Evaluación comparativa de modelos de traducción estadística y neuronal

Trabajo especial de la carrera Licenciatura en Ciencias de la Computación

Alumno: Jonathan David Mutal

Directora: Paula Estrella

5 de septiembre de 2018

Facultad de Matemática, Astronomía y Física

Universidad Nacional de Córdoba

Índice General

Índice de Figuras

Agradecimientos

A mi directora Paula Estrella, por su paciencia y predisposición. Por darme la gran oportunidad de trabajar a su lado junto con el grupo TIM-UniGe en la Universidad de traducción en Ginebra. Además por la gran capacidad de enseñar y hacerme entender, no sólo conceptos teóricos, si no también diferentes perspectivas en ciertas situaciones.

A mis compañeros de trabajo en la Universidad de traducción en Ginebra, Pierrete y Sabrina, quienes me ayudaron día a día en la estadía en Ginebra. Por su gran parte humana y sus abundantes conocimientos.

A mis padres, David y Eva, quienes con mucho esfuerzo siempre me apoyaron e incentivaron en el estudio dándome la oportunidad de hacerlo de manera exclusiva. También de apoyarme y estimularme en toda situación. A mi hermana Daniela, quien soportó cada etapa en mis años de estudiante.

A mis amigos de siempre, en especial los que siempre estuvieron conmigo en toda situación desde muy pequeño. A los de la facultad, quienes hicieron cada día en el cursado, un día único (y fuera también).

A la facultad y sus docentes, por la formación inigualable, pasión y entrega total. Gracias a ellos hoy en día soy lo que soy. A la educación pública, que me dio la oportunidad de ser hoy un profesional competente.

Resumen

Recientes investigaciones han demostrado que sistemas de traducción automática neuronales han superado, en gran medida, la calidad de sistemas estadísticos. Al mismo tiempo, grandes empresas dedicadas a brindar servicios relacionados con la traducción, tales como Systran, Google, DeepL y Microsoft, han anunciado la implementación de sistemas neuronales para uso público. La combinación de estos factores hicieron que rápidamente compradores y proveedores de servicios de traducción estén atentos a las nuevas oportunidades y analicen la posibilidad de agregar traducción automática a su flujo de trabajo. Además, la explosión en la cantidad de contenido publicado en los últimos tiempos, así como la necesidad de publicarlo rápidamente tiene como consecuencia inmediata que los actores del mercado busquen nuevas alternativas para reducir tiempo y costos.

En este trabajo evaluaremos la implementación de un sistema de traducción automática en un una empresa privada con necesidades particulares: el Correo Suizo, cuyo departamento de servicios lingüísticos está interesado en agregar traducción automática a su flujo de trabajo para diferentes tareas, y así eventualmente reducir costos y acelerar tiempos de publicación. Se analizaron distintas opciones como el desarrollo in-house (Moses Toolkit) o la contratación de un proveedor del servicio (Microsoft Translator Hub) y se realizaron distintas evaluaciones (automática y humana) para determinar cuál sería la solución más apropiada para este cliente.

Abstract

This study presents the preliminary results of an ongoing academia-industry collaboration that aims to integrate MT into the workflow of Swiss Post's Language Service. We describe the evaluations carried out to select an MT tool (commercial or open-source) and assess the suitability of machine translation for post-editing in Swiss Post's various subject areas and language pairs. The goal of this first phase is to provide recommendations with regard to the tool, language pair and most suitable domain for implementing MT.

Palabras claves: Computational Linguistics, Statistical Machine Translation, Neural Machine Translation, Industry Collaboration, Natural Lenguage Processing, Artificial intelligence.

Clasificación (ACM CCS 2012):

- **Computing methodologies∼Machine translation**
- **Computing methodologies∼Natural language processing**
- Computing methodologies∼Artificial intelligence

Capítulo 1

Introducción

1.1 Traducción automática

Muchos autores definen a la traducción automática (TA[1]) como el uso de la computadora en el proceso de traducir un lenguaje a otro (Jurafsky y Martin, 2008), otros como un campo de la lingüística computacional donde investiga métodos para traducir un texto o discurso en múltiples pares (una o más) de lenguas. Sin embargo, muchos la definen como el procesamiento de un texto origen para generar un texto objetivo, donde no sólo se basa en la traducción de dos idiomas diferentes, sino también traducir habla, lenguajes de señas, braille, textos complejos, entre otros.

A pesar de múltiples definiciones, todos coinciden que el proceso de traducción no es una tarea fácil, debido a que requiere un gran entendimiento del humano y de la comunicación. Por ende, no sólo requiere un conocimiento de las lenguas involucradas, sino también del dominio o materia; es totalmente diferente traducir textos legales que literarios.

1.2 ¿Porque es importante?

En los últimos tiempos la traducción automática creció inmensamente. Durante muchos años lenguas como Latín, Árabe, Ruso, Español, Inglés, entre otras han actuado como lenguajes de conocimientos por el que imponen creencias y formas de expresión en diferentes partes del mundo. La comunicación mudó en los últimos tiempos, dejando de lado la versión centrista para cambiar a una versión donde los actores se comunican entre sí, por lo que la traducción automática pasa a tener un

[1]A lo largo de la tesis nombraremos TA como traducción automática

rol principal, posibilitando una mejor interlocución entre los actores.

Traducir con ayuda de un sistema TA reduce tiempos; el volumen de datos incrementa de forma abrupta día a día, por lo que empieza a ser imposible para los humanos traducir gran cantidad de datos en un tiempo considerable. En los últimos treinta años se ha generado más información que en los cinco mil anteriores. En organismos multilingües, como la Unión Europea, tareas de traducción consumen alrededor de 50% en sus costes administrativos. En este contexto, avances de las nuevas tecnologías aplicadas a la traducción automática pueden contribuir a agilizar el volumen de información implicada y por ende los costos.

La globalización, definida como el proceso por el que la creciente comunicación e interdependencia entre los diferentes países del mundo unifica mercados, sociedades y culturas, lleva consigo múltiples repercusiones en el mundo donde es clara la necesidad de la TA. Además, la evolución tecnológica genera, de forma simultánea, más documentación que debe traducirse a múltiples idiomas.

Según el último reporte de TAUS (Andrew Joscelyne, 2017) (Translation Automation User Society), la organización dedicada a recomendar "mejores prácticas" en la industria de la localización y que nuclea a proveedores, usuarios y desarrolladores de tecnologías del lenguaje, la industria de la traducción cuenta con un volumen de 130 millones de dólares actualmente y se pronostican que crezca a 200 millones en los próximos 3 años. Esto muestra que la TA seguirá formando parte de los entornos productivos y se añadirán en nuevos entornos para poder procesar mayor volumen de textos, pares de lenguas, géneros textuales, etc., a la vez que la tecnología seguirá perfeccionándose y adaptándose para contribuir cada vez más a la automatización y consiguiente productividad.

1.3 Problemas usuales

Doug Arnold (Arnold, 2003) sostiene que la razón de que la traducción sea difícil en computadoras es porque también lo es para humanos. Una buena calidad en la traducción depende de varios factores, no sólo de la capacidad de traducir cada palabra, si no del contexto y del conocimiento cultural sobre el idioma. Además, múltiples frases equivalentes pueden ser traducidas de una misma oración origen, por lo que decidir cual es la correcta tiene cierta dificultad, sobre todo por la ambigüedad.

Por otra parte, hay una dificultad a causa de la percepción de cada individuo, de

ahí que es difícil determinar que es una buena traducción, es decir clara e inambigua, poética y elegante, persuasiva o todas las anteriores. Por lo tanto, cada objetivo es diferente, y así cada traductor debe entender el texto para ser capaz de aplicar reglas propias y lograr una traducción de calidad de acuerdo a lo esperado.

Además de los motivos nombrados anteriormente, hay grandes diferencias entre los idiomas[2]. No sólo diferencias léxicas si no también morfológicas, sintácticas, de estructuras argumentativas, y por los diferentes "marcos" (forma de expresar el sendero).

Morfológicamente hablando, una lengua se clasifica como *aislante* ó *sintética*, y como *aglutinante* ó *fusiónate*.

Se llama lengua *aislante* a aquellas donde las palabras tienden a ser monoformáticas (están formadas por un morfema) y presentan ninguno o muy pocos procedimientos derivativos, por lo que las palabras complejas son casi siempre el resultado de composición (unir palabras). El moderno chino estándar, las lenguas tai-kadai y las lenguas austronesias son ejemplos ilustrativos de lenguas aislantes. En contraste, una lengua es llamada *sintética*, si tiene una gran cantidad de morfemas por palabra. Entre las más conocidas están naturalmente las lenguas indoeuropeas, como el griego, el latín, el alemán, el español, el francés el italiano, el ruso, el inglés, entre otras. Se puede ver un claro ejemplo de diferencia entre estas lenguas en la Figura 1.1.

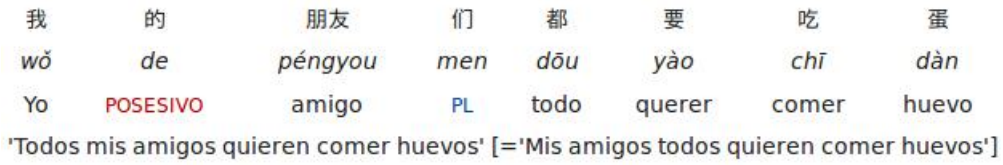

Figura 1.1: Ejemplo de diferencia entre lengua aislante, chino, y lengua sintética, español. Si comparamos el chino y el español, se observa que el español no comparte las características aislantes del chino, ya que el español emplea bastantes sufijos para marcar, por ejemplo, número/-s/ y género /-o/ en sustantivos y determinantes.

Por otro lado, a diferencia de lenguas *fusionantes*, una lengua *aglutinante* es aquella que las palabras se suelen formar uniendo monemas (es la unidad mínima de significado de la lengua) independientes. En japonés, una lengua aglutinante, el adjetivo omoshirokunakatta, traducido al español (lengua fusionante) como "No (era/fue/ha sido) interesante", se descompone en: omoshiro "interesante", kuna que

[2]Al estudio de diferencias y similitudes entre idiomas se denomina tipología.

indica negación y katta indica tiempo pasado.

Sintácticamente se puede ver que los lenguajes tienen diferentes estructuras, esto es, las lenguas difieren de acuerdo a la posición del verbo. Entre las más comunes, se puede ver lenguas *SOV* (sujeto-sustantivo-verbo), *SVO* (sujeto-verbo-sustantivo) o *VSO* (verbo-sujeto-sustantivo). Por ejemplo, para una oración en una lengua SVO, "ella come comida", la oración resultante con estructuras de lenguas SOV y VSO son "ella comida come" y "come ella comida" respectivamente.

Asimismo ocurre morfosintacticamente, que según (Nichols, 1986) se pueden diferenciar dos tipos: lenguajes por marcaje de complemento y de núcleo, explicadas en (contributors, 2018a) y (contributors, 2018b) respectivamente.

Según la forma de expresar el sendero (la dirección) o el modo de movimiento (tipo de movimiento) hay dos tipos de lenguas: de *marco verbal* ó *marco satélite*. Las lenguas de *marco verbal*, por ejemplo las romances, suelen usar verbos que indican el sendero, como entrar, salir, subir, bajar y cruzar. De lo contrario las lenguas de *marco satélite*, por ejemplo las germánicas, relegan la función del sendero a preposiciones como up, into y out of.

Estas diferencias entre idiomas se denomina divergencia de traducción. La computadora todavía no tiene la capacidad para lidiar con gran parte de ellas o es muy costoso computacionalmente, en consecuencia esta es uno de los principales motivos por la cual la TA es una tarea bastante compleja.

1.4 Historia

La primera aparición de la traducción automática fue mediados del año 1930, por George Artsrouni, donde hojas perforadas eran utilizadas para diccionarios bilingües. Luego Peter Troyanskii, nacido en Rusia, incluyó un método para lidiar con algunas diferencias gramaticales entre las lenguas. El método fue desconocido hasta mediados del año 1950.

Después de la primera computadora electrónica, sistemas de traducción automática en computadoras empezaron a desarrollarse en el año 1949 por Warren Weaver, como medio para descifrar códigos alemanes en la segunda guerra mundial. Evidentemente, este problema podía ser tratado como uno de traducción. Weaver publicó sus investigaciones unos años más tarde, lo que comenzó a investigarse en las universidades.

El 7 de enero de 1954, IBM hizo la primera demostración pública sobre sistemas de traducción automática con apenas 250 palabras, donde las oraciones del ruso fueron elegidas cuidadosamente para traducirlas al inglés. Esto estimulo el movimiento de inversiones hacía la TA.

La mayoría de los sistemas utilizaban un gran diccionario bilingüe y reglas escritas a mano para arreglar el orden las palabras en la salida del sistema (denominada traducción directa 2.5.1). Sistemas ya eran utilizados por fuerzas aéreas y por algunas asociaciones de Europa.

La gran actividad sobre traducción automática se vio perjudicada en el año 1966, cuando el ALPAC (Automatic Language Processing Advisory Committee) realizó un estudio sobre la traducción. La investigación mostró que la post-edición (PE) ayudada con sistemas era más costoso que una traducción totalmente humana, no sólo por la tecnología disponible, si no por la escasa literatura sobre el tema (la gran mayoría estaba en Ruso). Además, se pudo determinar que alrededor de 20 millones de dolares eran gastados en traducción anualmente en Estados Unidos (poco para un país). Por los motivos ya mencionados, el comité sugirió que no había ningún tipo de ventajas en utilizar TA. Consecuentemente, la mayoría de las inversiones iban al desarrollo en computación lingüística dejando así la traducción automática parada por décadas.

Systran fue fundada en 1968, a pesar de la gran negatividad. Sus sistemas de ruso a inglés fueron utilizados por la fuerza aérea de EEUU en 1970 y de francés a inglés por la comisión europea, el ALPAC. Luego sistemas para pares lenguajes europeos fueron desarrollados.

Debido a la aparición de las computadoras de escritorio durante la década de los 90, muchos sistemas de ayuda para la traducción humana fueron desarrollados (Trados por ejemplo). También hubo un gran esfuerzo en desarrollar sistemas por interlingua (explicado en 2.5.3). Por otro lado, en Japón habían comenzado a investigar sistemas basado en ejemplos (explicado en 2.5.4). En ese momento, la TA empieza a tomar un nuevo rumbo, comienza a dejar de usar numerosas reglas lingüísticas y se centra en usar memorias de traducción (traducciones pasadas) para así entrenar los modelos.

Después de un largo tiempo de investigación sobre la traducción estadística, IBM desarrolla diferentes modelos alrededor del 2000, lo que comienza a hacer viable este

tipo de sistemas. Fue el nuevo estado del arte hasta que surgieron nuevos modelos basados en aprendizaje profundo para la traducción automática, el cual mejoró la calidad de la traducción en algunos pares de idiomas.

1.5 ¿Quienes la usan?

La traducción automática es utilizada tanto por individuos, para eliminar la brecha entre idiomas, y empresas, para mantener o incrementar las ganancias y/o abrir nuevos mercados.

Según el TAUS hay cinco actores principales (referido a las empresas). Los que usan su propia tecnología para desarrollar nuevos sistemas de traducción, y así vender la licencia o el software como un servicio (SaaS).

Los que se enfocan en la preparación de datos para el entrenamiento y configuración de la TA para poder crear servicios personalizables y específicos para cada cliente en particular.

Muchos usan la TA internamente para diferenciarse y hacer más eficiente el servicio de traducción (LSPs[3]), donde, por lo general, utilizan sistemas de código abierto.

Los que agregan un valor agregado a sus negocios usando modelos ya creados o un servicio agregado a una aplicación, es decir, crean plataformas para ayudar al flujo de trabajo en la traducción.

Por último, los que ofrecen un servicio gratuito, donde a través de los visitantes recolectan datos de múltiples orígenes, y así incrementar sus ganancias de otros modos.

1.6 Nuestro caso de estudio

En este trabajo se estudia el caso de una empresa que requiere proveer distintos servicios a sus clientes, tanto externos como internos, a través de la implementación de un sistema de TA propio.[4]

En la actualidad, muchas empresas eligen incorporar traducción automática por diferentes razones: respuesta a un cliente en particular, una iniciativa para aumentar el valor de la empresa con un servicio nuevo, o bien para recortar costos y tiempos.

[3]Provedores de servicio de lenguaje.

[4]Por los acuerdos de confidencialidad firmados entre las partes y el alumno, no es posible mostrar ejemplos del corpus utilizado.

La tecnología de TA puede ser desarrollada y brindada por un tercero utilizando un servicio (SaaS) o puede ser desarrollada en dicha empresa. Cada solución tiene sus pros y contras.

El servicio del correo suizo esta interesado en integrar traducción automática en su flujo de trabajo en diferentes entornos, tanto para entender el contexto de un texto así como para el uso profesional en post-edición, entre otros, y así reducir costes y tiempos. Así, en un trabajo conjunto con la Universidad de Ginebra (grupo TIM-UniGe) indagamos sobre la posibilidad de elegir, configurar e implementar un sistema TA al flujo de trabajo. Este estudio preliminar se encarga de 1) seleccionar una herramienta TA (código abierto o comercial) y 2) determinar cuales pares de lenguajes y dominios serán los adecuados para emplearlos en el servicio. En particular, nos enfocamos en crear un sistema TA que sirva para la post-edición profesional de oraciones en el servicio.

Los datos de entrenamiento y evaluación del lenguaje origen en los diferentes pares de lengua son casi paralelos, es decir que comparten gran parte de las oraciones, haciendo posible comparar los resultados entre los diferentes dominios.

Además, cuando comenzamos a diseñar la parte experimental del estudio elegimos enfocarnos en los usuarios, profesionales traductores del correo suizo, proveyéndoles un entrenamiento especifico antes de involucrarlos en el proceso de evaluación. A la hora de reorganizar el flujo de trabajo tradicional, es importante dar un rol activo a los usuarios para promover la aceptación y evitar prejuicios durante la evaluación.

Se desarrollan experimentos para evaluar la factibilidad de tal solución usando dos herramientas open source: Moses (Koehn y cols., 2007) y openNMT (Klein, Kim, Deng, Senellart, y Rush, 2017), y un software privativo: Microsoft translator hub[5] (MTH).

1.7 Esquema de la tesis

Concluida la sección introductoria del *capítulo 1*, a continuación se detalla la estructura del contenido siguiente. El *capitulo 2* describe la traducción automática en sí, definiendo que es y que tipos hay. Daremos un pequeño recorrido por las diferentes arquitecturas que se fueron usando a lo largo de los años.

Los trabajos relacionados serán mencionados en el *capítulo 3*. En el *capítulo*

[5]https://www.microsoft.com/en-us/translator/hub.aspx.

4 describe la parte experimental, es decir el conjunto de datos, los diferentes pre procesamiento y modelos utilizados, y por último las evaluaciones y metodologías.

Finalmente, la tesis concluye en el *capítulo 5*, donde se concluirá con observaciones y detalles de trabajo futuro.

Capítulo 2

Traducción Automática

Traducción automática es un área de tecnología de lenguajes. Es un campo donde automáticamente se traduce un texto desde su origen hacía su objetivo. En esta sección, explicaremos la base de traducción automática y discutiremos diferentes tipos de sistemas de acuerdo a pares de idiomas, intervención humana, tarea a realizar y arquitecturas. Además, daremos un repaso acerca de los métodos más comunes de evaluación.

2.1 Diseños de sistemas

A la hora de diseñar un sistema, se decide principalmente su aplicación, por lo que determina las características de este. Por ende, los sistemas pueden clasificarse por diferentes aspectos (no mutuamente excluyentes) según:

- Pares de idiomas y direcciones tratadas.
- Intervención humana en el proceso.
- Tipo de tarea a realizar.
- Estrategias de traducción implementada/arquitectura.

A continuación se explicará cada ítem detalladamente.

2.2 Pares de idiomas y direcciones

Una dirección es **de** que lenguaje se traduce, llamado "lenguaje origen", **a cual** lenguaje, denominado "lenguaje objetivo". Los sistemas se pueden clasificar por el

número de idiomas, es decir si son bilingües o multilingües, y por la dirección de las traducciones. Ejemplos como la web necesitan traducciones multilingües, de una lengua a varias; organizaciones dentro de la Unión Europea, por lo general, requieren sistemas multilingües con direcciones múltiples entre varios idiomas.

2.3 Intervención humana

El proceso de traducir un texto es realizado con **intervención humana** o **sin** (totalmente automático). Dentro de la intervención humana hay múltiples categorías: **tradicional**, donde sólo la interpretación humana es utilizada en la traducción; **añadido humano** (HAMT), luego que un sistema de TA traduzca, humanos intervienen para desambiguar sentidos, elegir vocabulario, añadir palabras desconocidas, etc.; **añadido de la máquina en traducción humana** (MAHT), traducción humana con ayuda de una herramienta, usualmente llamada CAT[1], donde maneja terminología, memorias de traducción, chequeadores de ortografía y gramática, entre otros.

Además, la traducción automática se puede realizar de una manera **interactiva**, es decir, el usuario es involucrado durante el proceso de traducción para así tomar algunas decisiones, resolver ambigüedades, agregar vocabulario, proponer nuevas traducciones y demás, o **fuera del proceso**, con un trabajo de pre edición y/o post edición.

2.4 Tipo de tarea

Diferentes niveles de calidad son necesarios a la hora de implementar un sistema de TA, dependiendo el tipo de tarea. De acuerdo a la calidad deseada, la traducción automática se puede dividir principalmente en dos: **asimilación** y **diseminación**. La primera consiste en, a través del sistema, entender una idea general del texto o palabras claves para evaluar pertinencia de un documento en la búsqueda. Por lo general, no requieren una alta calidad de traducción. Por el contrario, diseminación apunta a publicar un texto sin editar o usarlo en alguna etapa del flujo del trabajo sin una modificación significativa.

[1] Computer-aided translation.

2.5 Arquitecturas existentes

En la siguiente sección se introducirá las principales arquitecturas. A lo largo de los últimos años, muchas arquitecturas han sido desarrolladas desde obsoletas como traducción directa hasta más modernas con redes neuronales y aprendizaje profundo. A pesar de que algunas arquitecturas están en desuso, es necesario explicarlas para comprender las arquitecturas modernas.

La traducción estadística revolucionó la TA en su momento, de modo que muchas arquitecturas quedaron obsoletas, entre otras: **traducción directa**, donde cada palabra del lenguaje origen es traducida al lenguaje objetivo sin importar el orden; **traducción por transferencia**, cada frase del lenguaje origen es reorganizada a la estructura del lenguaje objetivo (sintácticamente y gramaticalmente) a través de un conjunto de reglas; por **interlingua**, el lenguaje de origen es analizado y representado de alguna forma abstracta para luego ser traducido. En las siguientes secciones explicaremos muchas de las arquitecturas ya mencionadas y dos utilizadas en la actualidad: **estadística** (Brown y cols., 1990) y **neuronal** (Kalchbrenner y Blunsom, 2013; Sutskever, Vinyals, y Le, 2014; Cho, van Merrienboer, Bahdanau, y Bengio, 2014). Actualmente, muchos sistemas combinan múltiples arquitecturas.

Una forma de visualizar estas diferentes arquitecturas es a través del **triángulo de Vauquois** (Vauquois, 1968), donde muestra el incremento en la profundidad del análisis a medida que aumenta la complejidad de la arquitectura (ver Figura 2.1).

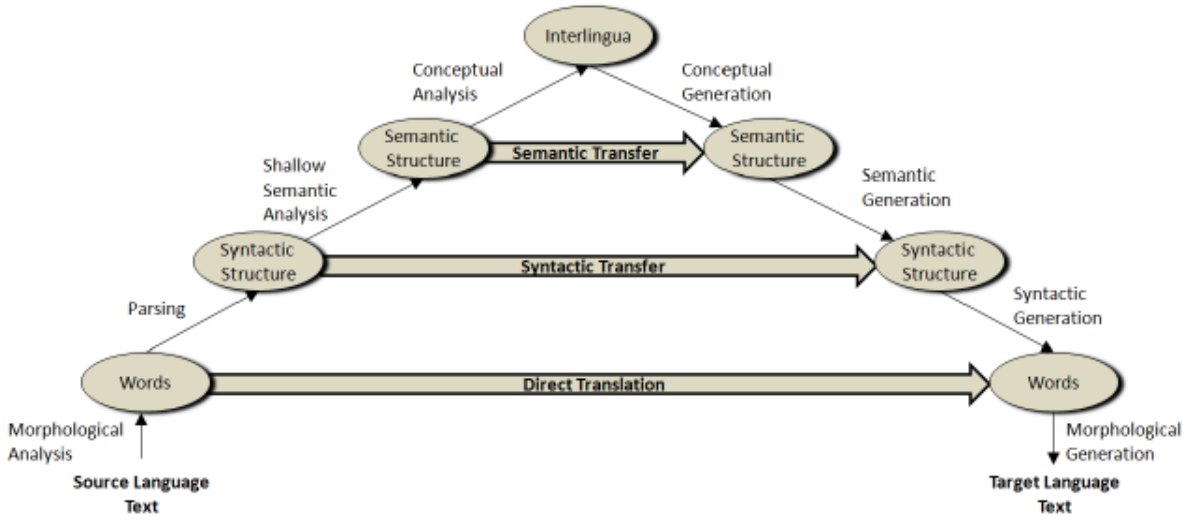

Figura 2.1: Triángulo de Vaquois.

Según este triangulo, las diferentes metodologías básicamente se categorizan en

traducción directa, por transferencia e interlingua. Estas se diferencian en la profundidad del análisis del lenguaje origen y el nivel de independencia lingüística en relación a la representación del significado. Por ejemplo, interlignua involucra el mayor nivel de análisis del lenguaje origen, en contraste con la traducción directa, que simplemente traduce palabras. En la figura 2.1 podemos ver que ilustra estos diferentes niveles.

Empezando en la base de la pirámide, el análisis es poco profundo, la traducción sólo se realiza analizando palabras. Previo a la cima se encuentra la traducción por transferencia (sintáctica y semántica), donde modifica la estructura del lenguaje origen al objetivo, que lleva consigo un mayor nivel de análisis en los textos y dependencia lingüística. Finalmente, en lo más alto se encuentra la traducción por interlingua, el cual representa el significado de ambos lenguajes con uno intermedio denominado interlingua, siendo así totalmente independiente de estructuras lingüísticas.

A medida que se acerca a la cima del triangulo, el trabajo para eliminar las diferencias entre diferentes lenguas se reduce (menor cambio de estructuras, ordenamiento de palabras, u otros), pero el costo de análisis y generación del lenguaje aumenta considerablemente. Así, por ejemplo, en la traducción directa luego de la traducción de palabras es necesario múltiples permutaciones para así lograr una buena calidad en la traducción, es decir hay muy poco análisis del texto de origen pero la diferencia de estructuras entre los lenguajes es mayor. Mientras que en interlingua, hay una representación semántica del lenguaje origen que abstrae totalmente las estructuras gramaticales de un lenguaje, por lo tanto hay un gran análisis del lenguaje origen, pero menor diferencia entre las lenguas.

2.5.1 Traducción directa

La **traducción directa** consiste en procesar palabra por palabra del texto origen al objetivo con mínimas modificaciones. No usa estructuras intermediaras, excepto por un superficial análisis *morfológico*; cada palabra origen se traduce directamente a su palabra objetivo. Se basa en un gran diccionario bilingüe para traducir, y luego aplica pequeñas reglas de ordenamiento, así como mover los adjetivos después de los sustantivos.

En la figura 2.2 muestra como es el flujo de la traducción directa, es decir es una

traducción incremental con diferentes etapas.

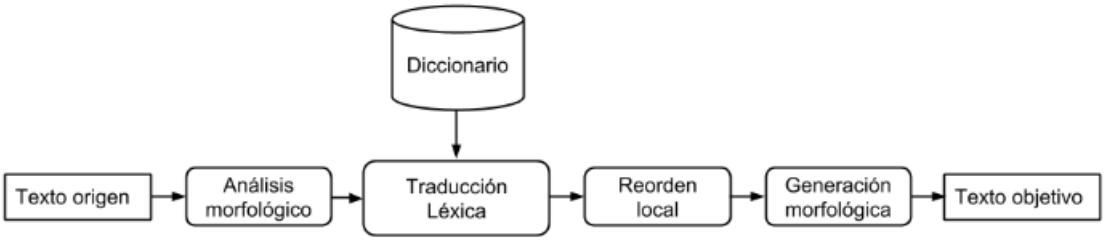

Figura 2.2: Traducción directa.

Se puede ver un claro ejemplo de traducción directa en la Figura 2.3. En el análisis morfológico se observa que el verbo *didn't* se transforma en su raíz junto con su tiempo verbal. Luego con el diccionario bilingüe se traducen todas las palabras, suponemos que *slap* se traduce como *dar una bofetada a*. Para realizar un orden local por lo general se utiliza diccionarios con reglas.

Entrada	Mary **didn't** slap the green witch
Análisis morfológico	**Mary DO**-pasado **not slap the green witch**
Traducción léxica	Maria PASADO no dar una bofetada a la **verde bruja**
Reorden local	Maria **PASADO** no dar una bofetada a la bruja verde
Generación morfológica	Maria no dió una bofetada a la bruja verde

Figura 2.3: Ejemplo de traducción directa. Cada etapa modifica las palabras que están en *"negrita"*.

2.5.2 Traducción por transferencia

Una estrategia para vencer las diferencias entre lenguajes es modificar la estructura del texto origen para que cumpla con las propiedades del objetivo a través de reglas lingüísticas.

Este modelo, por lo general, consiste en tres fases: *análisis* para determinar la estructura gramatical del texto origen, *transferencia* de la estructura origen a la objetivo y *generación*. Hay varios *análisis* que se pueden realizar: *morfológico* donde cada palabra es etiquetada según su propiedad morfológica (verbo, sujeto, etc); *sintáctico*, cada parte del texto es etiquetada de acuerdo a su posición en la estructura; o ambos, para esto se utiliza un analizador léxico y/o sintáctico.

La etapa de *transferencia* trata de eliminar la brecha lingüística entre los idiomas. Entre las diferentes transferencias, existen léxicas que se realiza con un diccionario, sintácticas a través de diferentes reglas (ver Figura 2.4), y semánticas.

$$\text{NP} \rightarrow \text{Adjective}_1\ \text{Noun}_2 \qquad \Rightarrow \qquad \text{NP} \rightarrow \text{Noun}_2\ \text{Adjective}_1$$

Figura 2.4: Regla sintáctica.

Una vez aplicada la etapa de transferencia, se *genera* (etapa de generación) el lenguaje objetivo con algunas reordenaciones. Podemos ver el siguiente ejemplo del inglés al español (ver 2.5).

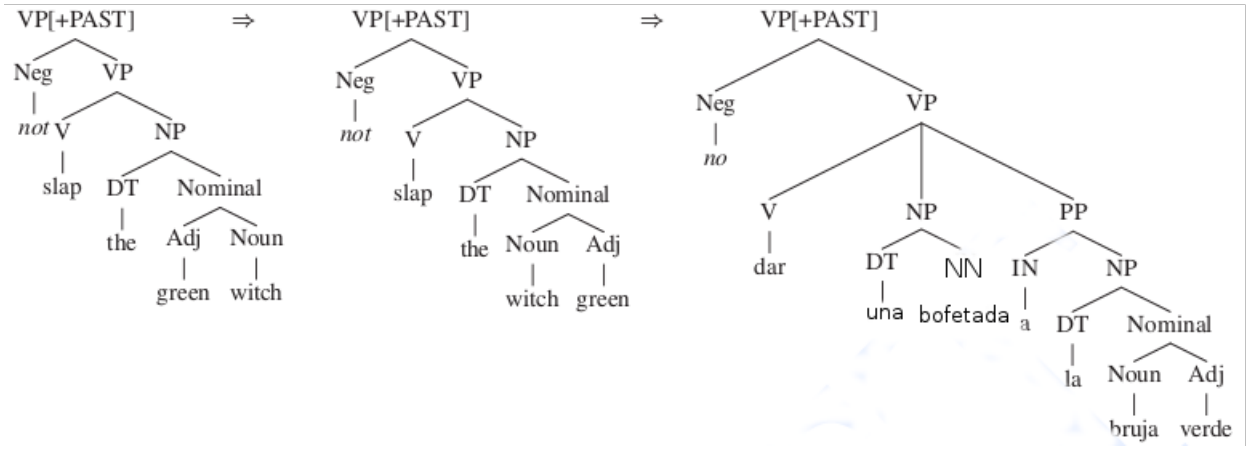

Figura 2.5: Ejemplo transferencia.

Las ventajas de este modelo es que permite un análisis más profundo, por lo que soluciona orden de palabras. Pero es muy costoso en el caso de múltiples pares de lenguas, sin contar que las reglas tienen que ser escritas a mano por una persona experta en el dominio.

2.5.3 Interlingua

Traducción automática por interlingua (Dorr, Hovy, y Levin, 2005) es una de las arquitecturas clásicas en traducción. La idea principal es representar las oraciones, tanto origen como objetivo, de forma que oraciones parecidas estén similarmente representadas en un lenguaje intermedio (denominado *interlingua*).

La traducción de este modelo consiste básicamente en analizar *semánticamente* el texto origen para representarlo en *interlingua*, y así *generar* la traducción del texto objetivo. Para ver las etapas gráficamente ver Figura 2.6.

Según (Dorr y cols., 2005) interlingua puede definirse con tres componentes: un

Figura 2.6: Interlingua: etapas básicas.

conjunto de *símbolos* donde cada uno de ellos representa un cierto significado, muchas veces son definidas como ontologías; una *notación*, el cual son la base para que los símbolos individuales tengan significados complejos, por ejemplo si las notaciones son proposiciones entonces los símbolos se emplean para su formulación; *lexicones*, que nombra una colección de palabras en alguna lengua por lo que cada una de ellas están asociadas directamente o indirectamente a un símbolo.

2.5.4 Basada en ejemplos (EBMT)

Debido a que es muy difícil encontrar arquitecturas puramente **basado en ejemplos** (muchas de ellas son el conjunto de varias) existe una dificultad en armar una definición concreta sobre esta. Sin embargo, la mayoría de la literatura coincide que básicamente consiste en usar segmentos del lenguaje origen para extraer ejemplos del lenguaje objetivo con similar significado (Hutchins, 2005), donde usualmente están almacenados en un gran base de datos. Sin embargo, esta definición superpone con arquitecturas basada en reglas con ejemplos, por lo que Somers (Somers, 1999) añade que esos ejemplos eran usados en tiempo de ejecución.

Originalmente se llamaba **traducción por analogía** (Nagao, 1984), que proponía usar diccionarios de palabras y *tesauros*[2] para simular el cerebro humano a la hora de realizar una traducción. El proceso consiste en varias etapas según (Nagao, 1984): *analizar frases a traducir* reduciendo a oraciones que son relevantes; *partir la entradas en fragmentos*, tanto caracteres como secuencias; *encontrar partes de la frase equivalentes* parecidos en el diccionario almacenado, para ello se puede utilizar el tesauro encontrando similitudes entre las palabras; si la frase completa no aparece en el diccionario, se toman partes y luego se *reordena* de acuerdo a la estructura del lenguaje objetivo.

En comparación con las arquitecturas ya mencionadas, esta es fácil de mejorar debido a que sólo implica agregar más corpus. Es bastante simple, ya que no es necesario crear reglas muy complejas, ni tampoco depende fuertemente de teorías

[2]Es un sistema que agrupa palabras con significado o naturaleza similar.

lingüísticas (como en el caso de basado en reglas). Además, en algunos casos los resultados son más legibles debido a que se basa plenamente en traducciones humanas anteriores. Sin embargo, es necesario un corpus de ejemplos y la mayoría de veces son complicados de encontrar (obviamente depende de la materia). Como es sabido, a medida que aumenta la longitud de la frase, la probabilidad de tener una traducción exacta es menor, ya que es difícil establecer una alineación entre las oraciones.

2.5.5 Modelos estadísticos

Hay múltiples **modelos estadísticos**, por lo que se explicará los más básicos para luego adentrarnos en los modelos más complejos. A pesar que estos modelos no constituyen más "el estado del arte", muchas ideas son utilizadas en modelos con aprendizaje profundo.

Como su nombre lo indica estos se basan en el uso de la estadística para la traducción. Para el entrenamiento todos ellos necesitan un *corpus paralelo*, es decir dos textos alineados en diferentes lenguajes. En la Figura 2.7 se puede visualizar oraciones extraídas de un corpus paralelo del alemán al inglés.

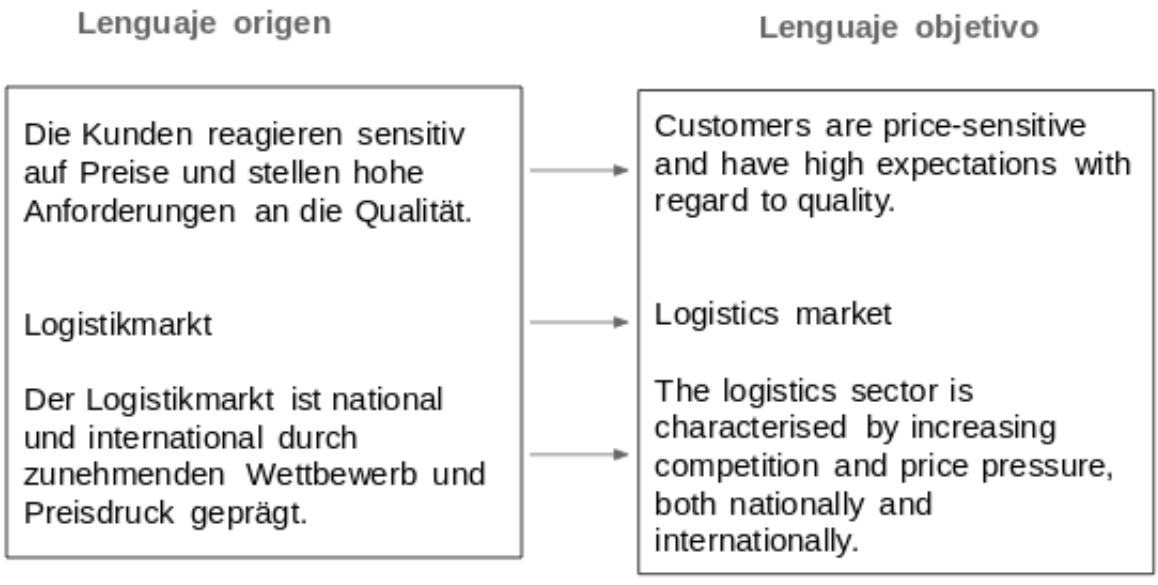

Figura 2.7: Ejemplo de oraciones en un corpus paralelo del alemán al inglés. Cada oración del alemán está alineada con su correspondiente traducción al inglés. Texto extraído del reporte anual del correo suizo (es público).

Estadísticos por palabra

El primer modelo en desarrollarse fue **basado en palabras** (Brown y cols., 1988), donde cada palabra es traducida de acuerdo a la distribución en el corpus. Por ejemplo, la palabra del alemán *Haus* se traduce a *casa* en un corpus donde la probabilidad de traducir a *casa* es mayor. Es claro que esta traducción es puramente léxica sin importar el contexto.

Agregando un poco de formalidad esta distribución será denominada léxica, definida como:

$$p_f : e \to p_f(e)$$

donde e es una palabra del lenguaje objetivo y p_f es la distribución de la palabra del lenguaje origen f. Es decir, dada una palabra del lenguaje origen f, $p_f(e)$ es la probabilidad de que f sea traducida como e. Esta función debería retornar una alta probabilidad si la palabra e es un buen candidato para f y 0 en caso contrario.

Esta primera aproximación no es del todo útil para todo par de idioma. Veamos el siguiente ejemplo:

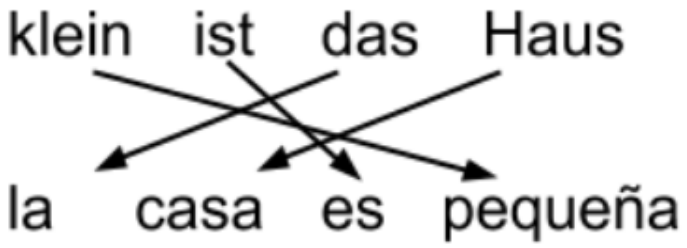

Figura 2.8: Ejemplo de alineación. $a = \{1 \to 3, 2 \to 4, 3 \to 2, 4 \to 1\}$.

Claramente en este par de idiomas habrá un problema de alineación entre palabras, por lo tanto es importante definir un modelo probabilístico para alinearlas $d(j|i, l_e, l_f)$, donde d es la probabilidad de que dada la palabra en la posición i del lenguaje origen predice la posición j del lenguaje objetivo, con l_e y l_f número de palabras en lenguaje objetivo e origen respectivamente. También, este tiene en cuenta las diferentes clase de palabras así como las diferentes vacantes en la traducción.

Además, es importante notar que una palabra origen se puede traducir como múltiples palabras objetivo (denominado *fertilidad* de una palabra) o que se agreguen nuevas palabras en la traducción. La *fertilidad* es modelada con una distribución probabilística $n(\phi|f)$ donde dada una palabra de origen f índica cuantas palabras

$\phi \in \{1, 2, 3, ...\}$ se traduce. Asimismo, modela las palabras que no tienen correspondencia con el objetivo con $\phi = 0$ y utiliza un token, usualmente *NULL*, para insertar nuevas palabras en la traducción (ver 2.9). Utilizaremos la notación ϕ_i con $i = 1, 2, ...$ para denotar la fertilidad de la palabra f_i y ϕ_0 la del token *NULL*.

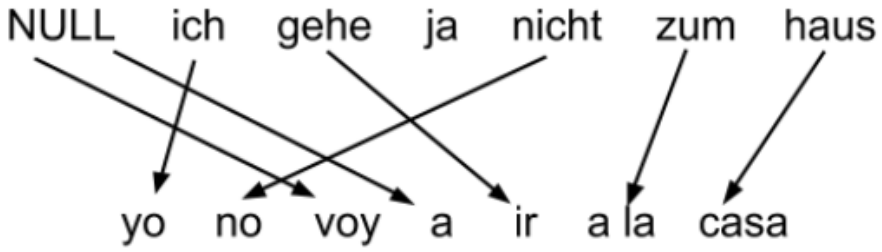

Figura 2.9: Ejemplo de fertilidad. $n(2|zum)$ es bastante alta.

Sea $\mathbf{e} = (e_1, .., e_{l_f})$ y $\mathbf{f} = (f_1, .., f_{l_f})$ oración objetivo y origen respectivamente, a una función de posiciones que representan la alineación entre las lenguas (ver ejemplo en Figura 2.8), $t(e|f)$ la probabilidad condicional de traducir la palabra f en e (denominada *probabilidad de traducción*) definiremos

$$\begin{aligned} p(\mathbf{e}|\mathbf{f}) &= \sum_a p(\mathbf{e}, a|\mathbf{f}) \\ &= \sum_{a(1)=0}^{l_f} ... \sum_{a(l_e)=0}^{l_f} \prod_{j=1}^{l_e} p(\phi_0) \prod_{i=1}^{l_f} \phi_i! n(\phi_i|e_i) * \prod_{j=1}^{l_e} t(e_j|f_{a(j)}) d(j|a(j), l_e, l_f) \end{aligned} \tag{2.1}$$

como la probabilidad de traducir la oración origen $\mathbf{f}$ a la objetivo $\mathbf{e}$.

Debido a que hay varios modelos que dependen entre sí para estimarse, se utiliza el algoritmo *EM* (expectation maximitation) que básicamente consiste en iniciarlos con valores aleatorios para luego poder aplicarlos a los datos y así entrenar nuevamente los modelos con los datos anteriormente estimados.

Un modelo de lenguaje, entrenado con el lenguaje objetivo, es utilizado para evitar que la traducción sea poco fluida. *Modelos de lenguajes de n-gramas* son comúnmente usados:

$$\begin{aligned} p(\mathbf{e}) &= p(e_1, e_2, ..., e_{l_e}) \\ &= \prod_{i=1}^{l_e} p(e_i|e_1, ..., e_{i-1}) \\ &\simeq \prod_{i=1}^{l_e} p(e_i|e_{i-(n-1)}, ..., e_{i-1}) \end{aligned} \tag{2.2}$$

Para encontrar la mejor traducción, con mayor probabilidad, es utilizado el modelo de traducción junto con el modelo de lenguaje de la siguiente manera:

$$\begin{aligned} \mathbf{e}_{mejor} &= argmax_{\mathbf{e}} p(\mathbf{e}|\mathbf{f}) \\ &= argmax_{\mathbf{e}} \frac{p(\mathbf{f}|\mathbf{e})p(\mathbf{e})}{p(\mathbf{f})} \\ &= argmax_{\mathbf{e}} p(\mathbf{f}|\mathbf{e})p(\mathbf{e}) \end{aligned} \tag{2.3}$$

La traducción consta en encontrar las mejores puntaciones de la formula 2.3 "buscando" la traducción con mejor probabilidad, denominada "búsqueda de error". La ecuación 2.3 es un problema de búsqueda de complejidad muy elevada para una computadora, por lo tanto usa múltiples heurísticas de búsqueda para encontrar la mejor solución ("beam search" Tillmann y Ney (2003)).

Usualmente este tipo de arquitectura no es utilizada para la traducción en sí, más bien para estimar los diferentes modelos útiles para el estadístico por frase.

Figura 2.10: Ejemplo de estadístico por palabra.

Estadísticos por frase

Una palabra en el lenguaje origen puede traducirse en múltiples palabras del lenguaje objetivo o viceversa, por ello las palabras no son la mejor unidad de traducción. Es así como esta arquitectura utiliza las frases, definida como un conjunto de palabras, como unidad de traducción. Esta forma de traducción también ayuda a resolver algunas ambigüedades.

Estadísticos por frase, a diferencia de palabras, no utiliza tantos modelos probabilísticos por defecto (fertilidad no es utilizada por ejemplo) para entrenar. A través de los modelos ya entrenados en estadísticos por palabra, alineación por ejemplo, "busca" la mejor traducción de un texto.

En la Figura 2.11 se puede ver un ejemplo de la noción básica de traducción estadística por frases del español al alemán.

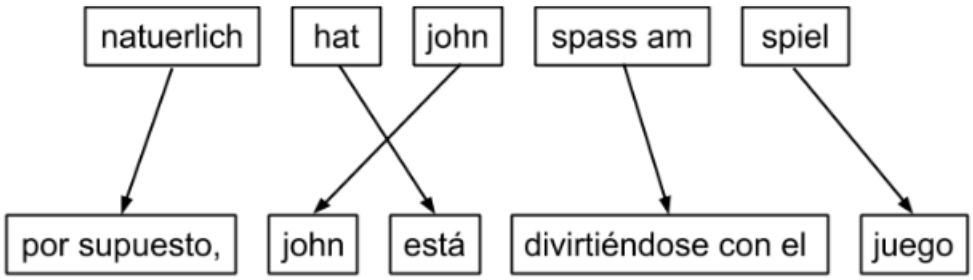

Figura 2.11: Ejemplo de estadístico por frases.

Para traducir estas frases es necesario un modelo probabilístico de traducción para frases, es decir sea $\bar{f}_i, \bar{e}_i$ frases del lenguaje objetivo e origen respectivamente, denotaremos como $\phi(\bar{f}_i|\bar{e}_i)$ la probabilidad de traducción de $\bar{f}_i$ a $\bar{e}_i$. Se podría pensar como una tabla de probabilidades de la siguiente forma:

Traducción	Probabilidad
por supuesto	0.5
por supuesto,	0.3
naturalmente	0.15
naturalmente,	0.05

Tabla 2.1: Tabla de oraciones de *natuerlich*. Ver ejemplo de la Figura 2.11.

Al igual que los estadísticos por palabras nombrados anteriormente, utiliza un decodificador para la traducción (ver fórmula 2.3). Sin embargo, la diferencia principal está en la definición de $p(\mathbf{f}|\mathbf{e})$ que se basa en la multiplicación de dos modelos:

la tabla de traducción (modelo ϕ) y de reordenación (modelo d), con sus respectivos pesos λ_ϕ y λ_d. Además, el decodificador agrega el modelo de lenguaje p_{ML} para asegurar fluidez con su peso λ_{ML}. Luego la fórmula 2.3 quedaría de la siguiente forma:

$$\mathbf{e}_{mejor} = argmax_{\mathbf{e}} \prod_{i=1}^{I} \phi(\bar{f}_i|\bar{e}_i)^{\lambda_\phi} d(start_i - end_{i-1} - 1)^{\lambda_d} * \prod_{i=1}^{|\mathbf{e}|} p_{ML}(e_i|e_1..e_{i-1})^{\lambda_{ML}} \tag{2.4}$$

donde I es la cantidad de frases divididas en $\bar{f}_i$, d es un modelo de reordenamiento basado distancias, $start_i$ esta definido como la posición de la primera palabra del lenguaje origen que traduce a la frase i (análogo con end_i).

El modelo de reordenamiento d se puede definir como una función de coste en decadencia exponencial, es decir $d(x) = \alpha^{|x|}$ con un parámetro $\alpha \in [0, 1]$. A diferencia de el resto de los modelos que se estiman con los datos.

Si agregamos exponencial y logaritmo en la ecuación 2.4 quedaría un **modelo linear logarítmico**, es decir:

$$\mathbf{e}_{mejor} = argmax_{\mathbf{e}} \exp \sum_{i=1}^{n} \lambda_i h_i(x) \tag{2.5}$$

por lo que n será el número de características (en este caso 3 por 2.4), variables aleatorias $x \in (e, f, start, end)$, y funciones de características $h1 = \log \phi$, $h2 = \log d$ y $h3 = \log p_{ML}$.

Gracias a esta nueva definición permite aumentar el número de características fácilmente (h_i funciones de modelos). Por ejemplo, en vez de utilizar una sola dirección $\phi(\bar{f}|\bar{e})$ se utiliza ambas $\phi(\bar{e}|\bar{f})$. Claramente, lleva a una mejora en la traducción ya que los errores en alineaciones disminuyen. Además, características como *peso léxico* y *penalidad en palabras y en frases* pueden ser agregadas, entre otras. La primera tiene en cuenta significados de palabras iguales, y la otra penaliza frases o traducciones de palabras de diferentes longitudes.

2.5.6 Neuronales

Traducción automática neuronal (NMT) es un enfoque *end-to-end* basado en aprendizaje profundo, es decir modela el proceso total de traducción automática a través de una gran red neuronal. El modelo es entrenado con un corpus paralelo al igual que el estadístico.

Diferentes tipos de arquitecturas han sido utilizadas para la traducción, sin embargo, el estado del arte en la actualidad son las arquitecturas *codificador-decodificador* (Cho y cols., 2014) con mecanismos de atención (Luong, Pham, y Manning, 2015; Bahdanau, Cho, y Bengio, 2014).

Como su nombre lo índica, esta compuesta por tres componentes: codificador, decodificador y atención. El codificador transforma las oraciones orígenes en una lista de vectores (uno por cada palabra). Dada la lista de vectores, el decodificador produce palabra por palabra hasta la palabra especial </s>. Ambos están conectados por atención, donde permite al decodificador enfocarse en diferentes regiones de los estados intermedios del codificador.

Explicaremos, de una forma general, una arquitectura básica *codificador - decodificador*. Sea (X, Y) oraciones del lenguaje origen y objetivo. Sea $X = x_1, x_2, ..., x_M$ secuencia de M palabras del lenguaje origen e $Y = y_1, y_2, ..., y_N$ N palabras del lenguaje objetivo. $\mathbf{X} = \mathbf{x}_1, ..., \mathbf{x}_M$ son vectores de dimensión fija (análogo con $\mathbf{Y}$), el cual representa cada x_i para $i = 1, ..., M$ (análogo con Y) resultado del codificador y decodificador respectivamente. Luego, los estados intermedios de las capas ocultas ("hidden layers") son calculadas como:

$$\begin{aligned} h_t &= CodificadorRNN(h_{t-1}, \mathbf{x}_t) \\ h_t &= DecodificadorRNN(h_{t-1}, c_t, \mathbf{y}_{t-1}) \\ y_t &= softmax(h_t) \end{aligned} \tag{2.6}$$

donde CodificadorRNN, DecodificadorRNN son redes RNNs, las cuales calculan estados de las capas ocultas (h_t) del codificador y decodificador respectivamente. y_t es la palabra generada por el decodificador en el tiempo t. Además, c_t es un vector

de contexto calculado como:

$$\begin{aligned} c_t &= \sum_s a_t(s) h_s \\ a_t(s) &= softmax(Atencion(h_{t-1}, h_s)) \end{aligned} \tag{2.7}$$

el cual h_s y h_t son estados de las capas ocultas del codificador y decodificador respectivamente. *Atencion* es una función de atención. Para funciones de atención ver Luong y cols. (2015); Bahdanau y cols. (2014). Para "softmax" ver Campbell, Dunne, y Campbell (1997).

Los argumentos de las funciones de las ecuaciones en 2.6 y 2.7 son una representación de lo que usa cada componente para calcularse.

Luego, el entrenamiento del sistema consiste en minimizar el error de "cross-entropy" (Li y Tam, 1998) de las oraciones resultantes de la generación, condicionadas de las oraciones orígenes, es decir:

$$\max_{\Theta} \frac{1}{N} \sum_{i=1}^{N} \log P_{\Theta}(Y^{*(i)}|X^{(i)}) \tag{2.8}$$

$X^{(i)}$ es la oración i de X e $Y^{*(i)}$ es la oración i de las oraciones de Y^* que son generadas por el modelo. Con P_{Θ} el modelo y Θ los parámetros. Esta primera aproximación es muy vaga, por lo que en (Wu y cols., 2016) muestran mejores funciones para minimizar el error, además de explicar con más detalle P_{Θ}.

Se puede ver en la imagen de la Figura 2.12 una clara representación de una arquitectura *codificador-decodificador* con mecanismos de atención. Las palabras de la oración en inglés son transformadas a vectores para luego ser procesadas por redes RNNs (celdas azules). Una vez visto el símbolo <s>, se inicializa las celdas rojas, repitiendo el proceso de las celdas azules. En la figura, se puede ver múltiples flechas desde las celdas azules, el codificador, al vector de contexto, representando así, el calculo de este vector. Asimismo, todas las celdas están conectadas con una flecha a las anteriores, lo que indica que todas usan el estado anterior para calcular su estado actual (propiedad de RNN). En las celdas rojas se puede apreciar que, además de utilizar el estado anterior, utiliza la palabra generada en el estado previo y el vector de contexto. Las celdas marrones son la capa de "softmax" para predecir la siguiente palabra hasta generar el símbolo de final de oración </s>.

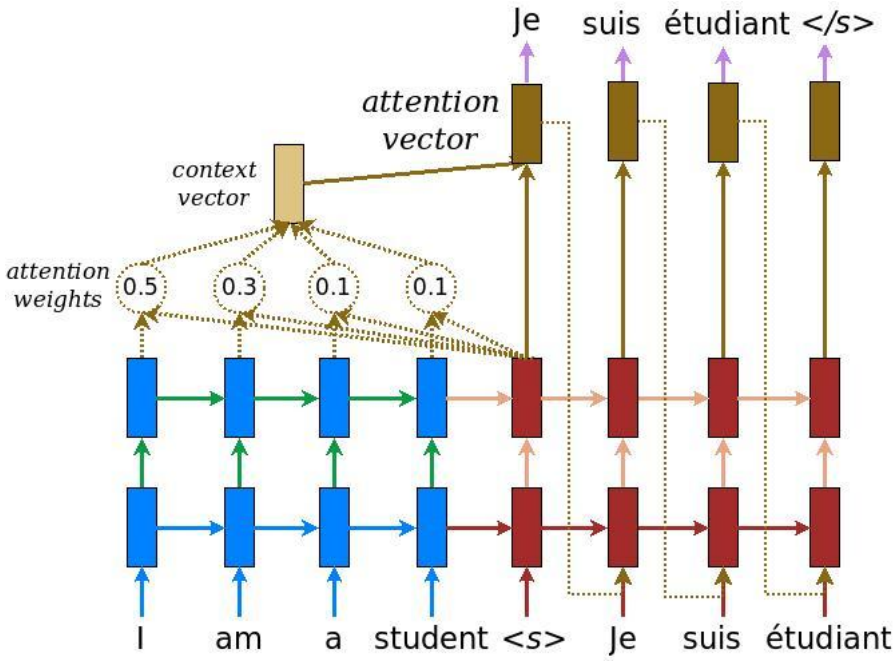

Figura 2.12: Visualización de una red neuronal codificador-decodificador con mecanismos de atención. Ejemplo de inglés a francés.

2.6 Evaluación de la TA

La evaluación de la calidad de una traducción automática es muy compleja debido a que no hay una única respuesta correcta, por lo que no existe una métrica perfecta. Por lo tanto, hay dos formas de evaluarlo, bien comparando la similitud de la respuesta de la traducción automática con una traducción humana[3], o bien con el juicio humano. Asimismo, se debe considerar el objetivo del sistema.

2.6.1 Evaluación humana

Un método para la evaluación de la traducción es observar la salida del sistema y con cierto nivel de juicio establecer si es correcta (denominada **por corrección**). Para ello se necesita un experto de dominio, evaluadores bilingües que entiendan tanto el lenguaje objetivo como el origen. Sin embargo, en el peor de los casos es posible realizarlo con evaluadores monolingües que logren comprender el idioma del lenguaje objetivo junto con un texto de referencia. En general, esta evaluación se realiza oración por oración.

[3]Necesita texto de referencia.

La evaluación por corrección es usada en gran medida, sin embargo solo sirve para sistemas donde la traducción es de alta calidad debido a que solo hay dos opciones para calificar una oración: correcta o incorrecta. Por ende, es usual usar métricas donde el criterio de evaluación es más gradual tales como **fluency**, donde se coloca un puntaje por gramática e uso del idioma, y **adequacy**, el cual clasifica de acuerdo al significado de la oración.

A pesar de que estas métricas son útiles, las definiciones de estas son muy vagas por lo que es bastante difícil que evaluadores sean consistentes, como por ejemplo en Koehn (Koehn y Monz, 2005). Claramente encontrar una métrica perfecta para evaluación humana es imposible debido a que no puede ser costosa, tanto económicamente como en tiempo involucrado. Además, debe tener la menor ambigüedad posible, es decir dos evaluadores deben evaluar la calidad de una manera equivalente sin subjetividad, y debería poder utilizarse para la etapa de optimización de un modelo.

A causa de la ambigüedad humana en la evaluación, existen métricas **internas de veracidad** (IRR) las cuales miden el porcentaje de concierto entre los evaluadores. La métrica tiene una precisión mayor si los evaluadores son entrenados previamente ya que es posible que el puntaje sea afectado por el sesgo de cada evaluador.

Según (E. Saal, Downey, y A. Lahey, 1980) hay múltiples definiciones para IRR diferenciando los diferentes puntos de vista sobre la definición de "acuerdo": 1) evaluadores estén de acuerdo con el desempeño "oficial" del sistema, 2) evaluadores establezcan un acuerdo entre sí y 3) evaluadores eligen cual desempeño es mejor y peor. Asimismo, depende de como definir el comportamiento, es decir los evaluadores se guían de acuerdo a puntajes establecido o en base a sus opiniones. Light's kappa (Light, 1971.) y Cohen's kappa (Cohen, 1968) son ejemplos de IRR.

2.6.2 Evaluación automática

Las métricas automáticas han sido creadas para solucionar parte del problema de la evaluación humana, tanto tiempo, ambigüedad y costo económico. Pese a que este tipo de evaluación soluciona mucho de los problemas ya mencionados, muchos sistemas siguen requiriendo especialistas del dominio para crear textos traducidos comparables con la traducción del sistema (denominados textos de referencia), y también para la evaluación.

Word error rate

Word Error Rate (WER) es una métrica automática de error y distancia, es decir mide la cantidad de errores en la traducción a través de una distancia. La distancia utilizada es la de *Levenshtein* (Levenshtein, 1966), definida como el número mínimo de operaciones requeridas para transformar una cadena de caracteres en otra. En este caso particular definiremos el mínimo número de operaciones para transformar una oración en otra a través de cambios de palabras. Se entiende por operación, bien una inserción, eliminación o la sustitución de una palabra.

Una vez dada la distancia de Levenshtein podemos computar WER de la siguiente manera:

$$WER = \frac{sustituciones + inserciones + eliminaciones}{|referencia|}$$

donde sustituciones, inserciones y eliminaciones son operaciones, y referencia es el texto de referencia.

Si la medición es 0 quiere decir que la traducción es igual a la de referencia, por otro lado si se encuentra cerca del 1[4], mayor cantidad de operaciones y por lo tanto menor calidad en la traducción.

Bilingual evaluation understudy

Por otro lado **bilingual evaluation understudy** (BLEU)(Papineni, Roukos, Ward, y Zhu, 2001) es una métrica que es similar a WER, pero que considera las coincidencias de n-gramas con la traducción de referencia. Por ello, es preferible tener múltiples textos de referencia (debido a que las no coincidencias penalizan mucho), sin embargo es posible hacerlo con uno sólo, siempre y cuando las traducciones no sean del mismo traductor. Es importante notar que mientras más textos de referencia haya, la métrica aumentará.

Al igual que WER oscila entre 0 a 1. Ninguna traducción tendrá 1 a menos que sea idéntica a su referencia, y tendrá 0 si no hay coincidencias de n-gramas. Está basada en la *precisión modificada de n-gramas* que se calcula tomando el número máximo de apariciones de n-gramas en el texto de referencia encontradas en la traducción dividido por la longitud de la traducción, como se muestra en el siguiente ejemplo.

[4]Para simplificar utilizamos diferentes evaluaciones que nos provee el conjunto de herramientas utilizadas, por lo tanto, los resultados para WER en este trabajo serán mostrados en %, es decir de 0 a 100.

Traducción: `El muchacho miraba el atardecer.`

Texto de referencia: `El hombre seguía mirando el atardecer.`

Sea $prec_n$ la precisión en el n-grama, entonces $prec_1$ es $\frac{3}{5}$, por lo que $prec_2$ es $\frac{1}{4}$.

Una vez obtenida la precisión hasta N-gramas BLUE se define como la media geométrica de las precisiones con una penalidad BP y pesos:

$$BLEU = BP * exp(\sum_{n=1}^{N} w_n * log(prec_n))$$

donde $\sum_{n=1}^{N} w_n = 1$ son los pesos, y BP es la penalidad para las frases cortas.

Sea t y r el largo de la traducción y referencia respectivamente, entonces:

$$BP = \begin{cases} 1 & si \quad t > r \\ e^{(1-r/t)} & si \quad t \leq r \end{cases}$$

La traducción es penalizada tanto por frases muy largas como pequeñas a causa de la precisión modificada y la breve penalidad BP. La unidad básica de esta métrica es la oración, sin embargo hay formas de evaluar todo un texto de traducción, como por ejemplo, calcular la media aritmética de los resultados de la evaluación de las oraciones.

Capítulo 3

Trabajos Relacionados

Desde hace varios años la TA ha ido ganando terreno en el ámbito industrial, primero en su versión estadística y actualmente en su versión neuronal, por lo que existen una gran cantidad de trabajos que reportan experiencias sobre la adopción de TA en el ámbito industrial. Para ilustrar esto, consideremos algunos de los trabajos presentados este año en una de las conferencias más relevantes del área, la EAMT (*European Association for Machine Translation*, s.f.).

3.1 How to Move to Neural Machine Translation for Enterprise - Scale Programs — An Early Adoption Case Study

En este trabajo (Schmidt y Marg, 2018) evalúan la posibilidad de migración de un sistema de traducción automático estadístico, para post edición, a uno neuronal. Utilizaron 28 pares de lenguas para comparar calidades, a través de evaluaciones automáticas, humanas y en producción, y así elegir las mejores combinaciones de sistema con sus pares de lenguas. El objetivo es proveer una mayor calidad a la traducción para post edición, y eventualmente incrementar la productividad y reducir costos.

3.2 Implementing a neural machine translation engine for mobile devices: the Lingvanex use case

Implementaron una versión neuronal móvil de traducción automática para maximizar la calidad de traducción y disminuir el tamaño del modelo. El objetivo era

utilizar este traductor para viajeros que necesiten traducciones básicas usadas en el día a día sin requerir conexión a internet (Parcheta, Sanchis-Trilles, Rudak, y Bratchenia, 2018).

Los datos que utilizaron para entrenar fueron extraídos del corpus OPUS (Skadiņš, Tiedemann, Rozis, y Deksne, 2014) y Tatoeba eligiendo cuidadosamente, a través de un algoritmo de selección, 740,000 oraciones paralelas del inglés al español. El sistema fue entrenado con openNMT basado en pytorch.

3.3 Toward leveraging Gherkin Controlled Natural Language and Machine Translation for Global Product Information Development

Un sistema de traducción automática es implementado para un lenguaje controlado (siglas en inglés CNL) Gherkin para asimilación, es decir sin ningún tipo de post edición, de inglés a italiano, portugués de Brasil y francés (O'Brien, 2018).

Esto permite una globalización de "historias de usuario" reduciendo el costo y entendimiento de los requisitos del software. El sistema utilizado fue Microsoft Translator Hub entrenado con diferentes textos CNL creado por desarrolladores y evaluadores de los productos McAfee.

Capítulo 4

Experimentos

4.1 Datos y dominios

El Servicio de Traducción del Correo Suizo traduce, principalmente, textos del alemán (de-DE) a francés(fr-CH), italiano(it-CH) e inglés(en-UK). Por simplicidad, denominaremos a los idiomas como DE, FR, IT y EN respectivamente. Asimismo, el servicio realiza múltiples tareas relacionadas con la traducción en diferentes materias por lo que posee memorias de traducción (denominadas MTs) en varios dominios: vocacional (denotado *Modulo*), servicios financieros (denotado *PF*), manual de procedimientos (denotado *PN*), y reporte anual (denotado *GB*). Cuenta con una memoria grande "master TM" (denotada *MTM*), la cual incluye todas las memorias anteriormente mencionadas y traducciones adicionales. La mayoría de estas memorias son paralelas entre los pares de lenguajes, es decir, al menos un 65% de las oraciones son comunes en todos los pares de idiomas, sin embargo, el porcentaje es levemente menor para los pares con inglés . Debido a que el volumen de oraciones en inglés es significativamente menor, sólo consideramos el dominio "annual report" *GB* y *MTM*. Las oraciones no se repiten dentro de cada memoria de traducción.

Estas memorias de traducción serán utilizadas como conjunto de entrenamiento. Los detalles de la cantidad de datos de entrenamiento son mostradas en la Tabla 4.1.

Los pares de lenguajes involucrados en este proyecto son bastante desafiantes debido a que son altamente flexivos (alemán, francés e italiano). Además, los pares de idiomas DE-IT y DE-FR son poco estudiados en la literatura de traducción automática donde la mayoría utilizan ejemplos con inglés (tanto objetivo como origen).

TMs	**DE-FR**	**DE-IT**	**DE-EN**
Modulo	99,612	107,128	–
PF	129,694	122,568	–
PN	23,131	23,447	–
GB	38,580	37,721	32,857
MTM	2,558,148	1,929,530	417,817

Tabla 4.1: Número de unidades de traducción por cada par de lenguajes

4.2 Selección del sistema estadístico

La primera fase del estudio fue dedicado a comparar y evaluar dos grandes sistemas de traducción automática estadístico por frases: una plataforma comercial en línea ofrecida por "Microsoft Translator Hub", MTH contra uno de código abierto Moses.

Estas soluciones son opciones comunes para muchas empresas que quieren realizar experimentos con sistemas de traducción automática; la primera es una plataforma privativa de un tercero (traducción como SaaS), que sólo requiere subir las MTs y luego pagar para el empleo del sistema, mientras que la otra solución, es totalmente propia y gratuita donde permite que el proceso total sea controlado, pero que requiere un gran nivel técnico de conocimientos y recursos computacionales.

A causa de que hay memorias de traducción de cada dominio en su respectivo par de lenguas, hemos intentado diferentes combinaciones para mejorar los resultados. En un principio entrenamos sistemas individuales para cada dominio usando las MTs individualmente. Cada conjunto de entrenamiento era pequeño, por lo que los resultados no fueron del todo deseables. Por consiguiente, decidimos realizar dos rondas de traducción con diferentes conjuntos de entrenamiento:

- Ronda 1 - Utilizar todas las memorias de traducción como un conjunto de entrenamiento mixto: conjuntos de test de cada dominio en particular fueron utilizados para evaluar la calidad del sistema. Tanto los modelos de Moses como los de MTH fueron entrenados para DE-IT/FR[1].

- Ronda 2 - Utilizar sólo *MTM*: debido a que mejores resultados dieron con Moses no se utilizo MTH. Además ahorramos el costo de anonimizar el gran conjunto de entrenamiento.

[1]El par DE-EN fue agregado en la siguiente fase.

Herramientas que provee Moses y MTH fueron utilizadas para el proceso de entrenamiento.

4.2.1 Entrenamiento del sistema: Moses

Después de seleccionar los mejores métodos de preproceso y n-grama indicados en las secciones 4.2.2 y 4.2.3, seguimos con el entrenamiento y evaluación del sistema. Esto es, entrenamiento del modelo de lenguaje (explicado en sección 4.2.4) y traducción, mejoramiento de parámetros y evaluación utilizando los diferentes dominios (sección 4.4.2).

4.2.2 Preprocesamiento

Debido a que los datos estaban expuestos en MTH, realizamos anonimización en algunos conjuntos de entrenamiento donde nombre de entidades, números (teléfonos, importes, cuentas, etc.), urls y emails fueron remplazados por una etiqueta en el corpus de entrenamiento y de evaluación.

Luego de tal anonimización exploramos algunas posibilidades para mejorar los resultados de la traducción. En la figura 4.1 se puede ver la cadena de pre procesamiento realizada.

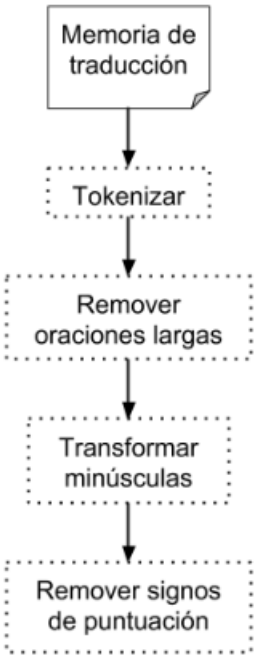

Figura 4.1: Pre procesamiento de las memorias de entrenamiento.

En la primera etapa del preprocesamiento el texto es tokenizado, es decir, agregar espacios entre palabras y signos de puntuación, así como la división de algunas palabras, por ejemplo "aren't" es divido en "aren" "'t"; hay diferentes métodos de tokenización dependiendo del lenguaje. Debido a que las oraciones muy largas no ayudan en el entrenamiento, por la gran dificultad de alinearlas, se eliminan las que están compuestas por más de noventa palabras (denominada "limpieza"). Las frases eliminadas para DE-IT, DE-FR y DE-EN son 5%, 3% y 1% respectivamente.

Una vez tokenizado y *"limpiado"*, exploramos diferentes alternativas de procesar el texto. Entre tantas, pasar todas las palabras a minúsculas pensando que la cantidad de vocabulario iba a reducirse, y por lo tanto la varianza de la distribución sería menor. Sólo probamos con la memoria *"MTM"* comparando diferentes resultados (ver Tabla 4.2), y así elegir la mejor opción de procesamiento.

En un primer intento las palabras del texto origen y objetivo de las memorias de traducción fueron cambiadas a minúsculas. Debido a que en el alemán mismas palabras tienen diferentes funciones sintácticas dependiendo de sus mayúsculas, palabras diferentes iban a traducirse iguales, por ejemplo *"tanzen"* y *"Tanzen"* donde la primera es *"bailar"* y la otra *"baile"*.

Por el motivo anteriormente mencionado, decidimos que sólo para el alemán íbamos a modificar la primera palabra de cada oración dependiendo su función sintáctica (llamado segundo intento). En otras palabras, entrenamos un modelo probabilístico que cuente la cantidad de apariciones de la palabra en mayúsculas y minúsculas, y así modificar la primera letra de la palabra de acuerdo a esa distribución.

	DE-IT		**DE-FR**	
Intento	*WER*	*BLEU*	*WER*	*BLEU*
Nulo	39,47	0,52	40,52	0,52
Primer	36,58	0,54	38,97	0,54
Segundo	35,63	0,55	37,68	0,55

Tabla 4.2: Diferentes intentos de cambiar mayúsculas a minúsculas y viceversa. El *"nulo"* intento es sin cambios; el primer intento es cambiando toda palabra a minúscula; el segundo intento es cambiando la primera letra del texto origen (alemán) y todas las palabras en el objetivo.

Una vez seleccionado el mejor intento, probamos eliminar diferentes puntuaciones. Esto mejora poco los resultados, por lo que no nos pareció relevante mostrarlos. Además, como es demasiado costoso volver a colocar la puntuación en su respectivo

lugar no lo tomamos como una alternativa viable.

En definitiva hemos decidido utilizar el segundo intento con sus respectivos signos de puntuación debido a que mejora la calidad de la traducción manteniendo la cadena de preproceso bastante simple y rápida.

4.2.3 Elección de n-gramas

Los modelos de lenguajes *"n-gramas"* fueron entrenados usando un conjunto de herramientas llamado KenLM (Heafield, 2011) en varios *"n"*. En teoría todos los modelos tienen el mismo concepto, sin embargo se diferencian en algunas optimizaciones con respecto a los recursos computacionales.

Empezamos utilizando el modelo SRILM (Stolcke, 2004), el cual es bastante usado en múltiples sistemas. Luego migramos a IRSTLM (Federico, Bertoldi, y Cettolo, 2008), que es una versión que consume menos recursos (memoria), para ver su desempeño en la memoria más grande *"MTM"*. Como ambos son modelos de lenguajes no hay una diferencia significativa respecto a la calidad, pero si en la velocidad.

Además de intentar con estos dos modelos, observamos que cambiando los diferentes *"n"* para los *"n-gramas"* llevan a una mejora en la calidad de traducción. Es cierto que a medida que aumente este número, la mejora es inevitable (en el sentido de métrica) ya que BLEU se basa en los aciertos de *n-gramas*, sin embargo el modelo de lenguajes consume más recursos y es más lento. Así que probamos con 3,4,5-gramas (ver Tabla 4.3) para *"MTM"* en *"DE-IT"* con un conjunto de test del dominio de *"GB"*.

Gramas	*WER*	*BLEU*
3-gramas	48,98	0,42
4-gramas	47,81	0,45
5-gramas	47,15	0,46

Tabla 4.3: Diferentes *"n-gramas"* con el modelo SRILM.

Podemos ver que hay bastante mejora a medida que aumentamos el *"n"*. Sin embargo, hemos decidido tomar 4-gramas debido a que hay una mayor brecha entre 3-gramas y 4-gramas que entre 4-gramas y 5-gramas, además los recursos computacionales para 5-gramas son mayores.

4.2.4 Modelos de traducción

Los modelos de traducción fueron entrenados con la ayuda de GIZA++ (Och y Ney, 2003) para las memorias más pequeñas, sin embargo para la memoria más grande *MTM* utilizamos mgiza (Gao y Vogel, 2008) a causa del tiempo y recursos.

Para estimar el modelo de traducción estadístico por frases se necesita que las palabras estén alineadas, es decir entrenar un modelo de alineación en el conjunto de entrenamiento (ver sección 2.5.5). Esta alineación es realizada por GIZA++/mgiza, para luego extraer tablas de oraciones (ver Figura 2.1) y otras informaciones para poder usarse en el decodificador.

Modelos IBM (Brown, Pietra, Pietra, y Mercer, 1993) y modelos de alineación HMM (Vogel, Ney, y Tillmann, 1996) son utilizados por la herramienta para estimar los parámetros. En cada iteración, se calcula la mejor alineación de palabras por cada oración en el corpus, acumulando conteos, y luego se normaliza para generar el modelo de traducción para la siguiente estimación. Esta etapa es una de las más costosas por el tiempo y recursos computacionales, sobre todo para una memoria de traducción grande. Cabe aclarar que en la etapa de alineación, todas las oraciones son alineadas independientemente para luego actualizar los parámetros de los modelos.

Luego de que las oraciones están alineadas, se extrae la tabla de traducción léxica, a través de máxima verosimilitud, y se estima el modelo de traducción directa e inversa, un modelo léxico directo e inverso y también se puede agregar modelos de penalidades por tamaño de frase y palabras, entre otros. Luego se logra calificar cada frase con las probabilidades de estos modelo agregando diferentes pesos. Seguidamente, el modelo de reordenamiento es estimado.

Finalmente, técnicas como *"beam search"* son empleadas en el decodificador para obtener la mejor traducción. Modelos de traducción, reordenamiento y lenguaje son utilizados para esta estimación.

Debido a que dos tercios de tiempo del entrenamiento es consumido por la alineación de palabras, mgiza utiliza la propiedad de independencia de alineaciones y paraleliza GIZA++ donde el tiempo y recursos computacionales son reducidos cuantitativamente. La idea básica es entrenar incrementalmente el modelo paralelizando las oraciones, es decir, crear un hilo para cada oración y utilizar métodos para el acceso a memoria eficaz.

4.3 Selección del sistema neuronal

La segunda parte de este estudio consistió en comparar los modelos estadísticos con los neuronales utilizando aprendizaje profundo.

Traducción automática neuronal es una nueva estrategia que emergió en estos últimos años propuesto por (Kalchbrenner y Blunsom, 2013; Sutskever y cols., 2014; Cho y cols., 2014).

Decidimos utilizar arquitecturas codificación-decodificación (Cho y cols., 2014) particularmente seq2seq (Bahdanau y cols., 2014) con mecanismos de atención (Luong y cols., 2015) debido a que la mayoría de la literatura afirma que es el nuevo estado del arte. Para ello utilizamos un framework de software libre: openNMT.

OpenNMT es una herramienta utilizada tanto en la academia como en la industria. Esta compuesto por una colección de implementaciones (arquitecturas neuronales) para que sea simple y extensible, manteniendo la eficiencia y la precisión.

Hay 3 implementaciones diferentes: en Lua[2], PyTorch[3] y TensorFlow[4], donde cada una de ellas tiene sus ventajas. Debido a que el sistema podría ser implementado en el servicio de traducción del correo suizo, es importante que pueda ser utilizado mediante un servicio, por ello la elección de TensorFlow.

El sistema es entrenado con la memoria *MTM* debido a que es el corpus paralelo con más oraciones. Asimismo, se podrá realizar una comparación con los mejores sistemas estadísticos.

4.3.1 Preprocesamiento

Al igual que el sistema estadístico, las memorias de traducción fueron tokenizadas previamente. Luego seguimos (Sennrich, Haddow, y Birch, 2015) para la segunda parte del preprocesamiento.

Debido a que a las arquitecturas modernas para traducción automática necesitan un vocabulario fijo, este estudio (Sennrich y cols., 2015) propone segmentar las "palabras poco vistas" en sub-palabras para así lograr reducirlas y mejorar la distribución de palabras. Esto se basa en que gran parte de este tipo de palabras son llamadas "palabras transparentes" clasificadas en tres tipos principalmente: nombre de entidades, palabras del mismo origen o "prestadas de otro idioma" y palabras

[2] `https://www.lua.org`
[3] `https://pytorch.org`
[4] `https://www.tensorflow.org`

complejas morfológicamente.

Las sub-palabras son agregadas al vocabulario inicial y así las memorias de traducción son segmentadas de acuerdo al nuevo vocabulario.

Consideramos que esta solución mejoraría bastante el desempeño del sistema debido a que el alemán es una lengua con palabras que se forman con múltiples lexemas. Por ejemplo la palabra *"Sonnensystem"* donde la traducción es *"sistema solar"* se puede dividir en *"sonnen"* y *"system"* donde cada palabra por separado tiene su respectiva traducción *"solar"* y *"sistema"*. Esto reduciría el número de palabras "poco vistas".

4.3.2 Arquitectura

Seguimos (Britz, Goldie, Luong, y Le, 2017) para elegir la arquitectura e hyper parámetros y realizamos algunas modificaciones[5]. Explicaremos algunos hyper parámetros elegidos producto de las modificaciones.

Nuestros modelos se basan en las arquitecturas codificador-decodificador con mecanismos de atención. El codificador f_{enc} toma como entrada un conjunto de tokens $\mathbf{x} = (x_1, ..., x_m)$ y produce una secuencia de estados intermedios $\mathbf{h} = (h_1, ...h_m)$. f_{enc} son dos LSTM (Hochreiter y Schmidhuber, 1997) bi direccionales (Bahdanau y cols., 2014), es decir son un tipo especial de RNN donde tiene más parámetros (entrada, olvido y contexto) por el que puede guardar información por largos períodos de tiempo, y se basa en pasadas y futuras entradas.

Por otro lado, el decodificador f_{dec} son dos LSTM con mecanismos de atención donde predice probabilidades $\mathbf{y} = (y_1, ..., y_k)$ basado en las capas ocultas (llamadas hidden layers). La probabilidad de cada token $y_i \in 1, .., V$ depende de los estados anteriores del decodificador, de la palabra anterior y del contexto (también llamado atención). El mecanismo de atención se encuentra entre las capas ocultas del decodificador.

4.3.3 Sobre-ajuste del modelo

Como es sabido, las redes neuronales son propensas a sobre ajustar los datos (overfitting), por lo que, decidimos extraer (aleatoriamente) del corpus paralelo de entrenamiento alrededor del 7% de las frases para validación, en cada par de idiomas, y

[5]Las modificaciones fueron realizadas siguiendo configuraciones por defecto de OpenNMT.

así poder emplear técnicas para evitarlo.

Una de las técnicas más utilizadas es "early stoping" (Prechelt, 1998), es decir, utilizar un criterio de parada para el entrenamiento. Nuestro criterio fue detener el entrenamiento cuando la evaluación con BLEU comienza a descender y la función de "error", cross-entropy (Li y Tam, 1998), empieza a ascender para el corpus de validación, por lo contrario al de entrenamiento.

Además, capas con "droupout" (Srivastava, Hinton, Krizhevsky, Sutskever, y Salakhutdinov, 2014) provee una forma simple para evitar el sobre-ajuste. La idea central es dejar algunos componentes, aleatoriamente, en la salida de la capa de la red que tenga "dropout". El resultado es que en cada capa, mayor cantidad de neuronas son forzadas a aprender las múltiples características de la red. "Droupout" fue utilizado en el codificador y decodificador, ambos con 0.3.

Regularización, otra técnica para evitar el sobre-ajuste, modifica la función de "error" para así penalizar diferentes parámetros. Utilizamos regularización L2 (Wang, Gordon, y Zhu, 2007) con 0.01 (configuración recomendable de OpenNMT).

Estimación de hyper parámetros no fue realizada por el momento, por lo que seguimos (Britz y cols., 2017) para otros hyper parámetros.

4.4 Evaluación automática para modelos estadísticos

Los modelos estadísticos fueron evaluados utilizando métricas estándares *BLEU* (Papineni y cols., 2001) y *Word Error Rate* (WER)[6], así como evaluación humana (sólo para estadístico por ahora).

4.4.1 Datos de evaluación

Diferentes conjuntos de evaluación, uno para cada dominio, fueron utilizados para evaluar la calidad del sistema con datos que no han sido vistos anteriormente, es decir no están en el corpus de entrenamiento. La cantidad de oraciones en el conjunto de evaluación es mostrada en la siguiente tabla 4.4.

[6] A pesar que MTH provee la métrica BLEU después del entrenamiento, utilizamos ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13.pl para calcularlo.

Dominio	**DE-FR**	**DE-IT**	**DE-EN**
PN	1736		–
Modulo	2034		–
PF	1919	2378	–
GB	1829	1718	704

Tabla 4.4: Número de unidades de traducción (oraciones) en el conjunto de evaluación por dominio y par de lenguas. Para *Modulo* y *PN*, italiano y francés comparten exactamente las mismas oraciones en el lenguaje origen alemán, mientras que en otros dominios, al menos un 58% es compartido. Este porcentaje es menor en inglés, ya que el corpus es bastante más chico.

4.4.2 Resultados

Resultados para la ronda uno son mostrados en la tabla 4.5 y 4.6: Moses supera los resultados de la evaluación en la todos los dominios donde los mejores resultados fueron para *PN*. En base a estos resultados, la ronda dos sólo fue entrenada en Moses para evitar anonimizar el gran conjunto de oraciones en el corpus. Los resultados mejoraron para todos los dominios (mirar tabla 4.7).

	Moses		**MTH**	
Dominio	WER	BLEU	WER	BLEU
PN	43.93	0.51	55.11	0.36
Modulo	45.94	0.46	60.17	0.31
PF	50.92	0.40	63.84	0.28
GB	58.49	0.34	71.91	0.23

Tabla 4.5: Resultados para DE-FR en el conjunto mixto de evaluación (ronda 1).

	Moses		**MTH**	
Dominio	WER	BLEU	WER	BLEU
PN	40.40	0.52	52.68	0.37
Modulo	44.16	0.46	55.55	0.35
PF	46.43	0.43	58.36	0.32
GB	51.94	0.40	62.66	0.31

Tabla 4.6: Resultados para DE-IT en el conjunto mixto de evaluación (ronda 1).

Debido a los resultados, concluimos que podemos utilizar solamente Moses con *MTM* como MT para proceder la evaluación humana y así verificar si es indicado usar el sistema para la post-edición (detallado en la sección 4.5).

Dominio	lang/pair	WER	BLEU
PN	DE-IT	33.01	0.6
	DE-FR	34.39	0.61
Modulo	DE-IT	40.96	0.5
	DE-FR	43.53	0.5
PF	DE-IT	43.07	0.48
	DE-FR	41.14	0.52
GB	DE-IT	47.41	0.45
	DE-FR	54.28	0.39
	DE-EN	34.48	0.62

Tabla 4.7: Resultados de la evaluación para DE-FR/IT/EN *MTM* (ronda 2).

4.5 Evaluación humana

El principal objetivo de la evaluación fue estimar la potencial usabilidad de traducción automática para la post-edición en el servicio del correo suizo en diferentes dominios con sus respectivos pares de lenguajes. Para ello, múltiples traductores, de diferentes procedencias, evaluaron la calidad de los diversos sistemas.

Antes de la evaluación final en el correo, traductores de la universidad de traducción de Ginebra evaluaron múltiples sistemas (entrenados diferentes uno de otros) y eligieron el más indicado. Luego, ante esa elección decidimos que los traductores del correo evalúen la calidad de algunas oraciones antes de involucrarlos en una tarea de post-edición en tiempo real, con el objetivo de dar una visión general de la calidad del sistema.

4.5.1 Datos de evaluación

A diferencia de la evaluación automática, utilizamos una pequeña parte de los corpus de evaluación. Seleccionamos aleatoriamente una muestra de 250 oraciones en alemán por cada dominio (en total 1000 oraciones) de diferentes textos de evaluación (descripto en la tabla 4.4), junto con sus traducciones objetivos en francés, italiano e inglés.

Este sub conjunto de textos de evaluación están completamente paralelizados, es decir que se han elegido exactamente las mismas 250 oraciones dentro de cada dominio en particular, en todos los pares de lenguajes. Cabe aclarar que tal como en la evaluación previa, sólo utilizamos "reporte anual" (*GB*) para el par alemán-inglés.

Este sub conjunto fue evaluado con métricas automáticas para así determinar contradicciones con evaluación humana, si es que las hay. Resultados pueden verse en la Tabla 4.8.

dominio	**leng/par**	**WER**	**BLEU**
PN	DE-IT	35.91	0.58
	DE-FR	35.20	0.59
Modulo	DE-IT	41.88	0.48
	DE-FR	47.52	0.46
PF	DE-IT	47.32	0.41
	DE-FR	47	0.43
GB	DE-IT	47.46	0.43
	DE-FR	58.77	0.34
	DE-EN	41.78	0.51

Tabla 4.8: Resultado de las métricas BLEU and WER para las 250 oraciones.

4.5.2 Metodología

Ocho traductores del servicio de lenguajes participaron en la evaluación: tres de ellos para DE-FR y DE-IT, y dos para DE-EN. Todos los traductores trabajaron al menos seis meses en el servicio de traducción del correo suizo, y poseen entre uno a diecinueve años de experiencia en traducción con diferentes dominios.

Días previos a la realización de la evaluación, se ha dado al equipo un día de capacitación para traducción automática y post-edición, involucrando diferentes ejercicios, tanto teóricos como prácticos, sobre el entrenamiento, evaluación y post-edición en sistemas de traducción automática, para así evitar prejuicios en la evaluación (dado al cambio en el flujo de trabajo).

Debido a que el propósito de esta evaluación fue evaluar la calidad de las diferentes oraciones traducidas para post-edición, a través de un sistema de traducción automática, decidimos utilizar una métrica adaptada para nuestro problema. Por lo tanto, para cada oración en el conjunto de evaluación origen, cada traductor evaluó la traducción objetivo preguntándose la siguiente pregunta: *¿Usarías la traducción para la tarea de post-edición?*, con ciertas respuestas posibles tales como *"Si, lo dejaría sin ninguna modificación"* (denotado como "Yes"), *"Si, pero con ciertas modificaciones"* (denotado como "YwC") y *"No, la traduciría nuevamente"* (denotado

como "No"). Como los evaluadores ya estaban familiarizados con el material a evaluar, no incluimos traducciones de referencia. Sin embargo, los traductores fueron conscientes del origen y del dominio de cada oración, por lo que pudieron evaluar si la terminología utilizada era la correcta.

La categoría será elegida en base al acuerdo de los traductores. En caso de no haber un acuerdo, esa oración no podrá ser contada. Por ejemplo, si hay tres traductores evaluando donde dos de ellos eligen "Yes", entonces la categoría será "Yes". En caso que los tres tengan opiniones diferentes, esta oración no será contada.

Estamos conscientes de que la categoría "YwC" es muy amplia, en el sentido de que oraciones pueden requerir un gran trabajo de post edición o simplemente una mínima modificación. Sin embargo, en esta evaluación preliminar estamos interesados mayormente en determinar si los traductores del servicio podrían aceptar los resultados de traducción automática para una posterior tarea de post-edición.

4.5.3 Resultados

La figura 4.2 muestra los resultados en porcentajes de las oraciones que son usables para post-edición, calculadas como la suma de todos los "Yes" y "YwC", donde la mayoría del jurado estuvo de acuerdo con esa decisión[7] divido por el total de oraciones en el corpus. Para FR e IT, los resultados son bastantes alentadores, oscilando entre 84% y 96% de oraciones usables para cada conjunto de evaluación. Se puede observar que el dominio *PN* obtiene los mejores resultados, tanto para IT como FR; estos resultados fueron confirmados por las métricas automáticas (mirar tabla 4.8).

El segundo dominio con mejor porcentaje de aceptación fue "reporte anual" *GB* donde los resultados del italiano fueron altos comparados con el francés y el inglés. Sin embargo, esto contradice con las métricas automáticas, siendo *GB* el dominio con el peor puntaje.

El gráfico en la Figura 4.2 también muestra que para el dominio *GB* en inglés hay menor porcentaje de usabilidad en las frases 62.80%. En cierto modo, consideramos que este porcentaje no es un buen indicio para establecer la calidad en EN, debido a que se realizó con sólo dos evaluadores y estos quedaron en desacuerdo en un 28% de las oraciones.

Un análisis interno de veracidad fue realizado para medir la consistencia de los

[7]Mayoría del jurado se refiere a dos personas. En el caso DE-EN la mayoría es igual al total, debido a que sólo hay dos traductores.

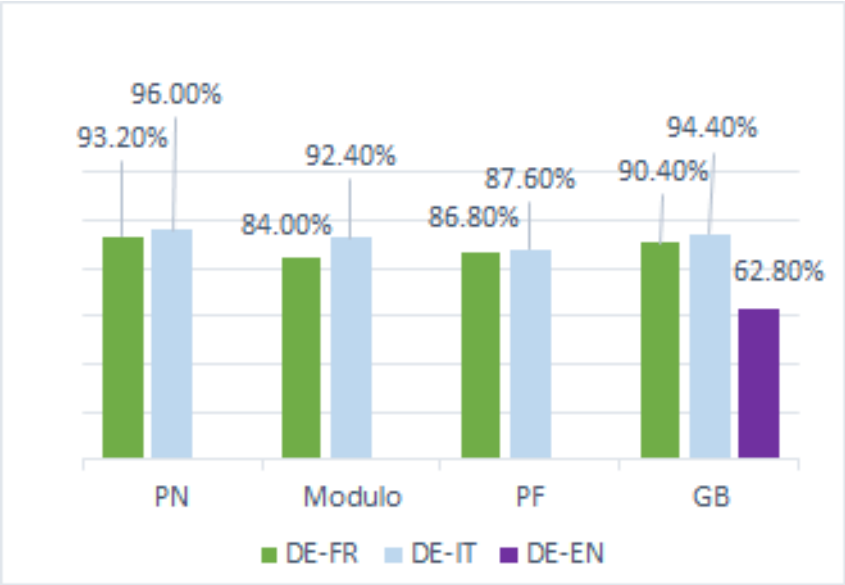

Figura 4.2: Porcentaje de las oraciones de traducción automática usables para post-edición para cada dominio y par de lenguajes.

puntajes de los evaluadores. Coeficiente kappa de light (Light, 1971.) y de Cohen (Cohen, 1968) fueron utilizados para DE-FR/IT y DE-EN respectivamente[8]. Los resultados son mostrados en la Tabla 4.9.

	DE-FR	**DE-IT**	**DE-EN**
PN	0.341	0.549	–
Modulo	0.411	0.547	–
PF	0.412	0.519	–
GB	0.340	0.562	0.430

Tabla 4.9: Resultados de Coeficiente kappa de light (DE-FR/IT) y de Cohen (DE-EN).

En general, los resultados muestran un moderado acuerdo entre los evaluadores con excepción de los dominio *PN* y *GB* para DE-FR, donde el acuerdo es "justo" según (Landis y Koch, 1977). Resultados son más confiables para DE-IT.

Las tablas 4.10, 4.11 y 4.12 muestran los resultados detallados de cada par de lenguajes. Estos resultados confirman que para DE-FR en todos los dominios, alrededor de 20-22% de las oraciones no requerirían ningún tipo de modificación (columna "Yes") y entre 63.3-71.2% podrían ser usadas con alguna pos-edición. Lo que queda por estudiar es el esfuerzo que tendrían que hacer los traductores para post-editar esos segmentos en la columna "YwC" para convertirlos en una traducción final. Só-

[8]Utilizamos kappa de light para DE-FR/IT debido a que fueron tres traductores.

DE-FR			
ratings %	Yes	YwC	No
*PN**	22	71.2	5.2
*Modulo**	20.4	63.6	15.6
PF	22	64.8	13.2
*GB**	20	70.4	9.2

Tabla 4.10: Resultados por dominio. Para el conjunto de evaluación *PN*, *Modulo* y *GB*, hay puntajes que no pudieron contarse, respectivamente, 2%, 0.4% y 0.4% de las oraciones.

DE-IT			
ratings %	Yes	YwC	No
*PN**	32.4	63.6	3.6
Modulo	31.6	60.8	7.6
*PF**	22.8	64.8	12
GB	26.8	67.6	5.6

Tabla 4.11: Detalle de la evaluación por dominio. Muchos de las evaluaciones para *PN* y *PF* no pudieron contarse para 0.4% de las oraciones.

lo 2% de los segmentos fueron calificados en desacuerdo (es decir, recibieron tres calificaciones diferentes).

La tabla 4.11 muestra resultados detallados para DE-IT. El porcentaje de oraciones en la categoría "Sí" fue mayor que en el par de idiomas DE-FR. En particular, el dominio PN tiene el porcentaje más alto de oraciones utilizables sin ninguna modificación ("Sí") y el porcentaje más bajo de oraciones no utilizables ("No").

Para el par de idiomas DE-EN, las frases se pueden utilizar principalmente con algunos cambios. También se vió un porcentaje muy parecido de "Yes" y "No". Sin embargo, cabe destacar que el 28% de las oraciones no pudieron ser contadas dado que sólo dos evaluadores de EN participaron en la tarea. Además la calificación de las oraciones se convirtió en un juicio casi unánime, es decir ambos debían calificar igual una oración para que se pudiera contar. Por lo tanto, no pudimos evaluar si ese tercio de las frases podrían utilizarse para la post edición. Es por ello que, en la Tabla 4.12, se reporta juicios mínimos, es decir, contamos cada vez en que cada categoría nominal recibió una puntuación. Al añadir las sentencias que faltan al recuento, se rechazan más sentencias y se aceptan menos sentencias sin ningún cambio. Sin embargo, en este caso en particular, necesitaríamos más análisis para confirmar si

DE-EN				
ratings %		Yes	YwC	No
GB	min.	14.8%	67.6%	17.6%
	max.*	9.2%	53.6%	9.2%

Tabla 4.12: Detalle de evaluación para *GB* en DE-EN tanto mínimo como máximo. Debido a que solamente dos evaluadores fueron involucrados en esta tarea, fue difícil el acuerdo. Falto el 28% de las oraciones.

DE-EN produce TA menos adecuada para la post edición, o si los evaluadores están menos inclinados a utilizar la salida de TA.

Evaluación tanto humana como automática confirmaron que para "manual de procedimientos" (*PN*) es el mejor dominio a implementar para cada par de lenguaje. Por ello, nos enfocamos en este dominio para ver que influye en los juicios subjetivos para el uso del sistema, y en un futuro estudio si hay correlación con alguna variable (tamaño de la oración, calidad de las memorias de traducción).

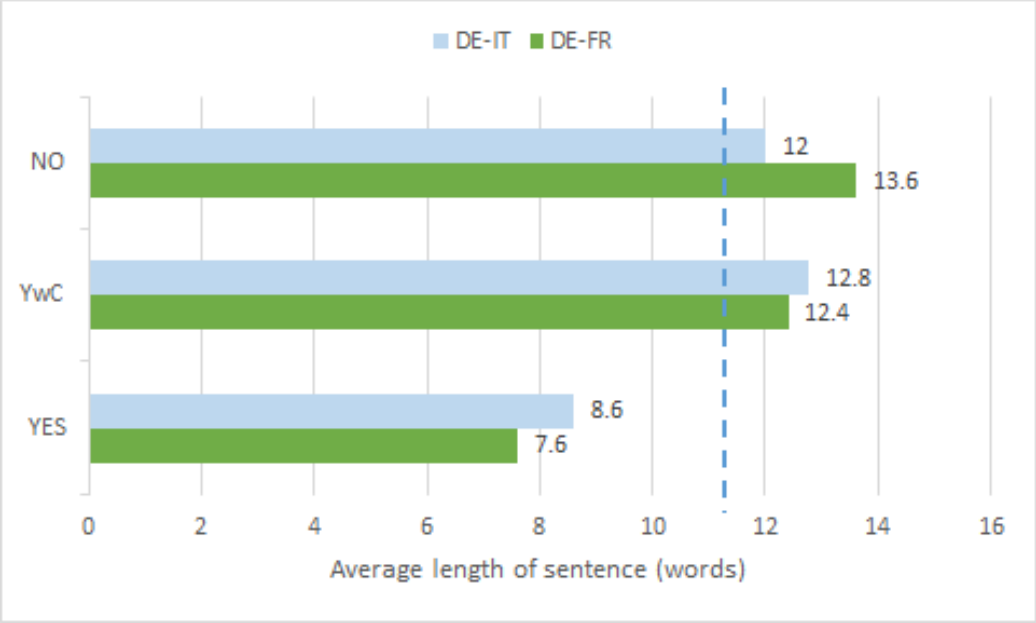

Figura 4.3: Promedio de longitud de oraciones de lenguaje origen que han sido evaluadas por traductores de IT y FR para el dominio *PN*. La línea azul es el promedio, es decir 11.37 palabras.

Como se muestra en la Figura 4.3, traducciones que se evaluaron "usables sin modificación" ("Yes") claramente son más cortas que el promedio de las oraciones en el corpus, mientras que las "oraciones no usables" ("No") son comúnmente más largas. La mayoría de las oraciones seleccionadas como "YwC", la categoría más elegida, generalmente son más largas que el promedio de las oraciones.

par lenguaje	métrica	**Yes**	**YwC**	**No**
DE-FR	*%*	22	71.2	5.2
	WER	20.40	37.34	65.16
	Kappa	0.462	0.282	0.235
DE-IT	*%*	32.4	63.6	3.6
	WER	22.99	42.68	71.29
	Kappa	0.64	0.514	0.339

Tabla 4.13: Detalles de los puntajes para *PN* comparando la evaluación humana (%) con la métrica WER y el coeficiente kappa de Light (k) en el sub conjunto de evaluación donde hubo acuerdo entre los evaluadores.

En la tabla 4.13 podemos ver que las puntuaciones WER también varían en función de la idoneidad y que la kappa de Light, calculada para cada categoría es inversamente proporcional a WER ("Yes" > "YwC" > "No").

Finalmente, descubrimos que el total de oraciones que se superponen en cada categoría para IT y FR es entre el 42%(IT) y el 62%(FR) para "Yes", frente a solamente entre 31%(FR) y 44%(IT) para "No". Estos últimos resultados son alentadores: confirman que los juicios subjetivos pueden estar relacionados a factores objetivos y en ese caso, en general, las oraciones con "Yes" son bastante confiables, mientras que para "No" y "YwC" pareciera que dependen tanto del lenguaje, elección de traductores y opiniones personales.

Capítulo 5

Conclusiones

Hemos presentado resultados preliminares del proyecto que tiene como objetivo integrar traducción automática dentro del flujo de trabajo en el servicio de lenguajes del correo suizo.

La primera parte del estudio consistió en elegir entre dos sistemas estadísticos: Microsoft Translator Hub, un sistema comercial pago que se utiliza a través de un servicio, y Moses, uno de código abierto, ambos entrenados con los múltiples materiales del servicio de lenguaje. Hemos optado por el último, entrenado con el dominio *MTM*, no sólo por las métricas automáticas en la mayoría de dominios, si no también por la anonimización. Esto nos permitió utilizar un sólo sistema por cada par de lenguaje, sin diferenciar cada dominio específico.

A continuación, una evaluación humana, realizada por un conjunto de traductores profesionales del servicio del correo suizo, fue llevada a cabo para determinar el porcentaje de oraciones que son percibidas aptas para post edición profesional en el sistema TA. En general, los resultados de esta evaluación fueron mejores para el dominio "manual de procesos" *PN*.

Las evaluaciones en el par DE-IT obtuvieron los mejores resultados con respecto a las oraciones usables (con o sin cambios) y mayor porcentaje en acuerdo entre los traductores. Sin embargo, en algunos casos encontramos contradicciones en los resultados humanos frente a resultados automáticos, tales como en el par DE-EN. Esta inconsistencia puede ser causada por los traductores que intervinieron en la evaluación de DE-EN, ya que la decisión era casi unánime (eran dos), sin contar el probable rechazo prejuzgado hacía el sistema.

Otra inconsistencia fue que en *GB* fue el peor dominio en relación con las mé-

tricas automáticas pero fue el segundo mejor en la evaluación humana. Queda por investigar esta inconsistencia entre las métricas automáticas y la evaluación humana.

En definitiva, consideramos nuestros resultados satisfactorios: el porcentaje de oraciones usables varían entre 85% (DE-FR) a 96% (DE-IT) lo cual es un buen umbral para comenzar a trabajar con TA en un contexto profesional. En cuanto a DE-EN, el porcentaje fue 62.80% por lo que el sistema podría ser usado pero con menos alcance. Un trabajo futuro sería investigar en esta dirección.

Este trabajo ha sido presentado en la European Conference for Machine Transaltion 2018 (Bouillon y cols., 2018).

5.1 Trabajo futuro

Actualmente estamos trabajando en replicar el experimento utilizando modelos neuronales y preparando una evaluación de productividad con lingüistas del Correo.

En las siguiente fase, llevaremos a cabo una evaluación de productividad en los traductores, con el fin de determinar si la implementación de TA en el servicio de lenguaje puede ser beneficioso (reducir costos).

Estas evaluaciones primero involucraran el dominio con las mayores puntuaciones *PN*, debido a que creemos necesario introducir cambios suaves en el flujo de trabajo para así no generar rechazo prejuzgado. Finalmente, una vez que los traductores estén familiarizados con el flujo de trabajo, nos gustaría hacer una evaluación comparativa de la traducción por frases frente a la traducción neuronal (el nuevo estado del arte), en el que está actualmente entrenándose. Esto permitirá comparar la productividad de los traductores y la satisfacción a la hora de utilizar diferentes arquitecturas TA.

Referencias

Andrew Joscelyne, J. v. d. M. A. R. A.-M. v. d. M., Anna Samiotou. (2017). *Taus machine translation market report.* Keizersgracht 74, 1015 CT: Translation Automation User Society (TAUS).

Arnold, D. (2003). Why translation is difficult for computers. En (first ed., Vol. xvi, p. 119-142). Amsterdam.: John Benjamins Publishing Company.

Bahdanau, D., Cho, K., y Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, *abs/1409.0473*. Descargado de `http://arxiv.org/abs/1409.0473`

Bouillon, P., Estrella, P., Girletti, S., Mutal, J., Bellodi, M., y Bircher, B. (2018). Integrating mt at swiss post's language service: preliminary results. En (p. 281-286). Proceedings of the 21st Annual Conference of the European Association for Machine Translation.

Britz, D., Goldie, A., Luong, M., y Le, Q. V. (2017). Massive exploration of neural machine translation architectures. *CoRR*, *abs/1703.03906*. Descargado de `http://arxiv.org/abs/1703.03906`

Brown, P. F., Cocke, J., Della-Pietra, S. A., Della-Pietra, V. J., Jelinek, F., Mercer, R. L., y Rossin, P. (1988). A statistical approach to language translation. En *Proceedings of the international conference on computational linguistics (coling).*

Brown, P. F., Cocke, J., Pietra, S. A. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., ... Roossin, P. S. (1990, junio). A statistical approach to machine translation. *Comput. Linguist.*, *16*(2), 79–85. Descargado de `http://dl.acm.org/citation.cfm?id=92858.92860`

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., y Mercer, R. L. (1993, junio). The mathematics of statistical machine translation: Parameter estimation. *Comput. Linguist.*, *19*(2), 263–311. Descargado de `http://dl.acm.org/citation.cfm?id=972470.972474`

Campbell, D., Dunne, R. A., y Campbell, N. A. (1997). *On the pairing of the softmax activation and cross–entropy penalty functions and the derivation of the softmax activation function.*

Cho, K., van Merrienboer, B., Bahdanau, D., y Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *CoRR*, *abs/1409.1259*. Descargado de `http://arxiv.org/abs/1409.1259`

Cohen, J. (1968, 11). Weighted kappa - nominal scale agreement with provision for scaled disagreement or partial credit. , *70*, 213-20.

contributors, W. (2018a). *Dependent-marking language — Wikipedia, the free encyclopedia.*

https://en.wikipedia.org/w/index.php?title=Dependent-marking_language&oldid=848090734.

contributors, W. (2018b). *Head-marking language — Wikipedia, the free encyclopedia.* https://en.wikipedia.org/w/index.php?title=Head-marking_language&oldid=841230457.

Dorr, B. J., Hovy, E. H., y Levin, L. S. (2005). *1 natural language processing and machine translation encyclopedia of language and linguistics, 2nd ed. (ell2). machine translation: Interlingual methods.*

E. Saal, F., Downey, R., y A. Lahey, M. (1980, 09). Rating the ratings: Assessing the psychometric quality of rating data. *Psychological Bulletin*, *88*, 413-428.

European association for machine translation. (s.f.). http://www.eamt.org/.

Federico, M., Bertoldi, N., y Cettolo, M. (2008, 01). Irstlm: An open source toolkit for handling large scale language models. *Proceedings of Interspeech*, 1618-1621.

Gao, Q., y Vogel, S. (2008, 01). Parallel implementations of word alignment tool. *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. En *Proceedings of the sixth workshop on statistical machine translation* (pp. 187–197).

Hochreiter, S., y Schmidhuber, J. (1997, 12). Long short-term memory. *Neural computation*, *9*, 1735-80.

Hutchins, J. (2005, 01). Towards a definition of example-based machine translation. *http://www.hutchinsweb.me.uk/MTS-2005.pdf*.

Jurafsky, D., y Martin, J. H. (2008). *Speech and language processing (2nd edition)*. Prentice Hall.

Kalchbrenner, N., y Blunsom, P. (2013, 01). Recurrent continuous translation models. *EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, *3*, 1700-1709.

Klein, G., Kim, Y., Deng, Y., Senellart, J., y Rush, A. M. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., ... Herbst, E. (2007, June). Moses: Open source toolkit for statistical machine translation. En *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions* (pp. 177–180). Prague, Czech Republic: Association for Computational Linguistics. Descargado de http://www.aclweb.org/anthology/P/P07/P07-2045

Koehn, P., y Monz, C. (2005, June). Shared task: Statistical machine translation between European languages. En *Proceedings of the acl workshop on building and using parallel texts* (pp. 119–124). Ann Arbor, Michigan: Association for Computational Linguistics. Descargado de http://www.aclweb.org/anthology/W/W05/W05-0820

Landis, J. R., y Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, JSTOR.*, 159–174.

Levenshtein, V. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals.

Soviet Physics Doklady, *10*, 707.

Li, C., y Tam, P. (1998, 06). An iterative algorithm for minimum cross entropy thresholding. , *19*, 771–776.

Light, R. (1971.). Measures of response agreement for qualitative data: Some generalizations and alternatives. *Psychological bulletin*, *76*(5).

Luong, M., Pham, H., y Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *CoRR*, *abs/1508.04025*. Descargado de `http://arxiv.org/abs/1508.04025`

Nagao, M. (1984). A framework of a mechanical translation between japanese and english by analogy principle. En *A.elithorn and r.banerji (eds.) artificial and human intelligence* (p. 173-180). (Amsterdam: North-Holland).

Nichols, J. (1986, 03). Head-marking and dependent-marking grammar. *Language*, *62*, 56-119.

Och, F. J., y Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, *29*(1), 19–51.

O'Brien, M. (2018). *Toward leveraging gherkin controlled natural language and machine translation for global product information development.*

Papineni, K., Roukos, S., Ward, T., y Zhu, W.-J. (2001, September 17). *BLEU: a method for automatic evaluation of machine translation* (Inf. Téc. no RC22176(W0109-022)). IBM Research Report. Descargado de `http://www.mt-archive.info/IBM-2001-Papineni.pdf`

Parcheta, Z., Sanchis-Trilles, G., Rudak, A., y Bratchenia, S. (2018). *Implementing a neural machine translation engine for mobile devices: the lingvanex use case.*

Prechelt, L. (1998, junio). Automatic early stopping using cross validation: Quantifying the criteria. *Neural Netw.*, *11*(4), 761–767. Descargado de `http://dx.doi.org/10.1016/S0893-6080(98)00010-0` doi: 10.1016/S0893-6080(98)00010-0

Schmidt, T., y Marg, L. (2018). *How to move to neural machine translation for enterprise-scale programs—an early adoption case study.*

Sennrich, R., Haddow, B., y Birch, A. (2015). Neural machine translation of rare words with subword units. *CoRR*, *abs/1508.07909*. Descargado de `http://arxiv.org/abs/1508.07909`

Skadiņš, R., Tiedemann, J., Rozis, R., y Deksne, D. (2014, May). Billions of parallel words for free: Building and using the eu bookshop corpus. En *Proceedings of the 9th international conference on language resources and evaluation (lrec-2014).* Reykjavik, Iceland: European Language Resources Association (ELRA).

Somers, H. (1999). *eview article: Example-based machine translation.* Springer.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., y Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*, 1929-1958. Descargado de `http://jmlr.org/papers/v15/srivastava14a.html`

Stolcke, A. (2004, 07). Srilm — an extensible language modeling toolkit. *SRI International*, *2*.

Sutskever, I., Vinyals, O., y Le, Q. V. (2014). Sequence to sequence learning with neural networks.

CoRR, *abs/1409.3215*. Descargado de http://arxiv.org/abs/1409.3215

Tillmann, C., y Ney, H. (2003, marzo). Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Comput. Linguist.*, *29*(1), 97–133. Descargado de http://dx.doi.org/10.1162/089120103321337458 doi: 10.1162/089120103321337458

Vauquois, B. (1968). A survey of formal grammars and algorithms for recognition and transformation in machine translation. En *IFIP Congress-68* (pp. 254–260). Edinburgh.

Vogel, S., Ney, H., y Tillmann, C. (1996). Hmm-based word alignment in statistical translation. En *Proceedings of the 16th conference on computational linguistics - volume 2* (pp. 836–841). Stroudsburg, PA, USA: Association for Computational Linguistics. Descargado de https://doi.org/10.3115/993268.993313 doi: 10.3115/993268.993313

Wang, L., Gordon, M., y Zhu, J. (2007, 01). Regularized least absolute deviations regression and an efficient algorithm for parameter tuning. , 690 - 700.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., ... Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, *abs/1609.08144*. Descargado de http://arxiv.org/abs/1609.08144

"Los abajo firmantes, miembros del Tribunal de Evaluación de tesis, damos Fe que el presente ejemplar impreso, se corresponde con el aprobado por éste Tribunal"