi-supervised training for statistical word alignment. In *In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 769 – 776, Sydney, Australia, 2006.

[54] P. Liang, B. Taskar, and D. Klein. Alignment by agreement. In *In Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 104 – 111, New York, United States, 2006.

[55] S. Venkatapathy and A. Joshi. Using information about multi-word expressions for the word-alignment task. In *In Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 20 – 27, Sydney, Australia, 2006.

[56] G. Mann and D.Yarowsky. Multipath translation lexicon induction via bridge languages. In *In Proceedings of NAACL 2001: 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 151–158, Pennsylvania, United States, 2001.

[57] G.W. Adamson and J. Boreham. *The use of an association measure based on character structure to identify semantically related pairs of words and document titles*. Information Storage and Retrieval, 1974.

[58] M.A. Covington. An algorithm to align words for historical comparison. *Computational Linguistics*, 4(22):481–496, 1996.

[59] J. Nerbonne and W. Heeringa. Measuring dialect distance phonetically. In *In Proceedings of the Third Meeting of the ACL Special Interest Group in Computational Phonology*, pages 11 – 18, Madrid, Spain, 1997.

[60] W.H. Vieregge, A.C.M.Rietveld, and C. Jansen. A distinctive feature based system for the evaluation of segmental transcription in dutch. In *In Proceedings of the l0th International Congress of Phonetic Sciences*, pages 654–659, Utrecht, the Netherlands, 1984.

[61] J.H. Connolly. Quantifying target realization differences. *Clinical Linguistics & Phonetics*, 11:267–298, 1997.

[62] G. Kondrak. A new algorithm for the alignment of phonetic sequences. In *In Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 288 – 295, Washington, United States, 2000.

[63] The uplug home page.
http://stp.lingfil.uu.se/ joerg/Uplug/home.html, 2006.
Consultado 09/12/10.

[64] J. Tiedemann. Corpora: Pwa - word alignment software available.
http://torvald.aksis.uib.no/corpora/2001-1/0088.html, 2001.
Consultado 08/10/11.

[65] F.J. Och. Giza++.
http://www.fjoch.com/GIZA++.html, 2003.
Consultado 08/10/11.

[66] A. Barbu. Simple linguistic methods for improving a word alignment algorithm. In *In Proceedings of the 7th International Conference on Statistical Analysis of Textual Data*, pages 88–98, Louvain-la-Neuve, Belgium, 2004.

[67] Ll. De Yzaguirre. L'etiquetador palic i el desambiguador ambilic. Jornades del Centre de Referència en Enginyeria Lingüística (CREL). IEC. Barcelona, 2000.

[68] J. Morel, S. Torner, J. Vivaldi, Ll. De Yzaguirre, and M.T. Cabré. El corpus de l'iula: etiquetaris. Technical report, Universitat Pompeu Fabra, Institut Universitari de Lingüistica Aplicada, 1998.

[69] J. Vivaldi, Ll. De Yzaguirre, X. Solé, and M.T. Cabré. Marcatge estructural i morfosintactic del corpus tecnic amb l'estandard sgml. Technical report, Universitat Pompeu Fabra, Institut Universitari de Lingüistica Aplicada, 1996.

[70] A. Alonso, M.T. Cabré, Ll. De Yzaguirre, and C. Tebé. La utilización de corpus paralelos alineados en la docencia de la traducción y de los lenguajes de especialidad. In *In Proceedings of the Second International Contrastive Linguistics Conference*, pages 71–82, Santiago de Compostela, Spain, 2002.

[71] D. Hiemstra. Multilingual domain modeling in twenty-one: automatic creation of a bi-directional lexicon from a parallel corpus. Technical report, University of Twente, Parlevink Group, 1998.

[72] B. Rassier and T. Pedersen. Alpaco aligner for parallel corpora. http://www.d.umn.edu/ tpederse/Code/Readme.Alpaco-v0.3.txt, 2003. Consultado 09/12/10.

[73] J. Tiedemann. Isa & ica: Interactive alignment of bitexts. http://www.let.rug.nl/ tiedeman/uplug-demo/, 2003. Consultado 09/12/10.

[74] E. Pianta and L. Bentivogli. Knowledge intensive word alignment with knowa. In *In Proceedings of the 20th international conference on Computational Linguistics*, pages 1086 – 1092, Geneva, Switzerland, 2004.

[75] B. Thomson and T. Pedersen. The duluth word alignment system. In *In Proceedings of the HLT-NAACL 2003 Workshop: Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 40 – 43, Edmonton, Canada, 2003.

[76] H. Wu, H. Wang, and Z. Liu. Alignment model adaptation for domain-specific word alignment. In *In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 467–474, Michigan, United States, 2005.

[77] A. Gispert, J. Mariño, and J. Crego. Phrase-based alignment combining corpus coocurrences and linguistic knowledge. In *In Proceedings of the International Workshop on Spoken Language Translation IWSLT 2004*, pages 107 – 114, Kyoto, Japan, 2004.

[78] K. Bonawitz, A. Kim, and S. Tardiff. An architecture for word learning using bidirectional multimodal structural alignment. In *In Proceedings of the HLT-NAACL 2003 Workshop on Learning word meaning from non-linguistic data*, pages 30–37, Edmonton, Canada, 2003.

[79] T. Okita, A. Maldonado, Y. Graham, and A. Way. Multi-word expression-sensitive word alignment. In *In Proceedings of the Fourth International Workshop On Cross Lingual Information Access*, pages 1–8, Beijing, China, 2010.

[80] K. Macherey J. DeNero. Model-based aligner combination using dual decomposition. In *In Proceedings of 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 420–429, Oregon, USA, 2011.

[81] B. Magnini and G. Cavaglia. Integrating subject field codes into wordnet. In *In Proceedings of the 2nd International Conference on Language Resources & Evaluation*, pages 1413–1418, Athens, Greece, 2000.

[82] S. Nakov, P. Nakov, and E. Paskaleva. Cognate or false friend? ask the web. In *In Proceedings of the RANLP2007 workshop: Acquisition and Management of Multilingual Lexicons*, pages 55–67, Borovets, Bulgaria, 2007.

[83] O.M. Frunza. Automatic identification of cognates, false friends, and partial cognates. Master thesis, University of Ottawa, 2006.

[84] A. Fraser and D. Marcu. Measuring word alignment quality for statistical machine translation. *Computational Linguistics*, 33(3):293 – 303, 2007.

[85] H.H. Do, S. Melnik, and E. Rahm. Comparison of schema matching evaluations. In *Proceedings of the GI-Workshop Web and Databases*, pages 221–237, Erfurt, Germany, 2002.

[86] L. Ahrenberg, M. Merkel, A.S. Hein, and J. Tiedemann. Evaluating word alignment systems. In *In Proceedings of the Second International Conference on Linguistic Resources and Evaluation*, pages 1255 – 1261, Athens, Greece, 2000.

Apéndice A

Pares correctos de K-Vec y GIZA para *Don Quijote de la Mancha*

En la Figura A.1 están representados los pares correctos propuestos por los alineadores versus el gold standard establecido por los anotadores y el topline.

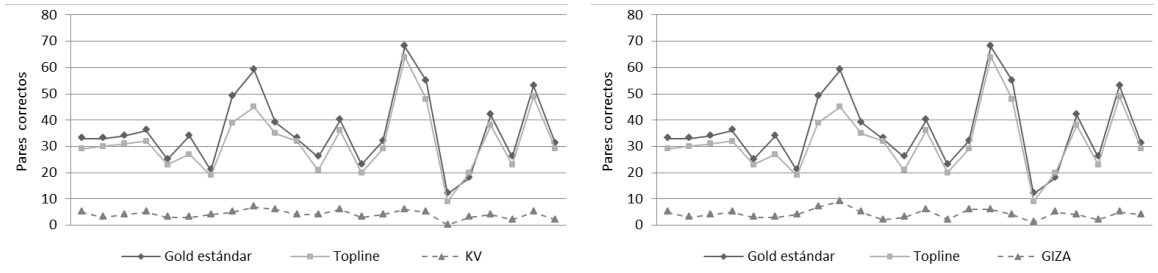


Figura A.1: Cantidad de pares correctos de K-Vec y GIZA para *Don Quijote de la Mancha*

Apéndice B

Pares correctos de K-Vec y GIZA para *Guerra y paz*

En la Figura B.1 están representados los pares correctos propuestos por los alineadores versus el gold standard establecido por los anotadores y el topline.

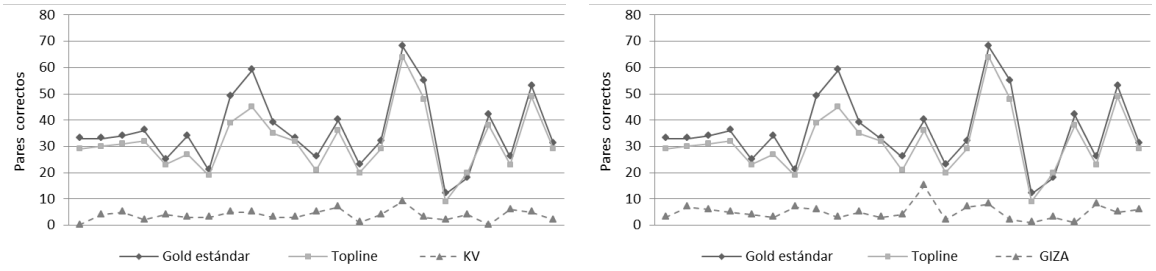


Figura B.1: Cantidad de pares correctos de K-Vec y GIZA para *Guerra y paz*

Apéndice C

Pares correctos para palabras de contenido de K-Vec y GIZA en *Don Quijote de la Mancha*

En la Figura C.1 están representados los pares correctos para palabras de contenido propuestos por los alineadores versus el gold standard establecido por los anotadores y el topline.

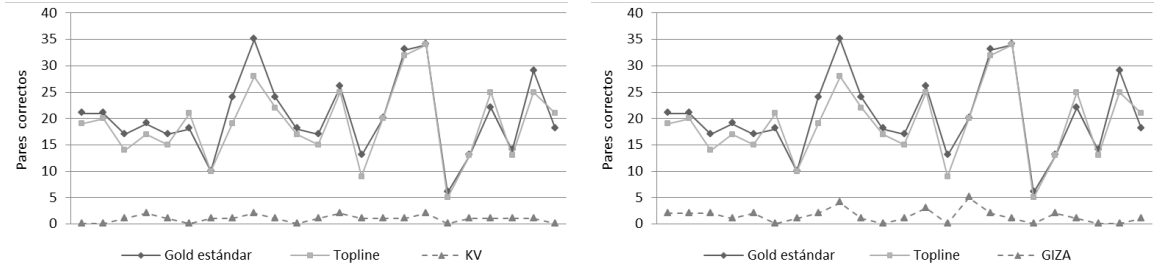


Figura C.1: Cantidad de pares correctos de K-Vec y GIZA para palabras de contenido en *Don Quijote de la Mancha*

Apéndice D

Pares correctos para palabras de contenido de K-Vec y GIZA en *Guerra y paz*

En la Figura D.1 están representados los pares correctos para palabras de contenido propuestos por los alineadores versus el gold standard establecido por los anotadores y el topline.

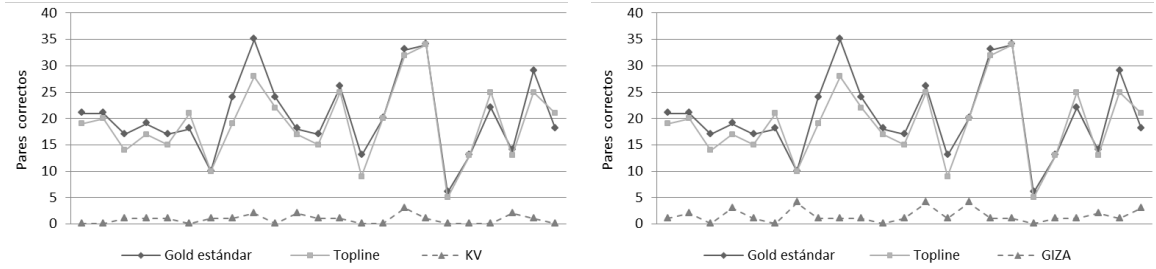


Figura D.1: Cantidad de pares correctos de K-Vec y GIZA para palabras de contenido en *Guerra y paz*