



**Benemérita Universidad Autónoma de Puebla**

Facultad de Ciencias de la Computación

**Técnicas de análisis semántico  
para el tratamiento automático  
de implicación textual**

Que para obtener el grado de  
Maestro en Ciencias de la Computación

PRESENTA

**Saúl León Silverio**

Asesor: Darnes Vilariño Ayala  
Coasesor: David E. Pinto Avendaño



Diciembre de 2014



## Índice General

<b>Capítulo 1: Introducción</b> .....	5
1.1 Planteamiento de la Investigación .....	5
1.2 Definición del problema .....	6
1.3 Objetivos de la investigación .....	6
1.4 Justificación de la investigación .....	7
1.5 Preguntas de investigación .....	8
1.6 Aportaciones de la investigación .....	8
1.7 Organización de la tesis .....	9
1.8 Conclusiones del capítulo .....	9
<b>Capítulo 2: Estado del arte del CLTE</b> .....	10
2.1 Visión general para abordar el CLTE .....	10
2.2 Técnicas de Pivote con algoritmos de aprendizaje automático .....	12
2.3 Técnicas de Pivote .....	15
2.4 Técnicas Cross-Lingüe .....	15
2.5 Técnicas Híbridas .....	17
2.6 Metodologías de Textual Entailment .....	17
2.5 Conclusión del capítulo .....	19
<b>Capítulo 3: Fundamentos de la investigación</b> .....	20
3.1 Metodologías empleadas para solucionar el CLTE .....	20
3.1.1 Sistemas Cross-lingüe .....	20
3.1.2 Sistemas de Pivote .....	21
3.2 Extracción de características .....	22
3.2.1 Características a Nivel Léxico .....	23
3.2.2 Características a Nivel Semántico .....	24
3.2.3 Características a Nivel Sintáctico .....	24
3.3 Algoritmos de aprendizaje automático .....	25
3.4 Técnicas de TE .....	26
3.4.1 Muestreo de plantillas .....	27
3.4.2 Similitud de Grafos .....	27



3.4.3 Evaluación Lógica .....	28
3.4.4 Alineación de Oraciones (Autómatas) .....	28
3.4.5 Híbridos .....	30
3.5 Recursos y herramientas .....	31
3.5.1 Colecciones de Datos Empleadas.....	31
3.5.2 Lematizadores y Truncadores .....	31
3.5.3 Etiquetados de partes del discurso .....	32
3.5.4 Parsers Sintácticos .....	32
3.5.5 Sinónimos, Antónimos, Hipónimos e Hiperónimos .....	33
3.5.6 Herramienta de aprendizaje supervisado .....	33
3.5 Conclusión del capítulo.....	33
<b>Capítulo 4: Metodología .....</b>	<b>34</b>
4.1 Modelo de Conteo Estadístico .....	34
4.1.1 Características Elegidas.....	35
4.1.2 Montaje del experimento .....	36
4.2 Modelo de similitud semántica.....	36
4.3.1 Características Elegidas.....	37
4.3 Modelos de Eliminación de Tokens .....	40
4.3.1 Eliminación de elementos comunes para detectar la implicación textual .....	41
4.3.2 Tokenización de oraciones.....	41
4.3.3 Criterio de similitud entre tokens .....	42
4.3.4 Montaje del experimento .....	43
4.4 Modelo de eliminación con análisis semántico .....	44
4.4.1 Criterio de similitud entre palabras .....	44
4.4.2 Montaje del experimento .....	45
4.5 Modelo de inferencia basado en anclas.....	46
4.5.1 Términos Anclas .....	46
4.5.2 Proceso de inferencia .....	47
4.6 Modelo de interpretación de oraciones basado en grafos .....	49
4.6.1 Extracción de hechos .....	51
4.6.2 Interpretación de hechos sobre grafos .....	53
4.6.3 Empatamiento de grafos para el descubrimiento de la implicación textual .....	56



4.6.5 Montaje del experimento .....	57
<b>Capítulo 5: Resultados experimentales</b> .....	<b>60</b>
5.1 Colecciones de Datos .....	60
5.2 Resultados del Modelo de Conteo Estadístico .....	62
5.3 Resultados del Modelo de Similitud Semántica .....	62
5.4 Resultados del Modelo de Eliminación de Tokens .....	63
5.5 Resultados del Modelo de Eliminación con Análisis Semántico .....	65
5.6 Resultados de Modelo de Inferencia Basado en Anclas .....	65
5.7 Resultados del Modelo de Interpretación de Oraciones Basados en Grafos .....	65
5.8 Comparativa de Desempeño entre Modelos .....	66
<b>Capítulo 6: Conclusiones y Trabajo a Futuro</b> .....	<b>71</b>
<b>Referencias</b> .....	<b>73</b>
<b>Anexo 1: Estudio preliminar realizado a la colección de datos</b> .....	<b>75</b>
1.1 Colecciones de Datos .....	75
1.2 Distribución del Corpus .....	75
<b>Anexo 2: Resultados del Modelo de Eliminación con Análisis Semántico, con diferentes medidas de similitud</b> .....	<b>81</b>
<b>Anexo 3: Resultados del Modelo de Inferencia Basado en Anclas</b> .....	<b>89</b>



## Capítulo 1: Introducción

La vasta cantidad de información disponible en Internet hoy en día supera nuestra capacidad de almacenarla, procesarla y por ende aprovecharla. En los últimos años grandes repositorios de información, como enciclopedias, periódicos, blogs, revistas electrónicas, multimedia y otras fuentes digitales de información han generado contenidos que pueden ser útiles para algún usuario en particular. Sin embargo la disponibilidad de información en varios idiomas, así como la variación léxico-sintáctica del lenguaje suponen un reto a los sistemas de Procesamiento de Lenguaje Natural (PLN) que existen actualmente.

Para explicar los fenómenos de la disponibilidad de información en varios idiomas y la variación léxico-semántica, se analizan los siguientes escenarios:

- A. Alger, un usuario alemán, desea buscar información acerca de la cultura Maya de México, hay una gran cantidad de artículos en idioma inglés que tratan de la cultura Maya, no obstante los artículos que son del interés de Alger se encuentran en español. Alger traduce del idioma alemán al idioma inglés sus inquietudes, pero como se ha notado no halla nada útil y desiste de ser arqueólogo de Chichenitza. Al igual que Alger, miles de usuarios son limitados por la cantidad de información que encuentran en su idioma.
- B. Roberto es un niño que tiene por tarea definir la lluvia, así que busca definiciones de "Lluvia", como buen estudiante, busca en más de una fuente para redactar una definición extensa, él se percata de que todas las definiciones denotan lo mismo, pero las palabras, y la forma en las que están escritas son diferentes. Así como Roberto, muchos usuarios buscan información concreta, pero la manera en que está escrita (léxico y sintaxis) impide que sea de fácil acceso. Roberto imagina un Google mejorado que pueda interpretar los textos que hablan de lo mismo, pero con diferentes palabras y escritos de manera diferente.

Estas dos problemáticas habían sido investigadas por separado hasta hace un par de años, ahora se estudian juntas en un problema denominado "Implicación Textual Cross-Lingüe" (ITCL o CLTE por sus siglas en inglés), cuyo fin es solucionar los escenarios dados en A y en B.

### 1.1 Planteamiento de la Investigación

A lo largo de esta sección se define el problema de investigación, así como objetivos de la tesis. Al final se detalla la estructura de este documento.



## 1.2 Definición del problema

En el año 2012, dentro del marco de la conferencia internacional SemEval-2012, se propone a la comunidad científica internacional resolver el problema CLTE<sup>1</sup>, éste consiste en determinar si un texto T y una hipótesis H, escritos en diferentes idiomas, se puede inferir el significado de H a partir del significado de T. Formalmente :

Dado un par de fragmentos de texto, tópicamente relacionados,  $T_1$  y  $T_2$  escritos en diferentes idiomas, la tarea consiste en asignar automáticamente uno de los siguientes juicios de implicación textual:

- Bidirectional: ( $T_1 \Rightarrow T_2 \ \&\& \ T_2 \Rightarrow T_1$ ) equivalencia semántica.
- Forward: ( $T_1 \Rightarrow T_2 \ \&\& \ !T_2 \Rightarrow T_1$ ) implicación unidireccional de  $T_1$  a  $T_2$ .
- Backward: ( $!T_1 \Rightarrow T_2 \ \&\& \ T_2 \Rightarrow T_1$ ) implicación unidireccional de  $T_2$  a  $T_1$ .
- No-Entailment: ( $!T_1 \Rightarrow T_2 \ \&\& \ !T_2 \Rightarrow T_1$ ) sin implicación entre  $T_1$  y  $T_2$ .

En esta tarea se asume que tanto  $T_1$  y  $T_2$  son declaraciones verdaderas (TRUE) y que no existen pares contradictorios.

Para desarrollar modelos que solucionan el problema de Implicación Textual Cross-Lingüe, los organizadores ponen a disposición de la comunidad de investigadores a nivel internacional un conjunto de datos, para entrenar y probar los mismos (entrenamiento / prueba). Se ofrecen las siguientes combinaciones de idiomas:

- Español / Inglés
- Alemán / Inglés
- Italiano / Inglés
- Francés / Inglés

Para darle solución a este problema se proponen varios modelos, siguiendo cada uno de ellos enfoques diferentes, lo que permite que se planteen los siguientes objetivos.

## 1.3 Objetivos de la investigación

### Objetivo General:

- Desarrollar modelos para detectar la implicación textual cross-lingüe.

---

<sup>1</sup> Una extensión Cross-Lingüe del problema de Implicación Textual



### Objetivos Particulares:

1. Estudiar todas las propuestas de solución del Foro SemEval 2012 y 2013, presentadas por los diferentes equipos participantes.
2. Comparar las metodologías desarrolladas por los equipos para analizar qué elementos permiten mejorar las soluciones presentadas.
3. Proponer diferentes características para representar ambos textos.
4. Desarrollar modelos que utilicen la técnica de pivote con metodologías de Textual Entailment (TE), considerando patrones léxico y semánticos.
5. Desarrollar un modelo para representar los textos utilizando teoría de grafos. Desarrollar una métrica para el empatamiento de los grafos y el descubrimiento del juicio de implicación.
6. Aplicar los modelos desarrollados a los datos ofrecidos en el SemEval 2012, 2013 y 2014.
7. Publicar los resultados de investigación obtenidos.

### 1.4 Justificación de la investigación

La calidad de un sistema de recuperación de información está totalmente relacionada con el grado de cumplimiento de la necesidad de información expresada por el usuario. Dotar a los sistemas de un módulo capaz de inferir cuando dos textos expresan lo mismo, enriquece las posibilidades de satisfacer totalmente a la consulta expresada por el usuario. Es importante destacar que esto aún se vuelve más relevante, si la consulta está en un idioma determinado y lo que se está buscando está en otro.

Otro resultado implícito que se obtiene con esta investigación, es la posibilidad de desarrollar corpus paralelos de manera no supervisada, que ayude a los sistemas que desarrollan modelos de traducción automática.

Los modelos desarrollados hasta el año 2014 no consideran el juicio de implicación textual *contradicción*, que en muchas ocasiones es muy difícil de detectar ya que solamente la presencia o ausencia de una palabra hace que dos textos sean totalmente contradictorios. El desarrollo de modelos para resolver este problema, puede ser incorporado a un módulo de procesamiento de lenguaje natural.

Detectar el juicio de implicación textual *cross lingüe* sin el desarrollo de modelos de traducción permite dar solución al problema sin necesidad de disponer de un corpus paralelo totalmente heterogéneo que permita el desarrollo de diccionarios estadísticos de dominio general y dominio particulares.



La mayoría de los modelos desarrollados para resolver este problema son supervisados, lo que obliga a disponer de un corpus de entrenamiento desarrollado por expertos. En muchas ocasiones no es tan fácil disponer de un corpus categorizado, por lo que es importante desarrollar modelos no supervisados, que ofrezcan buen comportamiento independientemente del corpus con el que se está trabajando.

## 1.5 Preguntas de investigación

Para dar solución a la problemática presentada se propone dar respuesta a las siguientes preguntas de investigación:

- ¿Cuáles son las fortalezas y debilidades de los modelos existentes?
- ¿Qué factores (características) influyen en la asignación de un juicio de implicación?
- ¿Se puede resolver el CLTE con algoritmos de aprendizaje automático?
- ¿Es posible modelar la estructura del lenguaje natural, para el descubrimiento del juicio de implicación?
- ¿Los modelos supervisados se comportan mejor que los modelos no supervisados para darle solución al problema?

## 1.6 Aportaciones de la investigación

Esta tesis ofrece las siguientes aportaciones:

- Modelos que permiten hacer el descubrimiento automático del juicio de implicación textual entre idiomas.
- Desarrollo de un modelo basado en el concepto de anclas para descubrir patrones semánticos en textos.
- Desarrollo de un modelo de representación de textos mediante grafos.
- Desarrollo de motor de extracción de tripletas para el empatamiento de los grafos.
- Diseño e implementación de una medida de empatamiento de los patrones extraídos de los grafos.
- Desarrollo de una base de conocimientos utilizando grafos y enriquecida con los conceptos extraídos de ConceptNet-5, WordNet y OpenOffice Thesaurus. Para extraer los conceptos de esta red semántica se implementó un algoritmo, para detectar si existe una relación entre un concepto y otro.
- Un modelo de eliminación que permite detectar cuando dos textos son equivalentes, introduciendo sinonimia e hiperonimia.



## 1.7 Organización de la tesis

El trabajo de tesis está estructurado en 6 capítulos, los cuales son descritos a continuación:

- Capítulo 1, Introducción: Plantea y define el problema a resolver, así como su importancia y la justificación del mismo.
- Capítulo 2, Estado del arte: Expone las diferentes metodologías existentes empleadas para solucionar el CLTE, se exploran los trabajos que ha realizado la comunidad científica.
- Capítulo 3, Fundamentos de la investigación: Se generalizan las técnicas expuestas en el estado del arte, para buscar cuáles de ellas son mejores y así proponer nuevas metodologías.
- Capítulo 4, Metodología: Explica los cinco modelos propuestos para resolver el problema del CLTE.
- Capítulo 5, Resultados experimentales: Expone los resultados alcanzados por las metodologías planteadas en el capítulo 4, y se discuten los comportamientos observados al modificar aspectos propios de cada modelo.
- Capítulo 6, Conclusiones y Trabajo a futuro: Se hace una reseña del comportamiento de cada metodología propuesta, y se hace contraste entre una y otra. Se presenta el trabajo a futuro.

## 1.8 Conclusiones del capítulo

A lo largo de este capítulo se ha definido el problema de la Implicación Textual Cross-Lingüe, se plantearon las preguntas de investigación y se han propuesto los objetivos, en aras de poder responder a dichas preguntas. Se ha justificado la necesidad del desarrollo de modelos para resolver este problema y se han presentado las aportaciones de la presente investigación.

## Capítulo 2: Estado del arte del CLTE

Este capítulo expone una visión general del CLTE, asimismo se comentan las investigaciones, más sobresalientes, que lo solucionan. A continuación se realiza una reseña de metodologías, herramientas y recursos empleados para solucionar el CLTE.

### 2.1 Visión general para abordar el CLTE

Antes de comenzar con la reseña de las metodologías, es necesario definir algunos criterios generales de las mismas. El CLTE asigna juicios de implicación a un par de sentencias en idiomas diferentes, en la imagen 1 se muestran 4 sentencias de la colección español / inglés, cada una etiquetada con su respectivo juicio de implicación.

Ejemplo de CLTE

```

<entailment-corpus languages="spa-eng">
  <pair id="1" entailment="bidirectional">
    <t1>Mozart nació en la ciudad de Salzburgo.</t1>
    <t2>Mozart was born in Salzburg.</t2>
  </pair>
  <pair id="2" entailment="forward">
    <t1>Mozart nació el 27 de enero de 1756 en Salzburgo.</t1>
    <t2>Mozart was born in 1756 in the city of Salzburg.</t2>
  </pair>
  <pair id="3" entailment="backward">
    <t1>Mozart nació en la ciudad de Salzburgo.</t1>
    <t2>Mozart was born on 27th January 1756 in Salzburg.</t2>
  </pair>
  <pair id="4" entailment="no_entailment">
    <t1>Mozart nació el 27 de enero de 1756 en Salzburgo.</t1>
    <t2>Mozart was born to Leopold and Anna Maria Pertl Mozart.</t2>
  </pair>
</entailment-corpus>

```

Imagen 1: Ejemplo de CLTE con idiomas Inglés/Español

El CLTE ha sido parcialmente resuelto utilizando algoritmos basados en reglas, que toman patrones repetitivos a niveles léxicos y/o sintácticos. Algunas de estas aproximaciones utilizan el conjunto de entrenamiento para poner a punto las reglas que evalúan sobre el conjunto de prueba.



Otra vertiente, quizás la más explotada, utiliza algoritmos de aprendizaje automático, que extraen características léxicas y/o sintácticas para descubrir patrones repetitivos en los 4 diferentes juicios de implicación. En este enfoque se construye un modelo para los datos de entrenamiento, que posteriormente permite evaluar los datos de prueba.

De este hecho se pueden distinguir dos perspectivas, la primera toma los cuatro juicios de implicación como clases, es decir que los modelos perciben 4 clases y por ello se nombra a este enfoque multiclase. El segundo punto de vista descompone las oraciones en sus formas más primitivas, reconociendo sólo 2 clases (implicación y no-implicación), esto quiere decir que para obtener el juicio final se realiza una composición de dos clases como exhibe la Tabla 1, este ajuste es denominado como composición.

Juicio de $T_1 \Rightarrow T_2$	Juicio de $T_2 \Rightarrow T_1$	Juicio final
<i>entailment</i>	<i>entailment</i>	<i>bidirectional</i>
<i>entailment</i>	<i>no_entailment</i>	<i>backward</i>
<i>no_entailment</i>	<i>entailment</i>	<i>forward</i>
<i>no_entailment</i>	<i>no_entailment</i>	<i>no_entailment</i>

**Tabla 1:** Tabla de composición de juicio de implicación

Independientemente de lo anterior, se tienen dos posibles tratamientos derivados del lenguaje del par de oraciones, el primero busca utilizar métodos cross-lingüe que rescatan las técnicas de las investigaciones del área de Machine Translation (MT). Mientras que el segundo enfoque utiliza un idioma como pivote para hacer que las dos sentencias se encuentren en el mismo idioma, una vez hecho esto, el problema pasa de ser CLTE a TE y gracias a ello se puede operar con metodologías del TE. Por su parte *TE data* del año 2005, por lo que utilizar el pivote supone una ventaja en comparación al CLTE.

Recapitulando lo expuesto con anterioridad se puede identificar las siguientes combinaciones de metodologías para resolver el CLTE:

- Cross-lingüe y multiclase
- Cross-lingüe y composición
- Pivote y composición
- Pivote y multiclase
- Cross-lingüe y multiclase con algoritmos de aprendizaje automático
- Cross-lingüe y composición con algoritmos de aprendizaje automático
- Pivote y composición con algoritmos de aprendizaje automático
- Pivote y multiclase con algoritmos de aprendizaje automático

A continuación se discuten los enfoques empleados por los participantes del SemEval-2012 y SemEval-2013, para solucionar el CLTE. A pesar de la existencia de múltiples técnicas, los participantes sólo emplean un pequeño conjunto de ellas, la imagen 2, nos presenta la clasificación de los trabajos más representativos, de acuerdo a las técnicas que usan.

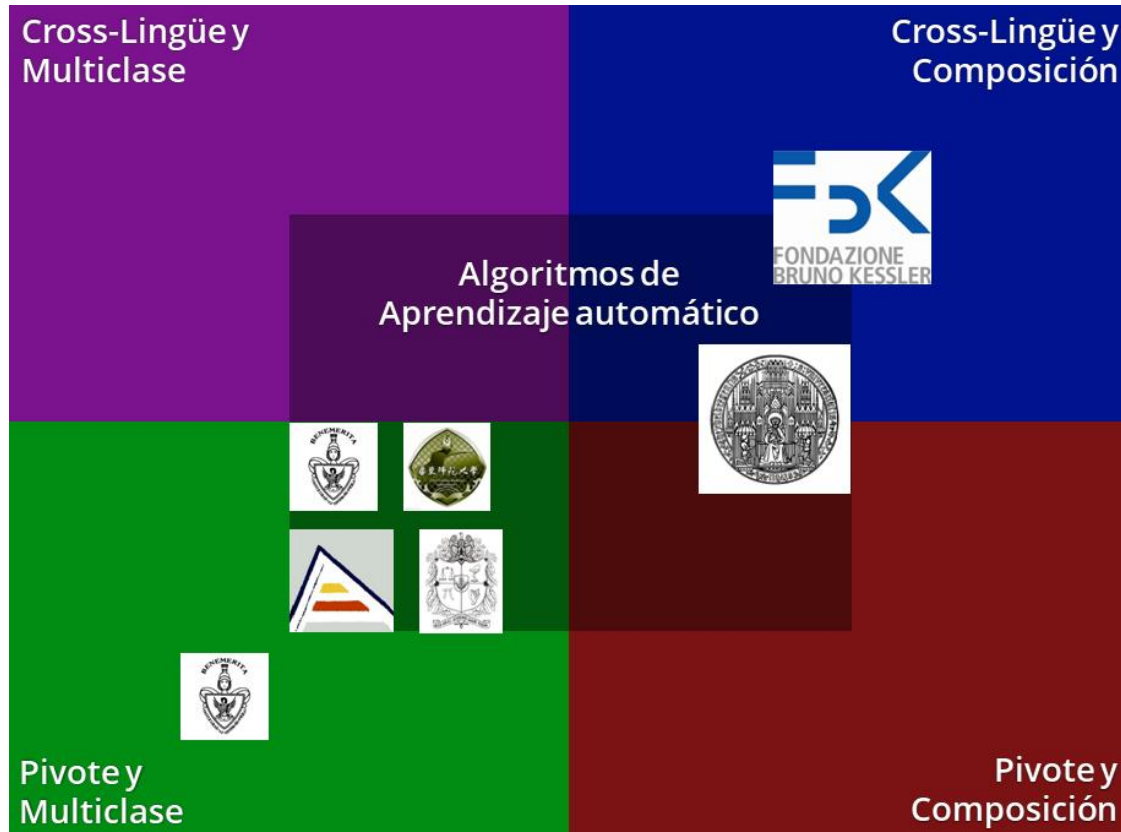


Imagen 2: Tendencias de metodologías para solucionar el CLTE

A continuación se explica con detalle cada una de las técnicas desarrolladas.

## 2.2 Técnicas de Pivote con algoritmos de aprendizaje automático

En [1], se emplea la técnica de pivote y en lugar de garantizar la correcta traducción, se explotan los avances de las Máquinas de Traducción (MT), ya que son eficientes. Los autores sostienen que la diversidad de las MT, así como sus metodologías, pueden ser de ayuda al momento de traducir. Manejan una técnica de pivote con un análisis de n-gramas, que van de 1 hasta la longitud de la oración. Para descubrir el juicio de implicación proponen utilizar una medida de overlap sobre los n-gramas y a través de una fórmula propuesta por ellos logran discernir el juicio.

En cuanto a los unigramas, los autores sostienen que aportan propiedades diferentes (porque son una sola palabra y no hay secuencia de tokens), por esta razón hacen un análisis por separado de ellos. Toman las medidas de los n-gramas junto con la longitud de los pares de



oraciones y se apoyan en el desarrollo de modelos de clasificación, mediante un esquema multiclase y un vector de 97 características.

Las mayoría de las propuestas utilizan metodologías netamente estadísticas, sin embargo existen enfoques que sostienen que trabajar con texto, demanda tener medidas suavizadas, en otras palabras el dinamismo de los textos requiere medidas de similitud que se adapten al contexto del mismo. Un trabajo que aborda esta perspectiva se desarrolla en [2], donde se establece que los términos de las oraciones tienen importancia al momento de hablar del contexto de la oración, los autores justifican este hecho mencionando que el significado de una palabra puede variar dependiendo de la posición en la que se encuentre.

Adicionalmente se expone que la cardinalidad de un par de oraciones denota balance en la información que ambas contienen. Los autores introducen la medida SoftCardinality (Cardinalidad suavizada) que utiliza la edit-distance, para justificar su uso, e ilustran con el siguiente ejemplo:

Sea  $A$  un conjunto tal que  $A = \{\text{"Sunday"}, \text{"Saturday"}\}$ , si se extrae su cardinalidad de forma clásica se obtiene  $|A| = 2$ , no obstante es posible observar que los términos son similares, por esta razón el SoftCardinality establece que  $|A| = 1.23$ .

La investigación utiliza las siguientes medidas:

- Conteo de términos lematizados
- Longest Common Subsequence (LCS)
- SoftCardinality

Finalmente los autores mencionan que también se utiliza una medida de asimetría, esto con el fin de dotar a los vectores con más información relevante. Se entrena un modelo de clasificación utilizando Máquina de Soporte Vectorial (SVM), con un vector de 60 características, cada característica corresponde a medidas de cardinalidad clásica y suavizada, así como medidas de similitud simétrica y asimétrica. El esquema de clasificación empleado es multiclase.

Otra investigación que emplea un enfoque de pivote-multiclase es la desarrollada en [3], donde a partir del par de oraciones en un mismo idioma se confecciona un vector característico con las siguientes métricas:



- Métricas Básicas (Cardinalidades de conjuntos):
  - $|A|$
  - $|B|$
  - $|A-B|$
  - $|B-A|$
  - $|A \cup B|$
  - $|A \cap B|$
  - $|A| / |B|$
  - $|B| / |A|$
  
- Métricas de Similitud
  - Jaccard coefficient
  - Dice coefficient
  - Overlap coefficient
  - Weighted overlap coefficient
  - Cosine similarity
  - Manhattan distance
  - Euclidean distance
  - Edit distance, a nivel de palabras
  - Jaro-Winker distance
  
- Similitud Semántica
  - Palabra a Palabra
  - Palabra a Oración
  - Oración a Oración
  
- Diferencia en las oraciones
  - Coincidencias de Palabra-EtiquetaDiscurso
  - No-Coincidencias de Palabra-EtiquetaDiscurso
  
- Relaciones Gramaticales
  - Métricas Básicas a dependencias funcionales (Stanford Parser)
  
- Relaciones Parciales
  - Número de Entidades en común

Para generar el modelo de clasificación se utiliza la máquina de soporte vectorial con diferentes combinaciones de características.

En la investigación desarrollada en [4] se retoma el esquema cross-lingüe, con la técnica de combinación. Para la construcción de sus vectores característicos utilizan los algoritmos de traducción estadística IBM-3 e IBM-4 disponibles en MGiza++. Ellos construyen sus vectores representativos con los siguientes valores:



- Porcentaje de palabras alineadas
- Número de palabras no alineadas
- Longitud del mayor n-grama alineado
- Longitud del mayor n-grama no alineado
- Promedio de secuencias de palabras alineadas
- Promedio de secuencias de palabras no alineadas
- Posición de la primera palabra no alineada normalizada con respecto a la longitud de la oración
- Porcentaje de n-gramas alienados de 1 a 5.

También desarrollan modelos de clasificación utilizando también la Máquina de Soporte Vectorial aplicando la metodología de composición de juicio de implicación textual.

## 2.3 Técnicas de Pivote

Aunque la mayoría de la comunidad resuelve esta tarea como un problema de clasificación y utilizan en particular al clasificador SVM, algunos investigadores usan metodologías alternas, un ejemplo de este tipo de enfoque es el que se presenta en el trabajo desarrollado en [5], donde se desarrolla un sistema basado en conocimiento. El sistema es capaz de inferir el juicio de implicación sin necesidad de tener un entrenamiento. Se utiliza un enfoque de pivote, por lo que la pareja de oraciones ( $T_1$  y  $T_2$ ) tienen sus respectivas traducciones ( $T_1'$  y  $T_2'$ ), inicialmente los autores determinan que la longitud de las oraciones es un primer indicador de implicación textual, después se calcula la similitud entre las oraciones utilizando un enfoque léxico y semántico, por separado.

Para el análisis léxico se emplea el coeficiente de Jaccard comparando  $T_1$  y  $T_2'$  (donde  $T_2'$  es la traducción de  $T_2$  al idioma en que se encuentra  $T_1$ ), de manera análoga se calcula el mismo coeficiente para  $T_1'$  y  $T_2$ . Finalmente los coeficientes son llamados  $sim_T$  y  $sim_S$  respectivamente.

Para el análisis semántico se emplea una medida de similitud semántica que internamente utiliza sinónimos para mejorar su precisión, de esta manera se obtienen  $sim_T$  al aplicar la medida de similitud semántica a  $T_1$  con  $T_2'$ , de igual manera se obtiene  $sim_S$  con  $T_1'$  y  $T_2$ . Finalmente los autores proponen un algoritmo que toma en cuenta la longitud de las oraciones, así como las similitudes anteriormente mencionadas.

## 2.4 Técnicas Cross-Lingüe

Otro tipo de técnica empleada para resolver este problema es la que se propone en [6], que explota los avances de las MT, esto es que la MT en lugar de traducir, también puedan detectar el juicio de implicación textual. La investigación realizada retoma un enfoque cross-lingüe y maneja análisis léxico, semántico y sintáctico, considerando estos tres análisis:



- Tabla de frases
- Relaciones de dependencia
- Tablas de frases semánticas

A continuación se explican en qué consisten cada una de las fases.

La *tabla de frases* es una especie de diccionario que contiene n-gramas con longitud de 1 a 5, se crea entrenando Giza++ con EuroParl (v4), News Commentary y United Nations (todos los corpus son paralelos). Adicionalmente se le da un tratamiento a cada uno, se utiliza TreeTagger para obtener la etiqueta del discurso y el lema; no obstante, los autores sostienen que es mejor hacer un truncamiento de términos en lugar de una lematización. Justifican para fines estadísticos que es mejor un término truncado que un término lematizado, ya que el lema de una palabra puede variar, mientras que un término truncado es uniforme, este pequeño cambio se ve recompensado después del entrenamiento de Giza++, una herramienta que genera diccionarios de traducción estadísticos. A partir del diccionario estadístico que crea Giza++, con ayuda de Moses toolkit, se crea un extracto. El resultado final es la tabla de frases, la cual actúa como un desambiguador del sentido de las palabras.

Las *relaciones de dependencia* surgen como respuesta a la interrogante de ¿Es lo mismo escribir “Microsoft compró a Apple” que “Apple acquired Microsoft”?, evidentemente las oraciones no son iguales, lamentablemente si se calcula el *overlap* de términos comunes el resultado será que las oraciones son Bidireccionales, cuando el juicio de implicación es No-Entailment. Para solucionar este detalle se proponen crear árboles de dependencia con ayuda de DepPattern, cada nodo del árbol guarda la información a través de conectores de dependencia como por ejemplo comprar(Microsoft, Apple) y acquire(Apple, Microsoft), así se garantiza que el juicio sea lo más exacto posible.

El modelo de clasificación lo construyen con un enfoque de dominio específico y por lo tanto, no se previene algún caso particular nuevo, ante esta problemática surge el concepto de *Tablas de frases semánticas*, las cuales son una generalización para contemplar nuevos posibles casos, de esta manera la oración “Microsoft compró a Apple” se generaliza a “ORG compró a ORG”, esto con ayuda de FreeLing, y posteriormente se une la etiqueta al término, de modo que se vea como un solo token para entrenar Giza++ con esa colección. El resultado final es la tabla de frases semánticas.

Para la etapa de clasificación utilizan un SVM con las tres características expuestas anteriormente en un sólo vector representativo con un enfoque composición de juicios de implicación.



## 2.5 Técnicas Híbridas

Un trabajo que adopta un enfoque híbrido entre cross-lingüe y pivote es el que se presenta en [7], para la técnica con pivote utilizan traducciones en ambos sentidos, es decir  $T_1'$  se encuentra en el mismo idioma que  $T_2$ , y de manera análoga  $T_2'$  se encuentra en el mismo idioma que  $T_1$ , ambas parejas de oraciones son evaluadas con una técnica llamada *METEOR* [8], la cual permite evaluar la concordancia de traducción, esto con el fin de aportar veracidad al pivote, aun así se normaliza la traducción para garantizar que ésta sea buena. El trabajo se basa en la hipótesis de que si se parafrasea un segmento de la oración, ya sea empleando sinónimos o no, existe implicación textual y que además la similitud no es simétrica, derivando de esta manera el juicio de implicación.

Como primera instancia, ya que tienen las traducciones  $T_2'$  y  $T_1'$ , realizan dos análisis por separado, el primero consiste en un análisis de la alineación monolingüe, mientras que el segundo es cross-lingüe. Como desean dar soporte a la medida de similitud, agregan medidas de similaridad como el coeficiente de Jaccard y Overlap (Contención), sobre términos truncados y auxiliándose de sinónimos, en los casos que los contengan. Además introducen una tabla de parafraseo, que son la mayor cantidad de n-gramas que comparten. Finalmente se construyen vectores representativos y emplean la técnica de composición. Dentro de sus conclusiones se menciona que la técnica de composición es mejor que la multiclase.

## 2.6 Metodologías de Textual Entailment

Aunque la comunidad científica prefiere la utilización de algoritmos de aprendizaje automático y como modelo de clasificación la máquina de soporte vectorial, el introducir el enfoque pivote permite retomar para el caso de CLTE las técnicas empleadas para resolver TE.

TE ha sido estudiado desde el año 2005, por lo que utilizar sus diferentes vertientes para solucionar el CLTE es una buena alternativa, se revisan algunas de las metodologías existentes que solucionan TE, y que se pueden utilizar para el CLTE con pivote. Para fines prácticos sólo se abordan las metodologías que son diferentes a las usadas en el CLTE.

La manera de conectar términos en una oración puede verse como un grafo, ahora si se tienen dos oraciones que hablan de lo mismo, éstas comparten enlaces a nodos equivalentes (ver Imagen 3). Esto es lo que se expone [9], quienes utilizan módulos de sinonimia e hiperonimia para detectar los nodos equivalentes en ambas oraciones y después se obtiene el costo de empatamiento usando ambos grafos. Esta técnica se apoya en el isomorfismo de grafos.

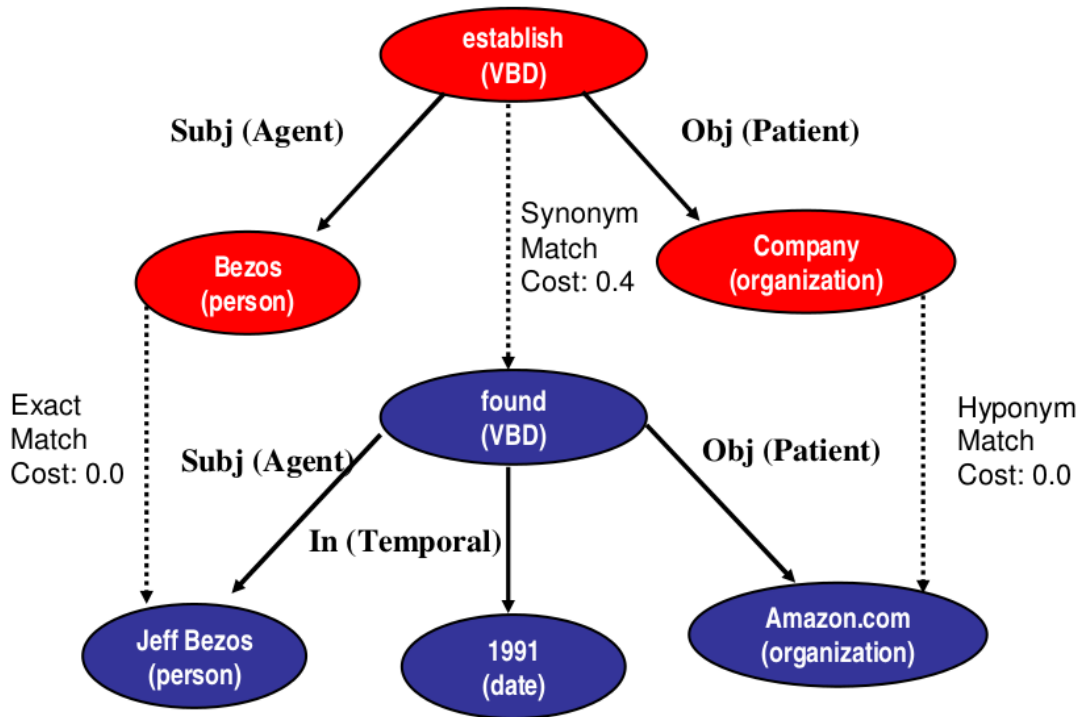


Imagen 3: Cotejamiento de Grafos

En la investigación reportada en [10] se retoma el enfoque anterior, pero en lugar de usar el isomorfismo de grafos, se introduce una distancia desarrollada para esta tarea en concreto. Utilizan un extractor de reglas que opera sobre los grafos de los pares de oraciones, de la colección de entrenamiento, y con ayuda de un modelo de clasificación desarrollado con la SVM, detectan cuando existe o no una implicación textual.

En las investigaciones desarrolladas en torno a la tarea de TE, las aproximaciones que obtienen mejores resultados son las que se enfocan en el empatamiento de grafos, sin embargo no siempre son correctas. Se ha observado que el uso de plantillas de empatamiento en muchos casos ofrece mejores resultados. Se propone el siguiente ejemplo:

Se tienen las oraciones: "Juan compró un CD de Mozart" y "Juan adquirió un CD de Mozart", es evidente que ambas oraciones denotan lo mismo, si se eliminan las entidades presentes, se obtiene "compró un" y "adquirió un". De esta manera se pueden construir grupos de plantillas que denotan un juicio de implicación textual.

Algunas investigaciones emplean plantillas, sin embargo extraerlas, ha sido un problema, hasta que en el trabajo desarrollado en [11] se introdujo un enfoque semi-supervisado para ello. La extracción de plantillas resultó ser un proceso simple, empieza con una semilla, como "X compró Y", y se busca en la web ese patrón, algunos posibles resultados son: "Disney compró a Indiana Jones", "Santander compró 8% del capital de Bank of Shanghai" y "Microsoft compró a Nokia".



Posteriormente se identifica la posición de la plantilla para dividir a la oración, por ejemplo: a partir del segundo ejemplo, se obtiene “*Santander*” y “*8% del capital de Bank of Shanghai*”, estos dos términos son denominados anclas y sirven para realizar una nueva búsqueda. Los resultados arrojados por el par de anclas anteriores son: “*Santander **compró** 8% del capital de Bank of Shanghai*”, “*Santander **adquirió** 8% del capital de Bank of Shanghai*”, “*Santander **toma el** 8% del capital de Bank of Shanghai*”, al quitar las anclas se genera la lista final de plantillas: “*X **compró** Y*”, “*X **adquirió** Y*”, “*X **toma el** Y*”. Los autores resaltan que el aspecto más importante a considerar es el contexto, e idean un mecanismo para verificarlo; esto les permite saber cuándo pueden aplicar un grupo de plantillas y cuando no.

## 2.5 Conclusión del capítulo

Como bien se ha podido apreciar a lo largo de este capítulo, las técnicas para resolver CLTE son muy variadas, y no todas las vertientes han sido explotadas, al analizar la imagen 2 se puede notar que existen enfoques que aún no han sido abordados, así mismo se aprecia que el enfoque que usa las técnicas de pivote con algoritmos de aprendizaje automático son muy utilizadas, y más la perspectiva del esquema de clasificación multiclase, en comparación con el esquema de combinación.

Por otra parte se presenta que el uso de la técnica de pivote para resolver el problema CLTE hace posible utilizar técnicas del TE. Algunas de estas son: técnicas basadas en grafos, técnicas basadas en plantillas con las que se busca lograr una mejor estrategia para detectar el juicio de implicación.



## Capítulo 3: Fundamentos de la investigación

Este capítulo es la piedra angular de la presente investigación, se hace un recorrido por las diferentes metodologías que ha utilizado la comunidad científica, con respecto al problema de detectar la Implicación Textual Cross-Lingüe. De la misma manera se revisan los diferentes enfoques que abordan el problema de la Implicación Textual. La idea de introducir estos dos problemas está dada por el objetivo de buscar métodos nuevos que permitan aprovechar los resultados obtenidos en cada una de ellas.

### 3.1 Metodologías empleadas para solucionar el CLTE

A pesar de que las posibles combinaciones existentes para resolver el CLTE son muchas, ver la sección 2.1, en la práctica sólo se llevan a cabo unas cuantas, a continuación se examinan los enfoques que son usados por la sociedad científica.

#### 3.1.1 Sistemas Cross-lingüe

Los trabajos que desarrollan sistemas cross-lingüe se apoyan en los fundamentos de la Machine Translation, principalmente en la correlación de traducciones, para ello se deben crear diccionarios de traducción a partir de corpus paralelos. Los diccionarios de traducción son usados para evaluar una relación cuantificable entre el par de sentencias. Este esquema genérico es mostrado a continuación, en la imagen 4.

Los corpus paralelos empleados, en las investigaciones de esta vertiente, son generalmente de dominio de información variado, como Wikipedia, ya que los diccionarios obtenidos comparten el dominio del corpus paralelo de donde derivan. Independientemente se generan diccionarios de diferentes tipos, los más empleados son de n-gramas [12], no obstante algunas investigaciones utilizan diccionarios de n-gramas de etiquetas de POS (Part Of Speech) y diccionarios de Entidades Nombradas. Algunos trabajos más elaborados realizan un postratamiento a los diccionarios para sustituir términos especiales por hipónimos [6], para que de esta manera, la evaluación de la relación sea lo más genérica posible.

En cuanto al uso del valor de la relación que es obtenido, se pueden crear vectores característicos que alimentan algoritmos de aprendizaje, ya que el uso de varios diccionarios provee varios valores, y varios valores conforman vectores característicos.

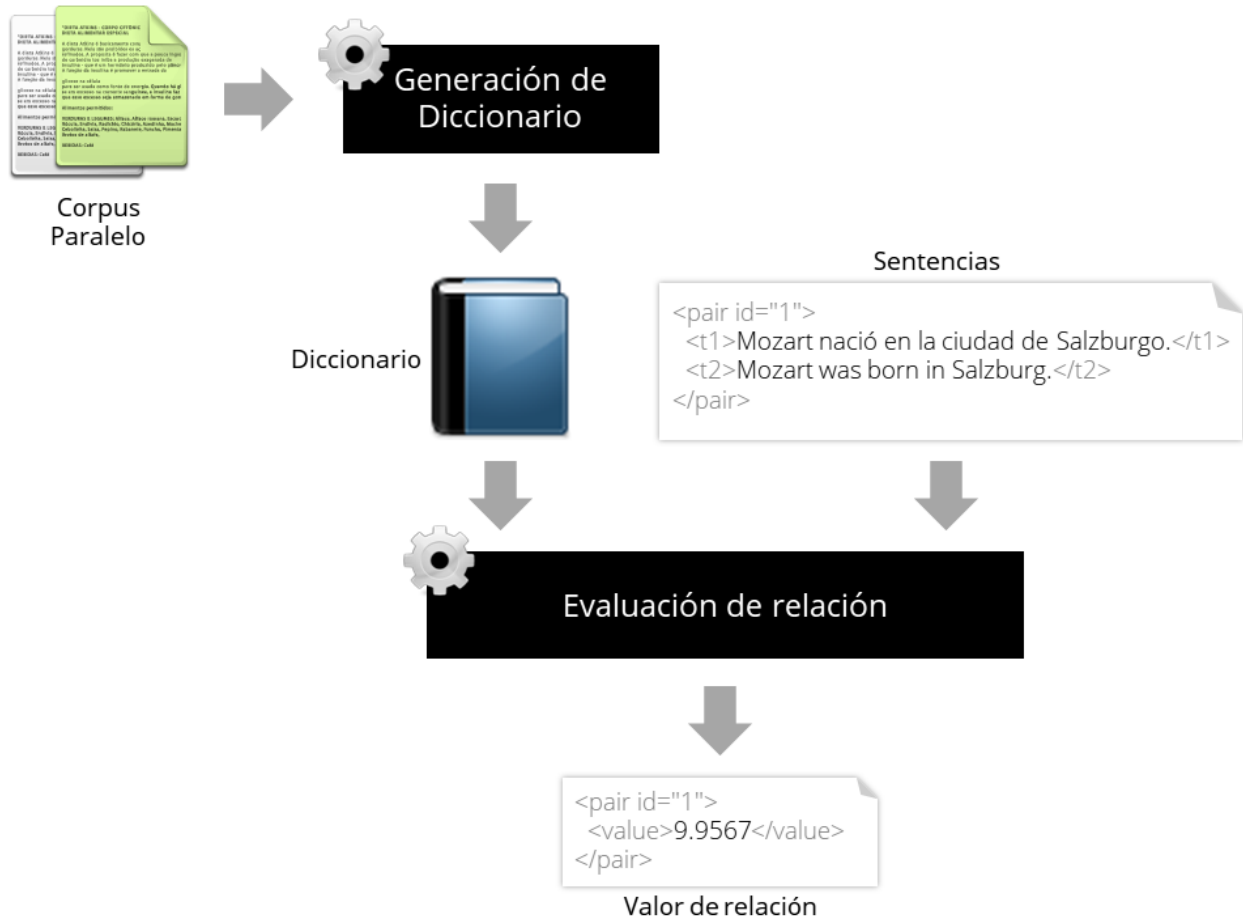


Imagen 4: Esquema genérico de sistemas Cross-lingüe

### 3.1.2 Sistemas de Pivote

En la perspectiva del pivote también se da un valor a la relación que guardan un par de oraciones. Dado que el pivote lleva el par de oraciones al mismo idioma, el uso de diccionarios es descartado. Para cuantificar la relación entre ambas oraciones, se recurre a medidas de similitud. El proceso es simple, tal como ilustra la imagen 5, primero se deben tener ambas oraciones en un mismo idioma, posteriormente se calcula alguna medida de similitud.

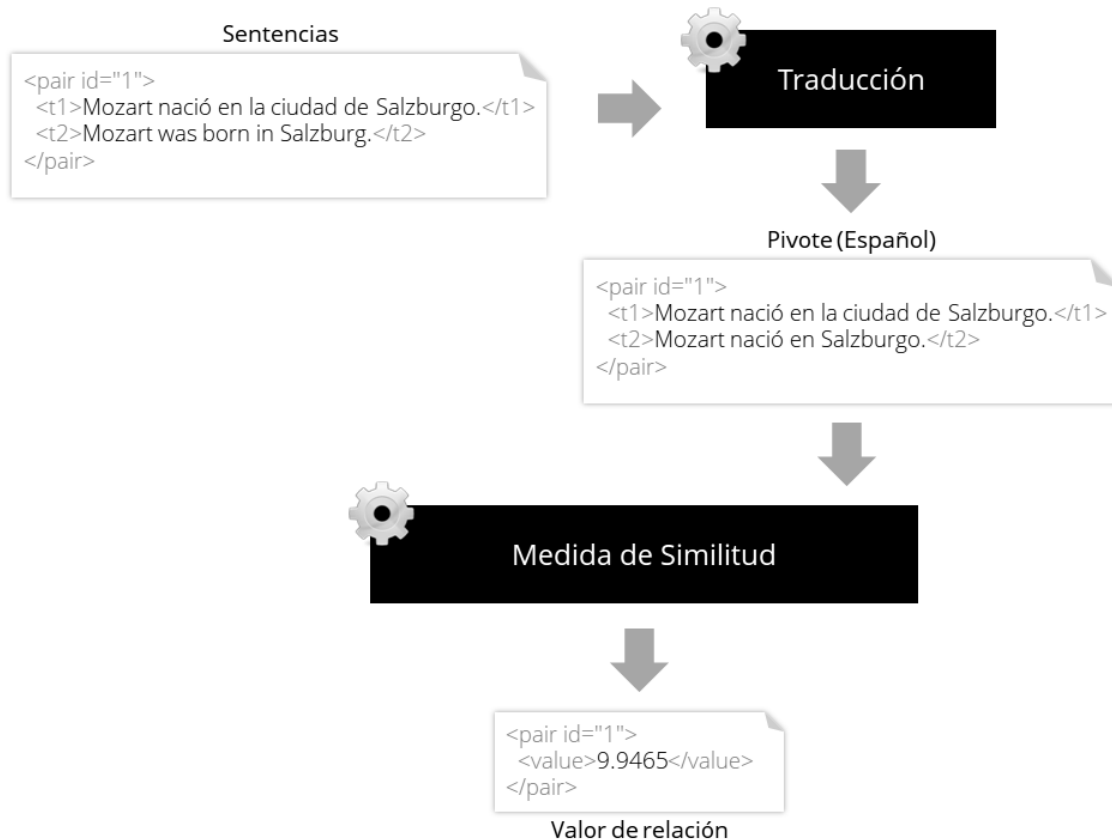


Imagen 5: Esquema genérico de sistemas de pivote

Los científicos han experimentado con un gran número de medidas de similitud, han usado: intersección, coeficiente de Jaccard, solapamiento, distancia euclidiana, por mencionar algunas. Así mismo han experimentado a nivel de tokens, n-gramas de palabras, n-gramas de caracteres y n-gramas de POS.

Los valores obtenidos por las medidas de similitud constituyen vectores característicos.

### 3.2 Extracción de características

En las dos secciones anteriores se han observado los esquemas genéricos que son abordados para solucionar el CLTE, se ha notado que, se tome la vía cross-lingüe o pivote, al final se generan valores de relación.

Al observar que los valores de relación generan vectores representativos, surge la incógnita ¿Qué puedo medir en un par de oraciones?, para contestar a la pregunta, regresemos un poco al par de oraciones. Tomando un enfoque lingüístico-computacional, una frase está compuesta por tres niveles:

- Nivel léxico
- Nivel sintáctico
- Nivel semántico



Cada nivel posee cualidades que pueden ser cuantificables, denominadas características. Antes de pasar a revisar las características empleadas por la sociedad científica para resolver el CLTE, es necesario recalcar que no todas las propiedades pueden ser aplicadas, para ello se considera la siguiente eventualidad.

Natalia, una estudiante de Ciencias de la Computación, desea contribuir con la sociedad científica y aportar ideas para resolver el problema del CLTE, ella opta por hacer uso de una Máquina de Vectores de Soporte (SMV) con un enfoque multiclase-pivote. Para generar sus vectores representativos emplea bolsa-de-palabras, ya que estuvo leyendo que esa metodología da buenos resultados en la clasificación de textos. Tras haber refinado su bolsa-de-palabras, obtiene una buena tasa de precisión en el entrenamiento, pero al momento de evaluar el conjunto de prueba se percató que su precisión no es tan buena como creía. Natalia, como buena investigadora, se pregunta en dónde está el error, así que realiza un análisis, y averigua que el vocabulario de la bolsa-de-palabras está presente en ambos conjuntos (entrenamiento y prueba), se plantea que la bolsa-de-palabras clasifica con respecto al texto, esto es, por ejemplo que si la palabra “bailar” es representativa del juicio “backward”, en el conjunto de prueba todos los pares de oraciones que contienen la palabra en cuestión se les atribuye un juicio de implicación erróneo. Natalia se percató que no puede usar la bolsa-de-palabras para inferir los juicios de implicación.

Para evitar lo que le ocurrió a Natalia, la comunidad científica usa características que no se encuentren en función del vocabulario o patrones repetitivos, estas características se expresan a continuación.

### 3.2.1 Características a Nivel Léxico

A nivel léxico analizan los n-gramas de fonemas, las sílabas, las letras y/o palabras; denominadas tokens. Dentro del CLTE normalmente se aplican medidas de similitud entre los tokens de los pares de sentencias, siguiendo la idea de que si un par de oraciones hablan de lo mismo, éstas deben tener un nivel alto de similitud. Otras investigaciones emplean una representación vectorial para aplicar otro tipo de medidas, como por ejemplo la distancia euclidiana, la similitud coseno entre otras. Además se pueden utilizar n-gramas de las etiquetas de las categorías gramaticales (*Part Of Speech*).



### 3.2.2 Características a Nivel Semántico

La semántica hace referencia al significado de la palabra de acuerdo al contexto. Para detectar el significado de las palabras se pueden utilizar sinónimos, hiperónimos e hipónimos. Por ejemplo, si la oración  $T_1$  contiene un sinónimo o hiperónimo presente en la oración  $T_2$ , se puede contar como si fueran el mismo token, cuidando de que la elección hecha esté respaldada por el sentido de ambas oraciones. Esto último es posible gracias al uso de los contextos, que recaen en este nivel característico.

Éste nivel característico refleja el sentido de la oración, pensemos en este par de oraciones “La vaca Margoth da leche” y “Margoth da leche a la vaca”, a pesar de que ambas oraciones contienen los mismo tokens, es fácil notar que el sentido de las oraciones es completamente diferente.

### 3.2.3 Características a Nivel Sintáctico

Este nivel busca detectar las relaciones entre las palabras (dependencias funcionales), a partir de las categorías gramaticales que poseen cada una de ellas. La imagen 6 muestra las dependencias funcionales del par de oraciones “La vaca Margoth da leche” y “Margoth da leche a la vaca”.

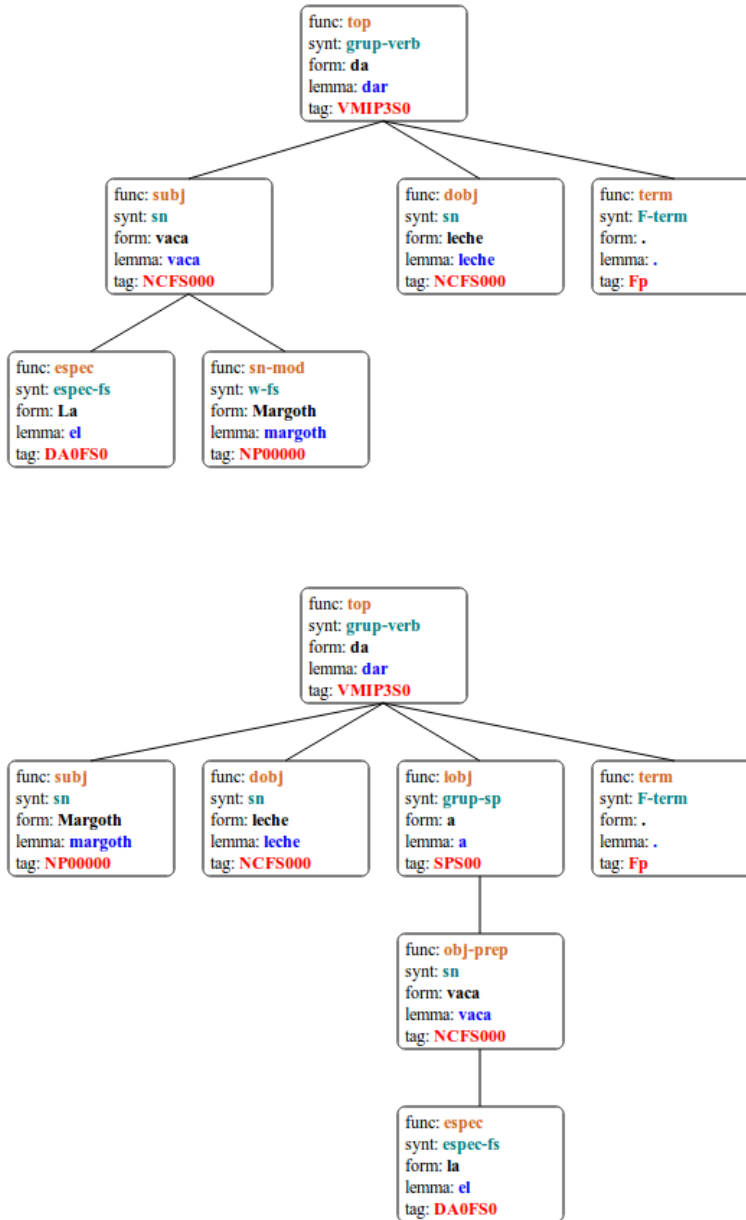


Imagen 6: Ejemplo de dependencias funcionales

### 3.3 Algoritmos de aprendizaje automático

Las características extraídas, a partir del par de sentencias, proporciona una percepción de cada instancia, la manera de aprovechar estas observaciones es mediante el uso de los algoritmos de aprendizaje automático, que son modelos que adquieren conocimiento a partir de observaciones, el modelo que los constituye se ajusta de acuerdo su percepción adquirida, a partir de muestras de entrada (entrenamiento).



Los algoritmos de aprendizaje automático se catalogan en clasificación-y-predictivos y agrupamiento [13]. La amplia gama de los algoritmos de clasificación existentes no es tema de este trabajo de investigación, sin embargo se mencionan algunos de los métodos que podrían emplearse para la tarea del CLTE.

**Árboles de Decisión:** Este tipo de algoritmos analizan las muestras en busca de cambios de ganancia de información, el modelo de aprendizaje es reajustado cuando la población de una clase presenta un rasgo muy distintivo, que lo hace ser diferente a los demás. Se busca expandir las ramas de los árboles que son prometedores y descartar las que no aportan información relevante (poda).

**Modelos Bayesianos:** Estos algoritmos se fundamentan en el teorema de Naïve Bayes, el cual hace un modelado a través de probabilidades. Son de rápido entrenamiento y variantes como el Modelo de Márkov analizan eventos previos para refinar la probabilidad final.

**Redes Neuronales:** Estos algoritmos, inspirados por las neuronas del cerebro, emplean un modelo matemático que actualiza pesos sinápticos dentro de una matriz, a medida que las iteraciones (épocas) avanzan, se reajusta su conocimiento con la técnica de propagación hacia atrás. Son de lento entrenamiento, pero de rápida respuesta.

**Algoritmos Flojos / Aprendizaje por Vecinos:** Se denominan flojos a estos algoritmos porque no requieren una fase de entrenamiento, cuando se procesa una instancia desconocida, simplemente se buscan sus  $k$  vecinos más cercanos y se analiza a cuántos de ellos es más parecido. Algunas variantes emplean árboles de decisión para determinar si un vecino es representativo o no, otros más ponderan la función de distancia para dar mayor importancia a los vecinos que realmente están más cercanos.

**Máquinas de Vectores de Soporte:** Abreviadas como SVM, por sus siglas en inglés, estos modelos se inspiran en hiperplanos vectoriales, y buscan soportes (intersecciones) entre planos, de tal modo que las instancias de una misma clase esté envuelta por hiperplanos, se pueden generar tantos hiperplanos como sean necesarios. Son unos de los pocos algoritmos que logran procesar clases sobrelapadas.

### 3.4 Técnicas de TE

Las aproximaciones del TE tienen lugar cuando se usa la técnica de pivote, ya que ambas sentencias se encuentran en el mismo idioma. Independientemente a las medidas de similitud entre oraciones, expuestas en la sección 3.2, la comunidad del TE ha desplegado investigaciones que no emplean algoritmos de aprendizaje automático [14]. Estos enfoques podrían utilizarse dentro del CLTE, ahora se realiza una reseña de ellos, sin contemplar las metodologías que usan algoritmos de aprendizaje automático, así como las que emplean enfoques léxico, porque ya se han mencionado en secciones pasadas.

### 3.4.1 Muestreo de plantillas

Usar plantillas para detectar TE es un mecanismo simple de usar. Se toma en cuenta el siguiente grupo de plantillas: “X escribió Y”, “Y fue escrito por X” y “Y es obra de X”. Para detectar si existe implicación textual, se cotejan las plantillas en las oraciones, de este modo se puede descubrir que las oraciones “Murakami escribió After Dark” y “After Dark es obra de Murakami” sostienen una implicación.

A pesar de que las plantillas solucionan al TE, la alta heterogeneidad de los lenguajes naturales, en cuanto a léxico y gramática, así como la difícil construcción de grupos de plantillas limita esta metodología.

### 3.4.2 Similitud de Grafos

Se pueden construir grafos a partir de las oraciones, con ayuda de los parser sintácticos o de dependencias funcionales. Y para detectar TE se recurren a medidas de similitud entre grafos. Este proceso se aprecia mejor en la imagen 7.

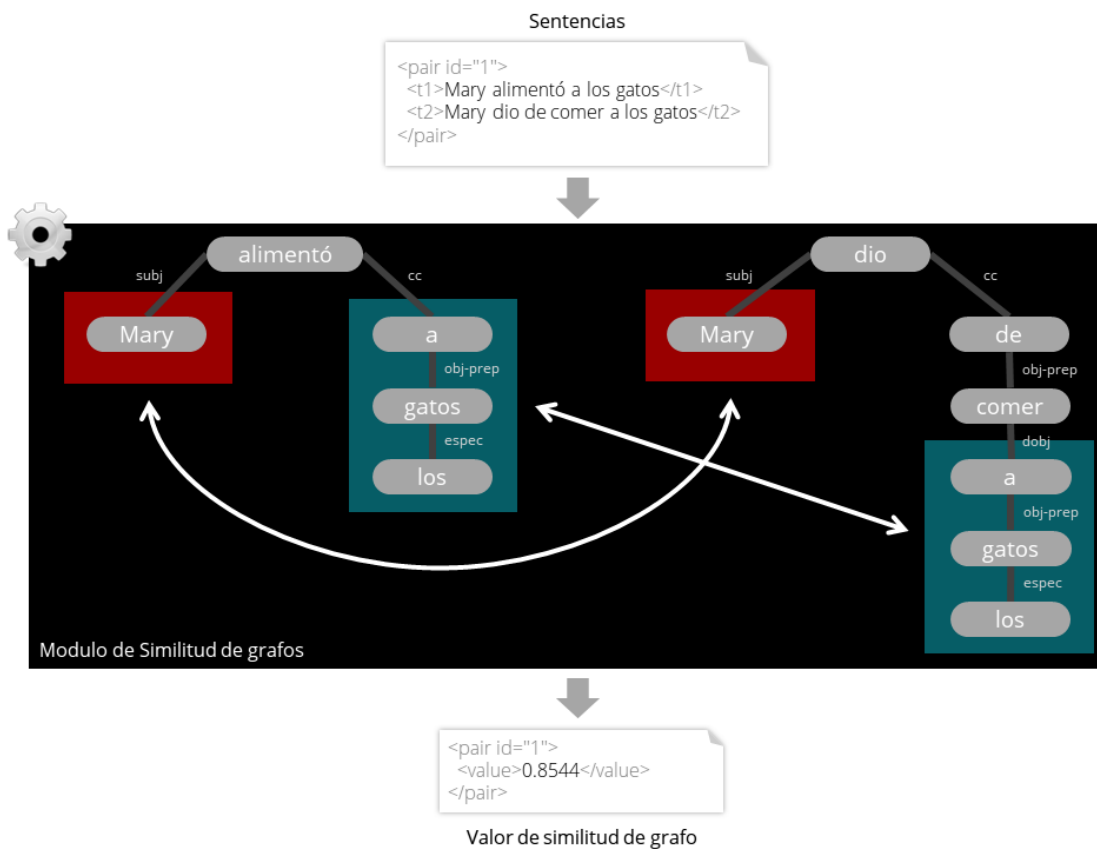


Imagen 7: Ejemplo de similitud de grafos

El uso del valor de similitud obtenido, puede formar parte de algún vector característico, o alimentar a algoritmos basados en reglas.

### 3.4.3 Evaluación Lógica

De manera similar al uso de plantillas, este enfoque busca una reciprocidad entre las sentencias, mediante el uso de funciones lógicas del estilo *relación(sujeto, objeto)*. Se lleva a cabo un proceso de generalización, esto es, a medida que se procesan instancias en el entrenamiento la base de conocimiento aumenta. Para generar este tipo de relaciones se utilizan parser sintácticos. Observemos el ejemplo que muestra la imagen 8, una vez que se generan las funciones lógicas, éstas pasan a ser genéricas y a través de operadores universales se genera la regla de empatamiento.

$$\begin{aligned}
 T & : \text{Leonardo da Vinci painted the Mona Lisa.} \\
 \phi_T & : \text{isPainterOf}(\text{DaVinci}, \text{MonaLisa}) \\
 H & : \text{Mona Lisa is the work of Leonardo da Vinci.} \\
 \phi_H & : \text{isWorkOf}(\text{MonaLisa}, \text{DaVinci}) \\
 \psi & : \forall x \forall y \text{ isPainterOf}(x, y) \Rightarrow \text{isWorkOf}(y, x)
 \end{aligned}$$

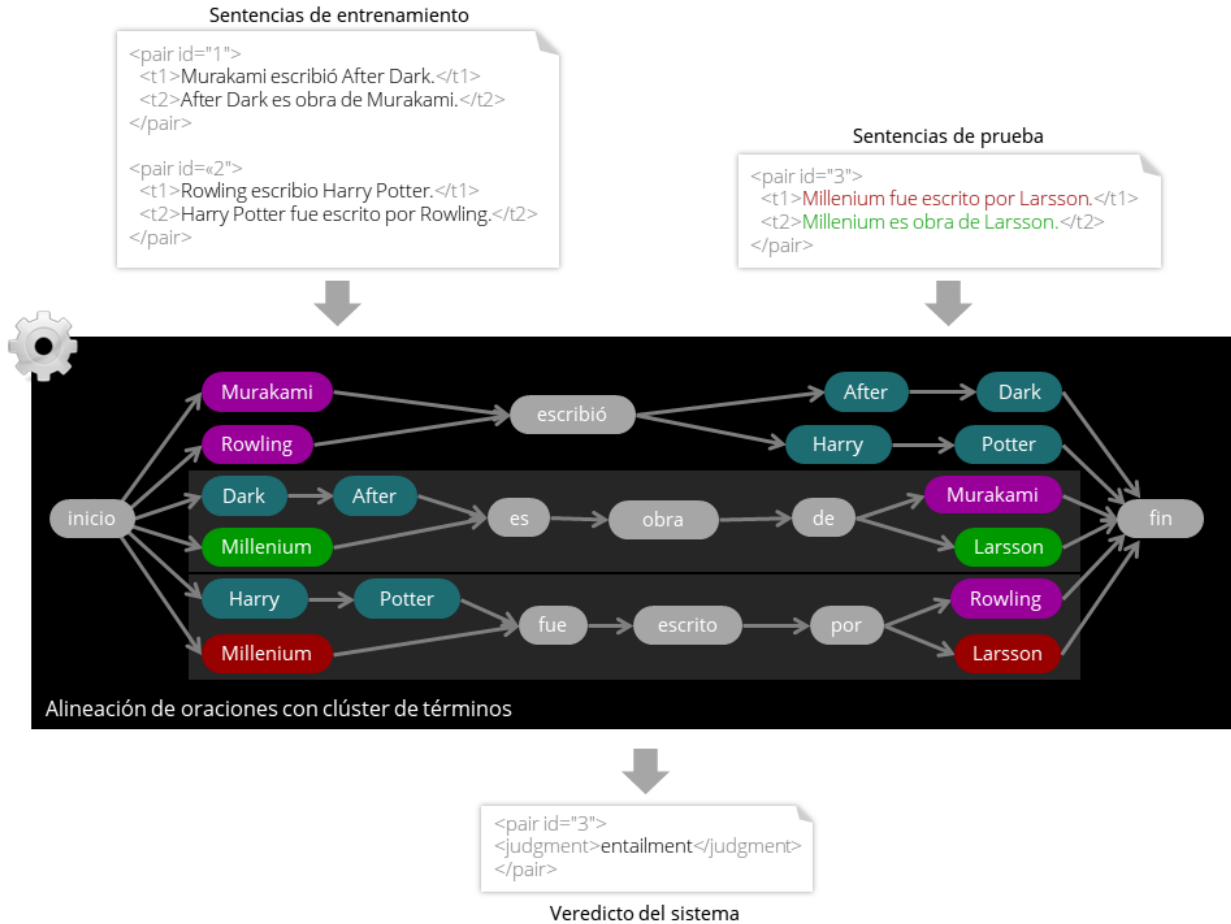
**Imagen 8:** Ejemplo del enfoque lógico.

El inconveniente observado en este enfoque deriva de la extensión de su base de conocimientos. Ya que si las nuevas oraciones al llevarlas a su representación lógica no forman parte de la base de conocimientos, no se podrá determinar el juicio de implicación.

### 3.4.4 Alineación de Oraciones (Autómatas)

Esta perspectiva tiene dos enfoques distintos, ambos generalizan las plantillas de empatamiento a través de autómatas.

El primer enfoque, a medida que procesa pares de oraciones con implicación textual, agrega los términos a un autómata, al finalizar todos los pares, empieza a buscar clusters de términos, después de que son hallados, se cotejan con otros aglomerados, si los clusters coinciden entre el inicio y el fin del autómata, se establece que para futuros pares de oraciones, si comparten el patrón en el autómata, se sostiene que poseen implicación textual. La imagen 9 muestra un ejemplo de lo que se explicó.



**Imagen 9:** Alineación de oraciones con clusters de términos.

El otro enfoque es inverso, se generaliza  $n$  autómatas como  $n$  pares de oraciones. Sólo se toman aquellas sentencias que tengan implicación textual, la idea es generar autómatas de implicación al fusionar los pares de oraciones, así cuando un par nuevo es reconocido por el mismo autómata, este dirá que existe una implicación textual. Aunque pareciera ser más fácil que el primer enfoque, su implementación requiere descomponer la oración en sujeto-verbo-objeto para establecer los criterios del alineamiento de las oraciones, la imagen 10 muestra cómo funciona este método.

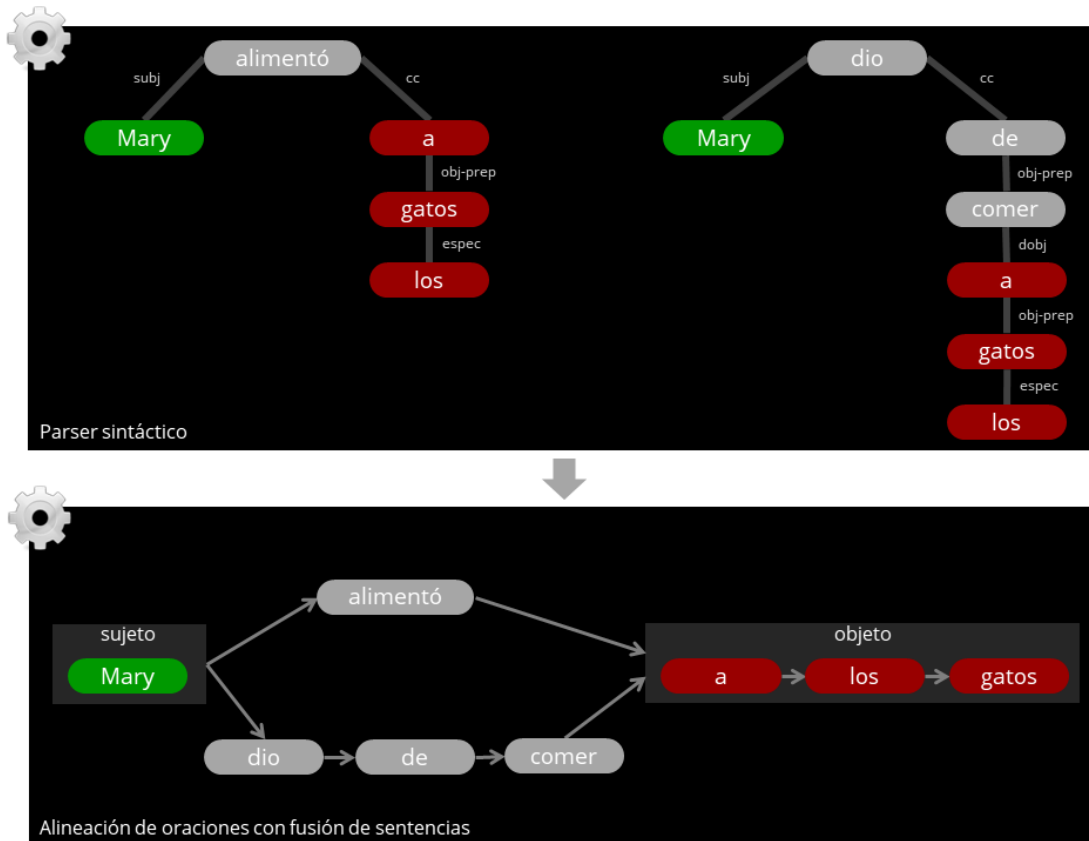


Imagen 10: Alineación de oraciones con fusión de sentencias.

### 3.4.5 Híbridos

Dentro de las vertientes que buscan solucionar al CLTE, existen trabajos considerados híbridos por la fusión de técnicas. El híbrido más sencillo es el de funcionar características cross-lingüe con características de pivote, mientras que los más elaborados buscan crear plantillas de empatamiento a través de gramáticas en conjunto con características léxicas y semánticas de cross-lingüe y pivote. Este tipo de propuestas tiende a obtener mejores resultados, pero no tan significantes en comparación a los enfoque normales.

Otras investigaciones usan varios clasificadores para establecer un sistema de votaciones, mientras que otras emplean metaclassificadores [15], que son clasificaciones que buscan eliminar juicios en cada etapa. Esta perspectiva es inversa a la habitual, en lugar de preguntar cuál juicio es el correcto, se busca que el modelo descarte juicios para que al final quede un sólo juicio de implicación.



## 3.5 Recursos y herramientas

Este epígrafe tiene como objetivo exponer las diferentes herramientas y recursos que pueden ser utilizados para resolver el problema del CLTE.

### 3.5.1 Colecciones de Datos Empleadas

Las colecciones principales para confeccionar metodologías que resuelven el CLTE, son obtenidas a través del SemEval-2012 y SemEval-2013. Se cuentan con una colección de entrenamiento y dos colecciones de prueba. Cada colección consta de 2000 pares de oraciones, divididas en 4 idiomas, y cada idioma contiene 125 pares de oraciones para cada categoría de implicación textual.

Además en el marco de la conferencia PASCAL (Pattern Analysis, Statistical Modelling and Computational Learning) se han confeccionado, una serie de corpus, desde el año 2008. Se manejan los siguientes conjuntos de datos:

- RTE (Recognizing Textual Entailment) -1: Conjunto de Entrenamiento y Conjunto de Prueba
- RTE-2: Conjunto de Entrenamiento y Conjunto de Prueba
- RTE-3: Conjunto de Entrenamiento y Conjunto de Prueba
- RTE-4: Conjunto de Entrenamiento y Conjunto de Prueba
- RTE-5: Conjunto de Prueba
- RTE-6: Conjunto de Prueba
- RTE-7: Conjunto de Prueba

Las colecciones del RTE son monolingües, por lo que emplear estos conjuntos de prueba para adecuar el comportamiento de los algoritmos implica utilizar la técnica de pivote.

### 3.5.2 Lematizadores y Truncadores

Los lematizadores y truncadores se emplean para homogeneizar los textos. El motivo es simple, al tener textos homogeneizados se puede hacer un análisis estadístico mejorado, ya que palabras como: *cantar*, *cantó*, *cantaría* y *cantaron* son tomados como un solo término.

Un lematizador, como su nombre indica, se encarga de buscar el lema de una palabra, de modo que las palabras *cantar*, *cantó*, *cantaría* y *cantaron* serían cambiadas por *cantar*.

Los truncadores toman los términos y eliminan el final de las palabras, este proceso de truncamiento sirve para que las palabras *cantar*, *cantó*, *cantaría* y *cantaron* sean cambiadas por *cant*.

El uso de una herramienta u otra está en función del estudio a realizar, en la literatura se reportan mejores resultados en investigaciones lingüísticas cuando se emplean lematizadores, mientras que en investigaciones de carácter estadístico-probabilístico los truncadores son más utilizados.

### 3.5.3 Etiquetados de partes del discurso

Las etiquetas del discurso son los roles que juegan las palabras en una oración, por ejemplo, el término “gato” es un sustantivo, al igual que el término “agua”. Ahora bien, si se conforma una oración con un sustantivo-verbo-sustantivo se puede obtener la oración “gato toma agua” o “Lisa escucha rock”, etc. Los etiquetadores de partes del discursos, también llamados Part of Speech (PoS) Tagger son utilizados para recuperar las etiquetas de los términos que conforman las oraciones.

### 3.5.4 Parsers Sintácticos

Al revisar la herramienta anterior, parecería ser que un PoS Tagger genera etiquetas Sujeto, Predicado. En realidad la herramienta que genera esas etiquetas son denominadas parsers sintácticos.

Un parser sintáctico toma las etiquetas generadas por algún PoS Tagger y devuelve las relaciones entre etiquetas, también llamadas dependencias funcionales. Por ejemplo en la oración “The tree of wisdom grew thanks to our efforts” generaría las dependencias funcionales que muestra la imagen 11.

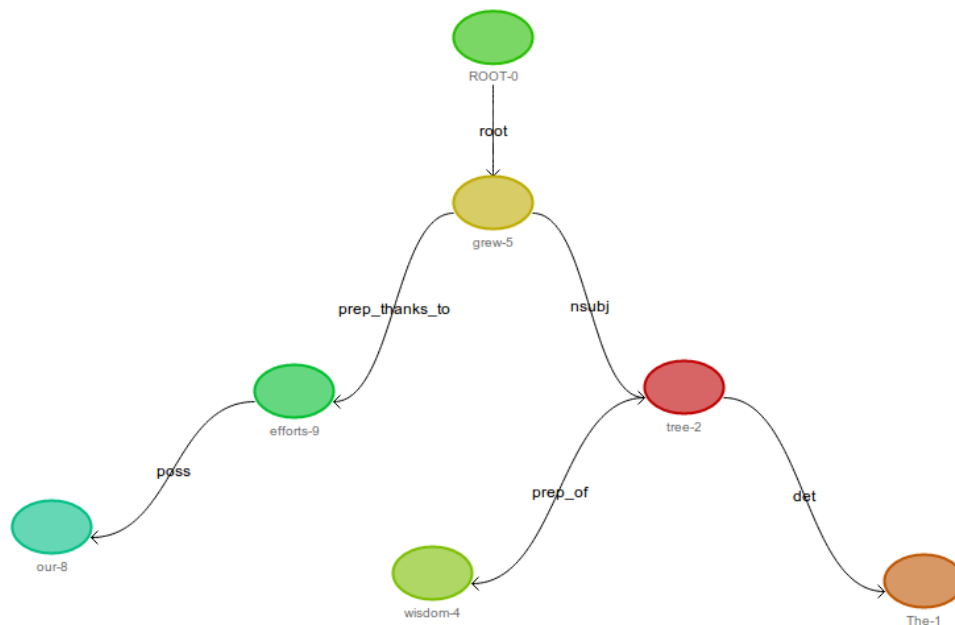


Imagen 11: Ejemplo de dependencias funcionales.



### 3.5.5 Sinónimos, Antónimos, Hipónimos e Hiperónimos

El título hace alusión al apartado 3.2.3, que habla de características semánticas. Para detectar las características semánticas de alguna sentencia, se recurren a diccionarios especializados, denominados tesauros. Un tesoro contiene información, detallada y organizada en contextos, de términos, este tipo de recursos son construidos por humanos y son de lenta construcción. Algunos de los más importantes son WordNet [17] y el diccionario de la RAE, otros trabajos como los diccionarios de OpenOffice son compilaciones de varias fuentes que su comunidad logró recabar.

### 3.5.6 Herramienta de aprendizaje supervisado

La herramienta mayormente utilizada, por su simplicidad de uso y por la amplia variedad de algoritmos de aprendizaje no supervisado y supervisado que ofrece es WEKA [16]. Esta herramienta es utilizada en varias investigaciones y por varios campos de la ciencia, que van desde el análisis económico, hasta modelos médicos, pasando por business intelligence.

## 3.5 Conclusión del capítulo

En este capítulo se han discutido todas las posibles técnicas que podrían utilizarse para resolver el CLTE, sin embargo se ha descubierto que no todas ellas son ejecutadas en la práctica. La comunidad científica se ha esforzado por solucionar el CLTE a través de algoritmos de aprendizaje automático, sin considerar los avances hechos por la sociedad del TE. Los puntos de vista del TE pueden ser llevados al CLTE, con el uso del pivote, por otro lado el muestreo de plantillas puede ser abordado de manera cross-lingüe. También comentamos acerca de la posibilidad de combinar técnicas en enfoques híbridos para mejorar el desempeño de los sistemas. Por último se mencionan algunas de las herramientas y recursos que podrían ser empleados para resolver el CLTE.

## Capítulo 4: Metodología

Antes de proponer los cinco modelos para resolver el problema se realizó un estudio de la colección CLTE, con el objetivo de medir el comportamiento de los textos de acuerdo al juicio de implicación; para posteriormente escoger los experimentos adecuados, que permitan determinar la calidad de un modelo dado. A continuación se discuten todos los modelos desarrollados.

### 4.1 Modelo de Conteo Estadístico

En esta sección se introduce un modelo que realiza conteos entre los pares de oraciones, posteriormente se construyen vectores representativos con estos conteos, finalmente los vectores generados alimentan a algún algoritmo de aprendizaje supervisado.

Este modelo se basa en la hipótesis de que oraciones que hablan de lo mismo, deben compartir una gran cantidad de información, por ejemplo el par de oraciones “Los chorros de vapor sugieren la existencia de un océano y según los científicos, donde hay agua puede haber o hubo vida” y “Las altas concentraciones de vapor, podría venir de un océano, que de acuerdo a los científicos, donde hay agua hay vida” denotan un juicio de implicación bidireccional, mientras que el par “La sonda Cassini de la NASA halló evidencia de un océano bajo la congelada corteza de la mayor luna de Saturno” y “La sonda Cassini captó la posible formación de una nueva luna en Saturno; La protuberancia se encuentra en el anillo más exterior del astro” denotan un juicio no-entailment. Esta regla puede apreciarse a través de diagramas de Venn, la Imagen 18 muestra la visión general de este modelo.



**Imagen 18:** Implicación textual desde el punto de vista de información compartida.

Es importante recalcar que para hacer un buen modelo estadístico, se debe homogeneizar al máximo posible el conjunto de datos. Por ello el preprocesamiento es vital en este tipo de aproximaciones. Para este modelo se propone eliminar palabras cerradas, lematizar el par de oraciones y buscar la sinonimia que pueden sostener, para llevar dos términos a la misma palabra, de este modo se garantiza que las características aplicadas sean lo más precisas posibles.



Este modelo emplea el uso de medidas de conteo estadístico, que de acuerdo con el estudio preliminar, es mejor utilizar el enfoque de pivote con composición de juicio. En el apartado siguiente se describen las características utilizadas.

#### **4.1.1 Características Elegidas**

Las características utilizadas son las siguientes:

##### **Porcentaje de n-gramas compartidos**

Para determinar la cantidad de información compartida entre ambas oraciones, se ha decidido emplear una similitud a nivel de n-gramas, tanto de palabras como de caracteres. Las longitudes de n-gramas van de 1 a 5.

##### **Porcentaje de skip-gramas compartidos**

Los skip-gramas de caracteres consideran un skip (salto) de alguna longitud, normalmente están conformado por dos letras. Por ejemplo un skip-grama de longitud 1 de la palabra “murciélago” es “ea”, ya que hay un salto de una letra, un skip-grama de longitud 2 es “ui”. Los skip-gramas se pueden aplicar tanto a palabras como a oraciones, y las longitudes también son variables. Para este modelo se consideran skip-gramas de longitudes que van de 1 a 5.

##### **Distancias vectoriales**

Tomando un enfoque vectorial, se puede llevar un conjunto de documentos a una representación de vector, y posteriormente se puede aplicar una medida de similitud para determinar qué documentos son similares. Siguiendo este razonamiento, se toma un par de oraciones y se llevan a una representación vectorial, para posteriormente aplicar cualquier medida de similitud. Las medidas de similitud aplicadas son las siguientes:

- Distancia Euclidiana
- Distancia Chevichev
- Distancia Manhattan
- Similitud Coseno

##### **Similitud de conjuntos**

En otra perspectiva, se ven a los documentos como conjuntos de palabras, esto permite aplicar métricas de similitud de conjuntos, y así determinar la similitud entre los documentos. El modelo propuesto utiliza esta metodología, en tiempo de ejecución se toma un par de oraciones y se genera un conjunto de palabras por cada oración (a y b). Posteriormente se aplican las siguientes métricas de similitud:



- Sobrelapamiento de a sobre b
- Sobrelapamiento de b sobre a
- Coeficiente de Jaccard
- Coeficiente de  $a / b$
- Coeficiente de  $b / a$

Todas las características anteriormente mencionadas alimentan a un vector que representa de alguna manera el grado de relación entre el par de oraciones.

#### 4.1.2 Montaje del experimento

Para este modelo en particular se ha considerado emplear un proceso de clasificación supervisada, con ayuda de la herramienta Weka (ver sección 3.5.6). Como las colecciones de datos han llevado a una representación vectorial y las clases se encuentran sobrelapadas, se considera que el mejor algoritmo para procesar este experimento es la Máquina de Vectores de Soporte (ver sección 3.3)

#### 4.2 Modelo de similitud semántica

La aproximación del modelo estadístico emplea métricas puramente léxicas, esto quiere decir que es totalmente dependiente de los términos que conforman el par de sentencias y su similitud. Producto de esta limitante se plantea que si en lugar de verificar la similitud de términos, que pasa si se verifica la similitud de ideas.

La tarea de medir el significado transmitido por una oración, con respecto a otra, es conocida como similitud semántica, y busca establecer medidas basadas en el uso de los términos de acuerdo a un contexto dado. En otras palabras busca asociar términos por campos semánticos, por ejemplo los términos avión, motocicleta, automóvil, autobús y tren corresponden al campo semántico de medios de transporte. Siguiendo esta idea en el problema de implicación textual, se pueden asociar términos que compartan un par de oraciones para determinar la relación semántica que guardan una con respecto a la otra y así descubrir el juicio de implicación textual entre ellas.

Este modelo se basa en la hipótesis de que un par de oraciones sostiene un juicio de implicación bidireccional y la similitud semántica entre ellas es alta. Así mismo si dos oraciones poseen una similitud semántica muy baja, se dice que el juicio de implicación textual es no-entailment.

Ahora que precisamos que la similitud semántica busca relacionar términos de acuerdo a su uso, y así poder establecer una métrica entre conceptos (significados), es preciso destacar que el uso de similitud semántica no interpreta el significado de las oraciones, se limita a relacionar términos de una manera más compleja que la sinonimia.



Para este modelo, el preprocesamiento de los datos consiste en remover palabras cerradas y lematizar los términos de las oraciones. A continuación se detallan las características utilizadas.

### 4.3.1 Características Elegidas

En el campo de las investigaciones enfocadas a la similitud semántica se destacan dos vertientes, métricas basadas en conocimiento y métricas basadas en corpus. Este modelo hace uso de estas dos métricas. A continuación se describe cada una de ellas:

#### Métricas basadas en conocimiento

Este enfoque también es llamado métricas basadas en conocimiento, las medidas propuestas en esta vertiente son muy precisas, y es debido a que para calcularse emplean conocimiento del mundo real. Las métricas se obtienen a través del uso de taxonomías y/o ontologías.

Las taxonomías son estructuras que almacenan información mediante las relaciones “es un” o “desciende de”, así los términos se encuentran representados en un árbol n-ario. Por ejemplo, pensemos en la taxonomía de los canis (ver Imagen 19), a simple vista no pareciera haber alguna relación entre un coyote (*canis latran*) y un lobo (*canis lupus*), sin embargo se puede observar que ambos son descendientes de un ancestro común. De manera análoga en el estudio de la similitud semántica, se hace uso de este tipo de estructuras, y para determinar el valor de similitud, se han propuesto una serie de medidas, que consideran la cantidad de enlaces y los niveles de profundidad que son necesarios para conectar a ambos nodos.

Ahora bien, las taxonomías se han estado creando y actualizando desde hace años, solo basta pensar en la publicación “*On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*” de Charles Darwin publicada en 1859, para darnos una idea, entonces caemos en cuenta de que este tipo de estructuras son difíciles de construir y que si quisiéramos capturar todo el conocimiento del mundo real en este tipo de estructura sería una tarea laboriosa. Lamentablemente este hecho deriva en los pocos recursos léxicos organizados taxonómicamente, hoy en día solo se cuenta con WordNet<sup>2</sup>.

Por otro lado, las ontologías son similares a las taxonomías, pero estas permiten relacionar los términos a través de otro tipo de relaciones, por ejemplo una relación entre los términos coyote y lobo puede ser que tienen-un-ancestro-común, esta manera de almacenar la información se asemeja a un grafo dirigido. La construcción de ontologías se realiza de forma supervisada, y es una tarea laboriosa, incluso más que las taxonomías, no obstante se han realizado trabajos enfocados en la construcción de ontologías de manera no supervisada, pero la verosimilitud de la información no está garantizada.

---

<sup>2</sup> [http:// wordnet.princeton.edu](http://wordnet.princeton.edu)

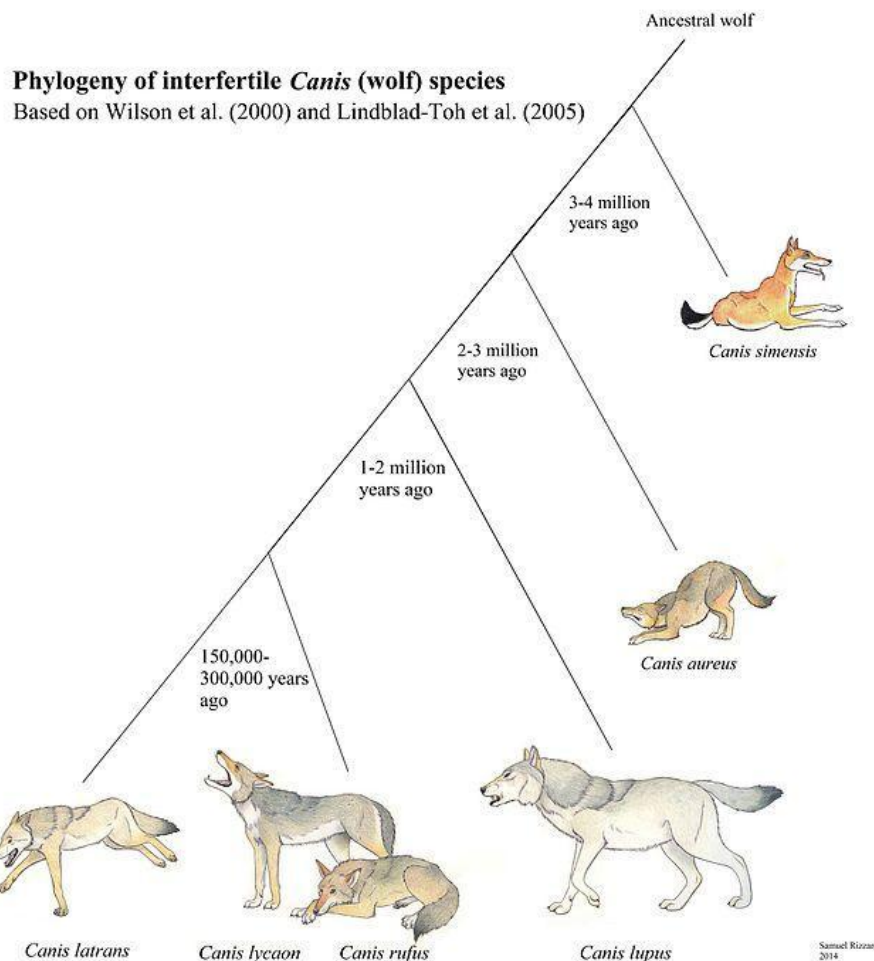


Imagen 19: Ejemplo de estructura taxonómica.

En el estudio de la similitud semántica se busca proponer algoritmos que toman la estructura taxonómica ofrecida por WordNet y generen una métrica de similitud, este tipo de medidas se apoyan en reglas de navegación entre nodos y profundidades entre ramas de la taxonomía para conectar dos nodos (términos) y así dar una medida. Algunas de estas métricas son [18 - 22]:

- Path Similarity
- Leacock-Chodorow Similarity
- Wu-Palmer Similarity
- Resnik Similarity
- Jiang-Conrath Similarity
- Lin Similarity

En este modelo se llevan un contador, por cada métrica aplicable a WordNet; el valor del contador es la suma de aplicar la métrica de todos los términos contra todos, descartando los términos iguales. El Algoritmo 1 muestra cómo se realiza la acumulación de métricas.



```

Funcion acumular_valor(Sentencia S1, Sentencia S2,
                        Funcion similitud){
    suma = 0
    para cada término T1 en S1 hacer
        para cada término T2 en S2 hacer
            si T1 != T2 hacer
                suma = suma + similitud(T1, T2)
    regresar suma
}

```

**Algoritmo 1:** Función para acumular valores de similitud

## Métricas basadas en corpus

En la actualidad la mayoría de la información que existe en fuentes digitales se encuentra en forma de texto, este tipo de información es plana y por ello se dice que es no-estructurada. Como hay una gran cantidad de información disponible, es lógico tratar de emplear estos recursos para generar medidas de similitud semántica. Dado que no es posible convertir información no-estructurada en estructurada, el paradigma cambia, ahora en lugar de buscar relaciones existentes se toman recursos no estructurados para generar métricas de similitud semántica. De aquí se rescatan dos propuestas: la Información Mutua y el Análisis Semántico Latente.

La Información Mutua o Pointwise Mutual Information (PMI) en inglés, es utilizada en la teoría probabilística para medir la dependencia de dos variables, esto quiere decir que si dos variables aparecen juntas en la mayoría de eventos observados, al evaluar su información mutua obtendremos un valor cercano a cero, en caso contrario decimos que estas variables son independientes.

Si se toma un corpus y se considera a cada documento como un evento del espacio muestral, se puede calcular la probabilidad de que dos términos aparezcan de forma continua. De esta manera se puede aplicar el PMI a un corpus para obtener un valor de similitud semántica que sea capaz de denotar que dos términos se escriben de manera continua, pero no si sostienen una relación semántica, para ello es necesario cotejar todos los bigramas que puedan generarse a partir de una oración, así al aplicar PMI a un par de términos se sabe si están semánticamente relacionados [23].

Por otro lado, el Análisis de Semántico Latente o Semantic Latent Analysis (LSA) en inglés, es una variante del modelo de espacio vectorial que, en este contexto, sirve para asociar palabras a conceptos, es decir que esta técnica aglomera los términos similares en grupos, esto es que si dos términos son utilizados en el mismo concepto deben pertenecer al mismo grupo. El LSA emplea una representación vectorial basada en bolsa de palabras, cada dimensión del vector representa un término del corpus, para cada documento representado de esta manera, se



dejan en cero o uno en función de la presencia o ausencia del término de esa dimensión, algunas variantes utilizan otras métricas como tf-idf o entropía. Una vez que el corpus es representado mediante vectores, se utiliza la Descomposición en Valores Singulares (DSV), el truco para determinar la cantidad de conceptos deseados, o sea grupos de palabras, es arbitrario y corresponde a la cantidad de valores singulares que se utilicen. Al final quien hace la agrupación es el proceso de DSV, el lector puede encontrar una explicación más detallada en [24].

Tanto PMI como LSA son dependientes de la colección que se emplee para la construcción de la métrica de similitud semántica, de modo que si se entrena con un corpus de dominio médico y se desea saber la similitud semántica de un texto que pertenece a un dominio de economía, ambos enfoques fallarán. Así que es recomendable utilizar un corpus de diversos dominios en la confección de ambas métricas, para garantizar un buen desempeño.

En el trabajo llevado a cabo por [23], se introduce la Similitud de Rada que emplea PMI como soporte para alimentar una métrica que dice si dos oraciones son semánticamente equivalentes. Como PMI y LSA retornan cero si los términos son semánticamente equivalentes, es posible aplicar la Similitud de Rada con PMI y LSA. Así mismo se aconseja un preprocesamiento del corpus que consiste en remover palabras cerradas y lematizar cada oración.

El modelo que se propone emplea un vector con 11 características. Las 6 medidas de similitud ofrecidas en WordNet, el número de términos que comparten ambos textos, el número de sinónimos que comparten, el número de hiperónimos, la similitud de Rada con PMI y las similitud de Rada con LSA.

Al igual que en el modelo estadístico, se desarrolla un modelo de clasificación supervisada utilizando la Máquina de Soporte Vectorial de la herramienta Weka.

### **4.3 Modelos de Eliminación de Tokens**

A diferencia de los modelos anteriores, esta propuesta es no supervisada. El principal motivo del cambio del paradigma radica en el solapamiento de las clases, si entrenamos con una clase solapada y luego aplicamos ese modelo, obtendremos juicios de implicación textual que no corresponden a los que realmente son, pero esto no quiere decir que las características elegidas sean malas, esto es consecuencia de la distribución de los conjuntos de datos.

A continuación se presenta un modelo que toma un par de oraciones, elimina términos comunes y con respecto al porcentaje de eliminación se asigna el juicio de implicación textual. El preprocesamiento de los datos en este modelo, consiste en llevar todo a un léxico común, y dejar las palabras cerradas.



### 4.3.1 Eliminación de elementos comunes para detectar la implicación textual

Para detectar el juicio de implicación textual que sostienen dos oraciones, se sigue el siguiente razonamiento: Las oraciones tienen información en común, unas más y otras menos, si se toman estos elementos comunes y los se eliminan de ambas oraciones, quedan términos únicos en cada oración. Los términos sobrantes al finalizar el proceso de eliminación se pueden cuantificar en relación a la longitud original de la oración, esto es que podemos medir el porcentaje de elementos eliminados y así determinar el juicio de implicación textual, por ejemplo si al terminar de eliminar los elementos comunes de un par de oraciones, se obtiene que se ha eliminado más del 80% en ambas oraciones, quiere decir que la cantidad de información compartida es mucha y por lo tanto se dice que el juicio de implicación es bidireccional. Se nota que este modelo emplea un umbral para poder determinar el juicio de implicación textual. A continuación se formaliza este principio.

Sea  $C$  el conjunto de elementos comunes entre las oraciones  $S_1$  y  $S_2$ .

Sea  $P_1$  el porcentaje restante de la oración  $S_1$  al eliminar los elementos de  $C$ .

Sea  $P_2$  el porcentaje restante de la oración  $S_2$  al eliminar los elementos de  $C$ .

Sea  $U$  un umbral de porcentaje.

Se dice que el juicio de implicación textual está dado por una de las siguientes reglas:

- Si  $P_1 > U$  &&  $P_2 > U$ : El juicio de implicación es bidireccional, ya que ambas oraciones comparten mucha información.
- Si  $P_1 < U$  &&  $P_2 > U$ : El juicio de implicación es backward, porque la oración  $S_2$  está contenida en la oración  $S_1$ , y  $S_1$  contiene mucha más información que  $S_2$ .
- Si  $P_1 > U$  &&  $P_2 < U$ : El juicio de implicación es forward, porque la oración  $S_1$  está contenida en la oración  $S_2$ , y  $S_2$  contiene mucha más información que  $S_1$ .
- Si  $P_1 < U$  &&  $P_2 < U$ : El juicio de implicación de no-entailment, ya que ambas oraciones comparten muy poca información.

### 4.3.2 Tokenización de oraciones

Como la aproximación propuesta elimina elementos comunes, es necesario precisar en qué consiste un elemento común. Desde un punto de vista lingüístico-computacional, un elemento de la oración es denominado token, que a simple vista pareciera ser un uni-grama, pero no lo es; la diversidad de tokens es muy amplia, por ejemplo una fecha está formada por el día, el mes y el año, un nombre de persona se compone por tres o más unigramas, por mencionar algunos ejemplos. En un enfoque lingüístico un elemento de una oración es una frase que transmite información relacionada entre los términos, por ejemplo en la oración "*Lisbeth tomó su vieja moto y emprendió su viaje a Nueva Zelanda*", sería tokenizada como {"*Lisbeth*", "*tomó*", "*vieja moto*", "*emprendió*", "*viaje*", "*Nueva Zelanda*"}



Para eliminar los elementos (tokens) comunes, es necesario llevar a cabo un proceso de tokenización. Afortunadamente hoy en día existen muchas herramientas para llevar a cabo esta tarea. Para este modelo se han considerado los siguientes métodos de tokenización:

- **Tokenización de unigramas:** Consiste en llevar ambas oraciones a un vocabulario común, eliminar palabras cerradas y asumir un uni-grama como un token.
- **Tokenización por palabras cerradas:** Consiste en recorrer la oración y a medida que se encuentra una palabra cerrada, se genera un token, esta propuesta es totalmente empírica.
- **Tokenización por árbol sintáctico:** Los parsers sintácticos procesan una oración y generan un árbol sintáctico, donde en cada nodo hoja se almacenan n-gramas que corresponden a los términos de la oración, se propone tomar como un token a los n-gramas que se encuentren en los nodos hoja.
- **Tokenización por herramienta:** Existen librerías que llevan a cabo el proceso de tokenización, de este modo sólo se toman los tokens generados por alguna herramienta.

A continuación se discute como se establece la similitud entre tokens.

### 4.3.3 Criterio de similitud entre tokens

Precisando que los tokens presentes en cada oración pueden ser n-gramas, es necesario determinar cuán similar es un n-grama con respecto a otro, el caso más trivial es que ambos sean el mismo, sin embargo no siempre es así. Por ello en este apartado se introduce un criterio para determinar si dos tokens son iguales.

Para determinar si un par de tokens son similares, se retoma el principio de este modelo, y se dice que son iguales si comparten más de cierto umbral. El Algoritmo 2, muestra la función de similitud entre tokens.

```

Funcion are_similar_tokens(Token t1, Token t2, Umbral k){
    len_min = MIN( length(t1), length(t2) )
    comun = 0
    para cada término ti en t1 hacer
        para cada término tj en t2 hacer
            si t1 == t2 hacer
                comun = comun +1

    si (comun / len_min) > k hacer
        regresar Verdadero
    si-no hacer
        regresar Falso
}

```

**Algoritmo 2:** Función que determina si dos tokens son similares



#### 4.3.4 Montaje del experimento

Este modelo utiliza el Algoritmo 3 para detectar el juicio de implicación, se nota que se puede variar el umbral de similitud entre oraciones y el umbral de similitud entre tokens, así como la función de tokenización.

```

Funcion obtener_juicio(Sentencia S1, Sentencia S2,
                      Umbral u_oracion, Umbral u_sim_token){
tokens_1 = tokenizar(S1)
tokens_2 = tokenizar(S2)

len_s1_inicial = length(tokens_1)
len_s2_inicial = length(tokens_2)

para cada token ti en tokens_1 hacer
  para cada token tj en tokens_2 hacer
    si are_similar_tokens(ti, tj, u_sim_token) hacer
      eliminar ti de tokens_1
      eliminar tj de tokens_2

len_s1_final = length(tokens_1)
len_s2_final = length(tokens_2)

p1 = 1 - (len_s1_final / len_s1_inicial)
p2 = 1 - (len_s2_final / len_s2_inicial)

segun el caso hacer
  si p1 > u_oracion && p2 > u_oracion : regresar "bidirectional"
  si p1 < u_oracion && p2 > u_oracion : regresar "backward"
  si p1 > u_oracion && p2 < u_oracion : regresar "forward"
  si p1 < u_oracion && p2 < u_oracion : regresar "no_entailment"
}

```

**Algoritmo 3:** Función para determinar el juicio de implicación textual mediante eliminación de tokens

Cada oración de la colección de datos es evaluada con el Algoritmo 3 para obtener el juicio de implicación textual.



## 4.4 Modelo de eliminación con análisis semántico

En el modelo anterior se lleva a cabo un proceso de eliminación de tokens, que comparten un par de oraciones, esto es que se eliminan los elementos comunes en ambas oraciones para determinar el juicio de implicación textual. El criterio de eliminación está en función de la similitud entre tokens, pero si se cuenta con palabras relacionadas de manera semántica, la aproximación no puede eliminar ese par de tokens. Dada esta observación es necesario plantear una modificación en el modelo.

Este modelo es una variante del modelo de eliminación, ahora se busca refinar la eliminación, al agregar una etapa posterior en la que se busca la similitud semántica entre un término con respecto a la oración inicial, si la palabra está relacionada semánticamente también debe ser eliminada, así se garantiza que la eliminación sea lo más justa posible.

Para este modelo, se contempla un preprocesamiento en los datos que consiste en lematizar y eliminar palabras cerradas para cada par de oraciones.

### 4.4.1 Criterio de similitud entre palabras

Considerando lo que se mencionó en la sección 4.2.1, la similitud entre términos no necesariamente es una sinonimia, hiperonimia o hiponimia; en ocasiones es necesario llevar a cabo un análisis semántico, en aras de detectar si los términos denotan el mismo concepto.

Para este modelo se toma la eliminación que realiza el modelo 4.3, y se realiza una eliminación posterior que considera la similitud de los conceptos. Esta eliminación posterior considera que si en el par de tokens a eliminar, aparece una pareja de términos que se encuentra con un valor de relación muy cercano a cero, entonces los tokens deben ser eliminados. El valor de la relación, bien puede obtener utilizando medidas basadas en conocimiento o medidas basadas en corpus, ver sección 4.2.1.

En el Algoritmo 4, se muestra la función que determina si dos tokens están relacionados semánticamente, se considera un umbral cercano a cero y un valor de relación, dado por alguna función genérica denominada *similitud\_semantica*.

```

Funcion are_semantic_similar_tokens(Token t1, Token t2,
                                   Umbral k){
  para cada término ti en t1 hacer
    para cada término tj en t2 hacer
      si similitud_semantica(t1, t2) < k hacer
        regresar Verdadero

  regresar Falso
}

```

**Algoritmo 4:** Función determinar si dos tokens son semánticamente similares



#### 4.4.2 Montaje del experimento

El algoritmo 5 es una variante del algoritmo 3, al igual que su predecesor, en este algoritmo se puede variar el umbral de similitud entre oraciones, el umbral de similitud entre tokens y la función de tokenización, además de que esta nueva aproximación puede variar la función de similitud semántica y el umbral que determina si dos términos son semánticamente similares.

```

Funcion obtener_juicio(Sentencia S1, Sentencia S2,
                      Umbral u_oracion, Umbral u_sim_token,
                      Umbral u_sim_sem_token){
tokens_1 = tokenizar(S1)
tokens_2 = tokenizar(S2)

len_s1_inicial = length(tokens_1)
len_s2_inicial = length(tokens_2)

para cada token ti en tokens_1 hacer
  para cada token tj en tokens_2 hacer
    si are_similar_tokens(ti, tj, u_sim_token) hacer
      eliminar ti de tokens_1
      eliminar tj de tokens_2

para cada token ti en tokens_1 hacer
  para cada token tj en tokens_2 hacer
    si are_semantic_similar_tokens(ti, tj, u_sim_sem_token) hacer
      eliminar ti de tokens_1
      eliminar tj de tokens_2

len_s1_final = length(tokens_1)
len_s2_final = length(tokens_2)

p1 = 1 - (len_s1_final / len_s1_inicial)
p2 = 1 - (len_s2_final / len_s2_inicial)

segun el caso hacer
  si p1 > u_oracion && p2 > u_oracion : regresar "bidireccional"
  si p1 < u_oracion && p2 > u_oracion : regresar "backward"
  si p1 > u_oracion && p2 < u_oracion : regresar "forward"
  si p1 < u_oracion && p2 < u_oracion : regresar "no_entailment"
}

```

**Algoritmo 5:** Variante del algoritmo 3 que considera eliminación por similitud textual



## 4.5 Modelo de inferencia basado en anclas

Al recapitular las propuestas anteriores, se nota que todas siguen una hipótesis común, si existe una gran cantidad de información compartida, quiere decir que las oraciones comunican la misma idea. Esta asunción no es del todo correcta, ya que puede presentarse el caso de que dos oraciones que compartan una gran cantidad de información, no transmitan la misma idea, por ejemplo el par de oraciones: “*La becerra de Manuel está en la calle*” y “*Esta es la calle de Manuel Becerra*” es un par equivalente a nivel de tokens, pero al momento de interpretar ambas oraciones, se descubre que hablan de cosas completamente diferentes.

Ante esta nueva evidencia, es necesario proponer un nuevo modelo, el cual sea capaz de detectar los tokens que comparten un par de oraciones, y descubrir si estos tokens se relacionan de manera similar en ambas oraciones. Como preprocesamiento sólo se lematizan las oraciones.

### 4.5.1 Términos Anclas

En la técnica del muestreo de plantillas (ver sección 3.4.1) se aborda un concepto de empatamiento por estructuras rígidas denominadas plantillas. Las plantillas de empatamiento garantizan la implicación textual, ya que son verdaderas y están validadas, el problema es generar estas plantillas, no obstante se puede ahondar un poco más sobre ellas. Consideremos el par de plantillas “*X es obra de Y*” y “*X escribió Y*”, veamos que en ambas se repiten las variables *X* y *Y*, estas variables pueden ser sustituidas por cualquier par de tokens, como por ejemplo “*Tokio Blues es obra de Murakami*”, “*Murakami escribió Tokio Blues*”; o incluso “*Esta tesis es obra de Saúl*”, “*Saúl escribió esta tesis*”. Ahora bien, estos tokens son denominados anclas, ya que al estar presentes en ambas oraciones, nos permiten detectar la posición de los elementos en cada oración, de esta manera se pueden analizar cómo es que se relaciona una ancla con la otra dentro de cada par de oraciones. Siguiendo la metodología del empatamiento de plantillas, una vez detectada la posición de las anclas se recurre a buscar un conjunto de plantillas que contenga el texto que está entre las anclas, si se encuentra un par de plantillas que satisfacen las posiciones de las anclas, así como su relación (el texto que hay entre ellas), se dice que hay implicación textual.

Ahora pensemos de manera inversa. A partir de una oración se buscan sus tokens anclas, si estos distan entre sí, a no más de dos tokens en cada oración, quiere decir que se encuentran relativamente cerca, y sospechamos que sostienen alguna relación. Entonces, si en lugar de buscar la relación en plantillas, pudiéramos inferir qué similitud existe entre ambas relaciones, estaríamos descubriendo la implicación textual sin el uso de plantillas, pero con el mismo principio.

En el siguiente apartado se explica cómo se puede llevar a cabo el proceso de inferencia.



### 4.5.2 Proceso de inferencia

La inferencia de dos relaciones, se puede abordar con diferentes técnicas, en los siguientes incisos se describen cada una de ellas:

- **Relaciones Directas:** Las relaciones directas corresponden a sinonimia, hiperonimia o hiponimia. Son relaciones tan obvias que basta con sustituir la relación con su sinónimo para detectar si son relaciones iguales.
- **Similitud Semántica:** Estas relaciones son afines, no necesariamente son sinónimos, pero son empleadas bajo contextos similares. Al tomar dos relaciones y detectar una relación semántica, entonces decimos que son iguales.

Para detectar el juicio de implicación que sostienen dos oraciones, se recurre al concepto de eliminación de información común, pero a diferencia de la eliminación de tokens, ahora se eliminarán relaciones de términos anclas, esta nueva variante garantiza que la información eliminada en ambas oraciones corresponde a un nivel interpretativo, más allá del léxico.

Se plantea el algoritmo de la siguiente manera, primero se obtienen todos los términos anclas, que por simplicidad serán los n-gramas más largos presentes en ambas oraciones, posteriormente se tokenizan ambas oraciones y se verifica la distancia a la que se encuentran las anclas, si los términos anclas no exceden un máximo de dos tokens entre ellos en ambas oraciones, se verifican las relaciones que sostienen en una y en otra. Si las relaciones son similares, se eliminan las relaciones y las anclas involucradas en cada oración, al final se asigna el juicio de implicación, en dependencia del porcentaje de eliminación, tal y como se expone en la sección 4.4.1. Los pasos anteriores se pueden apreciar en el Algoritmo 6.



```

Funcion obtener_juicio(Sentencia S1, Sentencia S2){
  anclas = obtener_anclas(S1, S2)
  tokens_1 = tokenizar(S1)
  tokens_2 = tokenizar(S2)

  len_s1_inicial = length(S1)
  len_s2_inicial = length(S2)

  para cada n-grama ai en anclas hacer
    para cada n-grama aj en anclas hacer
      si ai != aj and are_near(ai, aj, tokens_1) and
        are_near(ai, aj, tokens_2) hacer
        r1 = obtener_relacion(ai, aj, tokens_1)
        r2 = obtener_relacion(ai, aj, tokens_2)

        si are_similar_relations(r1, r2) hacer
          eliminar ai, r1 y aj de S1
          eliminar ai, r2 y aj de S2
          eliminar ai y aj de anclas

  len_s1_final = length(S1)
  len_s2_final = length(S2)

  p1 = 1 - (len_s1_final / len_s1_inicial)
  p2 = 1 - (len_s2_final / len_s2_inicial)

  segun el caso hacer
    si p1 > u_oracion && p2 > u_oracion : regresar "bidireccional"
    si p1 < u_oracion && p2 > u_oracion : regresar "backward"
    si p1 > u_oracion && p2 < u_oracion : regresar "forward"
    si p1 < u_oracion && p2 < u_oracion : regresar "no_entailment"
}

```

**Algoritmo 6:** Propuesta de eliminación de tokens basada en anclas y relaciones



Del Algoritmo 6 se detectan las funciones:

- *obtener\_anclas*: Esta función genera los n-gramas más largos que comparten ambas oraciones.
- *are\_near*: Esta función regresa verdadero si en una oración, los términos ancla se encuentran a no más de 2 tokens de distancia, en caso contrario el valor de retorno es falso.
- *obtener\_relacion*: Esta función regresa el segmento de texto de una oración, que se encuentra acotado por un par de anclas.
- *are\_similar\_relations*: Es una función que decide si los dos segmentos de texto que está comparando son similares o relacionados, internamente esta función hace uso de sinonimia y similitud semántica.

## 4.6 Modelo de interpretación de oraciones basado en grafos

A lo largo de los modelos anteriores se han propuesto algunas metodologías que intentan resolver el problema de la implicación textual a través de propuestas supervisadas y no supervisadas. Se han empleado medidas de similitud léxica, similitud semántica, eliminación de información en función de la similitud y eliminación de relaciones entre términos anclas. Pese al esfuerzo realizado los resultados no son suficientes, lo que nos lleva a replantear el problema.

Recordemos que en un lenguaje natural, las oraciones tienen tres niveles de composición, léxico, sintáctico y semántico. En cada nivel se evalúan aspectos diferentes, al descomponer una oración en sus tres niveles, se puede apreciar ciertas igualdades y diferencias.

Por ejemplo si se toman dos oraciones que comparten poco léxico, que una oración se encuentra escrita en voz pasiva y la otra en voz activa, y que en cuestión semántica difieren; para un modelo computacional que se basa en similitudes, este tipo de oraciones es claramente un *no\_entailment*, y sin embargo puede ser bidireccional. Sin embargo el humano es capaz de interpretar el contenido de cada oración, y gracias a su sentido común, es capaz de relacionar si las ideas son las mismas, de esta manera oraciones como “*Un hombre sostiene un aparato electrónico*” y “*Una cámara digital es aguantada por un transeúnte*” son oraciones que reflejan la misma idea.

Hasta el desarrollo de esta investigación, no se ha encontrado reportado en la bibliografía ningún sistema que resuelva TE y mucho menos CLTE a través de una interpretación de oraciones, por ello se considera de suma importancia proponer un modelo no supervisado que interprete las oraciones para identificar qué entidades se encuentran en juego y determinar las relaciones que sostienen en cada oración, posteriormente se debe detectar qué entidades son equivalentes y verificar si sus relaciones también son equivalentes, para así determinar si son la misma idea.



El proceso de interpretación de oraciones que se presenta en esta metodología, tiene su origen en las investigaciones de Preguntas y Respuestas. Dentro de los sistemas de preguntas y respuestas se encuentran sistemas que procesan preguntas factuales y preguntas complejas. Las preguntas factuales, corresponden a información verdadera que se encuentra escrita textualmente en algún lugar, mientras que las preguntas complejas, son preguntas que necesitan valerse de información extra para poder ser respondidas, por ejemplo, la pregunta “¿Dónde nació Albert Einstein?” es una pregunta factual, y la pregunta “¿Que medicamento le puedo dar a un paciente enfermo de gripe si es alérgico al paracetamol?” es una pregunta compleja.

Como en el descubrimiento de la implicación textual involucra un par de oraciones, es posible ver una oración como la fuente de conocimiento y la otra como las preguntas que se le hacen a un sistema de preguntas factuales. En un sistema de preguntas factuales se generan árboles sintácticos de la información que se desea indexar, esta estructura permite generar estructuras atómicas del tipo *Sujeto-Verbo-Objeto*, o lo que es lo mismo *Entidad<sub>1</sub>-Accion-Entidad<sub>2</sub>*, una vez generadas estas tripletas son agregadas al índice, cuando una pregunta es lanzada al sistema, este identifica las entidades y acciones presentes en la pregunta y lanza un buscador con operador OR sobre su índice de tripletas, de modo que todas las tripletas recuperadas son respuestas de la pregunta, por ejemplo consideremos el siguiente ejemplo:

Sean los siguientes incisos, el índice de tripletas de un sistema de preguntas y respuestas:

1. (“Los lobos”, “comen”, “conejos”)
2. (“Las tortugas”, “comen”, “ranas”)
3. (“Las tortugas”, “viven”, “el mar”)

Cuando se le pregunta al sistema “¿Dónde viven las tortugas?”, este identifica los términos “viven” y “tortugas”, los cuales son buscados en sus índices y encuentra que el índice 3 contiene ambos términos, finalmente el sistema asume que “el mar” es la respuesta de la pregunta. Esta investigación retoma este mecanismo de validación de información para interpretar las oraciones.

Este modelo es una aproximación que emplea parsers sintácticos para interpretar las oraciones, después con ayuda de una serie de reglas se extraen hechos de cada oración, estos hechos se utilizan en la confección de grafos. Cada nodo de un grafo generado corresponde a términos de la oración y las aristas son las relaciones entre los términos. Una vez que se construyen los grafos de cada oración, se aplica un algoritmo que busca empatar subestructuras de los mismos, las cuales denominamos hechos-extendidos, que con ayuda de una fuente de conocimiento, se puede determinar si los nodos son equivalentes o si tendrían alguna relación. Finalmente la cantidad de subestructuras compartidas, con respecto a la cantidad de subestructuras presente en cada oración nos proporciona el juicio de implicación textual.



A continuación se exponen cada una de las etapas involucradas en la confección de este modelo.

#### 4.6.1 Extracción de hechos

Para entender que es un hecho, se considera la oración “El lobo feroz derribó la casa de los cerditos, se comió al cazador y huyó a las Vegas”, a partir de ella podemos extraer la siguiente información: el lobo es feroz, el lobo derribó una casa, la casa es de los cerditos, el lobo se comió al cazador, el lobo huyó a las Vegas. Toda esta información obtenida es denominada hechos, y se define como información verdadera generada a partir de una oración.

Antes de continuar explicando esta etapa, es bueno comentar que para esta investigación no se toma en cuenta toda la información que arroja una oración, se propone trabajar con un pequeño conjunto de hechos, la razón principal de esta decisión es que es el primer modelo de interpretación de oraciones y para poder evaluar su comportamiento, este debería ser lo más sencillo posible.

El proceso de extracción de hechos consiste en tomar una oración y generar toda la información relevante a partir de ella, para realizar esta tarea se emplea el parser sintáctico de *Stanford*<sup>3</sup>, quien genera el árbol sintáctico. A partir del árbol sintáctico se visita cada nodo para poder generar nuevos nodos, a continuación se listan los nombres de los nuevos nodos:

- **ENTIDAD:** Son todos los términos de los nodos hijos que contengan algún sustantivo, se recuperan todos los términos asociados a las etiquetas de PoS del tipo: NN, NNS, NNP, NNPS y NAC.
- **ACTIVIDAD:** Son todos los términos de los nodos hijos que contengan algún verbo, se recuperan todos los términos asociados a las etiquetas de PoS del tipo: VB, VBG, VBZ, VBP, VBN, VBD y MD.
- **CALIDAD:** Son todos los términos de los nodos hijos que contengan algún adjetivo calificativo, se recuperan todos los términos asociados a las etiquetas de PoS del tipo: JJ y JJS.
- **PREPOSICIÓN:** Son todos los términos de los nodos hijos que contengan alguna proposición, se recuperan todos los términos asociados a las etiquetas de PoS del tipo: IN y TO.

Los nombres de los nodos reflejan los roles de la información recuperada (texto). Para relacionar la información recabada, se buscan los siguientes hechos:

- **Sujeto:** Este hecho es obtenido cuando una entidad está asociada a una, o más, actividades. Se asume que la ENTIDAD es el sujeto de la ACTIVIDAD.
- **Objeto:** Este hecho es obtenido cuando una actividad está asociada a una, o más, entidades. Se dice que la ENTIDAD es el objeto de la ACTIVIDAD.

<sup>3</sup> <http://nlp.stanford.edu/software/lex-parser.shtml>



- **Califica:** Este hecho es obtenido cuando se detecta una cualidad asociada a una entidad, entonces se dice que esa CALIDAD califica a la ENTIDAD.
- **Extensión:** Este hecho es obtenido cuando una ENTIDAD está asociada a una PREPOSICIÓN
- **Complemento:** Este hecho es obtenido cuando una PROPOSICIÓN está asociada a una ENTIDAD.

Estas reglas son extraídas al recorrer el árbol sintáctico, que contienen las nuevas etiquetas, y en cada nodo se aplican las reglas de la Tabla 2 para extraer los hechos.

Regla	Hecho
X ⇒ ENTIDAD && X ⇒ ACTIVIDAD	<i>sujeto</i> (ENTIDAD, ACTIVIDAD)
ACTIVIDAD ⇒ ENTIDAD	<i>objeto</i> (ACTIVIDAD, ENTIDAD)
X ⇒ ENTIDAD && X ⇒ CALIDAD	<i>califica</i> (CALIDAD, ENTIDAD)
ENTIDAD ⇒ PREPOSICIÓN	<i>extensión</i> (ENTIDAD, PREPOSICIÓN)
PREPOSICIÓN ⇒ ENTIDAD	<i>complemento</i> (PREPOSICIÓN, ENTIDAD)

Tabla 2: Reglas para la extracción de hechos

El proceso completo puede ser apreciado en la Imagen 20. Pese a que los hechos son un buen indicativo de la información que existe en una oración, no es suficiente hacer un estudio aislado de ellos. Por ello se decide emplear una estructura de grafo para representar la información, en la siguiente sección se describe cómo se genera un grafo a partir de un conjunto de hechos.

Stage	Sentence 1	Sentence 2
Original Sentence	The old man is skillfully playing the guitar	An adult person is playing an instrument
Stanford Syntactic-Parser output	<pre>( ROOT   ( S     ( NP       ( DT The )       ( JJ old )       ( NN man )     )     ( VP       ( VBZ is )       ( ADVP ( RB skillfully ) )       ( VP         ( VBG playing )         ( NP           ( DT the )           ( NN guitar )         )       )     )   ) )</pre>	<pre>( ROOT   ( S     ( NP       ( DT An )       ( NN adult )       ( NN person )     )     ( VP       ( VBZ is )       ( VP         ( VBG playing )         ( NP           ( DT an )           ( NN instrument )         )       )     )   ) )</pre>
Adding new tags	<pre>( ROOT   ( S     ( NP ( ENTITY man ) ( QUALITY old )       ( DT The )       ( JJ old )       ( NN man )     )     ( VP       ( VBZ is )       ( ADVP ( RB skillfully ) )       ( VP ( ACTIVITY playing )         ( VBG playing )         ( NP ( ENTITY guitar )           ( DT the )           ( NN guitar )         )       )     )   ) )</pre>	<pre>( ROOT   ( S     ( NP ( ENTITY adult person )       ( DT An )       ( NN adult )       ( NN person )     )     ( VP       ( VBZ is )       ( VP ( ACTIVITY playing )         ( VBG playing )         ( NP ( ENTITY instrument )           ( DT an )           ( NN instrument )         )       )     )   ) )</pre>
Facts extracted	<pre>subject(man, playing) object(playing, guitar) qualify(old, man)</pre>	<pre>subject(adult person, playing) object(playing, instrument)</pre>

Imagen 20: Ejemplo de extracción de hechos para un par de oraciones.

#### 4.6.2 Interpretación de hechos sobre grafos

Al principio de este capítulo, se mencionaba que la estructura empleada por los sistemas de preguntas y respuestas, para representar la información indizada, consiste en una tripleta de la forma *Entidad<sub>1</sub>-Actividad-Entidad<sub>2</sub>*. Ahora bien, bajo el contexto de los hechos se aprecia que corresponde a un par de hechos: *sujeto(Entidad<sub>1</sub>, Actividad) && objeto(Actividad, Entidad<sub>2</sub>)*, de modo que es necesario representar los hechos a través de una estructura que permita preservar tanto la información de los hechos como las relaciones entre ellos. Por ello se recurre a una estructura de grafo.

Al representar los hechos en un grafo surge el concepto de hechos ampliados, un hecho ampliado es la manera en la que se relacionan los términos de alguna oración, bajo el contexto de preguntas y respuestas corresponde a una tripleta, y para el contexto de grafos corresponden a subestructuras del grafo. A partir de la representación de los hechos mediante



grafos, se detectan tres tipos de subestructuras. La Tabla 3 describe las subestructuras y sus equivalentes en tripletas.

<b>Subestructura</b>	<b>Tripleta</b>
sujeto(Entidad <sub>1</sub> , Actividad) & objeto(Actividad, Entidad <sub>2</sub> )	(Entidad <sub>1</sub> , Actividad, Entidad <sub>2</sub> )
califica(Entidad, Calidad)	(Entidad, califica, Calidad)
extensión(Entidad <sub>1</sub> , Preposición) & complemento(Preposición, Entidad <sub>2</sub> )	(Entidad <sub>1</sub> , Preposición, Entidad <sub>2</sub> )

**Tabla 3:** Subestructuras y sus equivalencias en tripletas.

Para representar los hechos a través de grafos, basta con tomar a los textos y relacionarlos mediante sus hechos, la imagen 21 muestra un ejemplo de este proceso. Es conveniente mencionar que por cada oración se pueden generar n grafos.

Original Sentence

The strongest rain ever recorded in India shut down the financial hub of Mumbai, snapped communication lines, closed airports and forced thousands of people to sleep in their offices or walk home during the night, officials said today.

Facts

```

subject (rain~2, snapped~15)
subject (rain~2, closed~19)
subject (rain~2, shut~7 down~8)
qualify (strongest~1, rain~2)
extension (thousands~23, of~24)
complement (of~12, Mumbai~13)
object (shut~7 down~8, hub~11)
object (recorded~4, India~6)
object (forced~22, thousands~23)
subject (rain~2, recorded~4)
extension (hub~11, of~12)
qualify (financial~10, hub~11)
complement (of~24, people~25)
subject (thousands~23, sleep~27)
subject (officials~38, said~39)
object (said~39, today~40)
object (closed~19, airports~20)
object (sleep~27, their~29 offices~30)
object (snapped~15, C.~16 lines~17)
object (walk~32, night~36)
subject (rain~2, forced~22)
subject (thousands~23, walk~32)
object (walk~32, home~33)
    
```

Graph structure generate from facts

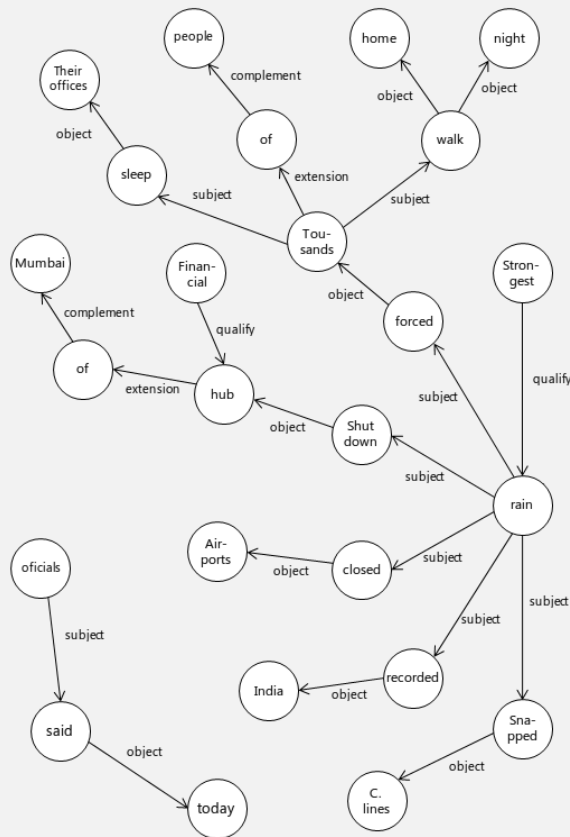


Imagen 12: Ejemplo de construcción de grafos a partir de hechos.



### 4.6.3 Empatamiento de grafos para el descubrimiento de la implicación textual

Ahora que se ha hablado de la representación de las oraciones, y cómo a través de subestructuras es posible interpretar la información que hay en ellas; es momento de plantearnos cómo medir la similitud de los grafos de las oraciones.

Dentro de la teoría de grafos, el mecanismo que denota que dos grafos tienen la misma estructura es denominado isomorfismo de grafos, que es una metodología rígida que da mayor peso a la forma del grafo que a su contenido. Por otro lado, en algunas ocasiones los grafos que se desean empatar varían en el número de nodos, y no es posible realizar un estudio de isomorfismo, en estos casos lo ideal sería aplicar un empatamiento de hipergrafos. Los hipergrafos son grafos que en cada nodo puede contener otro grafo, desafortunadamente esta última propiedad provoca que el empatamiento de grafos sea un problema NP-Completo.

Como empatar el grafo de forma matemática no es una opción viable, se propone un algoritmo ávido que recorre cada grafo para generar subestructuras, llamadas hechos-extendidos, que al final estas son validadas con otro conjunto de subestructuras. Para validar una subestructura contra otra es necesario decidir si dos segmentos de textos son equivalentes, llegados a este punto siempre se ha hablado de sinonimia, hiperonimia, hiponimia y similitud semántica. Sin embargo para este modelo se propone el uso de una fuente de información que busca capturar el sentido común del mundo real en una base de conocimientos, esta base es llamada Conceptnet<sup>4</sup>, adicionalmente se utiliza el tesoro de OpenOffice y WordNet, para conformar una base de conocimiento lo más sólida posible.

El empatamiento de subestructuras, que ahora se denomina empatamiento de hechos extendidos, consiste en visitar cada nodo de ambos grafos, generar una subestructura, y verificar si son equivalentes, para ello se emplea la base de conocimientos, se debe asegurar que los hechos extendidos, que se están analizando, posean la misma información para decir que son hechos equivalentes. Este proceso es mostrado en la imagen 22.

---

<sup>4</sup> <http://conceptnet5.media.mit.edu>

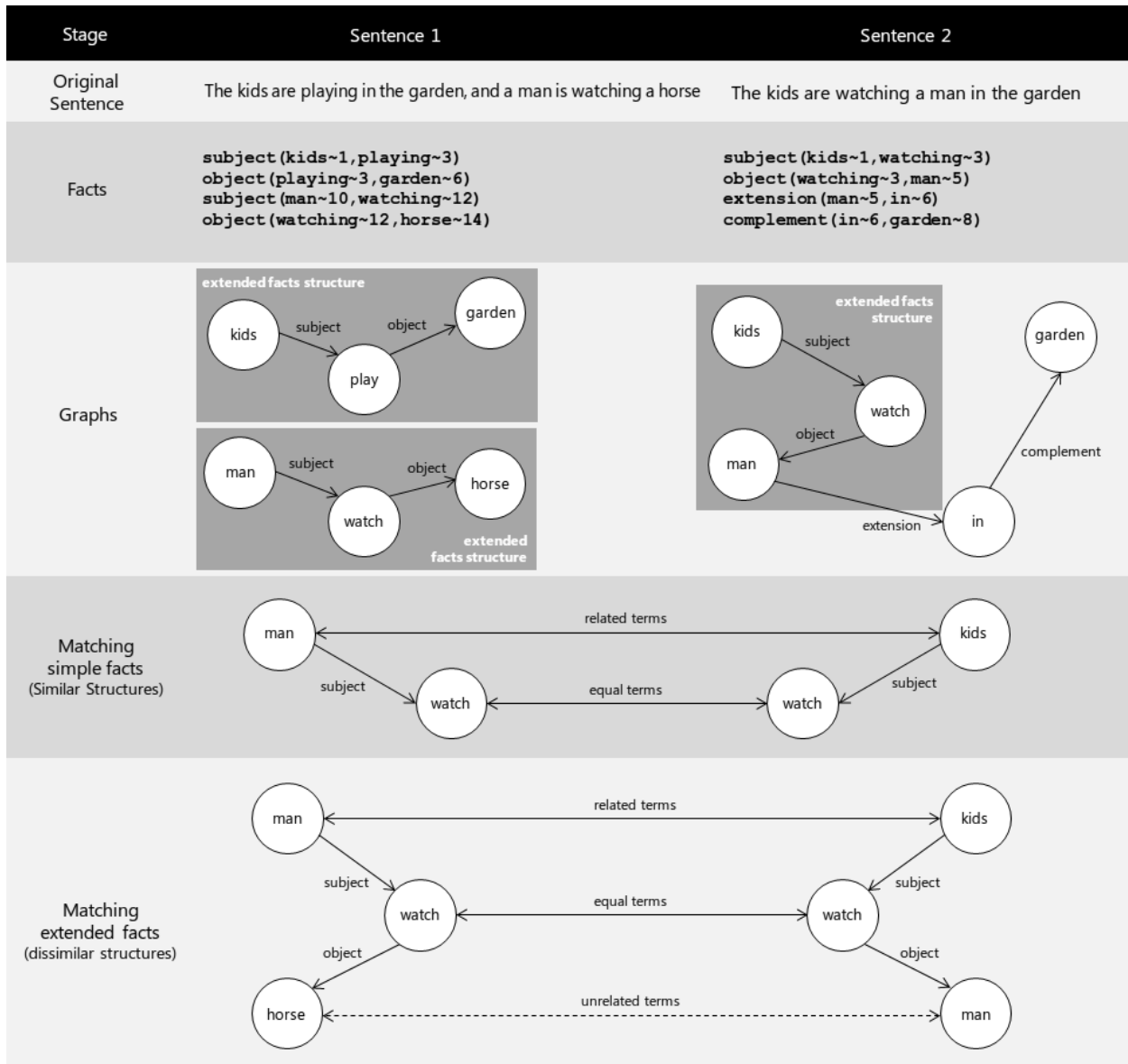


Imagen 22: Ejemplo de macheo de grafos utilizando hechos extendidos.

#### 4.6.5 Montaje del experimento

A lo largo de este capítulo se ha descrito un modelo que toma un par de oraciones, interpreta su contenido y éste se valida a través de los hechos extendidos de cada oración; para ayudar en el proceso de validación se emplea una base de conocimientos y al final se genera un porcentaje de similitud de ambas oraciones. Como puede observarse, este mecanismo de empatamiento es similar al esquema de eliminación, de modo que el juicio final está en función de la cantidad de información cotejada correctamente. A continuación se muestra el algoritmo 7, el cual corresponde al modelo presentado.



```

Funcion obtener_juicio(Sentencia S1, Sentencia S2, Umbral k){
  H1 = extraer_hechos(S1)
  H2 = extraer_hechos(S2)

  G1 = crear_grafo(H1)
  G2 = crear_grafo(H2)

  G1_num_nodos_ini = numero de nodos de G1
  G2_num_nodos_ini = numero de nodos de G2

  para cada nodo ni en anclas hacer
    para cada nodo nj en anclas hacer
      H_ext_i = obtener_subestructuras(ni)
      H_ext_j = obtener_subestructuras(nj)

      Para cada hi in H_ext_i hacer
        Para cada hj in H_ext_j hacer
          Si son_hechos_iguales(hi, hj) hacer
            de G1 eliminar los nodos involucrados en hi
            de G2 eliminar los nodos involucrados en hj

  G1_num_nodos_fin = numero de nodos de G1
  G2_num_nodos_fin = numero de nodos de G2

  p1 = 1 - (G1_num_nodos_fin / G1_num_nodos_ini)
  p2 = 1 - (G2_num_nodos_fin / G2_num_nodos_ini)

  segun el caso hacer
    si p1 > u_oracion && p2 > u_oracion : regresar "bidirectional"
    si p1 < u_oracion && p2 > u_oracion : regresar "backward"
    si p1 > u_oracion && p2 < u_oracion : regresar "forward"
    si p1 < u_oracion && p2 < u_oracion : regresar "no_entailment"
}

```

**Algoritmo 7:** Algoritmo que interpreta oraciones y valida la información equivalente para resolver el CLTE.



En el Algoritmo 7 se emplea la función *son\_hecho\_iguales*, a continuación es descrita en el Algoritmo 8.

```
Funcion son_iguales(Subgrafo SG1, Subgrafo SG2){
  si el numero de nodos de SG1 y SG2 son iguales hacer
    son_iguales = True
  para i de 0 a la cardinalidad de SG1 hacer
    son_iguales = son_iguales &
      tienen_relacion(nodo i de SG1, nodo i de SG2)
  regresar son_iguales
sino hacer
  regresar Falso
}
```

**Algoritmo 8:** Función que determina si dos subestructuras son similares.

El Algoritmo 8, emplea un llamado a la base de conocimientos mediante la función *tienen\_relacion*, si en la base de conocimientos se encuentran estos dos términos asociados se regresa verdadero, en caso contrario es falso.



## Capítulo 5: Resultados experimentales

En este capítulo se presentan los resultados alcanzados por cada uno de los modelos propuestos en el capítulo 4. Así mismo se introducen dos colecciones extras (RTE1 y SICK) que ayudarán al análisis de resultados de las metodologías desarrolladas.

### 5.1 Colecciones de Datos

A pesar de que los modelos desarrollados en esta investigación se basan en dos conjuntos de colecciones (ver sección Anexo 1), es posible extrapolar las metodologías propuestas a otros conjuntos de datos, esto gracias a que los enfoques de las propuestas toman inspiración de modelos que tratan de resolver TE.

Para probar el desempeño de los enfoques expuestos en el capítulo 4, se han considerado 3 colecciones de datos:

#### Corpus CLTE

La colección de datos del CLTE fue liberada a la comunidad científica en el foro internacional de evaluación semántica SemEval, una primera parte en su edición 2012, y una segunda en su edición 2013. El lector puede ver la el Anexo 1.1 para mayor información.

Como las colecciones del CLTE constan de 4 idiomas, se ha decidido solo reportar la subcolección de Ingles/Español, ya que es la colección en la que mejor se comportan todos los algoritmos del estado del arte. La distribución de esta colección puede apreciarse en la Tabla 4.

Conjunto	Pares de oraciones	Juicio Bidireccional	Juicio Backward	Juicio Forward	Juicio NoEntailment
CLTE12-Train	500	125	125	125	125
CLTE12-Test	500	125	125	125	125
CLTE13-Test	500	125	125	125	125

Tabla 4: Información de la subcolección Ingles/Español del Corpus CLTE.

#### Corpus RTE1

La colección de datos del RTE1 (Recognising Textual Entailment 1 [25]) fue liberada en el marco del Pattern Analysis, Statistical Modelling and Computational Learning (SemEval), donde se utilizó por primera vez el término Textual Entailment. El corpus RTE1 consta de 2 clases, las cuales se encuentran balanceada. La Tabla 5 muestra la distribución de las colecciones que conforman al corpus RTE1.



Conjunto	Pares de oraciones	Pares de oraciones con juicio "Entailment"	Pares de oraciones con juicio "NoEntailment"
RTE1-Train1	287	143	144
RTE1-Train2	280	140	140
RTE1-Test	800	400	400

Tabla 5: Información del Corpus RTE1.

## Corpus SICK

La colección de datos SICK (Sentences Involving Compositional Knowledge [26]) es una colección enfocada a evaluar Modelos de Semántica Distribucional, este tipo de modelos busca representar el significado de sentencias utilizando representaciones composicionales de las palabras presentes en una oración. El corpus SICK se compone de 2 atributos de evaluación, por un lado ofrece un valor de similitud semántica, el cual va de 0 a 5, siendo 5 una similitud semántica equivalente y 0 una similitud semántica nula. Por otro lado se ofrece una juicio de implicación textual que consta de tres juicios de implicación, *Entailment*, *Neutral* y *Contradiction*. Para evaluar los modelos propuestos, sólo se ha empleado la parte de implicación textual.

Se ha decidido emplear esta colección de datos ya que se encuentra bien estructurada, a diferencia de las dos colecciones anteriores, las cuales contienen datos reales, la colección SICK tiene ejemplos muy concretos, con un vocabulario controlado y pares de oraciones bien empatadas. Se cree que al utilizar este corpus, se puede apreciar el comportamiento de los modelos en un entorno más controlado, es decir que se prueba con ejemplos sencillos para cualquier modelo.

La colección SICK es un corpus no balanceado que consta de 2 colecciones, una colección de entrenamiento y una de prueba. La Tabla 6 muestra la distribución de la colección.

Conjunto	Pares de oraciones	Pares de oraciones con juicio "Entailment"	Pares de oraciones con el juicio "Neutral"	Pares de oraciones con el juicio "Contradiction"
SICK-Train	4500	1299	2536	665
SICK-Test	4927	1414	2793	720

Tabla 6: Información del Corpus SICK.

A continuación se exponen los resultados alcanzados por cada una de las metodologías propuestas en cada uno de los corpus.

## 5.2 Resultados del Modelo de Conteo Estadístico

El modelo del conteo estadístico sólo consta de una ejecución por cada colección de datos. Las siguientes tablas muestran los resultados alcanzados por este modelo.

Conjunto	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	0.616	0.688	0.656	0.512	0.618
CLTE13-Test	0.488	0.424	0.416	0.504	0.458

Tabla 7: Resultados del Modelo de Conteo Estadístico con el Corpus CLTE.

Conjunto	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	0.605	0.520	0.562

Tabla 8: Resultados del Modelo de Conteo Estadístico con el Corpus RTE1.

Conjunto	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
SICK-Test	0.570	0.4844	0.910	0.738

Tabla 9: Resultados del Modelo de Conteo Estadístico con el Corpus SICK.

Como este modelo es supervisado, su desempeño depende totalmente de la características de los datos de entrenamiento, si estos se encuentran desbalanceados, afecta totalmente el proceso de clasificación de un nuevo par de sentencias, ya que tiende a asignar la categoría predominante en dicho corpus. Este comportamiento se ve claramente sobre la colección del SICK, que tiene un número grande de instancias con la categoría de neutral dentro de los datos de entrenamiento.

## 5.3 Resultados del Modelo de Similitud Semántica

Para el modelo de similitud semántica, al igual que el modelo anterior, sólo consta de una ejecución por cada colección de datos. Las tablas que siguen a continuación, muestran los resultados alcanzados por este modelo.



Conjunto	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	0.496	0.424	0.320	0.464	0.426
CLTE13-Test	0.336	0.272	0.224	0.408	0.310

Tabla 7: Resultados del Modelo de Similitud Semántica con el Corpus CLTE.

Conjunto	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	0.548	0.594	0.571

Tabla 8: Resultados del Modelo de Similitud Semántica con el Corpus RTE1.

Conjunto	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
SICK-Test	0.022	0.57	0.872	0.661

Tabla 9: Resultados del Modelo de Similitud Semántica con el Corpus SICK.

## 5.4 Resultados del Modelo de Eliminación de Tokens

El proceso de eliminación está en función de la calidad de los tokens, por esta razón es necesario utilizar diferentes mecanismos de tokenización y analizar cuál es el comportamiento de este modelo para cada uno de ellos, en particular para este modelo, se aplican 3 tipos de tokenizadores:

- **Whitespace Tokenizer:** Es el tokenizador más básico, cada token se encuentra delimitado por un espacio en blanco.
- **Trebank Tokenizer:** Ofrecido por la librería NLTK<sup>5</sup>, este tokenizador emplea expresiones regulares para generar tokens.
- **Linguistic Tokenizer:** Es un tokenizador experimental propuesto en esta investigación. Este tokenizador delimita cada token por palabras cerradas, se cree que este tipo de tokenización preserva la riqueza de los términos, ya que en la lingüística la unidad léxica más primitiva es la frase.

<sup>5</sup> <http://www.nltk.org/>



A continuación se muestra el mejor resultado por cada tokenizador y cada colección estudiada. Para la colección del CLTE se obtiene:

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.520	0.544	0.304	0.672	0.510
CLTE13-Test	Whitespace	0.312	0.328	0.160	0.648	0.362
CLTE12-Test	Treebank	0.512	0.392	0.272	0.576	0.438
CLTE13-Test	Treebank	0.216	0.184	0.168	0.672	0.310
CLTE12-Test	Linguistic	0.512	0.392	0.272	0.576	0.438
CLTE13-Test	Linguistic	0.216	0.184	0.168	0.672	0.310

**Tabla 7:** Resultados del Modelo de Similitud Semántica con el Corpus CLTE.

Para esta colección los resultados muestran que es más conveniente utilizar el tokenizador *whitespace*. Se puede observar que este modelo ofrece mejor desempeño para la colección del 2012, sin embargo para la colección del 2013, la detección del juicio de forward se vio completamente afectada, para todos los tokenizadores. Esto nos lleva a concluir que aún no ha sido posible detectar el correcto sentido de cada texto de manera automática, ya que la eliminación está totalmente relacionada con el número de tokens que comparte cada texto.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.057	0.982	0.520
RTE1-Test	Treebank	0.257	0.780	0.518
RTE1-Test	Linguistic	0.305	0.732	0.518

**Tabla 8:** Resultados del Modelo de Similitud Semántica con el Corpus RTE1.

El modelo para esta colección logró detectar el juicio de implicación no entailment, pero no logró detectar cuando dos textos sostienen una implicación textual. Para este corpus en particular el mejor tokenizador fue el *linguistic*.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
SICK-Test	Whitespace	0.929	0.6308	0.628	0.673
SICK-Test	Treebank	0.929	0.617	0.652	0.683
SICK-Test	Linguistic	0.929	0.617	0.652	0.683

**Tabla 9:** Resultados del Modelo de Similitud Semántica con el Corpus SICK.



El modelo para este corpus superó una precisión global de 67%, pero esto ocurre por casi el 93% de precisión con que es detectado el juicio de contradicción, y esto no está relacionado con el modelo, sin embargo también logra detectar el juicio de implicación neutral y entailment en más del 60%, este comportamiento no se logra con ninguno de los modelos supervisados.

## 5.5 Resultados del Modelo de Eliminación con Análisis Semántico

Para este modelo se emplean los mismo mecanismos de tokenización expuestos en la sección anterior, y al momento de realizar el análisis semántico se aplican las 8 medidas de similitud semántica expuestas anteriormente, por esta razón los resultados son agrupados con respecto a la medida de similitud empleada en el proceso del análisis semántico. En el Anexo 1 se muestran todos los resultados obtenidos por este modelo utilizando los tres tokenizadores y las diferentes medidas de similitud expuestas en 4.2.1. Es importante destacar, que nos es posible afirmar que una medida de similitud semántica se comporte mejor que el resto, ya que el desempeño del modelo ha variado en función de los corpus estudiados, en la sección 5.8 se ha reportado este modelo con la mejor precisión alcanzada por esta metodología, para cada corpus.

## 5.6 Resultados de Modelo de Inferencia Basado en Anclas

Siguiendo el mismo análisis realizado para el modelo anterior se estudió el comportamiento de éste, utilizando los tres mecanismos de tokenización reportados. En el Anexo 2 se muestran los resultados obtenidos por este modelo con cada combinación de corpus, tokenizador y medida de similitud semántica.

Es importante comentar que para este modelo el tokenizador que se use es determinante, ya que las anclas cambian totalmente, para todos los corpus el tokenizador que permitió detectar las mejores anclas fue el linguistic, este modelo para todos los corpus, supera los resultados obtenidos por los dos modelos que utilizan el principio de eliminación.

## 5.7 Resultados del Modelo de Interpretación de Oraciones Basados en Grafos

Para este modelo solo se maneja una sola configuración, las tablas que a continuación se presentan, corresponden a los resultados obtenidos.

Conjunto	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	0.336	0.168	0.136	0.728	0.342
CLTE13-Test	0.200	0.096	0.080	0.768	0.286

Tabla 10: Resultados del Modelo de Interpretación de Oraciones con el Corpus CLTE.



Conjunto	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	0.380	0.667	0.523

Tabla 11: Resultados del Modelo de Interpretación de Oraciones con el Corpus RTE1.

Conjunto	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
SICK-Test	0.929	0.625	0.619	0.666

Tabla 12: Resultados del Modelo de Similitud Semántica con el Corpus SICK.

En general para los tres corpus reportados el comportamiento ha sido similar. Debido a que la metodología es muy novedosa, quizá haga falta realizar un estudio posterior, para detectar que debilidades pudiera presentar, ya que no logra empatar de manera correcta los grafos asociados a cada uno de los textos. A continuación se realiza una comparativa entre todos los modelos desarrollados, presentando el mejor comportamiento de cada uno con respecto a cada colección de datos procesada.

## 5.8 Comparativa de Desempeño entre Modelos

Ahora que se han expuesto los resultados obtenidos, se han seleccionado las mejores ejecuciones de cada modelo para hacer una comparativa de los desempeños de cada modelo con respecto a cada colección, a continuación se muestran las tablas comparativas.



**CLTE - 2012**

<b>Modelo</b>	<b>Precisión de Bidireccional</b>	<b>Precisión de Backward</b>	<b>Precisión de Forward</b>	<b>Precisión de NoEntailment</b>	<b>Precisión Global</b>
Conteo Estadístico	0.616	0.688	0.656	0.512	0.618
Similitud Semántica	0.496	0.424	0.320	0.464	0.426
Eliminación de Tokens	0.520	0.544	0.304	0.672	0.510
Eliminación con análisis semántico	0.608	0.528	0.328	0.592	0.514
Inferencia basada en anclas	0.544	0.496	0.424	0.840	0.576
Interpretación de oraciones basada en grafos	0.336	0.168	0.136	0.728	0.342

**Tabla 13:** Comparativa de resultados para el Corpus CLTE12-Test.

**CLTE - 2013**

<b>Modelo</b>	<b>Precisión de Bidireccional</b>	<b>Precisión de Backward</b>	<b>Precisión de Forward</b>	<b>Precisión de NoEntailment</b>	<b>Precisión Global</b>
Conteo estadístico	0.488	0.424	0.416	0.504	0.458
Similitud semántica	0.336	0.272	0.224	0.408	0.310
Eliminación de tokens	0.512	0.392	0.272	0.576	0.438
Eliminación con análisis semántico	0.488	0.280	0.152	0.600	0.3800
Inferencia basada en anclas	0.304	0.408	0.240	0.736	0.422
Interpretación de oraciones basada en grafos	0.200	0.096	0.080	0.768	0.286

**Tabla 14:** Comparativa de resultados para el Corpus CLTE13-Test.



Como puede apreciarse para ambos corpus, el comportamiento ha sido similar, los mejores resultados han sido ofrecidos por el modelo de conteo estadístico, sin embargo con ninguno de los 5 modelos se ha podido superar una precisión global del 0.65. El modelo de interpretación de oraciones basado en grafos no logra superar los resultados enviados al SemEval 2012<sup>6</sup>. Con respecto al SemEval 2013<sup>7</sup> el modelo de conteo estadístico supera los resultados alcanzados por la aproximación que obtuvo mejor precisión.

Los resultados obtenidos en ambos años para este corpus muestran que todas las aproximaciones enviadas logran detectar con mayor exactitud el juicio de implicación no-entailment y bidireccional, sin embargo la detección del juicio de implicación backward y forward en general no ha logrado superar en ninguno de los corpus con respecto al resultado obtenido en las otras dos categorías. Este comportamiento se presenta tanto para los modelos supervisados, como para los no supervisados; lo que nos conduce a pensar que los modelos desarrollados requieren un preprocesamiento más detallado para lograr detectar correctamente la dirección en la implicación.

El modelo supervisado “similitud semántica”, que incluye en el vector representativo de ambas sentencias a las medidas de similitud ofrecidas por WordNet, LSA y PMI fue el que ofreció los peores resultados en ambas colecciones, lo que nos conduce a pensar que el número de características utilizadas (8) no permiten que el clasificador pueda desarrollar un buen modelo.

Para los modelos 3, 4 y 5 su ventaja fundamental radica en que son no supervisados y los resultados que ofrecen son muy similares, superando significativamente a los resultados enviados a ambas conferencias por nuestro grupo, pero no superando a los resultados obtenidos por el modelo de conteo estadístico. Un caso particular es el comportamiento del modelo basado en grafos, el cual presenta el desempeño más bajo para estas dos colecciones. Se considera que el empatamiento de los grafos es lo que ha provocado este comportamiento.

---

<sup>6</sup> <http://www.cs.york.ac.uk/semeval-2012/task8/index.php?id=results>

<sup>7</sup> <http://www.cs.york.ac.uk/semeval-2013/task8/index.php?id=results>



## RTE1

Modelo	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
Conteo estadístico	0.605	0.520	0.562
Similitud semántica	0.548	0.594	0.571
Eliminación de tokens	0.305	0.732	0.518
Eliminación con análisis semántico	0.785	0.297	0.541
Inferencia basada en anclas	0.135	0.907	0.521
Interpretación de oraciones basada en grafos	0.380	0.667	0.523

**Tabla 15:** Resultados del Modelo de Interpretación de Oraciones con el Corpus RTE1.

Para este corpus en el que sólo hay que detectar dos juicios de implicación, no se evidencia que un modelo supera al resto, ya que las precisiones globales demuestran que se descubre aproximadamente más del 50% de los tipos de juicio. Esto conduce a concluir que un modelo logra detectar el juicio de entailment y sin embargo no el de no-entailment y viceversa. A pesar de que se han utilizado diferentes mecanismos de tokenización, lograr inferir el sentido de las sentencias es una tarea complicada.

El comportamiento de la interpretación de oraciones basada en grafos mejoró para este corpus, aumentando la detección de juicio de no entailment.

Para este corpus en particular el mejor comportamiento se obtuvo con el modelo de similitud semántica. Se esperaba que para este modelo, la precisión para el juicio de implicación aumentará, sin embargo este ha disminuido. Se considera que los términos que aparecen en los textos no se encuentran todos representados en la taxonomía de WordNet, y es por ello que esos pares de sentencias alimentan al juicio de implicación no entailment.



## SICK

Modelo	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Conteo estadístico	0.570	0.4844	0.910	0.738
Similitud semántica	0.022	0.57	0.872	0.661
Eliminación de tokens	0.929	0.617	0.652	0.683
Eliminación con análisis semántico	0.929	0.758	0.584	0.684
Inferencia basada en anclas	0.929	0.712	0.599	0.679
Interpretación de oraciones basada en grafos	0.929	0.625	0.619	0.666

**Tabla 16:** Resultados del Modelo de Similitud Semántica con el Corpus SICK.

Con respecto al corpus del SICK los cuatro modelos no supervisados logran detectar más del 92% cuando existe un juicio de contradicción, esto ocurre porque para este juicio se desarrollaron un conjunto de reglas particulares, y en general ofrecen mejores resultados con respecto a la detección del juicio de entailment. El modelo de conteo estadístico y el de similitud semántica obtienen mejor precisión para el juicio de neutral, porque existen más instancias con esta categoría en los datos de entrenamiento y provocan que el modelo tiende a irse por esta categoría. Este resultado apoya el hecho de que los modelos no supervisados a pesar de que no superan al de conteo estadístico son mejores para este corpus, ya que no necesitan entrenamiento. Para este corpus en particular el modelo de grafos ofreció muy buenos resultados, clasificando correctamente a más del 60% de las instancias para cada una de las 3 categorías, este comportamiento sugiere que este modelo en particular necesita que el corpus esté correctamente construido, ya que se alimenta del árbol de dependencia sintáctica ofrecido por el parser de Stanford.



## Capítulo 6: Conclusiones y Trabajo a Futuro

Se han desarrollado seis modelos para el tratamiento automático de la implicación textual. Tres de ellos utilizan similitud semántica en el proceso de detección del juicio de implicación que comparte dos segmentos de texto.

Se desarrollaron 163,622 experimentos diferentes, variando las medidas de similitud, el tokenizador, los modelos, el tipo de corpus, los umbrales para detectar la similitud de tokens y los umbrales empleados por el proceso de eliminación, lo que nos ha permitido llegar a las siguientes conclusiones:

1. El modelo de conteo estadístico es simple de implementar, detecta el grado de similitud textual entre los pares de textos, pero no incluye ningún mecanismo para apoyar el proceso de detección del juicio de implicación, puede ser utilizado para detectar el juicio de implicación en textos escritos en diferentes idiomas.
2. Las características del corpus influyen completamente en los resultados de cada uno de los modelos, todos los experimentos se comportaron mejor para el corpus del SICK, por la forma en que ha sido confeccionado, se han incluido elementos semánticos en su construcción.
3. El modelo de similitud semántica es una buena propuesta si el vocabulario de los corpora se encuentran incluidos en la taxonomía de WordNet.
4. El comportamiento de los modelos de eliminación de tokens, está totalmente condicionado a la forma en que se obtienen los tokens dentro de cada texto.
5. El modelo basado en anclas está también directamente relacionado al proceso de tokenización, porque de este depende la selección de las anclas de cada par de textos y permite la extracción de patrones de manera automática.
6. El modelo de grafos puede utilizarse como una herramienta para extraer información.
7. Se desarrolló una base de conocimientos utilizando ConceptNet5, OpenOffice Thesaurus y WordNet. Esta base de conocimientos se representó mediante un grafo y se diseñó un algoritmo para detectar si dos términos están relacionados dentro del grafo.
8. Los modelos no supervisados propuestos en el presente trabajo emplean metodologías novedosas, que aún es necesario mejorar en cuanto a los mecanismos de empatamiento de los patrones extraídos en cada uno de ellos.

Aunque el desempeño de los modelos no-supervisados no fueron los mejores, estos presentaron un nivel de competitividad alto, ya que logran superar a varios modelos reportados en la literatura, además todos los modelos de la literatura son supervisados, sin embargo no se logró superar el modelo estadístico. Lo anterior sugiere que dichos modelos requieren una maduración, y por ello se sugieren algunos aspectos que pueden contribuir a mejorar las metodologías no supervisadas. A continuación se muestran algunos puntos para trabajar a futuro:



1. El modelo de eliminación de tokens puede ser mejorado si antes de expandir los términos se realiza algún proceso que permita desambiguar el sentido de las palabras
2. La red de conceptos generada con Conceptnet5, los Tesoros de OpenOffice y WordNet, al ser una red fuertemente conexas, esto provoca que se extraigan relaciones incorrectas. Se puede refinar el proceso de búsqueda si se introduce en él un proceso de caminos aleatorios en la etapa de construcción, de esta manera el algoritmo tendría un valor de confianza que le permitiría podar el camino de búsqueda.
3. El tokenizador lingüístico construido para los modelos 3 y 4, puede ser mejorado considerando estudios de la gramática, que permita partir las frases sin perder el sentido y considerando la anáfora.
4. El modelo de inferencia basado en anclas, se vería beneficiado si se emplearan recursos que cubrieran el dominio de la colección que se procesa, de esta manera se garantiza que las palabras, o al menos la mayoría de ellas, estaría presente en el proceso de similitud semántica. Ya que las medidas de WordNet, que es una taxonomía de carácter general, contiene términos que no aparecen en la colección, mientras que LSA y PMI pueden construirse con un corpus de dominio particular.
5. Llevar el modelo de interpretación de oraciones basada en grafos al idioma español, ya que el cambio de idioma podría aportar una perspectiva diferente que pudiera contribuir a la mejora del modelo.



## Referencias

- [1] Miquel Esplà-Gomis, Felipe Sánchez-Martínez, y Mikel L. UAlacant: *Using Online Machine Translation for Cross-Lingual Textual Entailment*. In Proceedings of the 6th International Work-shop on Semantic Evaluation (SemEval 2012). Junio 2012.
- [2] Sergio Jimenez, Claudia Becerra y Alexander Gelbukh. *Soft Cardinality + ML: Learning Adap-tive Similarity Functions for Cross-lingual Textual Entailment*. In Proceedings of the 6th Interna-tional Workshop on Semantic Evaluation (SemEval 2012). Junio 2012.
- [3] Franz J. Och y Hermann Ney. *Improved Statistical Alignment Models*. In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics. 2000.
- [4] Marco Turchi, Matteo Negri. *ALTN: Word Alignment Features for Cross-lingual Textual Entailment*. Aun no impreso. Mayo 2013.
- [5] Darnes Vilariño, David Pinto, Mireya Tovar, Saul León y Esteban Castillo. *BUAP: Lexical and Semantic Similarity for Cross-lingual Textual Entailment*. In Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012). Junio 2012.
- [6] Yashar Mehdad, Matteo Negri y José G. C. *FBK: Cross-Lingual Textual Entailment Without Translation*. In Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval2012). Junio 2012.
- [7] Katharina Wäaschle y Sascha Fendrich. *HDU: Cross-lingual Textual Entailment with SMT Features*. In Proceedings of the 6th International Workshop on Semantic Evaluation 2012. junio 2012.
- [8] Satanjeev Banerjee y Alon Lavie. *METEOR: An Automatic Metric for MT Evaluation with Im-proved Correlation with Human Judgments*. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, páginas 65–72. 2005.
- [9] Haghghi, Aria D., Andrew Y. Ng, and Christopher D. Manning. *Robust textual inference via graph matching*. Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2005.
- [10] Paziienza, Maria Teresa, Marco Pennacchiotti, and Fabio Massimo Zanzotto. *Textual entailment as syntactic graph distance: a rule based and a SVM based approach*. Proceedings of the Pattern Analysis, Statistical Modelling, and Computational Learning (PASCAL) Challenges Workshop on Recognising Textual Entailment. 2005.
- [11] SZPEKTOR, IDAN, et al. *Unsupervised acquisition of entailment relations from the Web*. Natural Language Engineering: 1-45.
- [12] Negri, Matteo, et al. *SemEval-2012 Task 8: Cross-lingual textual entailment for content synchronization*. Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. Association for Computational Linguistics, 2012.
- [13] Han, Jiawei, Micheline Kamber, and Jian Pei. *Data mining: concepts and techniques*. Morgan kaufmann, 2006.
- [14] Androutsopoulos, Ion, and Prodromos Malakasiotis. *A survey of paraphrasing and textual entailment methods*. 2009.
- [15] Vilarino, Darnes, et al. *BUAP: N-gram based Feature Evaluation for the Cross-Lingual Textual Entailment Task*. Proceedings of the 7th International Workshop on SemanticEvaluation (SemEval 2013). 2013.
- [16] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten. *The WEKA Data Mining Software: An Update*; SIGKDD Explorations, Volume 11, Issue 1. 2009.
- [17] George A. Miller. *WordNet: A Lexical Database for English*. Communications of the ACM Vol. 38, No. 11: 39-41. 1995.
- [18] Leacock, C., Miller, G. A., & Chodorow, M. *Using corpus statistics and WordNet relations for sense identification*. *Computational Linguistics*, 24(1), 147-165. 1998.
- [19] Wu, Z., Palmer, M.: *Verb semantics and lexical selection*. In: 32nd. Annual Meeting of the Association for Computational Linguistics, pp. 133 –138. New Mexico State University. 1994.
- [20] Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: IJCAI, pp. 448–453. 1995
- [21] Jiang, J.J., Conrath, D.W.: *Semantic similarity based on corpus statistics and lexical taxonomy*. 1997



- [22] Lin, D.: *An information-theoretic definition of similarity*. In: Proc. 15th International Conf. on Machine Learning, pp. 296–304. Morgan Kaufmann, San Francisco, CA. 1998.
- [23] Mihalcea, Rada, Courtney Corley, and Carlo Strapparava. *Corpus-based and knowledge-based measures of text semantic similarity*. AAAI. Vol. 6. 2006.
- [24] Susan T Dumais; George W Furnas; Thomas K Landauer; Richard. *Indexing by Latent Semantic Analysis*. Journal of the American Society for Information Science (1986-1998); Sep 1990; 41, 6; ABI/INFORM Global pg. 391.
- [25] Dagan, Ido, Oren Glickman, and Bernardo Magnini. *The PASCAL recognising textual entailment challenge*. Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising textual entailment. Springer Berlin Heidelberg, 2006.
- [26] M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi and R. Zamparelli. *A SICK cure for the evaluation of compositional distributional semantic models*. Proceedings of LREC 2014, Reykjavik (Iceland). 2014.



## Anexo 1: Estudio preliminar realizado a la colección de datos

A continuación se muestra un estudio de los corpus utilizados y las características de cada uno de ellos.

### 1.1 Colecciones de Datos

Como se menciona en la sección 3.5.1, para el desarrollo de este trabajo de investigación se cuentan con las siguientes colecciones de datos:

- SemEval-2012-task8
  - Train-12: Conjunto de entrenamiento
  - Test-12: Conjunto de prueba
- SemEval-2013-task8
  - Test-13 Conjunto de prueba

Los tres conjuntos de datos fueron obtenidos a través del foro internacional SemEval, en sus ediciones 2012 y 2013. Cada conjunto consta de 2000 pares de oraciones, con 4 categorías de implicación textual, las cuales están divididas equitativamente en 500 pares de oraciones por idioma, formando 125 pares por idioma-categoría.

### 1.2 Distribución del Corpus

Como bien se menciona en la sección 2.1, no se puede resolver el CLTE con bolsa de palabras, ya que el juicio de implicación quedaría asociado totalmente a las palabras del corpus de entrenamiento. Para evitar este comportamiento en los modelos, se ha decidido trabajar con otro tipo de medidas que no se encuentran arraigadas a la etimología de las palabras, se cree que estos valores estadísticos pueden ayudar a determinar el juicio de implicación textual. A continuación se enumeran las características propuestas:

1. Longitud de oraciones con palabras cerradas
2. Longitud de oraciones sin palabras cerradas
3. Diferencia de longitudes con palabras cerradas
4. Diferencia de longitudes sin palabras cerradas
5. Número de palabras únicas de una oración sin cerradas
6. Número de palabras comunes entre el par de oraciones
7. Porcentaje de palabras repetidas de una oración
8. Coeficiente de Jaccard entre las oraciones, con palabras cerradas
9. Coeficiente de Jaccard entre las oraciones, sin palabras cerradas

Dado que las colecciones y los tipos de combinación entre idiomas, son muy extensas, se decide tomar la colección Español-Inglés de los conjuntos de datos: Train-2012, Test-2012, Test-2013. Además se analizan los histogramas con la medida del coeficiente de Jaccard, esta medida refleja la similitud de las oraciones, en términos de palabras.

Para este estudio preliminar, se han removido las palabras cerradas de cada par de oraciones; las palabras cerradas son consideradas como términos comunes que en estudios estadísticos, no aportan ninguna ventaja, de modo que se puede prescindir de ellas. Al eliminar las palabras cerradas se aprecia mejor la distribución de los histogramas.

Primeramente, se muestran los histogramas aplicados a pares de oraciones en idiomas diferentes, es decir que para este ejemplo particular se muestran el coeficiente de Jaccard entre una oración en inglés y otra en español.

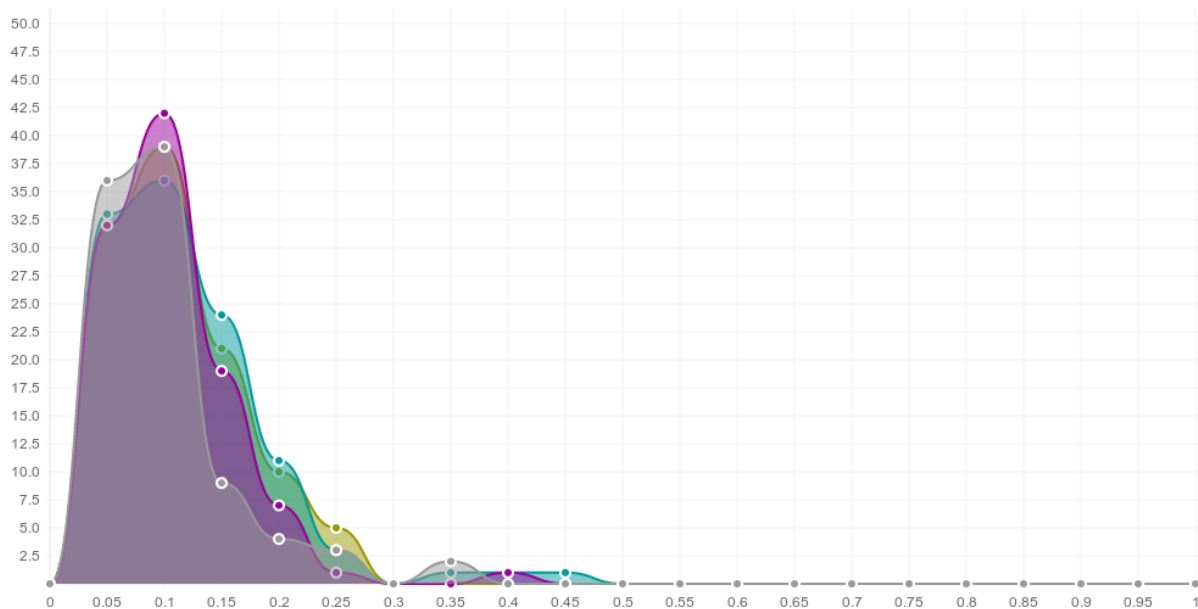


Imagen 12: Histograma de la colección Train-12, comparando 2 idiomas.

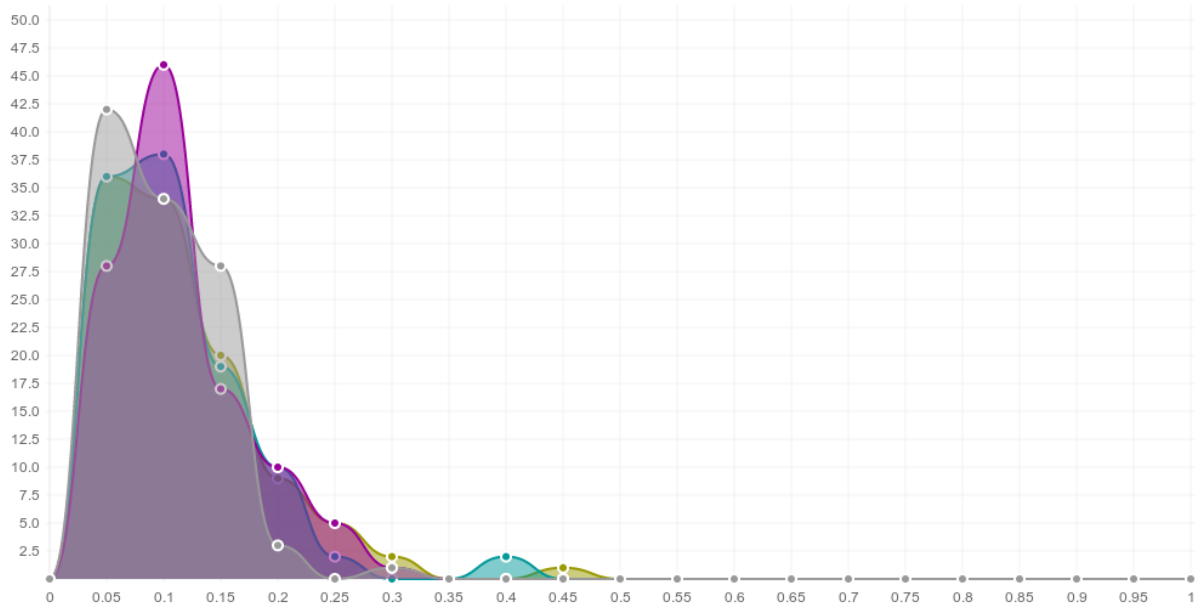


Imagen 13: Histograma de la colección Test-12, comparando 2 idiomas.

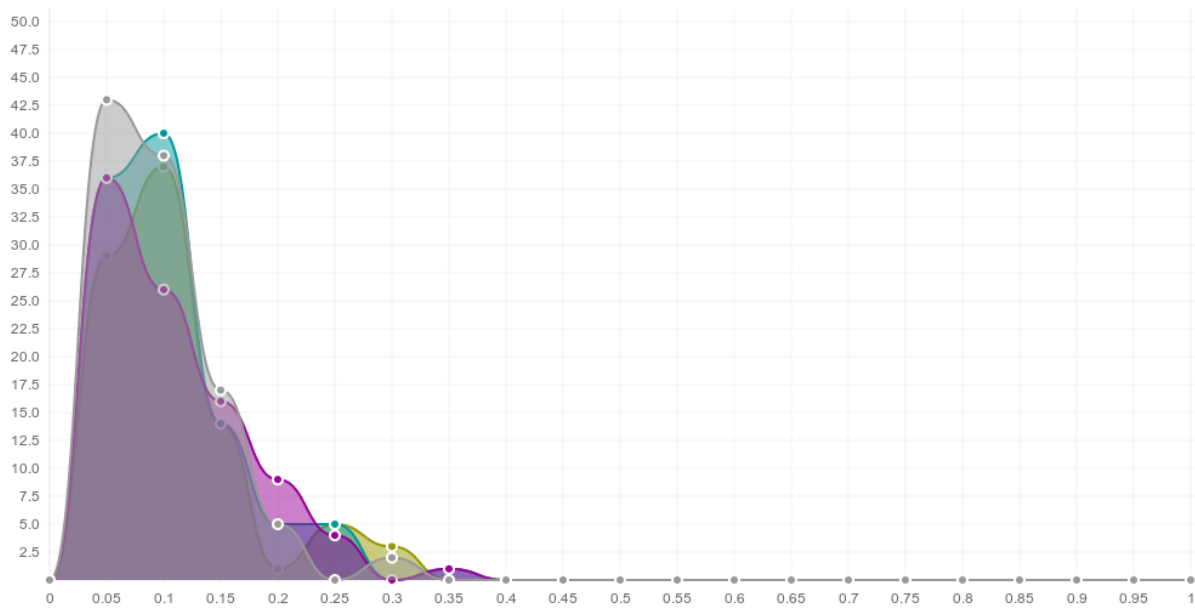
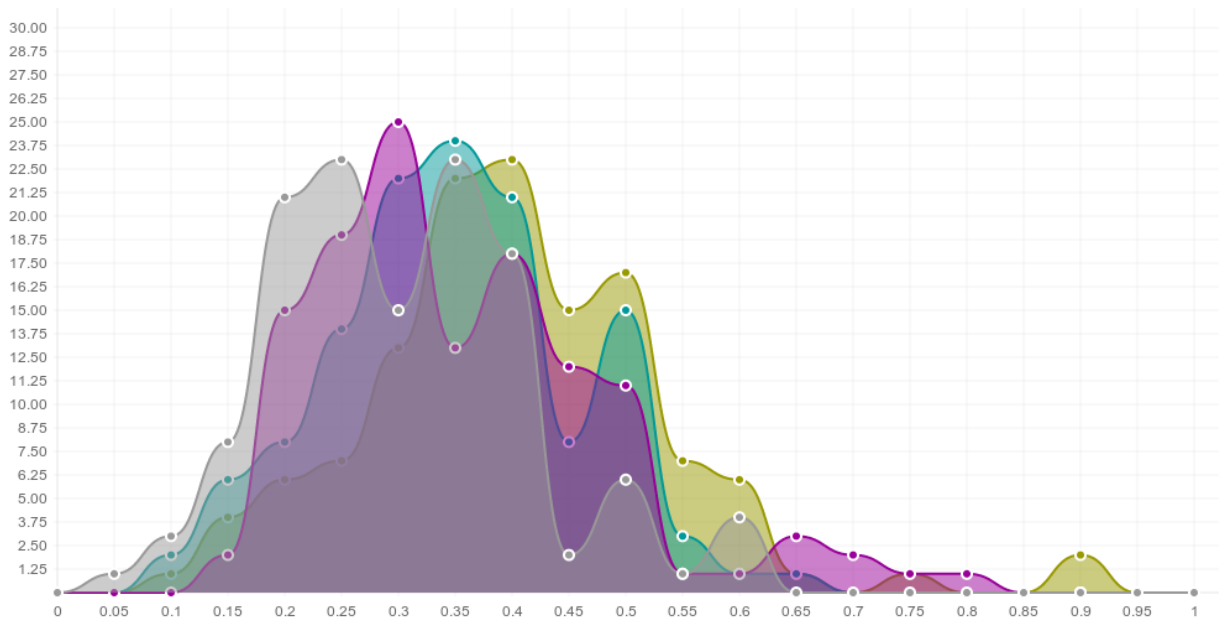


Imagen 14: Histograma de la colección Test-13, comparando 2 idiomas.

Como se puede apreciar en los histogramas anteriores, no es posible detectar una variación, todas las colecciones comparten la misma cantidad de información, esto deriva de que se analizan los pares de oraciones, es decir que la similitud de Jaccard entre una oración en inglés y otra en español, no es muy útil. Ahora bien, si se considera que ambas oraciones se encuentran en un mismo idioma (para ello es necesario traducir una oración), el histograma refleja las distribuciones que siguen cada uno de los cuatro juicios de implicación. A continuación se muestran los histogramas de las mismas colecciones, pero se ha llevado la oración de español al idioma inglés, de modo que se calcula su coeficiente de Jaccard en idioma inglés.



**Imagen 15:** Histograma de la colección Train-12, comparando 1 idiomas.

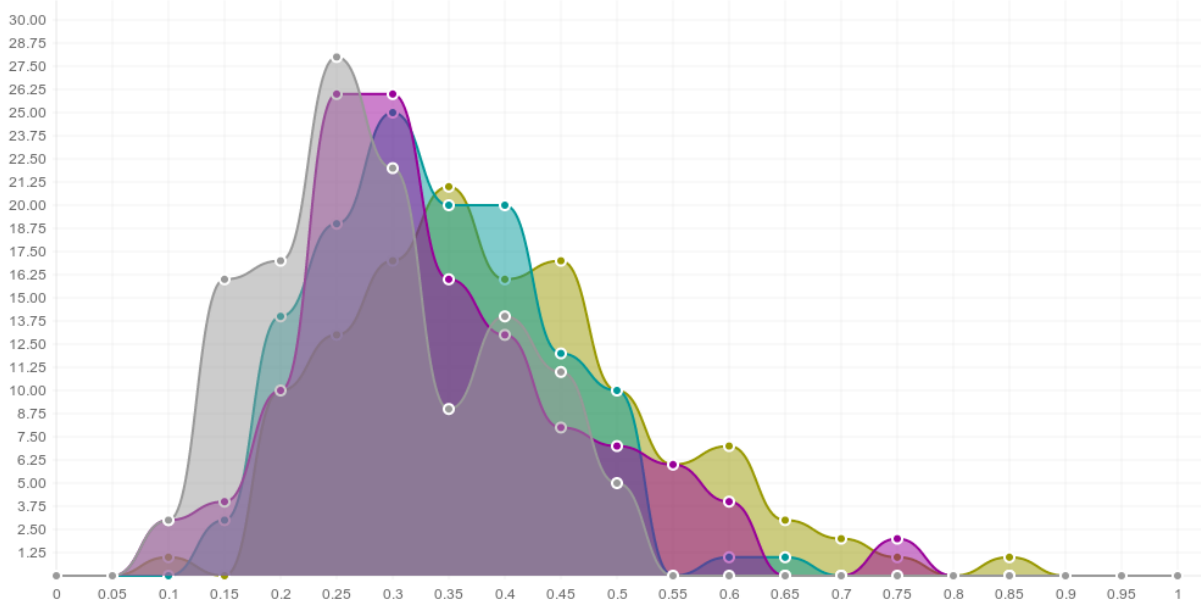


Imagen 16: Histograma de la colección Test-12, comparando 1 idiomas.

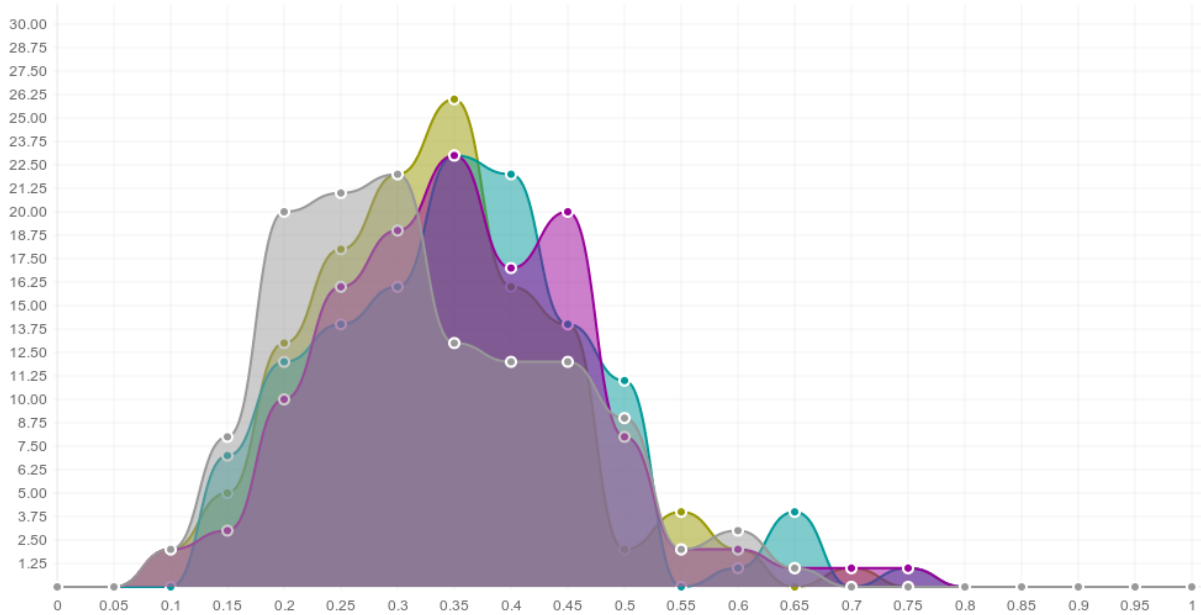


Imagen 17: Histograma de la colección Test-13, comparando 1 idiomas.



Ahora que las distribuciones de los juicios de implicación son claramente apreciables, en cada una de las colecciones de datos, notamos que las colecciones Train-12 y Test-12 son muy similares, esto quiere decir que probablemente fueron construidas a la par, mientras que la distribución de la colección Test-13 es completamente diferente. A partir de este hecho, se considera que si se generan modelos para detectar la implicación textual, que utilicen algoritmos de aprendizaje supervisado; y si se evalúan la colección Test-12 y Test-13, considerando en el entrenamiento la colección Train-12, el porcentaje de precisión de Test-12 sería superior a la del Test-13. En otro escenario, si entrenamos con la colección Test-13 y evaluamos las colecciones Test-12 y Train-12, el porcentaje de precisión sería muy bajo para ambos conjuntos de datos.

Con este estudio preliminar, se ha notado que si se realiza un estudio estadístico, **es mejor manejar las oraciones en un solo idioma, de esta manera, los modelos propuestos acogen la vertiente de pivote.** Por otro lado, se puede apreciar que las distribuciones comparten un área de solapamiento muy grande, esto quiere decir que los métodos propuestos van a confundir las clases, de modo que se puede predecir a priori, que el desempeño de los modelos propuestos para las colecciones Train-12 y Test-12 serán más elevados que el de la colección Test-13. Así mismo, notamos que los juicios No-Entailment y Bidirectional, en Train-12 y Test-12 son separables, mientras que Backward y Forward están más solapadas, esto quiere decir que se debe **emplear técnicas de composición de juicio, en lugar de utilizar los 4 juicios de implicación.**



## Anexo 2: Resultados del Modelo de Eliminación con Análisis Semántico, con diferentes medidas de similitud

### Usando Path como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.576	0.536	0.264	0.608	0.496
CLTE13-Test	Whitespace	0.192	0.280	0.152	0.800	0.356
CLTE12-Test	Treebank	0.632	0.288	0.216	0.440	0.394
CLTE13-Test	Treebank	0.616	0.144	0.152	0.328	0.310
CLTE12-Test	Linguistic	0.632	0.288	0.216	0.440	0.394
CLTE13-Test	Linguistic	0.616	0.144	0.152	0.328	0.310

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud Path.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.252	0.775	0.513
RTE1-Test	Treebank	0.762	0.282	0.522
RTE1-Test	Linguistic	0.762	0.282	0.522

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud Path.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.758	0.584	0.684
Sick-Test	Treebank	0.929	0.631	0.605	0.660
Sick-Test	Linguistic	0.929	0.631	0.605	0.660

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud Path.



### Usando Leacock-Chodorow como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.608	0.528	0.328	0.592	0.514
CLTE13-Test	Whitespace	0.320	0.312	0.176	0.640	0.362
CLTE12-Test	Treebank	0.576	0.328	0.168	0.608	0.420
CLTE13-Test	Treebank	0.296	0.152	0.128	0.672	0.312
CLTE12-Test	Linguistic	0.576	0.328	0.168	0.608	0.420
CLTE13-Test	Linguistic	0.296	0.152	0.128	0.672	0.312

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud Leacock-Chodorow.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.462	0.522	0.492
RTE1-Test	Treebank	0.645	0.402	0.523
RTE1-Test	Linguistic	0.732	0.340	0.536

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud Leacock-Chodorow.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.633	0.643	0.683
Sick-Test	Treebank	0.929	0.755	0.559	0.669
Sick-Test	Linguistic	0.929	0.755	0.559	0.669

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud Leacock-Chodorow.



### Usando Wu-Palmer como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.576	0.536	0.264	0.608	0.496
CLTE13-Test	Whitespace	0.536	0.248	0.112	0.544	0.360
CLTE12-Test	Treebank	0.616	0.264	0.144	0.568	0.398
CLTE13-Test	Treebank	0.384	0.152	0.080	0.648	0.316
CLTE12-Test	Linguistic	0.616	0.264	0.144	0.58	0.398
CLTE13-Test	Linguistic	0.384	0.152	0.080	0.648	0.316

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud Wu-Palmer.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.212	0.827	0.520
RTE1-Test	Treebank	0.405	0.650	0.527
RTE1-Test	Linguistic	0.785	0.297	0.541

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud Wu-Palmer.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.758	0.584	0.684
Sick-Test	Treebank	0.929	0.631	0.605	0.66
Sick-Test	Linguistic	0.929	0.631	0.605	0.660

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud Wu-Palmer.



### Usando Resnik como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.496	0.544	0.304	0.672	0.504
CLTE13-Test	Whitespace	0.312	0.328	0.168	0.648	0.364
CLTE12-Test	Treebank	0.520	0.296	0.256	0.536	0.402
CLTE13-Test	Treebank	0.184	0.120	0.152	0.768	0.306
CLTE12-Test	Linguistic	0.520	0.296	0.256	0.536	0.402
CLTE13-Test	Linguistic	0.184	0.120	0.152	0.768	0.306

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud Resnik.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.055	0.9825	0.518
RTE1-Test	Treebank	0.275	0.765	0.520
RTE1-Test	Linguistic	0.270	0.780	0.525

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud Resnik.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.727	0.575	0.670
Sick-Test	Treebank	0.929	0.676	0.543	0.637
Sick-Test	Linguistic	0.929	0.676	0.543	0.637

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud Resnik.



### Usando Jiang-Conrath como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.496	0.544	0.304	0.672	0.504
CLTE13-Test	Whitespace	0.312	0.328	0.168	0.648	0.364
CLTE12-Test	Treebank	0.520	0.296	0.256	0.536	0.402
CLTE13-Test	Treebank	0.184	0.120	0.152	0.768	0.306
CLTE12-Test	Linguistic	0.520	0.296	0.256	0.536	0.402
CLTE13-Test	Linguistic	0.184	0.120	0.152	0.768	0.306

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud Jiang-Conrath.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.145	0.885	0.515
RTE1-Test	Treebank	0.270	0.780	0.525
RTE1-Test	Linguistic	0.270	0.780	0.525

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud Jiang-Conrath.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.727	0.575	0.670
Sick-Test	Treebank	0.929	0.676	0.543	0.637
Sick-Test	Linguistic	0.929	0.676	0.543	0.637

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud Jiang-Conrath.



### Usando Lin como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.496	0.544	0.304	0.672	0.504
CLTE13-Test	Whitespace	0.312	0.328	0.168	0.648	0.364
CLTE12-Test	Treebank	0.520	0.296	0.256	0.536	0.402
CLTE13-Test	Treebank	0.184	0.120	0.152	0.768	0.306
CLTE12-Test	Linguistic	0.520	0.296	0.256	0.536	0.402
CLTE13-Test	Linguistic	0.184	0.120	0.152	0.768	0.306

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud Lin.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.145	0.885	0.515
RTE1-Test	Treebank	0.270	0.780	0.525
RTE1-Test	Linguistic	0.270	0.780	0.525

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud Lin.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.727	0.575	0.670
Sick-Test	Treebank	0.929	0.676	0.543	0.637
Sick-Test	Linguistic	0.929	0.676	0.543	0.637

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud Lin.



### Usando PMI como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.648	0.344	0.256	0.592	0.460
CLTE13-Test	Whitespace	0.488	0.280	0.152	0.600	0.380
CLTE12-Test	Treebank	0.664	0.264	0.184	0.544	0.414
CLTE13-Test	Treebank	0.352	0.128	0.080	0.632	0.298
CLTE12-Test	Linguistic	0.664	0.264	0.184	0.544	0.414
CLTE13-Test	Linguistic	0.352	0.128	0.080	0.632	0.298

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud PMI.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.1575	0.895	0.526
RTE1-Test	Treebank	0.250	0.812	0.531
RTE1-Test	Linguistic	0.250	0.812	0.531

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud PMI.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.715	0.598	0.680
Sick-Test	Treebank	0.929	0.659	0.566	0.646
Sick-Test	Linguistic	0.929	0.659	0.566	0.646

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud PMI.



### Usando LSA como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.616	0.488	0.296	0.624	0.506
CLTE13-Test	Whitespace	0.424	0.312	0.168	0.592	0.374
CLTE12-Test	Treebank	0.568	0.272	0.208	0.608	0.414
CLTE13-Test	Treebank	0.272	0.168	0.120	0.672	0.308
CLTE12-Test	Linguistic	0.568	0.272	0.208	0.608	0.414
CLTE13-Test	Linguistic	0.272	0.168	0.120	0.672	0.308

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus CLTE, empleando la medida de similitud LSA.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.112	0.930	0.521
RTE1-Test	Treebank	0.225	0.835	0.530
RTE1-Test	Linguistic	0.225	0.835	0.530

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus RTE1, empleando la medida de similitud LSA.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.637	0.645	0.684
Sick-Test	Treebank	0.929	0.611	0.616	0.660
Sick-Test	Linguistic	0.929	0.611	0.616	0.660

**Tabla:** Resultados del Modelo de Eliminación Semántica con el Corpus SICK, empleando la medida de similitud LSA.



## Anexo 3: Resultados del Modelo de Inferencia Basado en Anclas

### Usando Path como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.584	0.472	0.368	0.688	0.528
CLTE13-Test	Whitespace	0.216	0.392	0.216	0.776	0.400
CLTE12-Test	Treebank	0.568	0.464	0.36	0.688	0.520
CLTE13-Test	Treebank	0.272	0.376	0.240	0.720	0.402
CLTE12-Test	Linguistic	0.544	0.496	0.424	0.832	0.574
CLTE13-Test	Linguistic	0.328	0.440	0.264	0.632	0.416

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud Path.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.3375	0.692	0.515
RTE1-Test	Treebank	0.365	0.665	0.515
RTE1-Test	Linguistic	0.102	0.920	0.511

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud Path.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.667	0.603	0.669
Sick-Test	Treebank	0.929	0.692	0.577	0.661
Sick-Test	Linguistic	0.929	0.692	0.577	0.661

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud Path.



### Usando Leacock-Chodorow como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.576	0.480	0.360	0.624	0.510
CLTE13-Test	Whitespace	0.296	0.376	0.232	0.688	0.398
CLTE12-Test	Treebank	0.552	0.456	0.352	0.616	0.494
CLTE13-Test	Treebank	0.288	0.376	0.232	0.696	0.398
CLTE12-Test	Linguistic	0.504	0.480	0.424	0.864	0.568
CLTE13-Test	Linguistic	0.336	0.456	0.272	0.632	0.420

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud Leacock-Chodorow.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.3375	0.692	0.515
RTE1-Test	Treebank	0.7575	0.265	0.511
RTE1-Test	Linguistic	0.127	0.887	0.507

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud Leacock-Chodorow.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.692	0.577	0.661
Sick-Test	Treebank	0.929	0.692	0.577	0.661
Sick-Test	Linguistic	0.929	0.746	0.537	0.654

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud Leacock-Chodorow.



### Usando Wu-Palmer como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.544	0.472	0.384	0.704	0.526
CLTE13-Test	Whitespace	0.304	0.360	0.240	0.728	0.408
CLTE12-Test	Treebank	0.520	0.472	0.376	0.712	0.520
CLTE13-Test	Treebank	0.296	0.360	0.248	0.736	0.410
CLTE12-Test	Linguistic	0.544	0.496	0.424	0.840	0.576
CLTE13-Test	Linguistic	0.416	0.384	0.232	0.648	0.420

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud Wu-Palmer.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.732	0.297	0.515
RTE1-Test	Treebank	0.875	0.155	0.515
RTE1-Test	Linguistic	0.102	0.920	0.511

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud Wu-Palmer.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.666	0.611	0.673
Sick-Test	Treebank	0.929	0.666	0.611	0.673
Sick-Test	Linguistic	0.929	0.712	0.599	0.679

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud Wu-Palmer.



### Usando Resnik como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.520	0.456	0.352	0.704	0.508
CLTE13-Test	Whitespace	0.128	0.272	0.176	0.864	0.360
CLTE12-Test	Treebank	0.512	0.440	0.336	0.712	0.500
CLTE13-Test	Treebank	0.128	0.272	0.176	0.872	0.362
CLTE12-Test	Linguistic	0.576	0.568	0.312	0.760	0.554
CLTE13-Test	Linguistic	0.344	0.400	0.288	0.592	0.406

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud Resnik.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.332	0.700	0.516
RTE1-Test	Treebank	0.735	0.292	0.513
RTE1-Test	Linguistic	0.207	0.817	0.512

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud Resnik.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.664	0.611	0.673
Sick-Test	Treebank	0.929	0.664	0.611	0.673
Sick-Test	Linguistic	0.929	0.698	0.602	0.677

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud Resnik.



### Usando Jiang-Conrath como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.520	0.456	0.352	0.704	0.508
CLTE13-Test	Whitespace	0.128	0.272	0.176	0.864	0.360
CLTE12-Test	Treebank	0.512	0.440	0.336	0.712	0.500
CLTE13-Test	Treebank	0.128	0.272	0.176	0.872	0.362
CLTE12-Test	Linguistic	0.576	0.568	0.312	0.760	0.554
CLTE13-Test	Linguistic	0.344	0.400	0.288	0.592	0.406

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud Jiang-Conrath.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.347	0.682	0.515
RTE1-Test	Treebank	0.732	0.295	0.513
RTE1-Test	Linguistic	0.157	0.872	0.515

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud Jiang-Conrath.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.664	0.611	0.673
Sick-Test	Treebank	0.929	0.664	0.611	0.673
Sick-Test	Linguistic	0.929	0.698	0.602	0.677

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud Jiang-Conrath.



### Usando Lin como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.520	0.456	0.352	0.704	0.508
CLTE13-Test	Whitespace	0.128	0.272	0.176	0.864	0.360
CLTE12-Test	Treebank	0.512	0.440	0.336	0.712	0.500
CLTE13-Test	Treebank	0.128	0.272	0.176	0.872	0.362
CLTE12-Test	Linguistic	0.576	0.568	0.312	0.760	0.554
CLTE13-Test	Linguistic	0.344	0.400	0.288	0.592	0.406

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud Lin.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.140	0.897	0.518
RTE1-Test	Treebank	0.365	0.665	0.515
RTE1-Test	Linguistic	0.127	0.907	0.517

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud Lin.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.664	0.611	0.673
Sick-Test	Treebank	0.929	0.664	0.611	0.673
Sick-Test	Linguistic	0.929	0.698	0.602	0.677

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud Lin.



### Usando PMI como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.560	0.440	0.368	0.672	0.510
CLTE13-Test	Whitespace	0.232	0.296	0.224	0.752	0.376
CLTE12-Test	Treebank	0.544	0.488	0.312	0.632	0.494
CLTE13-Test	Treebank	0.224	0.264	0.232	0.800	0.3800
CLTE12-Test	Linguistic	0.608	0.568	0.376	0.744	0.574
CLTE13-Test	Linguistic	0.304	0.408	0.240	0.736	0.422

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud PMI.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.135	0.907	0.521
RTE1-Test	Treebank	0.402	0.625	0.513
RTE1-Test	Linguistic	0.142	0.882	0.512

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud PMI.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.667	0.603	0.669
Sick-Test	Treebank	0.929	0.667	0.603	0.669
Sick-Test	Linguistic	0.929	0.705	0.589	0.672

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud PMI.



### Usando LSA como medida de similitud semántica

Conjunto	Tokenizador	Precisión de Bidireccional	Precisión de Backward	Precisión de Forward	Precisión de NoEntailment	Precisión Global
CLTE12-Test	Whitespace	0.472	0.504	0.352	0.712	0.510
CLTE13-Test	Whitespace	0.320	0.400	0.192	0.648	0.390
CLTE12-Test	Treebank	0.448	0.488	0.352	0.720	0.502
CLTE13-Test	Treebank	0.312	0.400	0.208	0.656	0.394
CLTE12-Test	Linguistic	0.520	0.480	0.432	0.832	0.566
CLTE13-Test	Linguistic	0.472	0.456	0.224	0.504	0.414

**Tabla:** Resultados del Modelo de Inferencia con el Corpus CLTE, empleando la medida de similitud LSA.

Conjunto	Tokenizador	Precisión de Entailment	Precisión de NoEntailment	Precisión Global
RTE1-Test	Whitespace	0.345	0.692	0.518
RTE1-Test	Treebank	0.342	0.692	0.517
RTE1-Test	Linguistic	0.082	0.955	0.518

**Tabla:** Resultados del Modelo de Inferencia con el Corpus RTE1, empleando la medida de similitud LSA.

Conjunto	Tokenizador	Precisión de Contradiction	Precisión de Entailment	Precisión de Neutral	Precisión Global
Sick-Test	Whitespace	0.929	0.627	0.614	0.664
Sick-Test	Treebank	0.929	0.627	0.614	0.664
Sick-Test	Linguistic	0.929	0.7341	0.557	0.662

**Tabla:** Resultados del Modelo de Inferencia con el Corpus SICK, empleando la medida de similitud LSA.