UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Departamento de Sistemas Informáticos y Computación
Grupo de Reconocimiento de Formas y Tecnologías del Lenguaje Humano

TRABAJO FIN DE MÁSTER IARFID:

Aprendizaje online de los pesos del modelo log-lineal en traducción automática interactiva

Autor:
Francisco Javier López Salcedo

Revisores:
Germán Sanchis-Trilles
Francisco Casacuberta

7 de septiembre de 2012

Índice general

Agradecimientos

Me gustaría agradecer la ayuda de todas aquellas personas que han hecho posible realizar este trabajo fin de máster.

Para empezar querría agradecer al doctor Francisco Casacuberta la oportunidad que me ha dado de colaborar con el grupo de investigación PRHLT, así como la confianza depositada en mi para realizar este trabajo en un área de tanto interés como es la traducción automática estadística.

También me gustaría agradecer la ayuda y el apoyo recibido por parte de todo el laboratorio 1L05, especialmente a Guillem Gascó, Mauricio Maca, Martha Alicia Rocha y sobretodo al doctor Germán Sanchis con los cuales he tenido la oportunidad de compartir el día a día. Germán ha sido para mi una persona imprescindible, siendo mi guía y mi mentor a largo de todo el proceso de investigación. De él he aprendido algo cada día, desde las distintas herramientas disponibles para llevar a cabo los diversos experimentos hasta los conocimientos necesarios para saber como utilizarlas. Además, gracias a él he aprendido infinidad de conceptos sobre traducción automática estadística que me han ayudado a familiarizarme e interesarme más en este campo del área de reconocimiento de formas. Germán también me ha enseñado a superar las dificultades intrínsecas de la investigación, animándome a perseverar en el trabajo y a no rendirme ante un experimento fallido. A él, tampoco puedo dejar de agradecerle la paciencia y comprensión que ha tenido conmigo durante el desarrollo de todo este trabajo.

Para concluir, quisiera darle las gracias a Irene por haber confiado en mi desde el principio, así como también por su paciencia, por su apoyo y por su cariño en los momentos más difíciles.

Resumen

En muchas ocasiones las traducciones que un sistema de traducción automática genera no tienen la calidad deseada, por lo que es necesaria la intervención de expertos traductores humanos para corregir los posibles errores que estas puedan albergar, mejorando de esta forma la calidad de las mismas. Esta metodología, conocida como Post-Edición (PE), está siendo cada vez más utilizada por traductores humanos y está considerada como parte del estado del arte.

Debido al coste que supone aplicar dicha metodología, en este trabajo fin de máster (TFM) se plantea un esquema alternativo, denominado traducción automática interactiva (IMT, de sus siglas en inglés, Interactive Machine Translation), capaz de reducir el esfuerzo necesario por parte de un humano en el proceso de corrección.

Siguiendo este nuevo esquema, en combinación con una aproximación encargada de adecuar los pesos del modelo log-lineal a cada una de las traducciones propuestas mediante un algoritmo de aprendizaje online, se consigue que el sistema aprenda de los errores que el traductor humano ya ha corregido, favoreciendo a su vez la corrección de los próximos errores.

Para abordar esta problemática se han empleado tres estrategias diferentes. En primer lugar se plantea el uso del algoritmo de aprendizaje online denominado Regresión de Arista Discriminativa ya utilizado en post-edición con buenos resultados. Posteriormente se han utilizado dos estrategias más avanzadas, una en la que se realiza una primera y sencilla aproximación para adaptar $\boldsymbol{\lambda}$ dentro de un escenario IMT y por último una nueva formulación del algoritmo DRR exclusiva para trabajar con IMT.

Como se puede observar a lo largo de este TFM, esta nueva metodología genera diversos resultados con éxitos dispares, en donde el uso de la nueva formulación del algoritmo DRR ofrece resultados alentadores, abriendo un nuevo camino a explorar con la esperanza de obtener la mayor calidad posible en las traducciones mediante el menor esfuerzo por parte del traductor humano.

Descripción

Este trabajo fin de máster está estructurado en cuatro capítulos, los cuales relatan los aspectos más importantes de este trabajo.

El primer capítulo es una introducción a la traducción automática y al estado del arte de esta, más concretamente a la traducción automática estadística y a su vertiente traducción automática estadística interactiva (comúnmente denominada IMT por sus siglas en inglés, Interactive Machine Translation), la cual es la base de este trabajo. Por contra, en los tres siguientes capítulos se explican los experimentos realizados y las conclusiones obtenidas. En primer lugar se describen las estrategias seguidas para intentar mejorar el estado del arte actual a través de la adaptación de los pesos del modelo log-lineal mediante tres algoritmos. En segundo lugar los resultados obtenidos mediante estas tres estrategias de adaptación online. Por último, en el cuarto capitulo, se detallan las conclusiones obtenidas una vez evaluados los resultados de los experimentos y las futuras posibles mejoras y ampliaciones del trabajo realizado.

Capítulo 1

Introducción

1.1. Traducción automática

Según el estudio publicado en [Lewis, 2009] existen alrededor de 6909 lenguas vivas, lo que provoca en muchos casos la necesidad de mecanismos que faciliten la comunicación a través de una lengua común. Debido a esto, en los últimos años ha habido un importante crecimiento en la necesidad de traducir documentos, tanto por parte de instituciones oficiales como las Naciones Unidad o la Unión Europea, como del sector privado, traduciendo manuales, folletos, guías, páginas web o artículos. Una de las soluciones más empleadas son los traductores humanos, ya que en su gran mayoría son capaces de traducir un texto determinado en función del contexto con un gran resultado. A pesar de estas ventajas, los traductores humanos son costosos, requieren en muchos casos de una gran cantidad de tiempo para realizar las traducciones y son un recurso limitado. Por este motivo ha sido necesario automatizar el proceso introduciendo traductores automáticos. Debido a esto, durante los últimos años ha habido un gran interés por desarrollar traductores automáticos cada vez más precisos empleando diferentes aproximaciones con distinto éxito, pero todas ellas reportando un gran beneficio a la investigación.

La primera de estas aproximaciones fue denominada *traducción palabra por palabra* [Weaver, 1955], la cual realiza una traducción palabra a palabra para cada una de las palabras que aparece en un texto. Aunque esta aproximación es la mas sencilla e intuitiva de todas, no por ello está libre de inconvenientes, ya que provoca un mal reordenamiento de las palabras dentro de la frase traducida, no tiene en cuenta el contexto de la palabra a traducir, y genera errores en las perífrasis y en palabras que no tienen traducción en el idioma destino.

Con el objetivo de mejorar el reordenamiento dentro de las traducciones se propuso la *transferencia sintáctica* [Chomsky, 1956], que realiza un análisis previo de la oración a traducir y mediante reglas gramaticales voltea las ramas y las hojas, de forma que las palabras de la frase traducida aparezcan en el orden correcto. Esta técnica es la precursora del reordenamiento sintáctico utilizado en algunos sistemas actuales.

Mediante los modelos lógicos también se intentó definir una representación formal de

la lengua denominada *interlingua* [Vauquois, 1968], cuyo objetivo era crear un idioma intermedio a partir de varios, logrando de esta forma facilitar la comunicación entre los hablantes de los diferentes idiomas. En la práctica, la definición de una idioma intermedio está limitado tanto por el alto coste que provoca como por su inherente ambigüedad.

Hay ocasiones donde el texto a traducir pertenece a un ámbito muy limitado como pueden ser el caso de los manuales técnicos. En algunos casos donde nos podemos limitar a usar *lenguaje controlado* [Pym, 1990], en donde las palabras tienen un significado concreto y algunos tipos de oraciones no están permitidos, podemos utilizar *transferencia sintáctica* o *interlingua* con éxito. Por el contrario estas técnicas no tienen tanto éxito en contextos más amplios, no pudiendo cubrir la mayoría de necesidades diarias de traducción de textos que aparecen.

En las técnicas anteriores no se requería de una base de datos con ejemplos de oraciones bilingües, en donde se tiene la oración tanto en el idioma origen como en el idioma destino al que queremos traducir, pero con la aparición de corpus que contienen este tipo de oraciones, fue posible emplear otras metodologías en traducción automática, a la cual nos referiremos a lo largo de este trabajo fin de máster con sus siglas en inglés MT, *Machine Translation*. Bajo estas circunstancias apareció la *traducción basada en ejemplos* [Nagao, 1984], mediante la cual, dado un conjunto de ejemplo de un corpus bilingüe se buscan a partir de estos los segmentos adecuados para realizar la traducción por analogía, y a continuación se lleva a cabo la reordenación correspondiente. Esta aproximación se basa en la idea de que gran parte de las personas que no dominan completamente un idioma, al realizar una traducción, no realizan un análisis lingüístico profundo de la oración a traducir, sino que segmentan la oración en primer lugar y traducen posteriormente estos segmentos de forma que una vez estén traducidos se proceda a recomponer la traducción. La principal desventaja de la traducción basada en ejemplos es su complejidad para traducir oraciones completamente, aunque, por el contrario, sí permite traducir sus constituyentes.

Otra aproximación a la MT donde también se utilizan corpus bilingües es la traducción automática estadística (a la que nos referiremos a lo largo de este trabajo como *Statistical Machine Translation*, utilizando sus siglas SMT) [Brown et al., 1993], donde los sistemas basados en traducción automática estadística dependen en gran medida de la calidad de estos corpus, los cuales en la actualidad pueden encontrarse en múltiples idiomas y diversos tamaños. SMT emplea modelos matemáticos para describir el proceso de traducción de una forma precisa para posteriormente estimar las probabilidades de traducción y reordenamiento de forma automática a partir de los pares bilingües existentes en los datos de entrenamiento.

Los sistemas SMT en los que se basa el estado del arte tienen un gran potencial, aunque a día de hoy son capaces de proporcionar traducciones automáticas que puedan ser utilizadas directamente en aplicaciones del mundo real, como por ejemplo en la compra de billetes de avión o tren por internet, estas traducciones no pueden ser utilizadas cuando se requiere una alta calidad de las mismas, como por ejemplo ocurre en labores diplomática, ya que en la mayoría de casos es necesario un postproceso de las mismas. Debido a esto, muchos investigadores están trabajando en mejorar el estado del arte de la SMT.

1.2. Traducción automática estadística

Hay muchas alternativas para abordar un problema de traducción automática, siendo en la actualidad la traducción automática estadística una de las más empleadas. Mientras que años atrás los sistemas de traducción automáticas basados en reglas ofrecían grandes resultados, actualmente se han visto relegados en muchos casos en favor de los sistemas SMT [Callison-Burch et al., 2011], ya que estos ofrecen más flexibilidad a la hora de adaptarse a nuevos problemas. La aproximación del reconocimiento de formas a la traducción automática fue planteada en [Brown et al., 1993], en donde dada una oración a traducir $\mathbf{x} = x_1...x_j...x_{|x|}$, en un idioma origen, se pretende encontrar la oración $\hat{\mathbf{y}} = y_1...y_i...y_{|y|}$, en un idioma destino que maximice la probabilidad a posteriori, representada en la siguiente ecuación.

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}}\, Pr(\mathbf{y}|\mathbf{x}). \tag{1.1}$$

Este planteamiento requiere encontrar la oración $\hat{\mathbf{y}}$ con mayor probabilidad, dada una oración de entrada $\mathbf{x}$, por lo que previamente necesitaremos calcular la probabilidad conjunta para todos los pares de oraciones ($\mathbf{x}$,$\mathbf{y}$). Este cálculo de probabilidad es imposible ya que no disponemos de tal cantidad de corpus bilingües, y por lo tanto debemos emplear la regla de Bayes para abordar el problema:

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}}\, \frac{Pr(\mathbf{y}) \cdot Pr(\mathbf{x}|\mathbf{y})}{Pr(\mathbf{x})}. \tag{1.2}$$

Ya que la maximización es en función de $\mathbf{y}$ podemos omitir el denominador, obteniendo la siguiente fórmula:

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}}\, Pr(\mathbf{y}) \cdot Pr(\mathbf{x}|\mathbf{y}). \tag{1.3}$$

Al emplear la regla de Bayes hemos descompuesto $Pr(\mathbf{y}|\mathbf{x})$ en dos probabilidades diferentes, estimadas mediante un *modelo de lenguaje estadístico* para el idioma destino $Pr(\mathbf{y})$, y el *modelo de traducción inverso* $Pr(\mathbf{x}|\mathbf{y})$.

Mientras que el modelo de traducción $Pr(\mathbf{x}|\mathbf{y})$ capturará la relación entre palabras o segmentos de palabras entre el idioma desde el que estamos traduciendo hasta el idioma destino, el modelo de lenguaje $Pr(\mathbf{y})$, construido a partir de un corpus monolingüe, se asegurará de que la oración resultante de traducir la oración de entrada $\mathbf{x}$ esté bien formada según el idioma al que estamos traduciendo.

El modelo de lenguaje normalmente está basado en n-gramas [Shannon, 1948]. Un n-grama es una subcadena de n elementos, habitualmente palabras, en donde si $n = 1$ hablaríamos de unigramas, si $n = 2$ de bigramas, si $n = 3$ de trigramas, etc. Formalmente el modelo de n-gramas se define como:

$$Pr(\mathbf{w}) = \prod_{i=1}^{|\mathbf{w}|} Pr(w_i|w_1^{i-1}) \approx Pr(\mathbf{w}) = \prod_{i=1}^{|\mathbf{w}|} p_n(w_i|w_{i-n+1}^{i-1}), \tag{1.4}$$

en donde $\mathbf{w}$ representa la oración, $|\mathbf{w}|$ representa la talla de la oración $\mathbf{w}$, w_i representa la palabra i-esima del n-grama, y w_1^{i-1} y w_{i-n+1}^{i-1} son secuencia de palabras dentro del n-grama. La gran mayoría de sistemas de SMT utilizan 5-gramaspara definir el modelo de lenguaje.

Con el objetivo de modelar adecuadamente la probabilidad $Pr(\mathbf{x}|\mathbf{y})$ se propusieron varios modelos. En [Brown et al., 1993] se definieron los modelos de *alineamiento* de palabras conocidos como modelos de IBM, donde la correspondencia entre la oración origen y su traducción, se establece mediante variables de alineamiento ocultas $\mathbf{a} = a_1...a_i...a_{|y|}$, las cuales fueron definidas en función de todas las palabras de la oración en el idioma de destino. A cada palabra de la oración a traducir se le asigna una palabra como traducción. Pueden aparecer casos donde una palabra de la oración de entrada no tenga ninguna palabra en la oración de salida que represente su traducción y en estos casos no es posible realizar el alineamiento. Para solucionar estos problemas se introdujo la posición artificial cero o NULL. Los modelos de IBM indican como calcular $Pr(\mathbf{y}|\mathbf{x})$ mediante la ecuación

$$Pr(\mathbf{y}|\mathbf{x}) = \sum_{a \in \mathcal{A}(\mathbf{x},\mathbf{y})} p(\mathbf{y}, \mathbf{a}|\mathbf{x}), \tag{1.5}$$

en donde se define $\mathcal{A}(\mathbf{x}, \mathbf{y})$ como el conjunto de todos los posibles alineamientos entre $\mathbf{x}$ e $\mathbf{y}$.

En la práctica el modelado directo $p(\mathbf{y}|\mathbf{x})$ de la probabilidad a posteriori $Pr(\mathbf{y}|\mathbf{x})$ ha sido ampliamente adoptado y con este propósito varios autores [Papineni et al., 1998] [Och and Ney, 2002] proponen el uso del llamado modelo log-lineal para combinar los diferentes modelos de traducción, distorsión o reordenamiento y de lenguaje. Por tanto el modelo log-lineal se definiría de la forma:

$$p(\mathbf{y}|\mathbf{x}) = \frac{\exp \sum_{m=1}^{M} \lambda_m h_m(\mathbf{x}, \mathbf{y})}{\sum_{\mathbf{y}'} \exp \sum_{m=1}^{M} \lambda_m h_m(\mathbf{x}, \mathbf{y}')}, \tag{1.6}$$

donde la regla de decisión viene dada por la expresión

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}} \sum_{m=1}^{M} \lambda_m h_m(\mathbf{x}, \mathbf{y}). \tag{1.7}$$

Aquí, $h_m(\mathbf{x}, \mathbf{y})$ es la puntuación (comúnmente conocida en inglés como *score*) de la función que representa una característica importante en la traducción de $\mathbf{x}$ a $\mathbf{y}$, M es el número de modelos o características y λ_m representan un peso de la combinación log-linear. Por tanto, la ecuación 1.1 se puede ver como un caso especial de la ecuación 1.3, donde tanto $Pr(\mathbf{x}|\mathbf{y})$ como $Pr(\mathbf{y})$ son características importantes donde habría dos λ_m con valor 1. Por norma general, la función de características $h_m(\mathbf{x}, \mathbf{y})$ representa a un modelo. El propósito de los factores de escalado (comúnmente denominados con su traducción en ingles *scaling factors*) λ_m es ajustar el poder discriminante de su correspondiente función de características (comúnmente denominados por su traducción en ingles *feature functions*) $h_m(\mathbf{x}, \mathbf{y})$, de forma que cuanto mayor sea el valor de λ_m más influencia tendrá $h_m(\mathbf{x}, \mathbf{y})$ en la la decisión de la ecuación 1.7.

La ecuación 1.7 se puede expresar de una forma más compacta mediante su forma vectorial utilizando el producto interno

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}}\, \boldsymbol{\lambda} \cdot \mathbf{h}(\mathbf{x}, \mathbf{y}) = \underset{\mathbf{y}}{\operatorname{argmax}}\, g(\mathbf{x}, \mathbf{y}), \tag{1.8}$$

en donde $g(\mathbf{x}, \mathbf{y})$ representa la puntuación de la hipótesis $\mathbf{y}$ dada una oración de entrada $\mathbf{x}$. Tanto en este caso como en la ecuación 1.7, $g(\mathbf{x}, \mathbf{y})$ no representa una probabilidad ya que se ha omitido el término de normalización. Mientras que $\mathbf{h}(\cdot|\cdot)$ suelen obtenerse empleando el conjunto de entrenamiento, los valores $\boldsymbol{\lambda}$ son ajustados a partir del conjunto de desarrollo. Este corpus, conocido como *desarrollo*, *development* por su traducción en inglés o *tuning*, es generalmente de un tamaño inferior al de entrenamiento y está compuesto por pares de oraciones bilingües en los mimos idiomas que el corpus de entrenamiento.

1.3. Traducción automática estadística basada en secuencias de palabras

A pesar de que los modelos de traducción palabra por palabra parecen razonables, estos modelos no tienen en cuenta el contexto de las palabras a traducir. Por ello, con el objetivo de obtener la información del contexto, se introdujeron los modelos basados en segmentos o secuencias de palabras [Tomás and Casacuberta, 2001, Marcu and Wong, 2002, Zens et al., 2002, Zens and Ney, 2004, Koehn et al., 2003], comúnmente denominados *phrase-based (PB) models*, mejorando el rendimiento de los modelos basados en una palabra. Actualmente los modelos PB forman parte del estado del arte [Koehn and Monz, 2006, Callison-Burch et al., 2007, Fordyce, 2007] de la traducción automática y por tanto han sido empleados en este trabajo fin de máster.

A diferencia de los *modelos basados en palabras*, que emplean como unidad básica en la traducción una única palabra, los *modelos basados en secuencias de palabras* segmentan la oración de entrada $\mathbf{x}$ en bloques de secuencias de palabras, comúnmente llamados segmentos o *phrases*, lo que facilita la inclusión de información de contexto de una forma natural. Estos modelos aprenden la probabilidad de que una secuencia contigua de palabras de entrada, $\tilde{x}_k \in \mathbf{x}$, se traduzca por otra secuencia de palabras de salida $\tilde{y}_k \in \mathbf{y}$ de forma que el último paso sea reordenar estos segmentos de salida para obtener la oración $\mathbf{y}$ como traducción a la frase de entrada dada. Por tanto, en este caso los diccionarios estadísticos de pares de palabra se substituyen por diccionarios estadísticos de pares de segmentos bilingües. Al incluir el modelo basado en segmentos en el modelo log-lineal se mejora claramente la calidad de un SMT.

1.3.1. El modelo

La derivación de los modelos PB proviene del concepto de segmentación bilingüe, es decir, segmentar las oraciones de origen y destino en secuencias de palabras. En esta derivación sólo se consideran segmentos de palabras contiguas y no puede haber solapamiento entre ellos. Por tanto, el número de segmentos de la oración origen y el de la oración destino deben ser iguales, llamado K, y cada segmento de origen se alinea únicamente con un segmento de destino y viceversa.

Siendo J e I las longitudes de $\mathbf{x}$ e $\mathbf{y}$ respectivamente, definiremos la segmentación de la oración origen:

$$\gamma : \{1, \ldots, K\} \rightarrow \{1, \ldots, J\} : \gamma_k \geqslant \gamma_{k-1}\ 1 < k\ \leqslant K\ \&\ \gamma_k = J\ (\gamma_0 = 0),$$

y la segmentación de la oración destino:

$$\mu : \{1, \ldots, K\} \rightarrow \{1, \ldots, I\} : \mu_k \geqslant \mu_{k-1}\ 1 < k\ \leqslant K\ \&\ \mu_k = I\ (\mu_0 = 0).$$

Por tanto, la alineación de los segmentos de la oración $\mathbf{x}$ e $\mathbf{y}$, quedaría definida como:

$$\alpha : \{1, \ldots, K\} \rightarrow \{1, \ldots, K\} : \alpha(k) = \alpha(k')\ si\ k = k'.$$

Asumiendo que todas las posibles segmentaciones de $\mathbf{x}$ en K segmentos y todas las posibles segmentaciones de $\mathbf{y}$ en K segmentos tienen la misma probabilidad independiente de K, podemos definir $p(\mathbf{x}|\mathbf{y})$ como:

$$p(\mathbf{x}|\mathbf{y}) = p(J|I) \cdot \sum_{K} \sum_{\mu_1^K} \sum_{\gamma_1^K} \sum_{\alpha_1^K} \prod_{k=1}^{K} p(\alpha_k|\ \alpha_{k-1}) \cdot p(\mathbf{x}_{\gamma_{\alpha_k-1}+1}^{\gamma_{\alpha_k}} | \mathbf{y}_{\mu_{k-1}+1}^{\mu_k}), \tag{1.9}$$

donde normalmente se asume que el modelo de distorsión $p(\alpha_k|\ \alpha_{k-1})$ (la probabilidad de que un segmento destino k sea alinee con un segmento origen α_k) únicamente depende del alineamiento anterior α_{k-1} (modelo de primer orden).

1.3.2. Entrenamiento del modelo basado en secuencias de palabras

Finalmente, cuando aprendemos un modelo PB, el objetivo es calcular la tabla de traducción de segmentos (*phrase translation table*), con la forma

$$\{(x_j \dots x_{j'}), (y_i \dots y_{i'}), p(x_j \dots x_{j'}|y_i \dots y_{i'})\},$$

en donde $(x_j \dots x_{j'})$ representa un segmento en el idioma origen, $(y_i \dots y_{i'})$ un segmento en la idioma destino y $p(x_j \dots x_{j'}|y_i \dots y_{i'})$ la probabilidad asignada al modelo dado un par bilingüe de segmentos.

Durante la última década se han explorado e implementado una amplia variedad de técnicas para producir modelos PB [Koehn et al., 2003]. En primer lugar se propuso el aprendizaje directo de los parámetros de la ecuación $p(\tilde{x}|\tilde{y})$ [Tomás and Casacuberta, 2001, Marcu and Wong, 2002]. Otros enfoques sugeridos fueron explorar técnicas con mayor motivación lingüística [Sánchez and Benedí, 2006, Watanabe et al., 2003]. A pesar de esto, la técnica que ha sido ampliamente adoptada es la desarrollada por [Zens et al., 2002], en donde todos los pares de segmentos coherentes con un alineamiento de palabras dado son extraídos, en la mayoría de casos empleando uno de los alineamientos de IBM descritos en la sección 1.2. Ya que estos alineamientos son muy restrictivos debido a que cada palabra destino se asigna únicamente a cero o a una palabra origen se combinan heurísticamente los alineamientos origen-a-destino y destino-a-origen. Este procedimiento suele denominarse *simetrización*. Una vez hecho esto, el conjunto de segmentos consistentes con los alineamientos de palabras simetrizados se extraen para cada par de oraciones del conjunto de entrenamiento. Se puede ver un ejemplo de este procedimiento en la figura 1.1.

Concretamente, las características diferentes que se incluyen en el modelo de traducción son:

- Probabilidades de traducción inversa, dadas por la fórmula

$$p(\tilde{x}|\tilde{y}) = \frac{C(\tilde{x}, \tilde{y})}{C(\tilde{x})}, \tag{1.10}$$

  donde $C(\tilde{x}, \tilde{y})$ representa el número de veces que los segmentos $\tilde{x}$ e $\tilde{y}$ se extraen a lo largo de todo el corpus, y $C(\tilde{x})$ es el número de veces que aparece $\tilde{x}$.

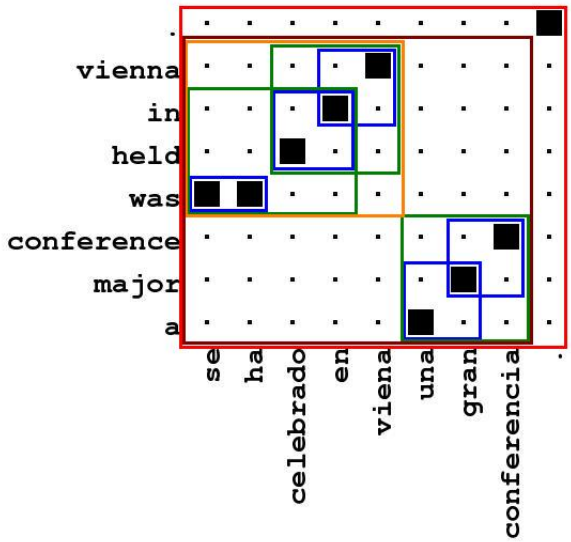

Figura 1.1: Ejemplo de la extracción de segmentos consistentes con el alineamiento de palabras.

- Probabilidad de traducción directa, $p(\tilde{y}|\tilde{x})$, que se obtiene análogamente.
- Características lexicalizadas directa e inversa, las cuales intentan explicar la solidez léxica de cada par de segmentos, estimando cuan bien cada palabra en un idioma se traduce por otra palabra en distinto idioma. Estas características lexicalizadas fueron definidas en [Zens et al., 2002].
- Una característica constante, o penalización del segmento, llamada habitualmente *phrase penalty*, cuyo propósito es evitar el uso de muchos segmentos cortos durante la traducción, en favor del uso de segmentos más largos.

1.3.3. Ajuste de los modelos log-lineales

Los modelos log-lineales normalmente consisten en una combinación lineal de modelos logarítmicos, los cuales pueden no estar definidos en el mismo rango por lo que también son conocidos como *feature functions*. La puntuación que reciben las hipótesis cuando se presenta una oración de entrada en el sistema es la combinación de las puntuaciones asignados por cada uno de los modelos. Hay que tener en cuenta que no todos los modelos tienen la misma importancia en la decisión global, lo que hace que se deba ajustar su influencia.

Por esta razón, mediante los factores de escalado se ajusta el poder discriminativo de cada modelo que participa en la combinación log-lineal. Estos factores de escalado

se ajustan típicamente a través de una pequeña cantidad de oraciones bilingües, el conjunto de desarrollo, debido a que la cantidad de parámetros cuyos valores necesitan ser aprendidos es pequeña, típicamente 14. El propósito es seleccionar la mejor configuración para estos pesos de forma que el error, dada una métrica de calidad, sea mínimo al utilizar el conjunto de desarrollo, empleando para ello la técnica llamada *minimum error rate training (MERT)* [Och, 2003a]. Aunque MERT está ideado para optimizar cualquier métrica de calidad, la más comúnmente empleada es el BLEU. El paso de *tuning* o ajuste más común a la hora de establecer un sistema SMT consiste en traducir las oraciones del idioma origen del conjunto de desarrollo y producir un conjunto de las N mejores hipótesis (llamadas comúnmente N-best) para la traducción de cada oración. Posteriormente, usando un algoritmo de optimización como el algoritmo propuesto por Powell [Powell, 1964], los valores para los factores de escalado son estimados de forma que las hipótesis con mayor puntuación dentro de la lista de N-best son empujadas hacia arriba de la lista. Inmediatamente después, se traduce de nuevo el conjunto de desarrollo empleando los nuevos factores de escalado, y por tanto, produciendo una nueva lista de N-best. Posteriormente esta nueva lista de N-best se combina con la anterior. Este proceso se repite de forma iterativa hasta que la lista de N-best no varíe de una iteración a otra, y por tanto el algoritmo converja, logrando de esta forma obtener todas las traducciones posibles para las oraciones del conjunto de desarrollo.

1.3.4. Traducción empleando los modelos basados en secuencias de palabras

Una vez que un sistema SMT ha sido entrenado es el momento de iniciar el proceso de traducción, mediante el cual las oraciones en un idioma origen se traducirán a un idioma destino. A este proceso se le denomina *decoding* y requiere el uso de un algoritmo para este fin. Se han sugerido diferentes estrategias de búsqueda para definir la forma en la que se organiza el espacio de búsqueda. En [Ortiz et al., 2003] se ha propuesto el uso del algoritmo $A^\star$, el cual adopta una estrategia *el primero mejor (best-first)* empleando colas de prioridad con el objetivo de organizar el espacio de búsqueda, siendo esta la estrategia más utilizada. Por otra parte también se ha sugerido la estrategia de *búsqueda en profundidad (depth-first)* en [Berger et al., 1996], la cual utiliza un conjunto de pilas para realizar la búsqueda.

1.4. Traducción Asistida por Ordenador

A día de hoy, las traducciones proporcionadas por los sistemas SMT del estado del arte aún siguen lejos de ser fiables. A pesar de ello, los sistemas actuales, en dominios generales, proporcionan traducciones con la calidad suficiente para hacernos una idea global del contenido de un texto. Por contra, sigue habiendo muchas otras tareas en las que estas traducciones no poseen la calidad necesaria y necesitan la colaboración de un traductor humano para garantizar resultados de calidad.

Actualmente, los traductores humanos emplean los ordenadores como una herramienta básica de trabajo, por lo que la interacción hombre-maquina es cada vez más necesaria, provocando que los diferentes casos de interacción hayan dado lugar a diferentes métodos en el marco de la traducción asistida por ordenador (comúnmente denominada *Computer-Assisted Translation* o CAT). Los diccionarios digitales fueron uno de los primeros y más rudimentarios elementos de la traducción asistida por ordenador, siendo las memorias de traducción (conocidas como *Translation Memory*, TM) la extensión natural de estos. Las memorias de traducción son bases de datos que almacenan pares de segmentos bilingües previamente traducidos para su uso posterior en caso de que estos vuelvan a aparecer. Las memorias de traducción colaborativas son un tipo de TM, en donde varios usuarios pueden agregar sus pares de segmentos bilingües. Debido a esta ventaja, las memorias de traducción colaborativas son uno de los recursos CAT más apreciados.

Por tanto, la comunidad científica cree que para aumentar la productividad en el proceso de traducción, las ordenadores deben tener un papel muy importante. Una forma sencilla de hacer que una máquina participe de forma activa en el proceso de traducción, es introducirla al comienzo de un proceso secuencial [Callison-Burch et al., 2004], de forma que el sistema SMT genere una traducción temporal que el experto traductor humano se encarga de aceptar, rechazar o corregir.

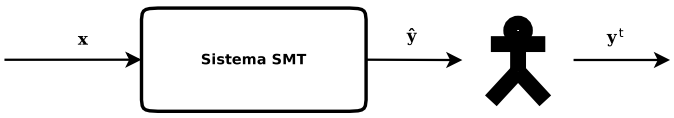

Figura 1.2: Esquema del paradigma de post-edición en la cual el traductor humano puede modificar la traducción dada por el sistema SMT

La aceptación del paradigma de post-edición por parte de los traductores humanos profesionales depende en gran medida de la calidad de las traducciones del sistema SMT. Si el sistema ofrece hipótesis de mala calidad el traductor requerirá de más tiempo para corregir la hipótesis dada del que necesitaría para traducir la oración desde cero y por tanto su productividad se verá reducida considerablemente. Por contra, si el sistema ofrece traducciones de calidad el usuario necesitará realizar pocos cambios en la hipótesis y verá aumentada su productividad, lo que ayudará a la aceptación del paradigma por parte del usuario.

Una mejora en el sistema de post-edición consiste en medir el grado de confianza que el sistema tiene en la hipótesis proporcionada de forma que sólo en el caso de que este supere cierto umbral se le proporcione al traductor la hipótesis para corregirla.

Además del paradigma de post-edición existen otras formas más sofisticadas de introducir una máquina en el proceso de traducción, siendo el paradigma de traducción automática interactiva [Barrachina et al., 2009, Casacuberta et al., 2009] una de ellas. Debido a que en las lenguas europeas la lectura se produce de izquierda a derecha, en el paradigma IMT el usuario acepta un prefijo (secuencia de palabras) de la hipótesis propuesta por el sistema como correcto, indicando que posteriormente a este segmento aparece un error. En función de este prefijo, en la siguiente iteración el sistema propondrá un sufijo para completar la traducción. Este procedimiento se repite hasta que el usuario acepte la totalidad de la oración.

1.5. Traducción Automática Interactiva

Debido a los grandes avances en las tecnologías de la información, en la actualidad se ha comenzado a necesitar métodos de traducción más eficientes y menos costosos. A pesar de ello, los actuales sistemas de MT no son capaces de producir traducciones con la suficiente calidad como para poder usarse directamente [Kay, 1997, Hutchins, 1999, Arnold., 2003] sin necesidad de ser previamente revisadas. Además los sistemas de MT habitualmente se encuentran limitados por la semántica específica del dominio y las traducciones que proporcionan, en la mayoría de casos, requieren de post-edición por parte de un humano para lograr obtener traducciones con una alta calidad.

Una forma de mejorar los sistemas de MT actuales es combinándolos con el conocimiento de un traductor humano experto, constituyendo el paradigma llamado *Computer-Assisted Translation* (CAT). CAT ofrece diferentes aproximaciones con el objetivo de beneficiarse de la sinergia entre los humanos y los sistemas de MT.

Una importante contribución a la tecnología CAT fue llevada a cabo dentro del proyecto TransType (TT) [Langlais et al., 2004, Foster et al., 2002, Foster, 2002, Och et al., 2003]. Este proyecto implicó un interesante cambio de enfoque, en el cual la interacción estuvo directamente dirigida a traducir texto, en lugar de a la desambiguación del texto a traducir como ocurría en los antiguos sistemas interactivos. La idea que el proyecto TT propuso fue integrar las técnicas de MT basadas en datos dentro de un entorno de traducción interactivo, con la esperanza de combinar lo mejor de los paradigmas CAT, en el cual el humano asegura una traducción de calidad, y MT, en donde la máquina asegura una ganancia significante en la productividad.

Continuando con las ideas del proyecto TT, en [Barrachina et al., 2009] se propone el uso de sistemas de traducción automática estadística completos para producir hipótesis de traducción completas, o porciones de estas, las cuales pueden ser aceptadas y modificadas por un traductor humano. Cada porción de texto, aceptada como correcta por el usuario, es usada por el sistema SMT como información adicional para generar el resto de la traducción, así como para mejorar la calidad de las futuras sugerencias. Concretamente, en cada iteración el prefijo de la oración traducida es de alguna manera fijado por el traductor humano y en la siguiente iteración el sistema predice el mejor o los mejores sufijos para completar dicho prefijo. Este proceso es comúnmente conocido como *traducción automática interactiva* o IMT por sus siglas en inglés, *interactive-predictive machine translation*. El paradigma IMT se ajusta bien dentro del marco de *Reconocimiento de formas interactivo* introducido en [Vidal et al., 2007].

En la figura 1.3 podemos ver un ejemplo del paradigma IMT. Inicialmente el usuario da una oración de entrada $\mathbf{x}$ para ser traducida. La referencia $\mathbf{y}$ proporcionada es la traducción que el usuario quiere lograr al final del proceso de IMT. En la iteración 0, el usuario no suministra ningún prefijo del texto al sistema, por este motivo $\mathbf{p}$ no contiene nada. Por tanto el sistema IMT tiene que proporcionar la traducción completa como sufijo $\mathbf{s}_h$, tal como si este fuera un sistema convencional de SMT. En la siguiente iteración el usuario valida el prefijo $\mathbf{p}$ como correcto posicionando el cursor en la posición en la que comienza la primera palabra errónea, que en este caso se corresponde con el final del segmento “gracias a la encuesta , esto”. Por otra parte implícitamente se está marcando el resto de la oración, es decir, el sufijo $\mathbf{s}_l$ (“cambie .”) como incorrecto.

ENTRADA (x):		thanks to the poll , this will change .
REFERENCIA (y):		gracias a la encuesta , esto ha cambiado .
ITER-0	$(\mathbf{p})$	( )
	$(\hat{\mathbf{s}}_h)$	*gracias a la encuesta , esto cambie .*
ITER-1	$(\mathbf{p})$	gracias a la encuesta , esto
	$(\mathbf{s}_l)$	*cambie .*
	(k)	ha
	$(\hat{\mathbf{s}}_h)$	*a cambiar .*
ITER-2	$(\mathbf{p})$	gracias a la encuesta , esto ha
	$(\mathbf{s}_l)$	*a cambiar .*
	(k)	cambiado
	$(\hat{\mathbf{s}}_h)$	.
ITER-3	$(\mathbf{p})$	gracias a la encuesta , esto ha cambiado .
	$(\mathbf{s}_l)$	( )
	(k)	(#)
	$(\hat{\mathbf{s}}_h)$	( )
FINAL	$(\mathbf{p} \equiv \mathbf{y})$	gracias a la encuesta , esto ha cambiado .

Figura 1.3: Ejemplo de funcionamiento del proceso IMT para traducir del inglés al español una oración Las hipótesis no aceptadas se muestran en cursiva mientras que los prefijos aceptados se muestran utilizando la fuente por defecto.

Posteriormente el usuario introducirá una nueva palabra k a continuación del prefijo aceptado, la cual se asume que es diferente de la primera palabra s_{l_1} en el sufijo $\mathbf{s}_l$ la cual no fue validada, $k \neq s_{l_1}$. A continuación el proceso anterior se repite, el sistema sugiere una nueva hipótesis como sufijo $\hat{\mathbf{s}}_h$, sujeto a $\hat{s}_{h_1} = k$, y el usuario valida el nuevo prefijo, proporciona una nueva palabra y así sucesivamente. Este proceso terminará en el momento en el que toda la oración haya sido validada como correcta y se introduzca la palabra especial "#".

Como se puede observar en el ejemplo anterior, a pesar de que un sistema IMT puede cometer errores en la predicción de los sufijos, ayuda a reducir el esfuerzo e incrementa la productividad de los traductores generando traducciones de alta calidad. Para el caso descrito, en la figura 1.3 únicamente ha sido necesario realizar cinco interacciones para obtener la traducción de referencia, las cuales constan de dos clicks de ratón y la introducción de 3 palabras. Por contra, sin emplear IMT el traductor habría tenido que introducir ocho palabras y siete caracteres de espacios.

1.5.1. Traducción Automática Estadística Interactiva

La traducción automática estadística interactiva se base en la formulación de SMT. Para establecer la ecuación fundamental para IMT es necesario modificar la Ecuación 1.1 de acuerdo al escenario IMT con el objetivo de tener en cuenta la parte de la oración destino que ya esta traducida, que es $\mathbf{p}$ y k:

$$\hat{\mathbf{s}}_h = \underset{\mathbf{s}_h}{\operatorname{argmax}} Pr(\mathbf{s}_h|\mathbf{x}, \mathbf{p}, k) \tag{1.11}$$

donde el problema de maximización se define sobre el sufijo $\mathbf{s}_h$, permitiéndonos reescribir la Ecuación 1.11 descomponiendo apropiadamente la parte derecha y eliminando los términos constantes, logrando el criterio equivalente

$$\hat{\mathbf{s}}_h = \underset{\mathbf{s}_h}{\operatorname{argmax}} Pr(\mathbf{p}, k, \mathbf{s}_h|\mathbf{x}). \tag{1.12}$$

Un ejemplo de la utilidad de estas variables puede verse en la Figura 1.3. Hay que tener en cuenta que debido a que $\mathbf{p}\,k\,\mathbf{s}_h = \mathbf{y}$, la Ecuación 1.12 es muy parecida a la Ecuación 1.1. La principal diferencia entre ellas reside en la búsqueda del argmax, ya que ahora se realizará sobre el conjunto de sufijos $\mathbf{s}_h$ que completen $\mathbf{p}\,k$ en lugar de la oración completa $\mathbf{y}$ como ocurre en la Ecuación 1.1. Esto hace que podamos usar los mismos modelos siempre y cuando el procedimiento de búsqueda se modifique correctamente [Barrachina et al., 2009].

1.5.2. IMT basada en segmentos

La aproximación basada en segmentos de palabra o phrases presentada anteriormente se puede adaptar fácilmente para ser usada en escenarios IMT. La modificación mas importante a realizar consiste en emplear grafos de palabras, los cuales representan las posibles traducciones para una oración. En [Barrachina et al., 2009] se estudió el uso de los grafos de palabras en IMT en combinación con dos técnicas de traducción, denominadas, *plantillas de alineamientos* conocidas en MT como *Alignment Templates* [Och et al., 1999, Och and Ney, 2004] y *Transductores de Estados Finitos*, conocidos como *Stochastic Finite State Transducers* [Casacuberta and Vidal, 2007].

1.5.3. IMT usando grafos de palabras

Un *grafo de palabras*, conocidos habitualmente en inglés como *word graph*, es un grafo dirigido, acíclico y ponderado, en el cual cada nodo representa una hipótesis de traducción parcial y cada arista etiquetada con una palabra de la oración destino está ponderada de acuerdo a la puntuación dada por el modelo SMT, lo cual puede verse con más detalle en [Ueffing et al., 2002].

En [Och, 2003a] se propone el uso de grafos de palabras como una interfaz entre los modelos de SMT basados en plantillas de alineamientos y el motor IMT. En este trabajo de maneara análoga usaremos el grafo de palabras construido durante el procedimiento de búsqueda realizado en el modelo de SMT basado en segmentos. Ya que este modelo podría generar un *grafo de segmentos* o *phrase-graph*, en lugar de un grafo de palabras, es necesario convertir este a un grafo de palabras. Sin embargo este procedimiento es bastante simple y se logra añadiendo nodos y aristas artificiales entre cada uno de las palabras que constituyen los segmentos y asignando la puntuación del segmento a la arista final. En la figura 1.4 podemos ver un ejemplo de este procedimiento. Debemos tener en cuenta en el proceso de conversión del grafo de segmentos

a grafo de palabras que las puntuaciones de las aristas no son probabilidades sino logaritmos de probabilidades ya que la maximización de la ecuación 1.5 se realiza sin normalización. Las puntuaciones de las aristas de los grafos depende de dos factores, en primer lugar de la función de características asociada a la oración que representa el grafo, y en segundo lugar a los pesos del modelo log-lineal asociados a esta función. Por ello, la puntuación de transición entre los nodos de un grafo dependen de ambos conjuntos de parámetros. Puesto que en este TFM únicamente pretendemos optimizar el valor de los pesos del modelo log-lineal para una oración dada $\mathbf{x}$, diremos que el grafo de palabras dependerá del conjunto de pesos $\boldsymbol{\lambda}^n$ y no de la función de características, denotándolo como $W_{\boldsymbol{\lambda}^n}(\mathbf{x})$.

Durante el proceso de IMT para una oración dada, el sistema hace uso del grafo de palabras generado para la oración con el fin de completar el prefijo aceptado por el traductor humano. Específicamente el sistema encuentra el mejor camino en el grafo de palabras asociado con el prefijo dado, de forma que permita completar la traducción, siendo capaz de proporcionar muchas sugerencias de terminación para cada prefijo.

Cuando el usuario define un prefijo que no aparece en el grafo de palabras, el sistema no puede encontrar el camino a través del grafo que proporcione un sufijo adecuado, dando lugar a uno de los problemas mas comunes en IMT. Para solucionar este problema se realiza una búsqueda tolerante en el grafo de palabras, la cual, como puede verse en [Och, 2003a], emplea la conocida distancia de Levenshtein para obtener el segmento más parecido al prefijo dado.

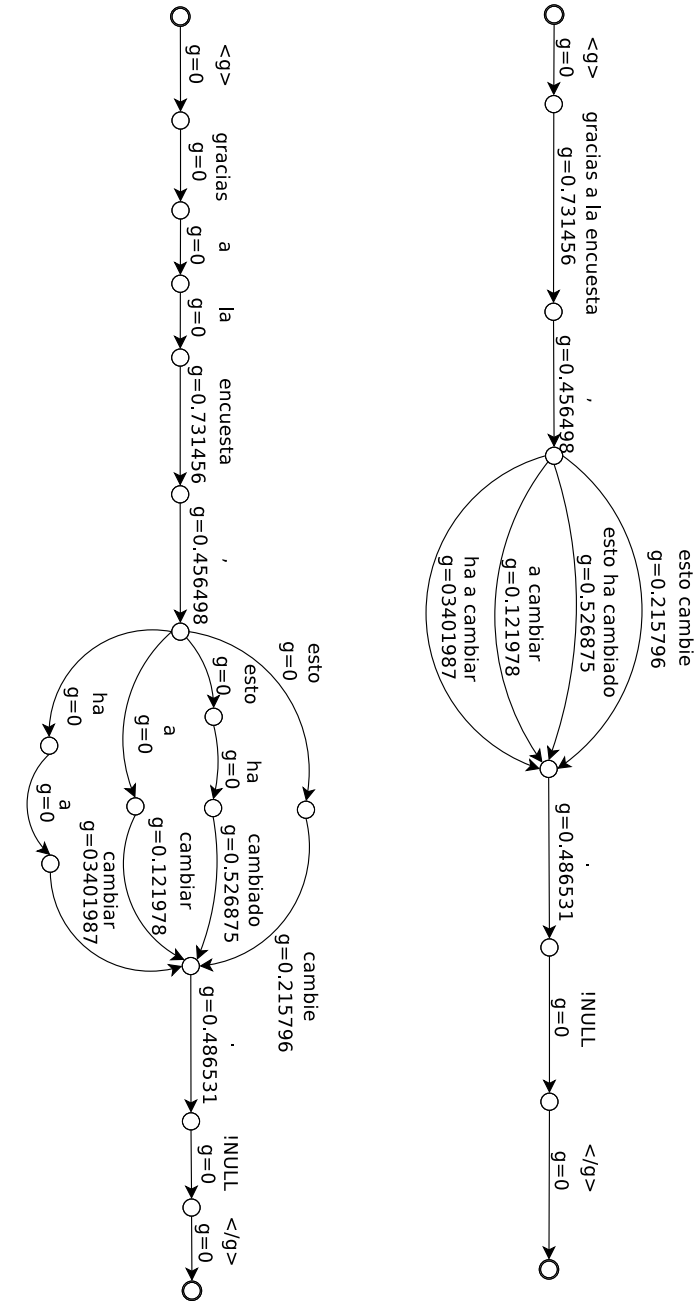

Figura 1.4: Ejemplo de conversión de un grafo de segmentos (arriba) a grafo de palabras (abajo). Los símbolos $<g>$ y $</g>$ representan a la aristas que parten del nodo inicial y a la arista que acaba en el nodo final del grafo respectivamente. El carácter g representa la palabra o segmentos correspondiente a una arista, es decir, la palabra o segmento perteneciente a una de las hipótesis del grafo.

1.6. Adaptación

Al traducir textos pertenecientes a un dominio distinto al de los corpus de entrenamiento o desarrollo utilizados en el sistema, la calidad de las traducciones disminuye significativamente [Callison-Burch et al., 2007], lo que provoca que la *adaptación* sea uno de los problemas más comunes en traducción automática estadística. El objetivo que persigue la adaptación es mejorar el rendimiento de los sistemas entrenados y calibrados mediante conjuntos de entrenamiento y desarrollo *out-of-domain* (fuera del dominio del texto a traducir) a partir de una cantidad muy limitada de datos *in-domain* (del propio dominio del texto a traducir) o que presentan otro tipo de restricciones sobre la cantidad de datos disponibles o limitaciones temporales.

Una de las aproximaciones para realizar la adaptación de forma eficaz es el *filtrado* [Moore and Lewis, 2010], en donde se exploran las oraciones de entrenamiento para buscar datos similares al dominio de interés. Las oraciones bilingües encontradas pueden emplearse para aumentar el tamaño del conjunto de desarrollo o bien para crear nuevos modelos con los nuevos datos dentro del dominio y finalmente interpolarlos con los datos fuera del dominio. Similarmente se puede realizar un nuevo muestreo en las oraciones del conjunto de entrenamiento para ponderar su relevancia en la tarea in-doman [Schwenk and Senellart, 2009, Shah et al., 2010, Gascó et al., 2012]. A pesar del interés que puedan tener estas aproximaciones una de sus limitaciones es que a día de hoy son exclusivas de la adaptación *batch*, en la cual, el sistema no es adaptado al procesar cada una de sus oraciones, sino al procesar la totalidad de las oraciones.

Para realizar adaptación de forma online, una de las estrategias existentes consiste en ajustar las funciones de características o los pesos log-lineales con el objetivo de mejorar el rendimiento de los sistemas modificando sus estimaciones de probabilidad. Por ello, en este trabajo fin de máster seguiremos una de estas estrategias: adaptar los pesos del modelo log-linal, para mejorar el rendimiento del sistema.

Mediante el uso de caches, se puede aprovechar el principio de localidad para modificar de esta forma las estimaciones de probabilidad de los sistemas. Para esta aproximación los modelos entrenados con datos fuera del dominio se pueden interpolar con modelos entrenados a partir de las ultimas Q oraciones procesadas [Clarkson and Robinson, 1997, Nepveu et al., 2004, Kuhn and De Mori, 1990], en donde el rendimiento en SMT es cuidadosamente evaluado en [Tiedemann, 2010].

Otros autores, sugieren diferentes formas de combinar datos de varios idiomas o simplemente combinar datos en un solo idioma pero de diferentes dominios [Koehn and Schroeder, 2007, Bertoldi and Federico, 2009]. Otros estudios [Zhao et al., 2004, Sanchis-Trilles et al., 2009] proponen el uso de técnicas de *agrupamiento* o *clustering* para extraer sub-dominios y construir modelos de lenguaje o traducción mas específicos. Algunos autores [Civera and Juan, 2007] proponen mixturas de modelos de alineamiento para llevar a cabo la adaptación de temas específicos.

Uno de los problemas más interesantes en CAT e IMT es adaptar el sistema a tareas cambiantes. La estrategia de adaptación importada desde la traducción automática estadística tradicional a CAT o IMT, se basa en el aprendizaje batch. El proceso de aprendizaje en la adaptación batch para CAT e IMT, se realiza mediante un conjunto de traducciones que el usuario ha producido a partir de un conjunto de oraciones de

entrada para un idioma origen dado, ayudado por un sistema de traducción automática adecuado 1.2. El conjunto de traducciones junto con las frases de entrada con las que se corresponden, forman el conjunto bilingüe utilizado para optimizar algunos parámetros que intervienen en el proceso de traducción.

En la traducción automática estadística tradicional esta optimización de parámetros requiere de cierta cantidad de recursos computacionales y una cantidad de tiempo considerable. A pesar de esto, el proceso no conlleva ningún problema ya que se realiza off-line antes de comenzar la tarea de traducción tradicional. Bajo el punto de vista de la adaptación, los paradigmas CAT e IMT son muy diferentes. Estos sistemas trabajan bajo la restricción marcada por la interacción del usuario, y en el caso de las tareas reales, la naturaleza del texto de origen a traducir puede variar constantemente de forma impredecible. Por tanto, para ambos problemas, no es adecuado emplear la adaptación batch ya que nos fuerza a realizar una optimización bastante costosa después de traducir cierto numero de oraciones.

El propósito es hacer el mejor uso posible de las correcciones proporcionadas por el usuario para adaptar los modelos de forma online, es decir, sin reentrenar completamente los parámetros del modelo, ahorrando de esta forma un gran coste. A lo largo de la historia se han propuesto diferentes aproximaciones para abordar el problema de la adaptación online basándose en traducción oración tras oración. Algunos autores [Ortiz-Martínez et al., 2010] proponen un sistema de aprendizaje online para IMT en donde los modelos que intervienen en el proceso de traducción se actualizan de forma incremental mediante una versión incremental del algoritmo *expectation-maximisation* (EM), permitiendo la inclusión de nuevos pares de frases en el sistema. Uno de los propósitos de este trabajo fin de máster es adaptar los mecanismos de predicción a IMT empleando varias estrategias de adaptación, las cuales emplean una lista en forma de N-best, independientemente de su origen, para ajustar los pesos del modelo log-lineal. Como podremos ver a lo largo de esta memoria, este tipo de adaptación nos permite comparar diferentes estrategias de aprendizaje online para adaptar $\boldsymbol{\lambda}$, en cualquier sistema empleando una lista en forma de N-best. En [España-Bonet and Marquez, 2010] se propone el uso del algoritmo Perceptron con el fin de obtener estimaciones más robustas para los *scaling factors* ($\boldsymbol{\lambda}$), iterando varias veces (*epochs*) sobre el conjunto de desarrollo mientras no se logre la convergencia deseada. A lo largo de este trabajo final de máster emplearemos varias estrategias de aprendizaje online, ya que en este caso, cada observación la procesaremos una única vez.

Los autores en [Gabriel Reverberi,] proponen el uso del algoritmo *Passive-Agressive* (PA) [Crammer et al., 2006] para actualizar las funciones de características $\mathbf{h}$. Ellos proponen integrar el uso de los sistemas MT basados en memorias y los sistemas SMT. En esta actualización el sistema SMT únicamente interviene cuando el sistema de traducción automática basado en memorias no ha realizado una traducción de calidad. En ambos casos el sistema combinado se realimenta a partir de la traducción final revisada por el usuario, de forma que el sistema de traducción automática basado en memoria incluye esto dentro de la memoria de traducción y el sistema SMT activa un procedimiento de aprendizaje online. Por esta razón el rendimiento de su aprendizaje online solo se evalúa mediante un pequeño subconjunto de las oraciones originales a

traducir. Las mejoras obtenidas fueron muy limitadas debido a que adaptar $\mathbf{h}$ es un problema muy disperso y para realizar la adaptación online se empleo un subconjunto de oraciones pequeño.

En los experimentos realizados en este trabajo no se han utilizado las memorias de traducción dentro del ciclo de interacción, de forma que el sistema IMT siempre propone una traducción al usuario permitiendo ver el comportamiento del aprendizaje online.

En este TFM, empleando varias estrategias para adaptar $\boldsymbol{\lambda}$ en combinación con técnicas de aprendizaje online en IMT, se pretende utilizar la información de la ultima interacción del usuario esperando mejorar la calidad de las posteriores traducciones.

Para abordar la tarea de adaptación online en un escenario IMT, todo este trabajo se basara en [Martínez-Gomez, 2010] donde se plantean diferentes estrategias para adaptar los pesos del modelo log-lineal y la función de características empleando un sistema CAT no interactivo.

1.7. Conclusiones

Debido al interés por comunicarnos con otras personas la traducción automática ha sido objeto de estudio desde el origen del ordenador. De entre todas las aproximaciones empleadas, la aproximación estadística en combinación con los modelos basados en segmentos ha sido la que ha ofrecido mejores resultados, siendo por tanto ampliamente adoptada tanto por grandes multinacionales como por las universidades de diferentes países. A pesar de tener tres claras etapas en el proceso de traducción como son, entrenamiento, ajuste y traducción, aparecen grandes dificultades a la hora de traducir textos pertenecientes a un dominio con el cual el sistema de traducción automática no ha sido debidamente entrenado. Esto hace que la adaptación sea uno de los mayores puntos de interés en el campo de la MT, y en especial cuando el sistema debe ser continuamente adaptado por ejemplo con cada interacción humana.

Para mejorar la calidad de las traducciones a lo largo de los años se han propuesto varias aproximaciones en donde la post-edición se ha convertido hoy en día en una buena herramienta para los expertos traductores humanos. Aún así las investigaciones no se han centrado exclusivamente en obtener traducciones de la máxima calidad posible, sino en obtener estas con el mínimo esfuerzo, lo que ha dado lugar a sistemas de traducción automática interactivos que permiten maximizar la calidad minimizando el esfuerzo humano.

En este trabajo fin de máster se han realizado las siguientes contribuciones. En primer lugar, se han propuesto tres nuevas aproximaciones enfocadas a un escenario IMT para ajustar los pesos del modelo log-lineal, $\boldsymbol{\lambda}$. Dos de ellas emplean el algoritmo de aprendizaje online Discriminative Ridge Regression (DRR) con variaciones en su implementación, por un lado siguiendo la implementación del DRR ya empleada en post-edición y por otro aplicando una nueva formulación del mismo para un escenario IMT. La tercera contribución es una primera aproximación, más sencilla que la planteada con el algoritmo DRR, para adaptar $\boldsymbol{\lambda}$, también en un escenario IMT. Todas las aproximaciones son independientes del algoritmo de aprendizaje empleado, de modo que podrían aplicarse otros siempre y cuando fueran correctamente adaptados a IMT. Por ultimo se realiza una exhaustiva comparación de los resultados para todos las estrategias planteadas.

Capítulo 2

Aprendizaje online en IMT

En este trabajo se presenta la adaptación del paradigma de aprendizaje online en el marco IMT, en donde se pretende adaptar los pesos del modelo log-lineal para mejorar la calidad de las traducciones. En el paradigma de aprendizaje online, las etapas de entrenamiento y predicción ya no están separadas. Esta característica es particularmente útil en IMT ya que de esta forma se logra que el sistema vaya aprendiendo de cada interacción con el usuario a partir de la realimentación que este proporciona. Los algoritmos tradicionales de ajuste, pretenden encontrar el conjunto de pesos del modelo log-lineal que maximice la calidad de las traducciones. Algunos de estos algoritmos emplean el procedimiento Minimum Error Rate Training (MERT), y son usados en modo batch, es decir, estos procesan todas las muestras del conjunto de desarrollo tantas veces como sean necesarias hasta lograr la convergencia del algoritmo. Sin embargo, en las tareas reales de traducción que realiza un sistema IMT o CAT, para llevar a cabo la adaptación del sistema, no es realista procesar todas las muestras cada vez que vaya apareciendo una nueva, ya que esto sería tanto computacionalmente como temporalmente algo muy costoso y poco eficiente. Debido a esto, generalmente en el marco del aprendizaje online, para adaptar un sistema, el algoritmo de aprendizaje procesa las muestras de forma secuencial, de modo que las muestras ya procesadas no vuelvan a procesarse, como si ocurre con el algoritmo MERT. Para ello, después de procesar una muestra, el sistema realiza la siguiente predicción realimentándose con información que el usuario proporciona de la última muestra procesada. La información que proporciona la realimentación puede ir desde una simple opinión sobre la calidad de la predicción que ha realizado sistema, por ejemplo, aceptando parte de la hipótesis como ocurre en IMT, o dando la etiqueta real de la muestra procesada en entornos completamente supervisados.

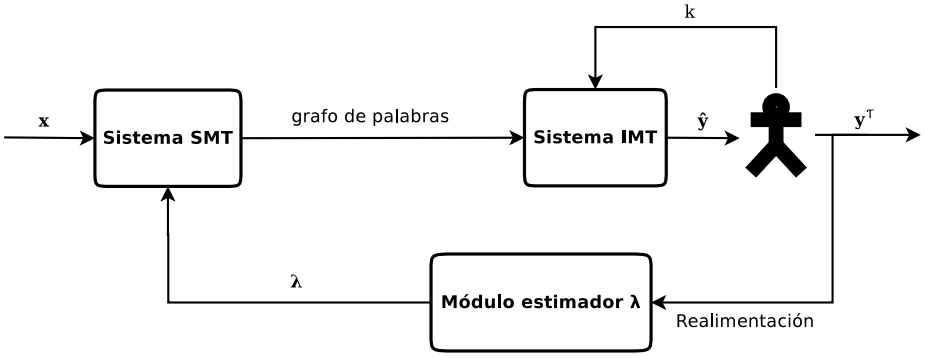

Figura 2.1: Esquema del paradigma de aprendizaje online dentro del marco de la traducción automática interactiva-predictiva (IMT) en donde la interacción del humano proporciona la realimentación de forma online.

En la figura 2.1 se muestra un esquema del paradigma de aprendizaje online, en donde se incorpora la realimentación del usuario dentro del módulo *estimador* $\boldsymbol{\lambda}$ y la interacción del usuario con el sistema IMT en cada interacción k. El objetivo del módulo estimador $\boldsymbol{\lambda}$ es generar el conjunto de pesos que se utilizara en la creación del grafo de palabras.

En un primer momento, el sistema SMT recibe una primera oración $\mathbf{x}$ en un idioma origen y genera un grafo de palabras o *word-graph* a partir de la misma.

A continuación, el grafo de palabras generado sirve como entrada al sistema IMT, de forma que el sistema genere el mejor sufijo posible a partir de él, es decir, devuelva al usuario el mejor camino del grafo. Una vez el sistema genera la traducción, como puede verse en la figura 1.3, el traductor humano deberá aceptar la totalidad de la traducción o un prefijo de la misma en cada una de las interacciones con el sistema. Este proceso finalizará cuando el usuario acepte totalmente la traducción propuesta por el sistema, convirtiéndose esta en la traducción de referencia $\mathbf{y}^{\tau}$. Esta oración puede ser usada como realimentación para el sistema IMT de forma que el algoritmo de aprendizaje online modifique el valor de $\boldsymbol{\lambda}$ con el objetivo de mejorar la calidad de las futuras traducciones. A partir de la segunda oración que recibe el sistema SMT, durante la generación del grafo de palabras, se modificara el valor de los pesos de los modelos por los del conjunto de pesos que el módulo *estimador* $\boldsymbol{\lambda}$ optimizó a partir de todas las interacciones anteriores del usuario con el sistema. Como consecuencia de esta modificación, la puntuación de las diferentes aristas del grafo de palabras también serán modificadas. Puesto que nuestro objetivo es aprender de cada interacción, la ecuación 1.7 se redefine como:

$$\mathbf{y}_t = \underset{\mathbf{y}}{\operatorname{argmax}} \sum_{m=1}^{M} \lambda_m^t h_m(\mathbf{x}_t, \mathbf{y}), \tag{2.1}$$

transformada en forma vectorial como:

$$\hat{\mathbf{y}}_t = \underset{\mathbf{y}}{\operatorname{argmax}}\, \boldsymbol{\lambda}_t \mathbf{h}(\mathbf{x}_t, \mathbf{y}), \tag{2.2}$$

en donde los pesos del modelo log-lineal $\boldsymbol{\lambda}_t$ varían de acuerdo a las muestras $(\mathbf{x}_1,\mathbf{y}_1^\tau),\ldots,(\mathbf{x}_{t-1},\mathbf{y}_{t-1}^\tau)$ vistas antes del momento t. Con el fin de simplificar la notación, y siguiendo la notación descrita en [Martínez-Gómez et al., 2012], trabajo previo sobre el que se sustenta este trabajo fin de máster, a partir de ahora omitiremos el subíndice t para la oración de entrada $\mathbf{x}$, aunque dicho subíndice lo asumiremos siempre.

Nuestro objetivo es obtener traducciones con la mayor calidad posible, es decir que se asemejen lo máximo posible a la traducciones de referencia. Aún así, es frecuente que la hipótesis $\hat{\mathbf{y}}$ que maximiza la verosimilitud no tiene por qué ser la hipótesis de mayor calidad, es decir, la mejor según la perspectiva de un traductor humano o para una medida de calidad dada. Por tanto, es posible que la hipótesis con mayor calidad $\mathbf{y}^*$ no coincida con la hipótesis con mayor verosimilitud. Hay que tener en cuenta que $\mathbf{y}^*$ puede no coincidir con $\mathbf{y}^\tau$, la oración de referencia, debido a eventuales problemas de cobertura. Como ya se realizó en [Martínez-Gómez et al., 2012], se propone adaptar los parámetros del modelo de forma que $\mathbf{y}^*$ obtenga la mayor puntuación de acuerdo a la ecuación 1.7.

Con este fin se define la *diferencia* entre la calidad de la hipótesis propuesta por el sistema $\hat{\mathbf{y}}$ y la mejor hipótesis $\mathbf{y}^*$ en función de la medida de calidad $\mu(\cdot)$:

$$l(\hat{\mathbf{y}}) = |\mu(\hat{\mathbf{y}}) - \mu(\mathbf{y}^*)| \tag{2.3}$$

Debido a que los SMT pueden emplear diferentes medidas de calidad para evaluar su sistema, es decir, TER [Snover et al., 2006] representa el ratio de error (mejor cuanto más bajo), mientras que BLEU [Papineni et al., 2002] representa una medida de precisión (mejor cuanto más alto), se incluyó el valor absoluto en la ecuación 2.3 para preservar la generalidad. Además, la diferencia entre los scores de $\hat{\mathbf{y}}$ e $\mathbf{y}^*$ ha sido definida como

$$\phi(\hat{\mathbf{y}}) = g(\mathbf{x}, \mathbf{y}^*) - g(\mathbf{x}, \hat{\mathbf{y}}) \tag{2.4}$$

Tanto en la ecuación 2.3 como en la ecuación 2.4, para simplificar la notación, se omitieron las dependencias de la oración de entrada $\mathbf{x}$, la mejor hipótesis del sistema $\mathbf{y}^*$ y la traducción de referencia proporcionada por el usuario $\mathbf{y}^\tau$.

La intención es correlacionar $l(\cdot)$ y $\phi(\cdot)$ de forma que las diferencias de una se correspondan con las de la otra. Por lo tanto, si la hipótesis candidata $\mathbf{y}$ tiene una calidad de traducción $\mu(\mathbf{y})$ muy similar a la calidad de la traducción proporcionada por $\mu(\mathbf{y}^*)$, esperamos que $g(\mathbf{x}, \mathbf{y})$ sea muy similar a $g(\mathbf{x}, \mathbf{y}^*)$.

2.1. Aproximación

En [Martínez-Gómez et al., 2012] se proponen dos aproximaciones, adaptar $\mathbf{h}_t$ o $\boldsymbol{\lambda}_t$, es decir, adaptar en el instante de tiempo t la función de características o los pesos

del modelo log-lineal, respectivamente. En este trabajo únicamente nos centraremos en la adaptación de los pesos del modelo log-lineal, ya que esta aproximación ofrece mejores resultados, debido probablemente a que únicamente deben adaptarse el mismo número de parámetros que modelos intervienen en la combinación log-lineal, es decir, normalmente alrededor de 14 parámetros, frente a los cerca de tres millones en el caso de adaptar $\mathbf{h}_t$.

2.1.1. Adaptación de los pesos del modelo log-lineal

En el presente trabajo vamos a emplear una técnica para adaptar los pesos del modelo log-lineal $\boldsymbol{\lambda}$, también llamados factores de escalado, para abordar el problema del aprendizaje online en IMT.

Una vez el sistema haya recibido la oración de entrada $\mathbf{x}_t$ y su correspondiente oración de referencia $\mathbf{y}_t^\tau$ se procede a calcular el mejor conjunto de pesos $\hat{\boldsymbol{\lambda}}_t$ para el par de oraciones observado en el instante t, $(\mathbf{x}_t, \mathbf{y}_t^\tau)$ el cual se utilizará para calcular el término de actualización $\check{\boldsymbol{\lambda}}_t$.

Una vez se ha calculado $\check{\boldsymbol{\lambda}}_t$ podemos actualizar $\boldsymbol{\lambda}$ de la siguiente forma:

$$\boldsymbol{\lambda}_t = (1-\alpha)\boldsymbol{\lambda}_{t-1} + \alpha\check{\boldsymbol{\lambda}}_t, \tag{2.5}$$

empleando un ratio de aprendizaje α. Como puede observarse, para calcular $\boldsymbol{\lambda}_t$ previamente es necesario emplear el vector de pesos calculado para el par de oraciones observadas en el instante $t-1$, $\boldsymbol{\lambda}_{t-1}$ mediante la ecuación 2.5.

El objetivo es ajustar el poder discriminativo de cada modelo de forma que la puntuación resultante de la combinación log-lineal [Stauffer and Grimson, 2000] de estos sea mayor para la hipótesis más similar a la oración de referencia dada $\mathbf{y}_t^\tau$, es decir, $\mathbf{y}_t^*$, que la puntuación de cualquier otra hipótesis.

El proceso del cálculo de $\boldsymbol{\lambda}_t$ puede entenderse como una corrección de la estimación del $\boldsymbol{\lambda}_{t-1}$ anterior.

A pesar de que la información empleada al calcular $\check{\boldsymbol{\lambda}}_t$ es general e imprecisa, la variación en el score de la ecuación 1.8 puede ser alto ya que se están modificando los factores de escalado del modelo log-lineal. De esta forma, se permite adaptar el sistema a una nueva tarea ajustando la importancia que tiene cada uno de los modelos de forma online.

2.2. Algoritmos de aprendizaje online

Dada un oración $\mathbf{x}$ en un idioma origen, un sistema IMT generará un grafo de palabras que contendrá todas las posibles traducciones para la oración de entrada $\mathbf{x}$ en un idioma destino. De entre todas las hipótesis contenidas en el grafo de palabras, el sistema escogerá como traducción la "mejor" hipótesis, aquella cuyo camino en el grafo obtenga mayor puntuación. Dada la medida de calidad planteada en la ecuación 2.3 podemos comprobar que la hipótesis dada como traducción del sistema baseline, sistema del estado del arte que se utiliza como referencia y que emplea siempre el conjunto de

pesos obtenidos inicialmente, no tiene por qué ser la hipótesis con mayor calidad. Pretendemos que el sistema IMT genere las traducciones con la máxima calidad posible. Por ello emplearemos diversos algoritmos de aprendizaje online para intentar ajustar los pesos del modelo log-lineal de forma online con el objetivo de que la hipótesis del grafo con mayor puntuación sea además la hipótesis con más calidad.

En esta sección veremos como calcular $\check{\boldsymbol{\lambda}}_t$ para, mediante la ecuación 2.5, obtener $\hat{\boldsymbol{\lambda}}_t$ empleando el algoritmo DRR con dos formulaciones, una ya conocida basada en post-edición y otra planteada en ese TFM basada en IMT.

Para cada uno de los algoritmos de aprendizaje online, en primer lugar daremos una breve descripción para posteriormente detallar su aplicación para adaptar los pesos del modelo log-lineal. Cabe destacar que por simplicidad, siguiendo con la nomenclatura presentada en [Martínez-Gómez et al., 2012], se omitirán los subíndices y superíndices t en caso de que no se requiera de una clara distinción temporal.

Por coherencia, se describirán ambas definiciones del algoritmo DRR de forma continuada, aunque la estrategia *Primera aproximación* fue planteada antes de crear la nueva definición para el algoritmo DRR.

2.2.1. Discriminative ridge regression

A diferencia de otros algoritmos como Passive-Agressive [Crammer et al., 2006] y Percentron Like [España-Bonet and Marquez, 2010], empleados con éxito en un escenario de post-edición, y que intentan encontrar el conjunto de pesos tal que las "buenas" hipótesis dentro de la lista de N-best tengan una puntuación alta, el algoritmo DRR [Martínez-Gómez et al., 2012] además fuerza a que las "malas" hipótesis tengan una puntuación baja. El algoritmo DRR emplea técnicas de regresión de arista[1] para desarrollar un algoritmo de adaptación online discriminativo.

2.2.1.1. Adaptación de los factores de escalado mediante DRR en post-edición

El algoritmo DRR emplea una lista de N-best en orden decreciente de verosimilitud calculada a partir de los diferentes modelos dada una oración de entrada $\mathbf{x}$. El primer paso para adaptar $\boldsymbol{\lambda}$ consiste en definir una matriz N x M, $\mathrm{H}_{\mathbf{x}}$, en donde M es el número de características que contiene las funciones de características $\mathbf{h}$ para cada hipótesis en la ecuación 1.7:

$$\mathrm{H}_{\mathbf{x}} = [\mathbf{h}(\mathbf{x}, \mathbf{y}_1), ..., \mathbf{h}(\mathbf{x}, \mathbf{y}_N)]'. \tag{2.6}$$

Consecuentemente, definimos $\mathrm{H}^*_{\mathbf{x}}$ como una matriz de la siguiente forma:

$$\mathrm{H}^*_{\mathbf{x}} = [\mathbf{h}(\mathbf{x}, \mathbf{y}^*), ..., \mathbf{h}(\mathbf{x}, \mathbf{y}^*)]', \tag{2.7}$$

la cual contendrá todas sus filas idénticas e iguales al vector de características de la mejor hipótesis $\mathbf{y}^*$ de la lista de N-best.

A continuación definiremos $\mathrm{R}_{\mathbf{x}}$ como:

[1] También conocida como regularización de Tikhonov.

$$\mathrm{R}_{\mathbf{x}} = \mathrm{H}^{*}_{\mathbf{x}} - \mathrm{H}_{\mathbf{x}}. \tag{2.8}$$

El objetivo de DRR es encontrar el vector $\check{\boldsymbol{\lambda}}_t$ que refleje las diferencias en scores como diferencias en la calidad de las hipótesis, es decir

$$\mathrm{R}_{\mathbf{x}} \cdot \check{\boldsymbol{\lambda}}_t \propto \mathbf{l}_{\mathbf{x}}, \tag{2.9}$$

siendo $\mathbf{l}_{\mathbf{x}}$ un vector columna de N filas representado como

$$\mathbf{l}_{\mathbf{x}} = [\mathbf{l}(\mathbf{y}_1), \ldots, \mathbf{l}(\mathbf{y}_i), \ldots, \mathbf{l}(\mathbf{y}_N)]', \ \forall \mathbf{y}_i \in nbest(\mathbf{x}). \tag{2.10}$$

Por lo tanto definimos el vector $\check{\boldsymbol{\lambda}}_t$ a buscar como:

$$\check{\boldsymbol{\lambda}}_t = \underset{\boldsymbol{\lambda}}{\operatorname{argmin}} \, |\mathrm{R}_{\mathbf{x}} \cdot \boldsymbol{\lambda} - \mathbf{l}_{\mathbf{x}}| \tag{2.11}$$

$$= \underset{\boldsymbol{\lambda}}{\operatorname{argmin}} \, ||\mathrm{R}_{\mathbf{x}} \cdot \boldsymbol{\lambda} - \mathbf{l}_{\mathbf{x}}||^2, \tag{2.12}$$

en donde $||\cdot||^2$ representa la norma Euclídea.

A pesar de que las ecuaciones 2.11 y 2.12 son equivalentes, la ecuación 2.12, gracias a la regresión de arista, $\check{\boldsymbol{\lambda}}_t$ puede resolverse como la solución al sistema sobredeterminado $\mathrm{R}_{\mathbf{x}} \cdot \check{\boldsymbol{\lambda}}_t = \mathbf{l}_{\mathbf{x}}$ dada de la siguiente forma:

$$\check{\boldsymbol{\lambda}}_t = (\mathrm{R}'_{\mathbf{x}} \cdot \mathrm{R}_{\mathbf{x}} + \beta \mathrm{I})^{-1} \mathrm{R}'_{\mathbf{x}} \cdot \mathbf{l}_{\mathbf{x}}, \tag{2.13}$$

donde un valor pequeño de β representa el termino de regularización para estabilizar el producto $\mathrm{R}'_{\mathbf{x}} \cdot \mathrm{R}_{\mathbf{x}}$ y asegurar que este sea invertible.

2.2.1.2. Adaptación de los factores de escalado mediante DRR en IMT

Al aplicar el algoritmo DRR en un escenario IMT, la métrica de calidad que empleamos no es inherente a una sola hipótesis sino a todo un grafo de palabras. Es bastante común evaluar la calidad de un sistema IMT calculando el número de interacciones que un usuario necesita para modificar la hipótesis hasta que se obtenga la oración de referencia y para medir esto utilizamos el WSR [Toselli et al., 2011], una métrica empleada para medir la calidad de un sistema IMT. Cuando se introduce una palabra el sistema IMT modifica el sufijo, provocando que el número de interacciones no se pueda calcular en función de la hipótesis y por tanto se debe calcular simulando el procedimiento de interacción con la ayuda de un grafo de palabras. Esto hace que el DRR tal cual se ha descrito en la sección 2.2.1.1, no se pueda aplicar directamente dentro de un marco IMT.

Es posible pensar que al optimizar cierta métrica de calidad también se optimizaría el número de interacciones necesarias para conseguir obtener la oración de referencia. A pesar de ello, los experimentos descritos en la sección 3.4.1 demuestran que esto no es completamente cierto. Por lo tanto, puesto que la métrica que optimizamos mediante aprendizaje online no depende únicamente de la mejor hipótesis, es necesario modificar la formulación del algoritmo DRR descrita en la sección 2.2.1.1.

En primer lugar sería razonable considerar una lista de N-best grafos de palabras en lugar de una lista de N-best. No obstante el concepto de N-best grafos de palabras es un poco confuso, ya que no hay una forma clara de medir la calidad de un grafo de palabras. Esto provoca que en lugar de calcular una verdadera lista de N-best grafos de palabras calcularemos el grafo de palabras $W_{\boldsymbol{\lambda}^n}(\mathbf{x})$ asociado a cada oración de entrada $\mathbf{x}$ a partir de uno de los conjuntos de pesos $\boldsymbol{\lambda}^n$, de entre los obtenidos previamente de forma semi-aleatoria $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}^1, \ldots, \boldsymbol{\lambda}^n, \ldots, \boldsymbol{\lambda}^N\}$.

Puesto que los pesos se han obtenido de forma semi-aleatoria, los grafos de palabras generados no constituyen una verdadera lista de los N-best grafos de palabras, sino una aproximación de esta, la cual mejoraría con cuantos mas conjuntos de pesos se hayan generado.

Ya que el objetivo del algoritmo DRR es premiar en este caso a un grafo de palabras (realmente premiamos un conjunto $\boldsymbol{\lambda}^n$) con una buena puntuación y penalizar aquellos con una puntuación baja lo que es realmente importante es tener grafos de palabras ($W_{\boldsymbol{\lambda}^n}(\mathbf{x})$) con ambos tipos de puntuaciones. De esta forma propondremos el vector columna l_y cuyas N filas son

$$l_y = [l(W_{\boldsymbol{\lambda}^1}(\mathbf{x})) \ldots l(W_{\boldsymbol{\lambda}^n}(\mathbf{x})) \ldots l(W_{\boldsymbol{\lambda}^N}(\mathbf{x}))]. \tag{2.14}$$

Otro aspecto a considerar de la formulación original del DRR dentro un escenario IMT es la matriz $H_{\mathbf{x}}$, la cual debe ser redefinida debido a que las características a considerar ya no se corresponden a las hipótesis de la lista de N-best, sino con los grafos de palabras creados empleando $\boldsymbol{\Lambda}$. Puesto que un grafo de palabras $W_{\boldsymbol{\lambda}^n}(\mathbf{x})$ no tiene un único conjunto de características sino más bien un vector de características para cada uno de los caminos del grafo de palabras, empleamos el vector de características $\mathbf{h}$ del mejor camino en $W_{\boldsymbol{\lambda}^n}(\mathbf{x})$, es decir, el vector de características de la mejor hipótesis de $W_{\boldsymbol{\lambda}^n}(\mathbf{x})$, para definir $\mathrm{H}_{\mathbf{x}}$. Manteniendo la notación, nombraremos a este vector de características $\mathbf{h}_{\boldsymbol{\lambda}^n}$ y definiremos por tanto $\mathrm{H}_{\mathbf{x}}$ para IMT de la forma

$$\mathrm{H}_{\mathbf{x}} = [\mathbf{h}_{\boldsymbol{\lambda}^1}, \ldots, \mathbf{h}_{\boldsymbol{\lambda}^N}]'. \tag{2.15}$$

Del mismo modo definiremos $\mathrm{H}^*_{\mathbf{x}}$ como

$$\mathrm{H}_{\mathbf{x}^*} = [\mathbf{h}_{\boldsymbol{\lambda}^*}, \ldots, \mathbf{h}_{\boldsymbol{\lambda}^*}]', \tag{2.16}$$

en donde $\mathbf{h}_{\boldsymbol{\lambda}^*}$ es el vector de características perteneciente a la mejor hipótesis del grafo de palabras $W_{\boldsymbol{\lambda}^*}(\mathbf{x})$ y $W_{\boldsymbol{\lambda}^*}(\mathbf{x})$ es el grafo de palabras con mayor calidad de entre los calculados empleando los diferentes $\boldsymbol{\lambda}$ del súper conjunto $\boldsymbol{\Lambda}$, de acuerdo a la métrica empleada correspondiente a IMT, en este trabajo el WSR.

Los conjuntos de pesos semi-aleatorios han sido generados mediante una distribución gaussiana, cuya media se corresponde con el conjunto de pesos obtenido en la etapa de tuning mediante la técnica MERT (Minimum Error Rate Training).

El conjunto de pesos empleado en la creación del grafo de palabras tiene una influencia muy importante en la obtención del mejor camino del mismo, es decir, en la obtención de la traducción de la oración a partir de la cual se creó el grafo de palabras. Debido a esto, partimos del conjunto de pesos baseline, el calculado por MERT, para

generar un número finito de nuevos conjuntos de pesos similares al de MERT. De esta forma evitamos emplear conjuntos de pesos aleatorios que con certeza nos darían en su gran mayoría grafos de palabras con una calidad muy baja e intentamos obtener conjuntos de pesos que nos ofrezcan grafos de palabras de mayor calidad que los dados por el sistema baseline.

El uso de la distribución gaussiana se basa en la posibilidad de que el conjunto de pesos generado mediante MERT no sea el mejor posible para el dominio de las oraciones a traducir debido a que el conjunto de desarrollo empleado para ajustar los pesos mediante MERT pertenece a un dominio diferente.

Por otra parte, todos los grafos de palabras pertenecientes a una misma oración han sido generados a partir de un grafo de palabras creado mediante un software empleando el conjunto de pesos dado por MERT a partir de dicha oración.

2.2.2. Primera aproximación

Esta estrategia, a la que hemos denominado *Primera aproximación* por ser la primera estrategia definida para un escenario IMT, es la más sencilla de todas.

En esta aproximación, el término de actualización $\check{\boldsymbol{\lambda}}_t$ para la oración procesada $\mathbf{x}$ en el instante de tiempo t se corresponderá con el conjunto de pesos $\boldsymbol{\lambda}^*$ al grafo de palabras $W_{\boldsymbol{\lambda}^*}(\mathbf{x})$. $W_{\boldsymbol{\lambda}^*}(\mathbf{x})$ es el grafo de palabras con mayor calidad, de acuerdo a la métrica de calidad WSR, de entre todos los creados para la oración $\mathbf{x}$

Los grafos de palabras son creados empleando para cada uno de ellos un $\boldsymbol{\lambda}$ distinto dado un súper conjunto $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}^1, \ldots, \boldsymbol{\lambda}^n, \ldots, \boldsymbol{\lambda}^N\}$. Cada uno de los conjuntos de pesos pertenecientes a $\boldsymbol{\Lambda}$ han sido generados de forma semi-aleatoria mediante una distribución gaussiana, cuya media se corresponde con el conjunto de pesos obtenido en la etapa de tuning mediante la técnica MERT. Generar los pesos de forma semi-aleatoria permite que los pesos empleados en la generación de los grafos tenga en su promedio una calidad aceptable y permitan, en teoría, mejorar la calidad de las traducciones dadas por el sistema baseline.

2.3. Conclusiones

Aunque existen varias aproximaciones para llevar a cabo el proceso de adaptación online, una de las que mejor rendimiento ha dado consiste en adaptar los pesos del modelo log-lineal. Por este motivo, ha sido la aproximación escogida en este trabajo fin de máster.

Existen muchos algoritmos de aprendizaje online, Passive-agressive [Crammer et al., 2006], Perceptron like [España-Bonet and Marquez, 2010], Discriminative ridge regression [Martínez-Gómez et al., 2012], Bayesian predictive adaptation [Sanchis-Trilles and Casacuberta, 2010], etc. De entre todos ellos, como se puede ver en [Martínez-Gómez et al., 2012], el algoritmo DRR ha sido el que mejores resultados ha dado en un problema de adaptación, por lo que ha sido el algoritmo de aprendizaje online escogido para llevar el problema de adaptación online a un escenario IMT.

Puesto que el algoritmo DRR descrito en la sección 2.2.1.1 utiliza un sistema de reranking para premiar a hipótesis con buena calidad y penalizar a las que tengan mala calidad, necesita de una lista de N-best junto con el WSR asociado de cada una de las hipótesis para poder realizar el reordenamiento.

Por tanto, asumiendo una fuerte correlación entre el TER (o cualquier otra métrica de calidad) de una hipótesis y el WSR con el que se mediría la calidad de un grafo de palabras, se describe el uso del algoritmo DRR según se planteó en un escenario de post-edición, cuyos resultados pueden verse en 3.4.

Del mismo modo la estrategia *Primera aproximación* vista en la sección 2.2.2 se plantea como una alternativa sencilla al algoritmo DRR definido para un escenario de post-edición, en donde esta vez se aborda el problema de adaptación directamente desde una perspectiva de IMT, en donde los resultados pueden verse en la sección 3.4.2.

Por terminar, la última de las estrategias intenta abordar el problema de adaptación online combinando lo mejor de las dos anteriores estrategias. De esta forma se combina el rendimiento del algoritmo DRR y el planteamiento directo para IMT de la estrategia *Primera aproximación*. Para ello en la sección 2.2.1.2 se ha creado una nueva definición del algoritmo DRR cuyos resultados pueden verse en la sección 3.4.

Dentro de un escenario IMT la lista de N-best es sustituida por una lista de N-best grafos de palabras y, puesto que reordenar los grafos de palabras como se haría con una lista de N-best es un concepto un tanto confuso se utiliza como representación del grafo de palabras el mejor camino de este, es decir, la traducción que daría el grafo de palabras para la oración de entrada mediante la cual fue generado.

Aún con todo esto, estamos asumiendo que la mejor hipótesis de un grafo de palabras es capaz de representar correctamente la calidad de un grafo de palabras, algo que a priori puede no ser completamente cierto, tal como que podremos ver en la sección 3.4.

Capítulo 3

Experimentos

Este capítulo del trabajo fin de máster detallara los aspectos más relevantes de los experimentos realizados. En primer lugar se describirán los diferentes corpus empleados. En segundo lugar se detallará la configuración del sistema empleada para realizar los diferentes experimentos. En tercer lugar se definirán las principales métricas de calidad empleadas a lo largo de los experimentos. En cuarto lugar se mostraran los resultados experimentales. Por último se extraerán las principales conclusiones de los experimentos realizados.

3.1. Corpus

En esta sección van a definirse los corpus empleados en los experimentos junto con sus principales características, las cuales podrán visualizarse en forma de tabla.

3.1.1. Europarl

El corpus Europarl [Koehn, 2005] ha sido construido a partir de documentos del Parlamento Europeo e incluye versiones en 11 lenguas europeas: románicas (francés, italiano, español, portugués), germánicas (inglés, holandés, alemán, danés, sueco), griego y finlandés.

El objetivo de la creación de este corpus fue generar un texto de oraciones alineadas para sistemas de traducción automática estadística. Hoy en día, Europarl es un corpus de referencia en SMT y ha sido usado en muchos proyectos de traducción automática. Por este motivo, el corpus Europarl ha sido utilizado para entrenar nuestro sistema. Las principales características del corpus Europal pueden verse en el cuadro 3.1.

Cuadro 3.1: Características del corpus Europarl en donde utilizamos "Entrenamiento" para indicar el corpus de entrenamiento empleado, Europarl inglés-español en su partición del *EMNLP 2011*, "Desarrollo" para el corpus de desarrollo, empleando, Europarl inglés-español en su partición del *NAACL 2006*, "Palabras OoV." para las palabras "fuera del vocabulario", "Long. media oraciones" para la longitud media de las oraciones, "Núm. palabras" para indicar el número de palabras del corpus, k para miles de elementos y M para millones de elementos. Los datos estadísticos fueron recolectados después del preproceso del corpus, tokenizado, lowercasing (paso del corpus a minúsculas) y filtrado de oraciones de más de 40 palabras.

		En	Es
Entrenamiento	Oraciones	1.4M	
	Núm. palabras	28.9M	29.9M
	Long. media oraciones	20.9	21.6
	Vocabulario	85.3k	129.8k
Desarrollo	Oraciones	2000	
	Núm. palabras	58.7k	60.6k
	Long. media oraciones	30.3	29.3
	Palabras OoV.	99	164

3.1.2. News Commentary

El corpus News Commentary está formado a partir de trozos de noticias pertenecientes a un dominio diferente al del corpus Europarl. Puesto que el corpus está generado a partir de diferentes fuentes es ampliamente usado para comprobar el rendimiento de los sistemas en tareas de adaptación, como por ejemplo ocurre en la conferencia *EMNLP2011*. En el cuadro 3.2 podemos ver las estadísticas de este corpus.

Cuadro 3.2: Características del corpus News Commentary utilizado como prueba (*EMNLP 2011*), en donde utilizamos "Palabras OoV" para indicar las palabras "fuera del vocabulario", "Long. media oraciones" para la longitud media de las oraciones, "Núm. palabras" para indicar el número de palabras del corpus, k para miles de elementos y M para millones de elementos. Los datos estadísticos fueron recolectados después del preproceso del corpus, tokenizado y lowercasing (paso del corpus a minúsculas).

		En	Es
Prueba	Oraciones	3003	
	Núm. palabras	79.4k	74.7k
	Long. media oraciones	24.9	26.5
	Palabras OoV.	1708	1549

3.2. Configuración inicial del sistema

Para realizar las experimentaciones propuestas en un escenario IMT real sería necesario disponer de un grupo de traductores humanos profesionales que interactuaran con el sistema IMT. Debido a que corregir cada una de las hipótesis de la forma mostrada en la figura 1.3 supondría un coste muy elevado, resulta imposible emplear traductores humanos para la realizar la evaluación del sistema IMT y es necesario emplear una alternativa que simule la interacción del traductor con el sistema.

La alternativa aquí propuesta consiste en utilizar la traducción de referencia del conjunto de prueba para calcular la calidad de la hipótesis propuesta por el sistema IMT como traducción e imitar la evaluación que un traductor humano haría de dicha traducción.

Para poder discernir las mejoras que nuestro sistema IMT aporta en la calidad de las traducciones lo hemos comparado con un sistema de referencia de SMT, al que llamaremos *baseline*. Este sistema de referencia ha sido entrenado utilizando el conjunto de entrenamiento Europarl [Koehn, 2005] inglés-español, con la partición utilizada en el sexto taller sobre traducción automática estadística de la conferencia EMNLP (*Empirical Methods in Natural Language Processing*) 2011[1] realizado en Edimburgo. Hemos usado el toolkit para MT de código abierto *Moses* [Koehn et al., 2007] en su configuración estándar no monótona (la cual incluye el modelo de reordenamiento *msd-reordering-fe* [Koehn et al., 2005]) y estimado $\boldsymbol{\lambda}$ usando MERT [Och, 2003a] junto con el corpus Europarl en los idiomas inglés-español en su partición establecida en el taller de SMT perteneciente a la conferencia NAACL (*North American Chapter of the Association for Computational Linguistics: Human Language Technologies*) 2006[2] empleado como conjunto de desarrollo. Moses es un sistema de traducción automática que realiza el entrenamiento de los modelos de traducción para cualquier par de lenguas

[1] http://www.statmt.org/wmt11/
[2] http://www.statmt.org/wmt06/

dado un conjunto de oraciones bilingües. Además, Moses implementa un algoritmo de decodificación que permite encontrar de forma eficiente la traducción más probable para una oración de entrada. Los grafos de segmentos requeridos para llevar a cabo las diversas experimentaciones también se crearon utilizando Moses.

Se entrenó un modelo del lenguaje de 5-gramas mediante la herramienta SRILM [Stolcke, 2002] empleando interpolación y suavizado Kneser-Ney [Kneser and Ney, 1995].

Por último, y dado que el propósito es analizar el rendimiento de las diversas estrategias de adaptación online, además del corpus Europarl se utilizó un conjunto de prueba diferente que no pertenece al dominio del Europarl, News Commentary (NC) procedente del taller de traducción automática de la conferencia *EMNLP 2011* [1].

Se realizaron unos experimentos preliminares, con el objetivo de ver el funcionamiento de la estrategia basada en el algoritmo DRR definido para post-edición en la sección 2.2.1.1. En estos experimentos se han empleado diferentes tamaños de la lista de N-best, con valores de N = 500, 5k y 10k para un sistema inicial entrenado sin modelo de reordenamiento y que por tanto cuenta con un vector de 8 características en $\mathbf{h}$ y que sirve como una primera aproximación del algoritmo DRR basado en post-edición a IMT. Además, en el apéndice 4, pueden verse los resultados de los experimentos realizados empleando este sistema y que optimizan una mayor variedad de métricas de calidad, pero que únicamente emplean una porción del conjunto de prueba, concretamente 1.000 de las 3.003 oraciones que lo componen. Por otra parte, una vez analizados los resultados de este sistema inicial, se ha optado por realizar un análisis más extenso para la métrica de calidad escogida, TER, y emplear un sistema esta vez con modelo de reordenamiento y que cuenta con 14 características como los sistemas SMT del estado del arte. Por tanto, para este sistema, se ha utilizado una lista de N-best con valores de N = 200, 500, 1k, 2k, 5k y 10k.

La selección del tamaño de la lista de N-best es un aspecto muy importante, ya que influye en dos aspectos:

1. La selección de la mejor hipótesis $\mathbf{y}^*$. Es posible que la mejor hipótesis respecto a la referencia $\mathbf{y}^\tau$ dada una lista de N-best no aparezca en una lista de menor tamaño N', $\forall N' < N$, es decir, la hipótesis $\mathbf{y}^*$ para una lista de tamaño N puede no estar entre sus primeras N' posiciones provocando que la hipótesis $\mathbf{y}^*$ pueda ser distinta para una lista de tamaño N a la de una de tamaño N', y por tanto su calidad pueda ser mayor.

2. La dificultad de la tarea de predicción se incrementa con el número de posibles hipótesis candidatas. Como se describe en [Martínez-Gomez, 2010] y también hemos observado en este trabajo, la mayoría de hipótesis de una lista de N-best tiene menor calidad que la propuesta por el sistema SMT baseline. Además, debido a que el sistema SMT produce las hipótesis en orden decreciente de puntuación, conforme aumenta el tamaño de la lista de N-best se incrementa el número de hipótesis de baja calidad y se incrementa por tanto la diferencia entre el número de "buenas" y "malas" hipótesis.

[1] http://www.statmt.org/wmt11/

Para la estrategia descrita en la sección 2.2.2 y la aproximación que emplea el algoritmo DRR adaptado a IMT, el cual puede verse en la sección 2.2.1.2, se ha empleado $\mathbf{\Lambda}$ con un tamaño de 501 conjuntos de pesos, o lo que es lo mismo, de 501 grafos de palabras, incluyendo el grafo de palabras que emplea el conjunto de pesos dado por MERT.

La creación de la lista de "N-best" grafos de palabras para cada una de las oraciones a traducir del conjunto de prueba a partir de estas dos estrategias es, con diferencia, el proceso con mayor coste de los experimentos realizados. Esto es debido a que se deben generar las características de cada una de las hipótesis además de simular al usuario tantas veces como talla de la lista de "N-best" grafos de palabras por el número de oraciones del conjunto de prueba, generando por tanto para este trabajo cerca de un millón y medio de traducciones (junto con su consecuente generación del grafo de palabras y cálculo del mejor camino) con su posterior cálculo de WSR.

Los valores de BLEU, TER y WSR de los experimentos reportados se corresponden con el valor dado para todo el conjunto de prueba. Además todos los experimentos se han realizado dentro del marco IMT, por lo que las oraciones de referencia fueron empleadas para adaptar los parámetros del sistema después de haber realizado la traducción de la oración de entrada y evaluado su calidad. Esto hace que la calidad de la traducción reportada para todo el conjunto de prueba se corresponda con la media de cada oración del conjunto de prueba, incluidas las primeras traducciones en las cuales el sistema IMT no estaba todavía adaptado.

Para los experimentos preliminares que utilizan la estrategia descrita en la sección 2.2.1.1 se han empleado en la mayoría de experimentos 8 características en el modelo log-lineal, aunque también se ha obtenido un resultado con 14 para verificar los resultados de la aproximación.

Por otra parte, los diferentes algoritmos de la sección 2.2 utilizan un cierto número de parámetros libres los cuales han sido obtenidos basándose en los experimentos previos de [Martínez-Gómez et al., 2012]:

- Algoritmo DRR definido para post-edición en la sección 2.2.1.1: $\alpha = 0{,}02$ y $\beta = 0{,}01$.
- Algoritmo DRR definido para IMT en la sección 2.2.1.2: $\alpha = 0{,}02$ y $\beta = 0{,}01$, muestreo basado en una distribución gaussiana con $\sigma^2 = 0{,}01$ y $\mu = \boldsymbol{\lambda}_{MERT}$, en donde $\boldsymbol{\lambda}_{MERT}$ es el conjunto de pesos obtenido empleando la técnica MERT en el conjunto de desarrollo descrito arriba.
- Algoritmo *Primera aproximación* en la sección 2.2.2: $\alpha = 0{,}02$, muestreo basado en una distribución gaussiana con $\sigma^2 = 0{,}01$ y $\mu = \boldsymbol{\lambda}_{MERT}$, en donde $\boldsymbol{\lambda}_{MERT}$ es el conjunto de pesos obtenido empleando la técnica MERT en el conjunto de desarrollo descrito arriba.

3.3. Evaluación del sistema IMT

Hoy en día la evaluación automática de las traducciones es un problema muy difícil en traducción automática, provocando que la evaluación automática haya evolucionado a

un campo de investigación con identidad propia. Esto se debe al hecho de que dada una oración de entrada puede existir un gran número diferente de oraciones correctas como salida. Por lo tanto, no hay una oración que pueda ser considerada como exclusivamente correcta, como si ocurre en el caso de reconocimiento del habla o texto. Del mismo modo este problema es aplicable al marco IMT, en el cual se basa este trabajo fin de máster.

En este TFM, las métricas de calidad empleadas para evaluar el rendimiento de los distintos sistemas, o empleadas a lo largo de las diferentes estrategias de adaptación como métrica a optimizar por el algoritmo DRR definido para post-edición, son las siguientes:

- **WSR (Word Stroke Rate)** [Toselli et al., 2011]: WSR se calcula como el cociente entre el número de *número de palabras tecleadas* o *word-stroke*, es decir, las palabras que el usuario necesitaría introducir para obtener la traducción que tiene en mente y el número total de palabras en la oración. En este contexto, una palabra tecleada se interpreta como una sola acción, en la cual el usuario teclea una palabra completa que se asume de coste constante. Además cada palabra tecleada también incluye el coste incurrido por el usuario cuando lee el nuevo sufijo proporcionado por el sistema.

- **WER (Word Error Rate)**: WER calcula el mínimo número de ediciones (sustituciones, inserciones y borrados) necesarios para convertir las oraciones dadas como traducción en la oración de referencia. Suele ser una medida pesimista al aplicarla a traducción automática.

- **BLEU (BiLingual Evaluation Understudy)** [K. Papineni and Zhu, 2002]: Esta puntuación mide la precisión de unigramas, bigramas, trigramas y 4-gramas con respecto al conjunto de traducciones de referencia, aplicando una penalización a las oraciones demasiado cortas, denominada *brevity-penalty*. BLEU no es una tasa de error, por lo que es mejor cuanto mayor es el valor. BLEU puede ser multi-referencia o mono-referencia, pero debido a las restricciones del corpus hemos empleado BLEU mono-referencia.

- **TER (Translation Edit Rate)** [Snover et al., 2006]: TER es una métrica de error empleada en traducción automática que mide el número de ediciones necesarias para modificar la salida del sistema de modo que se convierta en la referencia. Se calcula como el mínimo número de ediciones requeridas para modificar las hipótesis del sistema de forma que estas coincidan con la traducción de referencia, normalizado por el número de palabras de la referencia. En este caso, las posibles ediciones incluyen inserciones, borrados, sustituciones por una sola palabra o reordenamiento de secuencias de palabras. En el artículo original, los autores afirman que el TER mono-referencia se correlaciona tan bien con el juicio humano sobre la calidad de la traducción automática como la variante 4-referencia del BLEU. Al igual que el BLEU, el TER puede ser multi-referencia, aunque hemos empleado en este trabajo TER mono-referencia.

Cabe destacar que los resultados en los que se utiliza el WER como métrica para el algoritmo DRR definido para post-edición, únicamente pueden verse en el apéndice 4 de esta memoria.

A pesar de emplear diferentes métricas de calidad a lo largo de este trabajo, el objetivo principal es maximizar la calidad de los sistemas en función del *Word Stroke Ratio* (WSR). A pesar de ello, en IMT también se han utilizado otras métricas como el *Key Stroke Ratio* (KSR), la cual, a diferencia del WSR que trabaja a nivel de palabra, el KSR trabaja a nivel de carácter. El motivo por el que decidimos emplear WSR en lugar de KSR, es que está última es claramente una medida optimista, ya que en un escenario IMT el usuario se vería abrumado, ya que recibiría una opción de traducción cada vez que pulsara una tecla y además no se tendría en cuenta el tiempo necesario para leer cada una de las hipótesis propuestas por el sistema.

En este trabajo, para realizar el cálculo de y^* de la forma descrita en la sección 2, utilizamos BLEU, TER, WSR, o cualquier otra medida de evaluación para dicho cálculo a pesar de poder reportar posteriormente otra medida de calidad, como ocurre por ejemplo en la sección 3.4.1 al optimizar TER para intentar minimizar el WSR. Una de las consideraciones a destacar es que BLEU suele utilizarse a nivel de documento, ya que de esta forma muestra una alta correlación con la valoración que tendría por parte de un humano. Por contra, al calcularse como una media geométrica para todo un corpus no está bien definido a nivel de oración y por tanto su valor puede ser cero, lo que ocasiona que no siempre es posible calcular y^* al tener todas las hipótesis el mismo valor de calidad, cero. Un ejemplo de este comportamiento al emplear BLEU a nivel de oración puede verse al analizar la oración en inglés "The green house", la cual tendría como oración de referencia en español "La casa verde". En este ejemplo, aunque el sistema pueda detectar la oración de referencia la puntuación de BLEU sería cero debido a que no tiene ningún 4-grama en común con la referencia. Por este motivo este tipo de muestras no han sido consideradas dentro del procedimiento online. Como puede verse en [Martínez-Gomez, 2010], otra consideración a tener en cuenta es que tanto BLEU como TER pueden no estar correlados, es decir, mejoras en BLEU no siempre implican mejoras en TER y viceversa, y como se ve en este trabajo, WER, BLEU o TER tampoco tienen una fuerte correlación con WSR.

Los intervalos de confianza sobre las puntuaciones del baseline han sido calculados usando la técnica *bootstrapping resampling*[3] empleando 1.000 repeticiones y remplazamiento para el cálculo de TER y BLEU. Para el calculo del WSR se ha empleado el mismo procedimiento que el utilizado en [Barrachina et al., 2009]. Por realizar el cálculo de WER y el de las métricas creada a partir de las combinaciones de las cuatro medidas de calidad ya comentadas se ha empleado un simple script.

3.4. Resultados experimentales

En este apartado se mostrarán los resultados más destacados en forma de gráfica o tabla para cada una de las tres estrategias seguidas con el fin de adaptar los pesos del modelo log-lineal de forma online en un escenario IMT. Para ello, dividiremos la

[3]Para mas detalles sobre la técnica de *bootstrapping* puede verse [Koehn, 2004]

sección en cuatro partes, las tres primeras dedicadas a cada una de las tres estrategias propuestas: adaptación mediante el algoritmo DRR visto en la sección 2.2.1.1 y definido para un escenario CAT no interactivo, la estrategia *Primera aproximación* definida en la sección 2.2.2 y la nueva formulación del algoritmo DRR definida en la sección 2.2.1.2 para un escenario interactivo. El último punto tratará la correlación de cada una de las tres métricas de calidad más utilizadas en este trabajo fin de máster: TER, BLEU y WSR.

Antes de mostrar los resultados de las distintas estrategias hay que destacar la importancia del parámetro α en cada una de ellas. El valor de α o ratio de aprendizaje puede verse como el encargado de controlar el tamaño del paso de actualización al procesar una muestra. Este tamaño puede verse como la influencia que tiene la muestra procesada en el valor de los parámetros a predecir. Pequeños valores del ratio de aprendizaje no permiten que valores atípicos, es decir que muestras no representativas, influyan negativamente en el valor de los parámetros, pero provocan que el proceso de adaptación sea más lento. Por contra, valores altos del ratio de aprendizaje permiten que la adaptación se realice a mayor velocidad, pero causa que el algoritmo se vuelva más sensible a las muestras no representativas. Puesto que seleccionar correctamente el valor adecuado para el ratio de aprendizaje no es una tarea sencilla, esto debe realizarse de forma empírica.

3.4.1. Minimizando el WSR mediante el algoritmo DRR definido para post-edición

En esta sección se presentan dos grupos de experimentos en función del tamaño del vector **h**. En primer lugar se mostraran los resultados realizados con un sistema IMT inicial que no emplea modelo de reordenamiento. Por contra, en segundo lugar se presentaran resultados de la misma estrategia pero esta vez obtenidos mediante un sistema IMT con modelo de reordenamiento. Estos nuevos experimentos se realizaron con el objetivo de obtener mayores mejoras que las reportadas con el sistema inicial y tener una referencia para comparar los resultados de las posteriores estrategias de adaptación.

3.4.1.1. Resultados con el sistema inicial: empleando 8 características

Esta estrategia se basa en la hipótesis de que existe una gran correlación entre alguna de las métricas conocidas, o bien en alguna combinación de estas, y el WSR, concretamente entre TER y WSR. Por ello el objetivo es minimizar el WSR mediante la minimización por parte del algoritmo DRR descrito en la sección 2.2.1.1 de alguna métrica de calidad diferente. En un primer momento, para probar el funcionamiento del algoritmo DRR definido para post-edición, se creó un sistema SMT básico entrenado sin el modelo de reordenamiento. De este modo, de una forma rápida y sencilla era posible analizar los resultados al emplear las diferentes métricas como término de minimización dentro del funcionamiento del algoritmo DRR tal como se describe en 2.2.1.1. Además así también era posible ver su variación en función del tamaño de la lista de N-best. Asumiendo que los resultados de optimizar una métrica mediante el algoritmo

DRR en este sistema tiene un comportamiento similar en un sistema del estado del arte, teóricamente permite comprobar la métrica que daría mejores resultados respecto al WSR al intentar optimizarla mediante el algoritmo DRR en un sistema del estado del arte, evitando de esta forma un alto coste computacional y temporal. Del mismo modo, sería posible averiguar el tamaño más apropiado de la lista de N-best a emplear en el algoritmo DRR. Por este motivo, en la primera columna del cuadro 3.3 se pueden ver distintas métricas de calidad, las cuales serán las diversas métricas que el algoritmo DRR intenta minimizar para consecuentemente optimizar el WSR y hallar el mejor conjunto de pesos para la oración dada. En la segunda columna del cuadro 3.3 pueden verse los distintos tamaños de lista de N-best empleados. La elección de estos tamaños se basa en los resultados previamente obtenidos al realizar diferentes experimentos con un subconjunto del mismo corpus de prueba de 1.000 oraciones y distintas métricas de calidad como medida a minimizar mediante el algoritmo DRR. En los resultados empleando dicho subconjunto, los cuales pueden verse en el apéndice 4 de este TFM, se puede observa que pasar de un valor de $N = 5.000$ a $N = 10.000$ no permiten mejorar el WSR del sistema de referencia, por lo que se fija como cantidad recomendada para realizar los experimentos empleando la totalidad del conjunto de prueba un valor de $N = 5.000$, evitando de esta forma un alto coste computacional y temporal. A pesar de ello, al realizar los experimentos empleando todo el conjunto de prueba se intento verificar que 5.000 era también el tamaño adecuado para mejorar el WSR. Por ello, en el la segunda columna del cuadro 3.3 se pueden ver otros tamaños de lista de N-best diferentes a este, en donde se observa que pasar de $N = 500$ a $N = 5.000$ sí mejora notablemente los resultados y que un valor de $N = 10.000$ sigue sin mejorar el WSR del sistema de referencia.

A continuación, en el cuadro[4] 3.3, pueden verse los resultados más relevantes del sistema entrenado sin modelo de reordenamiento y empleando la totalidad del conjunto de prueba.

Cuadro 3.3: Resultados del sistema SMT inicial para el algoritmo DRR definido bajo un escenario de post-edición utilizando diversas métricas para las 3.003 oraciones del conjunto de prueba del corpus News Comentary 2011. k indica miles de elementos. N-best indica el tamaño de la lista de N-best empleada y BP indica la brevity-penalty.

Métrica	**Núm. *N*-best**	**TER**	**WSR**	**BLEU**	**BP**
baseline	-	57.36	59.36	25.34	0.9974
TER	10K	56.37	61.27	20.60	0.82
TER	500	58.57	63.10	17.5311	0.7928
$\frac{\text{TER}}{\text{BP}}$	10K	55.5209	60.1352	23.2713	0.9040
$\frac{\text{TER}}{\text{BP}} - \text{BLEU}$	5K	57.5024	61.4409	21.5282	0.9987
$\frac{\text{WER}}{\text{BP}}$	5K	57.3418	59.3647	25.3651	0.9980

[4]Pueden verse más resultados en forma de tabla en el apéndice de este trabajo fin de máster

Los anteriores experimentos, realizados empleando el sistema SMT inicial y utilizando la totalidad del conjunto de prueba, originalmente intentaban minimizar el WSR a partir del TER. Como se puede observar en el cuadro 3.3, los resultados dependen del tamaño de la lista de N-best, pero aún con un gran valor de N el WSR sigue sin mejorar los resultados del sistema de referencia. A pesar de ello, puede verse como los valores de TER mejoran notablemente los resultados ofrecidos por el sistema de referencia. Por tanto, asumir una fuerte correlación entre WSR y TER, premisa de esta estrategia de adaptación online, no parece del todo correcto. Debido a ello se probaron múltiples combinaciones de métricas para intentar identificar una correlación entre alguna de ellas y el WSR mostrando aquí las mas relevantes. Aún así, no se ha encontrado ninguna combinación de métricas cuya correlación sea lo suficientemente fuerte como para mejorar el WSR a partir de la optimización de esta mediante la estrategia de adaptación utilizada.

Como se puede observar, empleando el sistema preliminar ninguna de las métricas ni combinación de estas ha obtenido mejores resultados que los generados por el sistema de referencia. A pesar de ello, se intento verificar que estos resultados también eran validos en un sistema del estado del arte, ya que se podían reportar más mejoras utilizando un sistema con modelo de reordenamiento. Además, de esta forma también se tendría una referencia solida para comparar los resultados de otras estrategias.

3.4.1.2. Resultados con el sistema final: empleando 14 características

A pesar de que los resultados observados en la sección 3.4.1.1 no consiguieran mejorar los resultados del sistema de referencia, se intentó asegurar que estos resultados no se verían modificados por un sistema IMT del estado del arte, el cual cuenta con un modelo de reordenamiento y por tanto con un vector de 14 características como $\mathbf{h}$. Por otra parte también se pensó que para poder valorar correctamente las mejoras de posibles futuras estrategias era necesario disponer de resultados de un sistema IMT del estado del arte. Para ello, esta vez, viendo que ninguna de las métricas empleadas en la sección 3.4.1.1 lograban el objetivo propuesto, se intento optimizar el valor de $\boldsymbol{\lambda}$ a través de la optimización de la métrica de calidad TER mediante el algoritmo DRR definido en la sección 2.2.1.1, intentando consecuentemente acabara minimizándose el WSR tal como se había propuesto originalmente.

Como puede verse en la figura 3.1, el WSR se comporta mejor aumentando el tamaño de la lista de N-best como ya pasaba con los experimentos vistos en el cuadro 3.3, pero no mejoran el resultados del sistema de referencia, ya que únicamente se solapa con él cuando el valor de α, el cual representa el ratio de aprendizaje, es muy pequeño, es decir, cuando la adaptación online esta prácticamente desactivada. Respecto al BLEU, los resultados tampoco muestran mejoras sobre el sistema de referencia independientemente del tamaño de la lista de N-best empleado, aunque sí que se puede observar que el BLEU se comporta mejor cuanto más grande es el valor de N. El hecho de que el BLEU no mejore respecto al sistema de referencia quizás pueda deberse a que BLEU no es la métrica que el algoritmo DRR está utilizando para medir la calidad de las distintas hipótesis, sino TER cuyos resultados se pueden ver en la figura 3.2, y por tanto no es la métrica que el algoritmo está minimizando. Además, puede ser que el

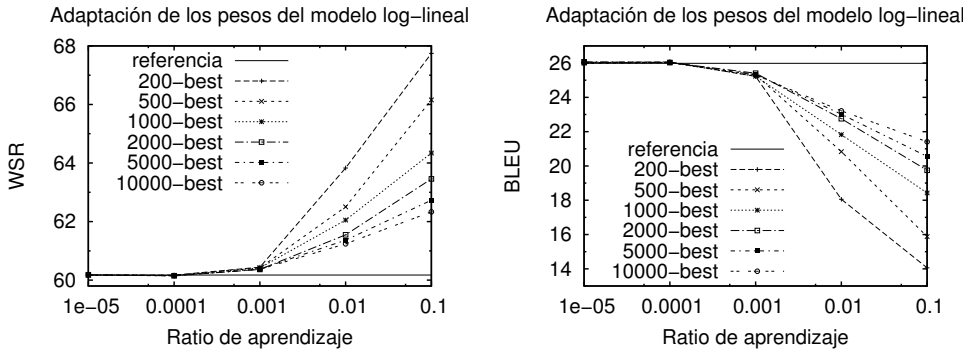

Figura 3.1: Influencia de α en el rendimiento del algoritmo para el conjunto de prueba NC11 en función del tamaño de la lista de N-best para WSR y BLEU.

TER y BLEU no estén fuertemente correlacionados para este experimento.

Por otra parte, en la figura 3.2 pueden verse tres gráficas que muestran resultados de TER. Aquí, conviene destacar varios aspectos clave. En esta estrategia de adaptación online la lista de N-best tiene dos posibles funcionalidades. En primer lugar, mediante estas hipótesis el algoritmo DRR es capaz de optimizar el valor de $\boldsymbol{\lambda}$. En segundo lugar, la hipótesis $\mathbf{y}^*$ es posible calcularla de dos formas, seleccionando la mejor hipótesis de la lista de de N-best o escogiendo el camino más probable de un grafo de palabras creado a partir de los $\boldsymbol{\lambda}$ calculados por el DRR y la oración $\mathbf{x}$ a traducir. Por este motivo es posible calcular el TER a partir de dos conjuntos de hipótesis propuestas como traducción, las obtenidas seleccionando la mejor hipótesis de la lista de N-best para cada una de las oraciones del conjunto de prueba y las obtenidas a partir del mejor caminos de cada uno de los grafos de palabras. Puesto que intentamos adaptar los pesos del modelo log-lineal en un sistema IMT asumiremos que todos los resultados son calculados a partir de las traducciones generadas a partir de los grafos de palabras si explícitamente no se especifica lo contrario.

Por tanto, en al figura 3.2, la gráfica superior izquierda muestra resultados en función de la calidad de las mejores hipótesis de la lista de N-best que el algoritmo DRR ha seleccionado al optimizar $\boldsymbol{\lambda}$ para cada una de las oraciones del conjunto de prueba. Por otra parte, la gráfica superior derecha muestra los resultados obtenidos al emplear las hipótesis extraídas a partir de los grafos de palabras, los cuales han sido generados utilizando los $\boldsymbol{\lambda}$ optimizados por el algoritmo DRR para cada una de las oraciones.

Los resultados de TER que pueden verse en ambas gráficas de la figura 3.2, como ya ocurrían en la sección 3.4.1.1, presentan bastante mejor comportamiento, llegando a mejorar 1.1 puntos el sistema de referencia. Como puede verse, al medir la calidad empleando TER, de nuevo hay una gran influencia por parte del tamaño de la lista de N-best en el rendimiento del algoritmo DRR, comportándose nuevamente mejor la métrica empleada para medir la calidad, en este caso TER, cuanto mayor es el valor de N. Además pueden verse notables diferencias entre los valores de TER calculados a

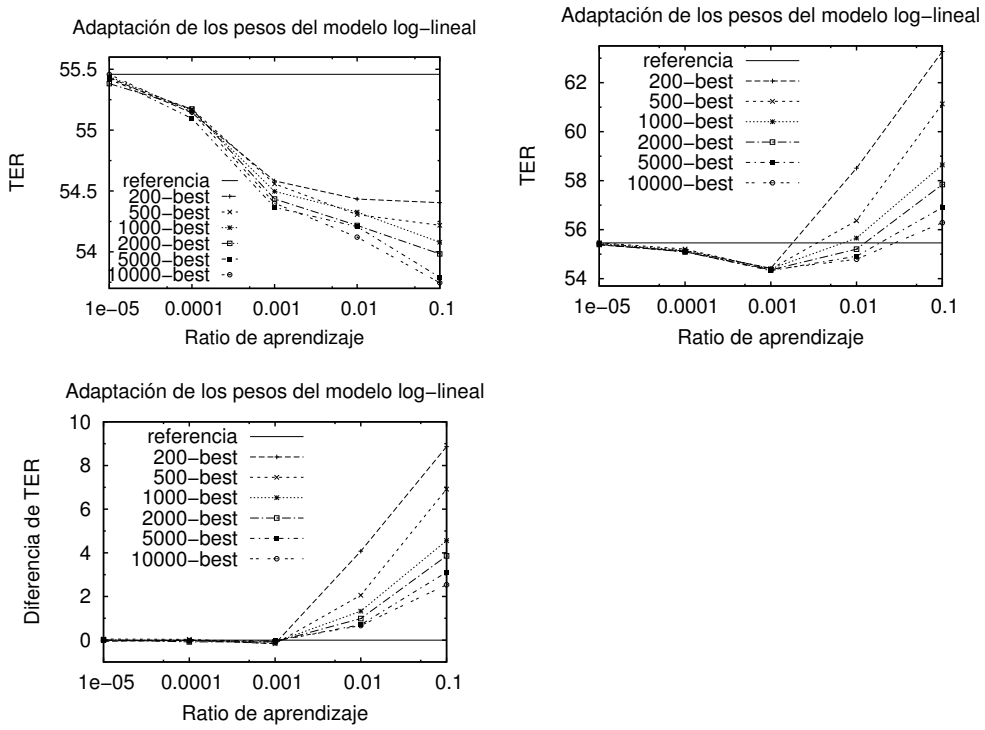

Figura 3.2: Influencia de α en el rendimiento del algoritmo para el conjunto de prueba NC11 en función del tamaño de la lista de N-best para TER. La gráfica superior izquierda muestra resultados de calcular el TER sobre las mejores hipótesis de la lista de N-best, mientras que la gráfica superior derecha muestra resultados de calcular el TER de las mejores hipótesis extraídas a partir de los grafos de palabras. La gráfica inferior muestra la diferencia de TER entre las dos gráficas superiores, es decir, entre el TER de la lista de N-best y de los grafos de palabras en función de alfa y del valor de N.

partir de la lista de N-best y el calculado a partir de los grafos de palabras. En primer lugar puede verse en la gráfica superior izquierda que los resultados de TER para todos los valores de α siempre mejora el obtenido mediante el sistema de referencia, a excepción de cuando el valor de α es muy pequeño y la adaptación online esta prácticamente desactivada. Por contra, los resultados de la gráfica superior derecha muestran que el ratio de aprendizaje si tiene un valor determinante para poder mejorar los resultados del sistema de referencia, ya que a diferencia de la gráfica superior izquierda, aquí el TER no se comporta bien para valores de α grandes. Estas diferencias de TER entre los dos sistemas, las cuales pueden verse en la gráfica inferior de la figura 3.2, se deben principalmente a que mientras que en la gráfica superior izquierda

el TER depende únicamente de las traducciones, en la gráfica superior derecha el TER dependerá de los grafos de palabras y no únicamente de la traducción extraída de cada uno de estos, es decir, su camino más probable. Como puede observarse en la figura 3.2, esto provoca diferencias de TER de hasta 9 puntos cuando el ratio de aprendizaje es grande, siendo este un buen valor para mejorar el TER calculado a partir de la las mejores hipótesis de la lista de N-best, pero no para mejorarlo mediante las traducciones generadas a partir de los grafos de palabras, lo cual no es en ninguno de los dos casos el objetivo de este trabajo.

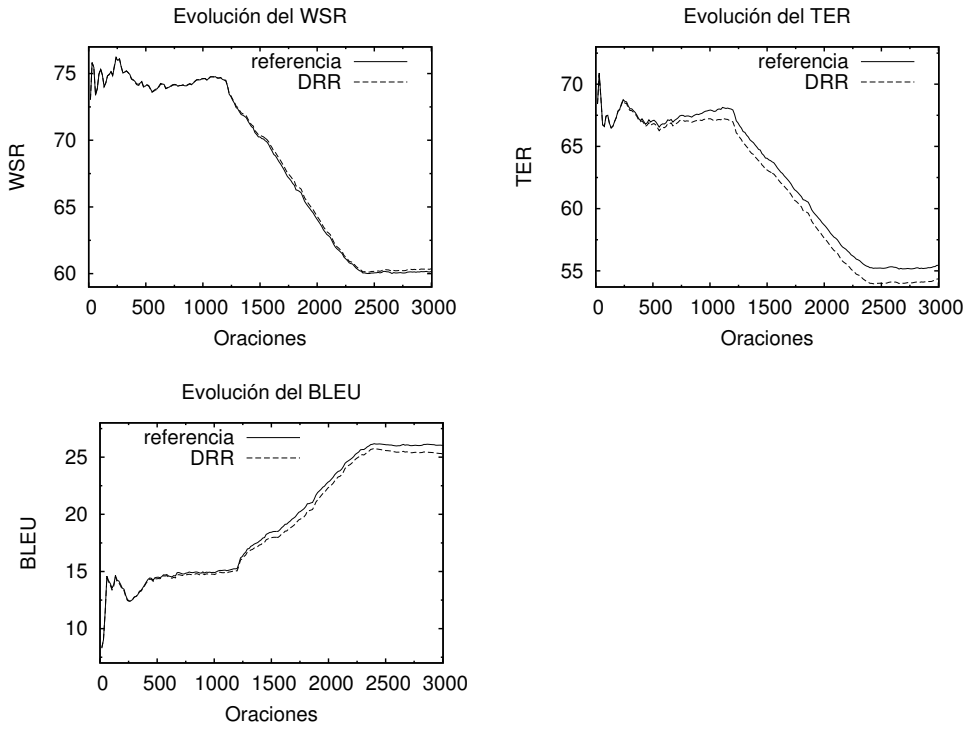

Figura 3.3: Evolución del WSR, TER y BLEU cuando adaptamos $\boldsymbol{\lambda}$ dentro del conjunto de prueba NC11. Solamente se han dibujado 1 de cada 15 puntos para facilitar la visualización de las gráficas. El valor de α empleado ha sido de 0.001. Los resultados han sido generados empleando un sistema IMT.

En las gráficas de la figura 3.3 puede verse la evolución de la calidad de un sistema IMT medido en WSR, TER y BLEU durante el procesado de cada una de las oraciones del conjunto de prueba. Estas gráficas muestran el valor medio acumulado de cada una de estas métricas para el subconjunto de oraciones procesadas hasta el momento. Para estas tres gráficas, el valor de α empleado ha sido de 0.001, es decir, el valor con el cual se han obtenido los mejores resultados de WSR. Como puedo observarse, las curvas de

evolución del sistema propuesto y el sistema de referencia tienen un comportamiento muy similar, aunque poseen ciertas diferencias. En cuento a la evolución del WSR, puede verse que en ambos sistemas el comportamiento es muy similar hasta procesar la oración 1.250 en donde las diferencias comienzan a ser visibles. A partir de dicha oración, el sistema IMT ligeramente empeora los resultados. Sin embargo, en la gráfica de evolución del TER, los resultados de los dos sistemas empiezan a diferenciarse a partir de la oración procesada número 300, y en este caso el sistema IMT paulatinamente va mejorando su comportamiento superando en todo momento al sistema de referencia. En cuanto a la gráfica de BLEU, se puede apreciar que las diferencias comienzan aproximadamente a partir de la oración 400, en donde el sistema IMT va empeorando ligeramente.

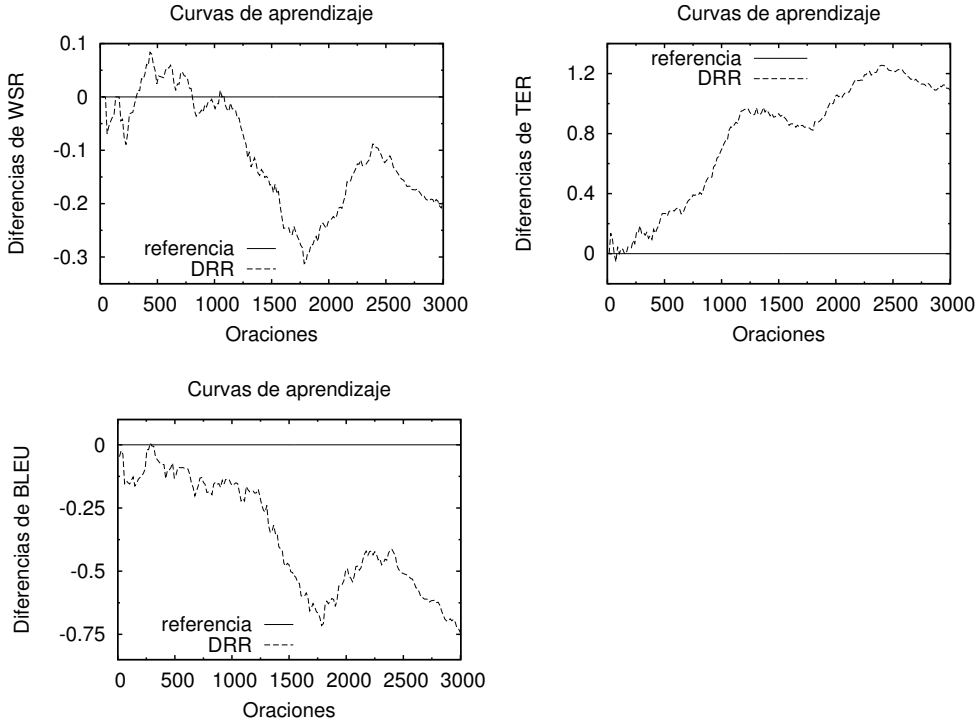

Figura 3.4: Curvas de aprendizaje cuando adaptamos $\boldsymbol{\lambda}$ dentro del conjunto de prueba NC11. Solamente se han dibujado 1 de cada 15 puntos para facilitar la visualización de las gráficas.El valor de α empleado ha sido de 0.001. Los resultados han sido generados empleando un sistema IMT.

La figura 3.4 muestra tres gráficas con las curvas de aprendizaje del sistema IMT propuesto, en donde los valores positivos representan mejoras del sistema IMT respecto al sistema de referencia y valores negativos representan perdidas de calidad del

sistema IMT respecto al de referencia. Tanto las mejoras como las perdidas logradas hasta una oración determinada representan una media de las mejoras o perdidas hasta dicha oración inclusive. Como puede verse en la gráfica superior izquierda, el WSR en el sistema IMT obtiene mejoras despreciables en el intervalo de oraciones aproximado [400,1.250]. Posteriormente a este intervalo, el sistema IMT comienza a generar traducciones ligeramente de menor calidad que el sistema de referencia. En la gráfica que muestra las mejoras de calidad en términos de TER, se puede ver como el comportamiento de este es bueno, por lo que el sistema IMT ha ido aprendiendo al procesar cada una de las oraciones, mejorando casi desde el principio la calidad dada por el sistema de referencia. La gráfica inferior de la figura 3.4, la cual mide la calidad del sistema en BLEU, presenta una situación casi opuesta a la gráfica que muestra la curva de aprendizaje evaluada mediante TER, ya que en este caso el sistema no aprende al procesar cada una de las oraciones del conjunto de prueba, sino que ligeramente va empeorando. Como ya se ha comentado, esto posiblemente se debe a que el sistema IMT esta optimizando TER y en este caso, este parece no estar bien correlacionado con el BLEU.

Por otra parte, en el cuadro 3.4 pueden verse los resultados más significativos en términos de la calidad medida empleando WSR, TER y BLEU. Estos resultados se corresponden con los experimentos realizados empleando el valor de α que mejores resultados dio respecto a la calidad evaluada mediante WSR, el cual fue de 0.001.

Cuadro 3.4: Efecto de variar α, medido mediante el WSR, TER y BLEU, en el esfuerzo de un traductor humano para generar una traducción de calidad en IMT. Los resultados han sido obtenidos mediante el algoritmo DRR definido bajo un paradigma de post-edición. TER *NB* indica el valor de TER medido a partir de las traducciones extraídas de la lista de N-best, mientras que TER *GP* indica el valor de TER calculado sobre las traducciones extraídas a partir de los grafos de palabras. α indica el ratio de aprendizaje. El tamaño de la lista de N-best utilizada en los resultados mostrados es de 10.000.

Método de optimización	α	WSR	TER *NB*	TER *GP*	BLEU
baseline	-	60.2	55.5	55.5	26
DRR (sección 2.2.1.1)	0.001	60.4	54.4	54.4	25.3

En el cuadro 3.5 se muestra dos ejemplos de traducciones propuestas tanto por el sistema IMT como por el de referencia. En ellos se puede ver como, concretamente para estas dos oraciones, el esfuerzo por parte del corrector humano disminuye al emplear la estrategia de adaptación propuesta. De esta forma se puede observar que el número de ediciones necesarias para convertir las traducciones propuestas por el sistema en las traducciones de referencia, es de 7 y 5 para el primer y segundo ejemplo respectivamente. De esta forma las ediciones se ven reducidas en 2 y 4 respectivamente para ambos ejemplos respecto al sistema de referencia.

Cuadro 3.5: Comparación de dos traducciones generadas por el sistema SMT de referencia y el sistema IMT que integra el algoritmo DRR definido para un escenario de post-edición. "In." indica el número de interacciones necesarias para convertir la oración dada por el sistema en la oración de referencia en un sistema IMT, "Oración" es la oración de entrada del sistema $\mathbf{x}$, "Referencia" se corresponde con $\mathbf{y}^{\tau}$, "baseline" es la oración $\mathbf{y}$ del sistema de referencia, "DRR" es la oración $\mathbf{y}$ obtenida mediante la técnica de adaptación de $\boldsymbol{\lambda}$ vista en 2.2.1.1. Tanto en DRR como en baseline sólo se muestra la primera hipótesis de un proceso de IMT.

Métodos	**Oraciones**	**In.**
Oración	they also plan to co-finance two movies based on " the hobbit , " along with warner bros .	
Referencia	también planean cofinanciar dos películas basadas en " el hobbit " , junto con warner bros .	
baseline	son también dos películas plan basado en la cofinanciación de la hobbit ", "junto con bros warner .	9
DRR	también previsto cofinanciar dos películas hobbit basado en " la " , junto con bros warner .	7
Oración	" finally , ordinary criminals and corrupt inspectors are at the lowest level . "	
Referencia	" finalmente , los criminales comunes y los inspectores corruptos están en el nivel más bajo " .	
baseline	por último , los delincuentes y " inspectores corruptos se encuentran en el nivel más bajo . "	9
DRR	" finalmente , los delincuentes y los inspectores corruptos se al mínimo nivel . "	5

3.4.2. Minimizando el WSR mediante la estrategia Primera aproximación

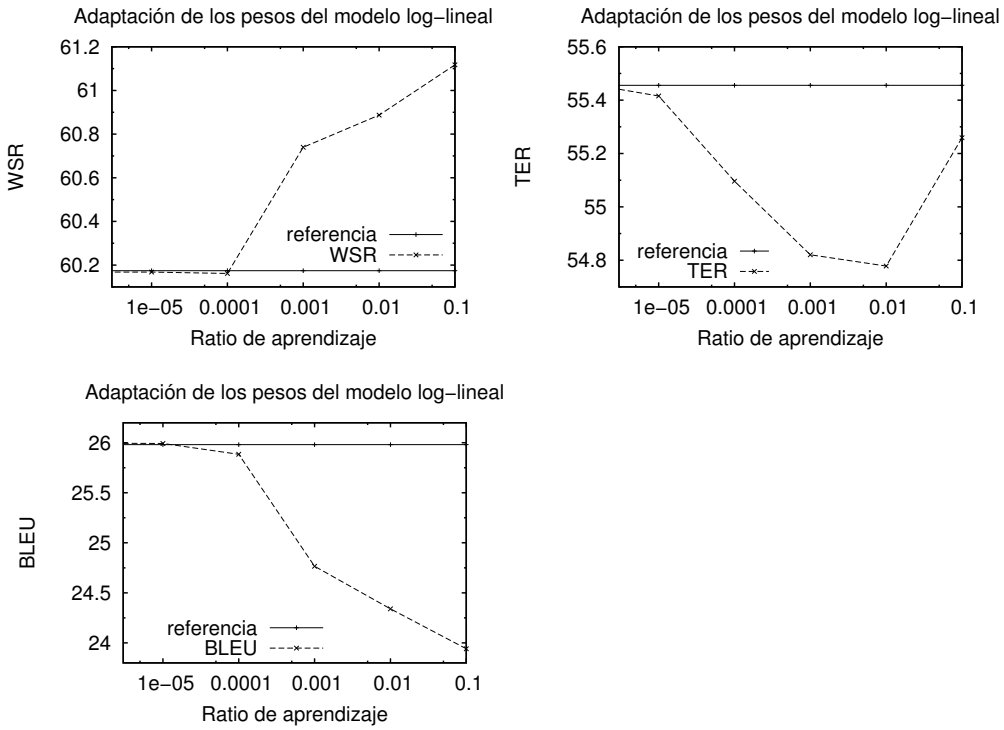

Figura 3.5: Influencia de α en el rendimiento del algoritmo para el conjunto de prueba NC11. La cantidad de conjuntos de pesos semi-aleatorios utilizados fue de 501, incluyendo el conjunto de pesos dados por MERT.

Una vez analizados los experimentos realizados empleando la estrategia de adaptación basada en el algoritmo DRR descrita en la sección 2.2.1.1, se vio que quizás el enfoque era erróneo, ya que si el objetivo era minimizar el WSR, esta era la métrica que directamente se debería intentar minimizar.

Para realizar una primera aproximación a esta nueva asunción, en la sección 2.2.2 se definió esta nueva estrategia.

Como puede verse a continuación en la figura 3.5 que muestra la influencia del ratio de aprendizaje en esta estrategia, los resultados no fueron los esperados. La gráfica superior izquierda muestra la calidad del sistema IMT medida en WSR. Aquí se puede ver como a medida que se decrementa el valor de α el WSR mejora su comportamiento, llegando a mejorar, aunque de una forma despreciable, el WSR del sistema de referencia con un $\alpha = 0.0001$, es decir, cuando la actualización de los $\boldsymbol{\lambda}$ es sumamente pequeña en cada actualización. Por otra parte el valor de TER del sistema IMT sí que mejora

respecto al de referencia para varios valores de α, obteniendo el mejor resultado con $\alpha = 0.01$ en donde se mejora en más de 4 décimas los resultados del sistema de referencia. El comportamiento del BLEU parece ser similar al del WSR, mejorando el comportamiento de este conforme se va decrementando el valor de α hasta llegar a solaparse con los resultados del sistema de referencia para valores muy pequeños, en donde la adaptación online se encuentra casi desactivada.

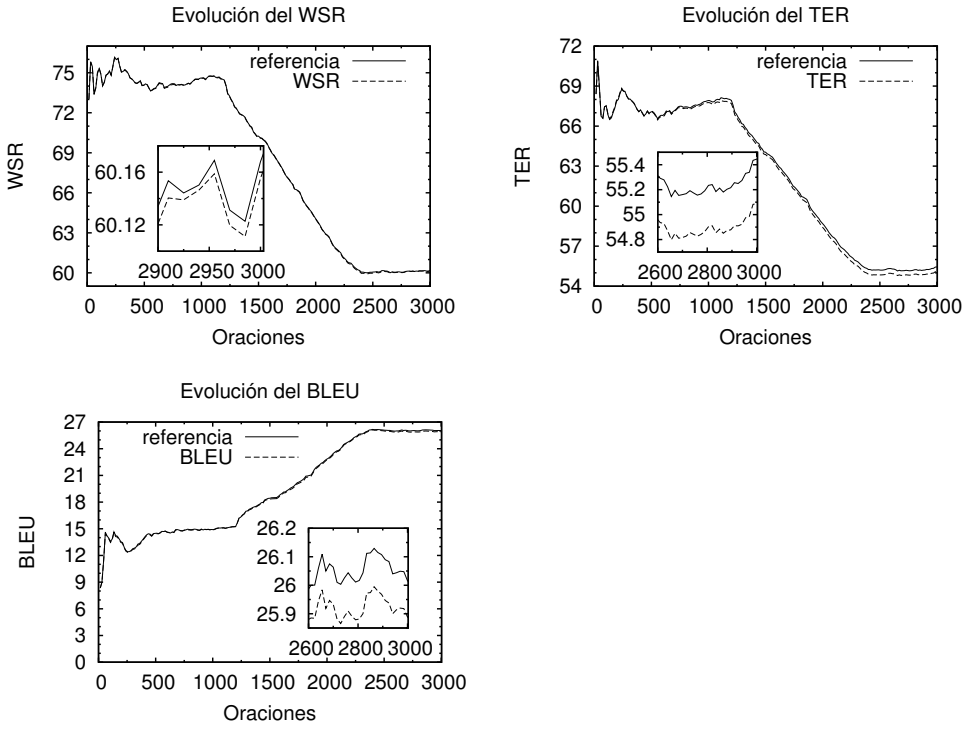

Figura 3.6: Evolución del WSR, TER y BLEU cuando adaptamos $\boldsymbol{\lambda}$ dentro del conjunto de prueba NC11. Solamente se han dibujado 1 de cada 15 puntos para facilitar la visualización de las gráficas. El valor de α empleado ha sido de 0.0001.

La figura 3.6 muestra la evolución del WSR, TER y BLEU durante el procesado del corpus de prueba. Las tres gráficas han sido generadas empleando un $\alpha = 0.0001$, ya que éste es el valor con el que se obtuvo un menor WSR. En la gráfica que muestra la evolución del WSR puede verse como el comportamiento de ambos es casi idéntico. Esto se debe a que la actualización de los parámetros del modelo log-lineal modifica estos de una forma muy sutil, provocando que apenas se pueda ver una mejora insignificante de menos de media décima. Los resultados de la gráfica de TER muestran que el comportamiento de éste para ambos sistemas es muy parecido, aunque puede verse

como paulatinamente el TER del sistema IMT aumenta ligeramente la mejora respecto al sistema de referencia, llegando a superarlo en 4 décimas al acabar de procesar todo el conjunto de prueba. La gráfica de la evolución del BLEU no muestra mejoras del sistema IMT respecto al de referencia y de nuevo puede verse como el comportamiento de ambos sistemas es muy parecido, comportándose mejor el BLEU en ambos casos conforme se van procesando más muestras.

Las curvas de aprendizaje pueden verse en la figura 3.7, donde las tres gráficas muestran las mejoras del sistema IMT en sus respectivas métricas de calidad como valores positivos en el eje y. De nuevo para dibujar estas tres gráficas se empleo el valor de α que mejores resultados dio de WSR, 0.0001. Aquí puede verse como el WSR del sistema IMT, aunque de forma poco significativa, mejora los resultados del sistema de referencia a partir de la oración 550 aproximadamente. En la gráfica que muestra la curva de aprendizaje del TER puede verse como a partir de la oración 350 aproximadamente, el comportamiento de este en el sistema IMT mejora considerablemente, mostrando una tendencia siempre ascendente, y por tanto, aumentando con cada oración procesada la mejora respecto al sistema de referencia. La curva de aprendizaje del BLEU presenta muchas más irregularidades que la del TER y no muestra una tendencia clara, aunque puede verse que el BLEU del sistema IMT no mejora el sistema de referencia prácticamente en ningún momento.

El cuadro 3.6 sintetiza los resultados más significativos para esta estrategia de adaptación online.

Cuadro 3.6: Efecto de variar α, medido mediante el WSR, TER y BLEU, en el esfuerzo de un traductor humano para generar una traducción de calidad en IMT. Los resultados han sido obtenidos mediante la estrategia *Primera aproximación*. α indica el ratio de aprendizaje y PA las siglas de *Primera aproximación*. La cantidad de conjuntos de pesos semi-aleatorios utilizados fue de 501, incluyendo el conjunto de pesos dado por MERT. Tanto en PA como en baseline sólo se muestra la primera hipótesis de un proceso de IMT.

Método de optimización	α	WSR	TER	BLEU
baseline	-	60.2	55.5	26
PA (sección 2.2.2)	0.0001	60.2	55.1	25.9

En el cuadro 3.7 se pueden ver dos ejemplos de traducciones propuestas por el sistema IMT que sigue la estrategia de adaptación *Primera aproximación* y el sistema de referencia. Como puede verse en el primer ejemplo, la traducción generada por el sistema IMT propuesto coincide con la dada por el sistema baseline o de referencia. Además, el número de interacciones necesarias para obtener la traducción de referencia coincide en ambos sistemas lo que indica que el sistema IMT no ha logrado encontrar mejores sufijos en las posteriores interacciones con el humano, por tanto para esta oración el sistema IMT no logra disminuir el esfuerzo necesario para corregir la tra-

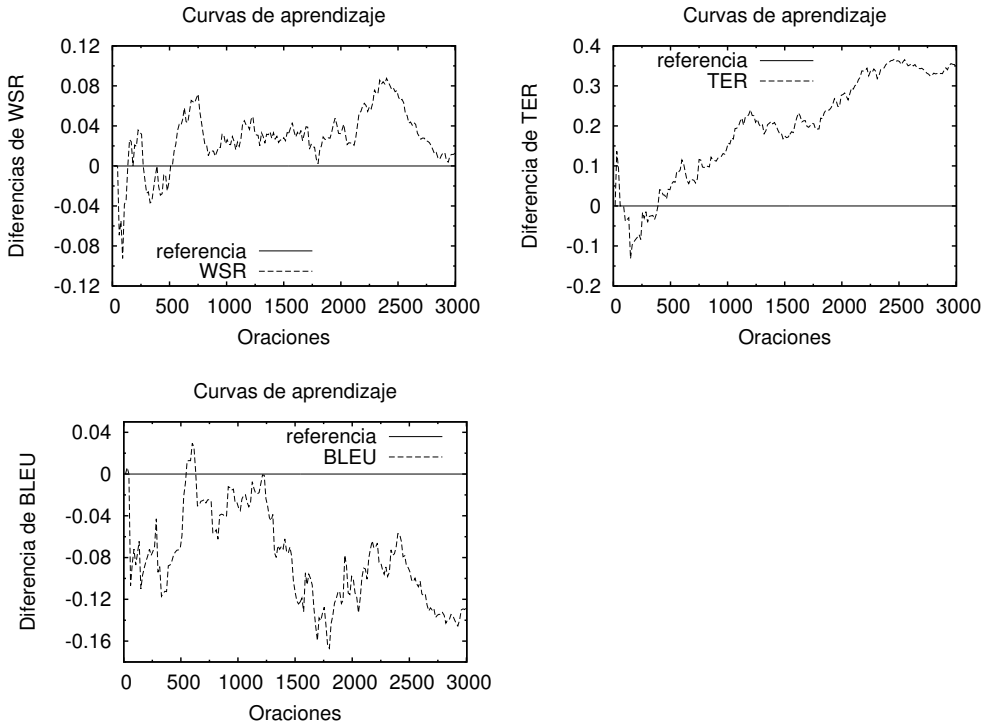

Figura 3.7: Curvas de aprendizaje cuando adaptamos $\boldsymbol{\lambda}$ dentro del conjunto de prueba NC11. Solamente se han dibujado 1 de cada 15 puntos para facilitar la visualización de las gráficas. El valor de α empleado ha sido de 0.0001.

ducción. Por contra, en el segundo ejemplo, a pesar de que la traducción propuesta por ambos sistemas vuelve a ser la misma, el esfuerzo necesario para corregir la traducción generada por el sistema IMT es menor que el necesario para corregir la oración propuesta por el sistema de referencia. Esto se debe a que a pesar de haber modificado las puntuaciones de las aristas de los grafos de palabras mediante la adaptación online empleando la estrategia *Primera aproximación*, el mejor camino, es decir, el camino más probable y por tanto la traducción propuesta por el sistema es la misma. A pesar de ello, al haber optimizado las puntuaciones de transiciones de las aristas se consigue corregir la traducción con menor esfuerzo, ya que a pesar de que el mejor camino del grafo de palabras es el mismo, el grafo resultante es mejor.

Cuadro 3.7: Comparación de dos traducciones generadas por el sistema SMT de referencia y el sistema IMT creado siguiendo la estrategia *Primera aproximación*. "In." indica el número de interacciones necesarias para convertir la oración dada por el sistema en la oración de referencia en un sistema IMT, "Oración" es la oración de entrada del sistema $\mathbf{x}$, "Referencia" se corresponde con $\mathbf{y}^{\tau}$, "baseline" es la oración $\mathbf{y}$ del sistema de referencia, "PA" es la oración $\mathbf{y}$ obtenida mediante la técnica de adaptación de $\boldsymbol{\lambda}$ vista en la sección 2.2.2.

Métodos	**Oraciones**	**In.**
Oración	they also plan to co-finance two movies based on " the hobbit , " along with warner bros .	
Referencia	también planean cofinanciar dos películas basadas en " el hobbit " , junto con warner bros .	
baseline	son también dos películas plan basado en la cofinanciación de la hobbit ", "junto con bros warner .	9
PA	son también dos películas plan basado en la cofinanciación de la hobbit ", "junto con bros warner .	9
Oración	" finally , ordinary criminals and corrupt inspectors are at the lowest level . "	
Referencia	" finalmente , los criminales comunes y los inspectores corruptos están en el nivel más bajo " .	
baseline	por último , los delincuentes y " inspectores corruptos se encuentran en el nivel más bajo . "	9
PA	por último , los delincuentes y " inspectores corruptos se encuentran en el nivel más bajo . "	8

3.4.3. Minimizando el WSR mediante el algoritmo DRR definido para IMT

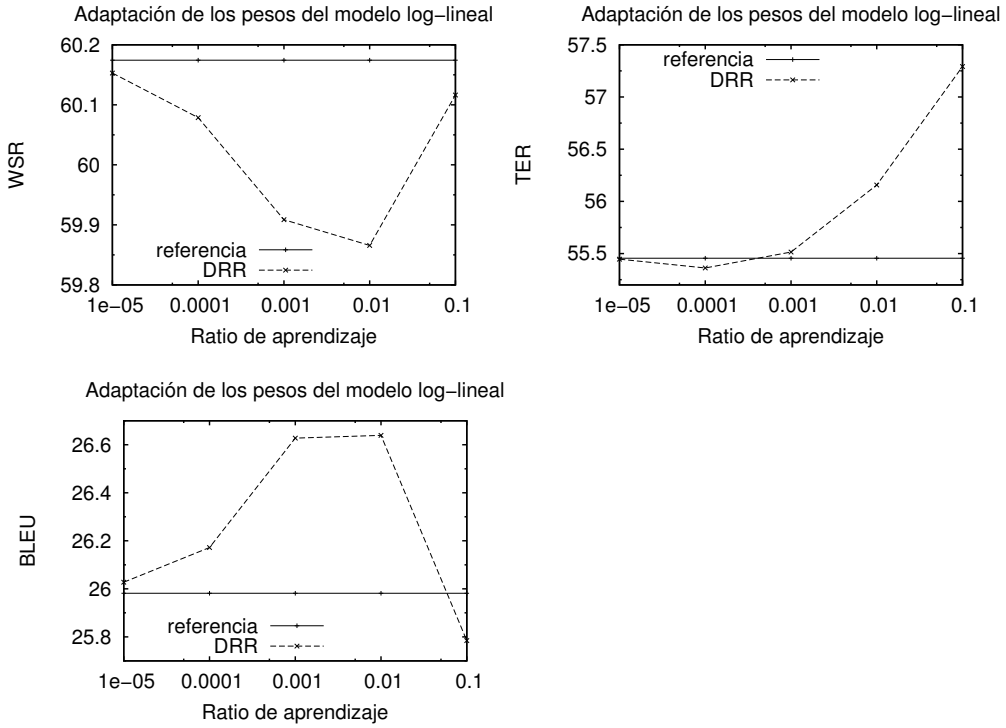

Figura 3.8: Influencia de α en el rendimiento del algoritmo para el conjunto de prueba NC11. La cantidad de conjuntos de pesos semi-aleatorios utilizados fue de 501, incluyendo el conjunto de pesos dado por MERT.

Hasta el momento se han empleado dos estrategias de adaptación online, las cuales pueden verse en las secciones 2.2.1.1 y 2.2.2. Mediante estas dos aproximaciones no se ha logrado mejorar el WSR. Para intentar subsanar las deficiencias de las dos anteriores estrategias, es decir, intentar minimizar el WSR mediante otras medidas y emplear un paso de actualización muy simple para llevar a cabo la optimización de los pesos, se ha propuesto el algoritmo DRR definido directamente para IMT. Con esto se pretende conseguir mejoras parecidas a las obtenidas en TER en la sección 3.4.1.2.

En las gráficas de la figura 3.8 se puede ver que mediante esta técnica el WSR ha mejorado alrededor de 3 décimas empleando un valor de $\alpha = 0.01$ el resultado dado por el sistema de referencia y aproximadamente 5 los resultados obtenidos al utilizar la definición original del algoritmo DRR como se puede ver en la sección 3.4.1.2, lo que puede considerarse como algo muy alentador y da pie a futuras investigaciones en esta linea. Esto se debe principalmente a que en esta ocasión la métrica minimizada por el

algoritmo DRR es el WSR y no ninguna otra como ocurría en la sección 3.4.1 con el uso del TER. Esto provoca que el TER obtenido por el sistema IMT únicamente obtenga ligeras mejoras respecto al sistema de referencia comparándolas con las diferencias de más de un punto que pueden verse en la sección 3.4.1.2. Además, también pueden verse diferencias en los resultados de BLEU con los obtenidos hasta ahora, los cuales en esta ocasión mejoran hasta en 6 décimas, empleando un valor de $\alpha = 0.01$, los resultados obtenidos por el sistema de referencia. En las tres gráficas puede verse como el ratio de aprendizaje tiene una importante relevancia en los resultados, ya que por ejemplo los resultados de TER varían más de dos puntos en función de este, mientras que en BLEU y WSR, las variaciones son de 8 y 3 décimas respectivamente.

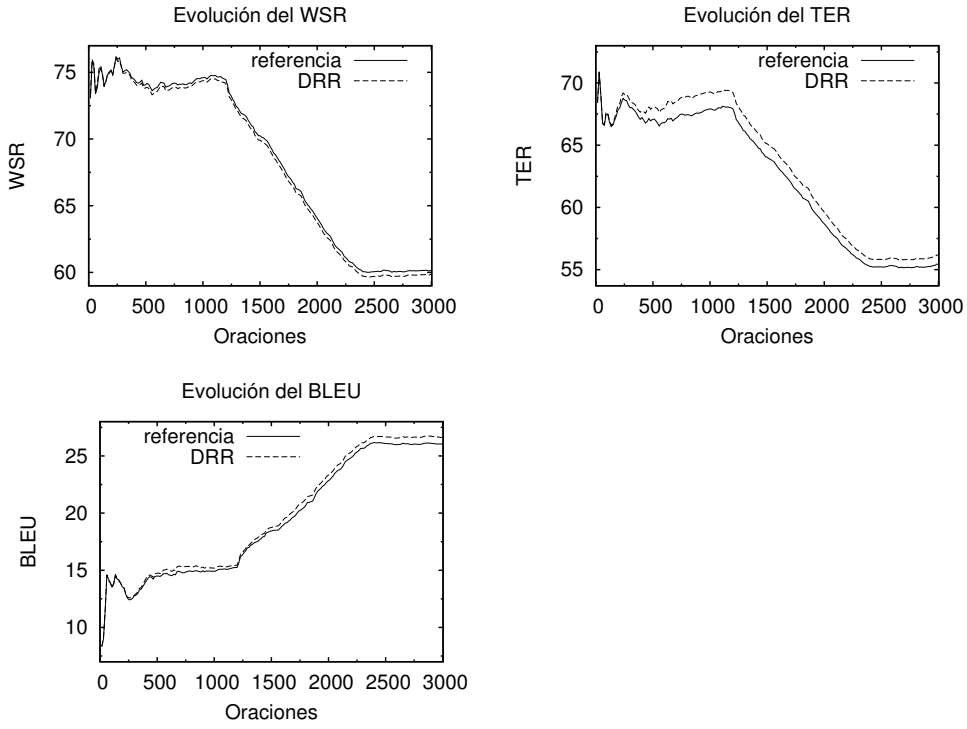

Figura 3.9: Evolución del WSR, TER y BLEU cuando adaptamos $\boldsymbol{\lambda}$ dentro del conjunto de prueba NC11. Solamente se han dibujado 1 de cada 15 puntos para facilitar la visualización de las gráficas. La cantidad de conjuntos de pesos semi-aleatorios utilizados fue de 501, incluyendo el conjunto de pesos dado por MERT.

En la figura 3.9 pueden verse las tres gráficas correspondientes a la evolución del WSR, TER y BLEU en función del número de oraciones procesadas. Para estas tres gráficas, el valor de α empleado ha sido de 0.01, es decir, el valor con el cual se han

obtenido los mejores resultados de WSR. El la gráfica que representa la evolución del WSR puede verse que hasta la oración 250 aproximadamente, el comportamiento del WSR en el sistema IMT definido y el de referencia es prácticamente idéntico. Por el contrario, conforme se procesan más oraciones del conjunto de prueba, las diferencias entre los dos sistemas se acentúan hasta llegar a las 3 décimas al procesar el conjunto de prueba en su totalidad. De forma similar ocurre en la gráfica de TER, en donde los resultados de ambos sistemas parecen solaparse hasta la oración 250 en donde comienzan a hacerse visibles las diferencias, llegando a empeorar el sistema IMT en 7 décimas aproximadamente respecto al sistema de referencia. En la gráfica que muestra la evolución del BLEU puede verse como las diferencias entre el comportamiento del BLEU entre los dos sistemas empieza a distinguirse a partir de la oración 250 aproximadamente, y como a partir de este valor, del mismo modo que ocurre en las otras dos gráficas, comienza a mejorar su comportamiento. Además, al procesar la totalidad del conjunto de prueba, el BLEU obtenido por el sistema IMT puede verse que mejora en 6 décimas aproximadamente el generado por el sistema de referencia.

La figura 3.10 muestra la curva de aprendizaje del sistema IMT y del sistema de referencia, en donde en cada una de las tres gráficas se puede ver el rendimiento de cada uno de ellos en función de una métrica de calidad distinta, WSR, TER y BLEU. Para estas tres gráficas, el valor de α empleado ha sido de nuevo de 0.01 ya que es el valor con el cual se han obtenido los mejores resultados de WSR. En la primera de las gráficas se puede ver como con un pequeño número de oraciones procesadas del conjunto de prueba el sistema IMT mejora respecto al baseline, y que al acabar de procesar todo el conjunto de prueba, la mejora lograda es de aproximadamente 3 décimas. La gráfica que muestra la curva de aprendizaje de TER presenta como a partir de la oración procesada número 1.000 la diferencia entre los dos sistemas empieza a acortarse. La curva de aprendizaje de la gráfica que muestra el BLEU, a pesar de las irregularidades que muestra, parece indicar que el sistema IMT mejora notablemente en función del número de oraciones procesadas, alcanzando aproximadamente las 6 décimas de mejora una vez procesado todo el corpus.

Por último, en el cuadro 3.8 pueden verse los resultados más significativos en términos de la calidad medida empleando WSR, TER y BLEU. Los resultados de este cuadro han sido obtenidos empleando el valor de α que mejores resultados dio en términos de calidad medida mediante WSR, que en este caso fue 0.01.

El cuadro 3.9 muestra dos ejemplos de los resultados obtenidos empleando la estrategia de adaptación que implementa el algoritmo DRR definido en la sección 2.2.1.2. Aquí se puede apreciar como el sistema IMT ofrece notables mejoras en dos oraciones de prueba, minimizando de esta forma el esfuerzo de un humano para corregir la traducción que originalmente proporciona el sistema de forma notable.

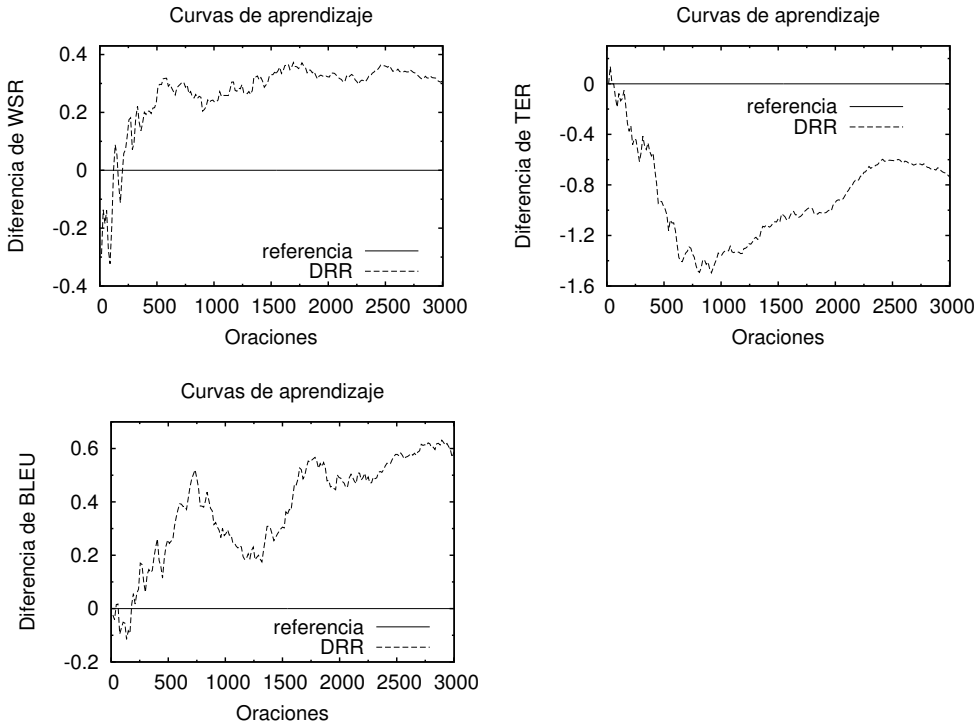

Figura 3.10: Curvas de aprendizaje cuando adaptamos $\boldsymbol{\lambda}$ dentro del conjunto de prueba NC11. Solamente se han dibujado 1 de cada 15 puntos para facilitar la visualización de las gráficas. La cantidad de conjuntos de pesos semi-aleatorios utilizados fue de 501, incluyendo el conjunto de pesos dado por MERT.

Cuadro 3.8: Efecto de variar α, medido mediante el WSR, TER y BLEU, en el esfuerzo de un traductor humano para generar una traducción de calidad en IMT. Los resultados han sido obtenidos mediante el algoritmo DRR definido bajo un paradigma de IMT. α indica el ratio de aprendizaje. La cantidad de conjuntos de pesos semi-aleatorios utilizados fue de 501, incluyendo el conjunto de pesos dado por MERT.

Método de optimización	α	WSR	TER	BLEU
baseline	-	60.2	55.5	26.0
DRR (sección 2.2.1.2)	0.01	59.9	56.2	26.6

Cuadro 3.9: Comparación de dos traducciones generadas por el sistema SMT baseline y el sistema IMT que integra el algoritmo DRR definido para un escenario IMT. "In." indica el número de interacciones necesarias para convertir la oración dada por el sistema en la oración de referencia en un sistema IMT, "Oración" es la oración de entrada del sistema $\mathbf{x}$, "Referencia" se corresponde con $\mathbf{y}^{\tau}$, "baseline" es la oración $\mathbf{y}$ del sistema baseline, "DRR" es la oración $\mathbf{y}$ obtenida mediante la técnica de adaptación de $\boldsymbol{\lambda}$ vista en 2.2.1.2.

Métodos	**Oraciones**	**In.**
Oración	they also plan to co-finance two movies based on " the hobbit , " along with warner bros .	
Referencia	también planean cofinanciar dos películas basadas en " el hobbit " , junto con warner bros .	
baseline	son también dos películas plan basado en la cofinanciación de la hobbit ", "junto con bros warner .	9
DRR	también se plan para cofinanciar dos películas basado en " la hobbit " , junto con bros warner .	5
Oración	" finally , ordinary criminals and corrupt inspectors are at the lowest level . "	
Referencia	" finalmente , los criminales comunes y los inspectores corruptos están en el nivel más bajo " .	
baseline	por último , los delincuentes y " inspectores corruptos se encuentran en el nivel más bajo . "	9
DRR	" , por último , los delincuentes y los inspectores corruptos se encuentran en el nivel más bajo " .	5

3.4.4. Correlación WSR, TER y BLEU

Una de las estrategias de adaptación seguidas en este trabajo final de máster se basa en la adaptación de los pesos del modelo log-lineal mediante el algoritmo de aprendizaje online DRR definido para un escenario de post-edición en la sección 2.2.1.1.

Esta estrategia se basa en la correlación entre las diferentes métricas más usadas o combinaciones de estas y el WSR para intentar minimizar este último. Como se ha podido observar en la sección 3.4.1, no siempre hay una fuerte correlación entre las diferentes medidas de calidad, por lo que optimizar una de ellas no implica siempre la optimización de cualquier otra.

Este fenómeno ha sido estudiado en múltiples ocasiones por la comunidad científica de traducción automática, existiendo varios documentos sobre la correlación de diversas métricas de calidad y el criterio de un humano [Denkowski and Lavie, 2010, Paula Estrella et al., 2004].

En este apartado vamos a analizar la correlación entre las métricas de calidad WSR, TER y BLEU dentro de un entorno no adaptativo, es decir, empleando el mismo $\boldsymbol{\lambda}$ para traducir todo el conjunto de prueba NC11 inglés→español. Para obtener estos valores se han utilizado las traducciones extraídas a partir de los grafos de palabras generados al realizar las experimentaciones definidas en las secciones 2.2.1.2 y 2.2.2. Estos grafos de palabras han sido creados a partir de 501 $\boldsymbol{\lambda}$ distintos, los cuales fueron generados de forma semi-aleatoria como ya se vio en la sección 2.2.1.2.

Para medir la correlación entre las distintas métricas usaremos el coeficiente de correlación de Pearson, el cual se define de la siguiente forma:

$$\rho_{C_1,C_2} = \frac{cov(C_1, C_2)}{\sigma_{C_1}\sigma_{C_2}}, \tag{3.1}$$

en donde C_1 y C_2 representan las dos métricas de calidad de las cuales se desea calcular el coeficiente de correlación.

Cuadro 3.10: Coeficiente de correlación de Pearson para las medidas de calidad WSR, TER y BLEU obtenidos al traducir el conjunto de prueba NC11 utilizando 501 conjuntos de pesos semi-aleatorios $\boldsymbol{\lambda}$, incluyendo el conjunto de pesos dados por MERT. Se utilizan las iniciales W para indicar WSR, T para TER y B para BLEU.

$\rho_{W,T}$	$\rho_{W,B}$	$\rho_{T,B}$
0.8719285	-0.9424423	-0.8183807

Los resultados que pueden verse en el cuadro 3.10 indican una gran correlación entre WSR, TER y BLEU. Esta correlación significaría que mediante la optimización de cualquiera de las tres métricas sería posible optimizar cualquiera de las otras, pero como hemos visto en la sección 3.4.1 esto no es del todo cierto en experimentaciones reales en donde se lleva a cabo adaptación online.

Esta fuerte correlación puede deberse en parte a la forma en la que los $\boldsymbol{\lambda}$ han sido extraídos, ya que estos no son conjuntos de pesos generados de forma aleatoria, sino que han sido extraídos mediante un criterio de selección semi-aleatorio acotando el espacio de búsqueda de estos y seleccionando conjuntos de pesos de cierta calidad.

Otra de las posibles causas de esta alta correlación puede deberse a la escasa cantidad de valores empleados en el calculo del coeficiente de Pearson, ya que 501 conjuntos de pesos es una muestra poco significativa. Como se aprecia en las gráficas de la figura 3.11 que muestran la correlación de TER-BLEU y WSR-TER, en estas aparece una nube de puntos fuera de la linea de mayor densidad para valores altos de WSR ,TER y bajos de BLEU que podría aumentar al incrementar el número de $\boldsymbol{\lambda}$ y por tanto hacer disminuir el valor del coeficiente de Pearson.

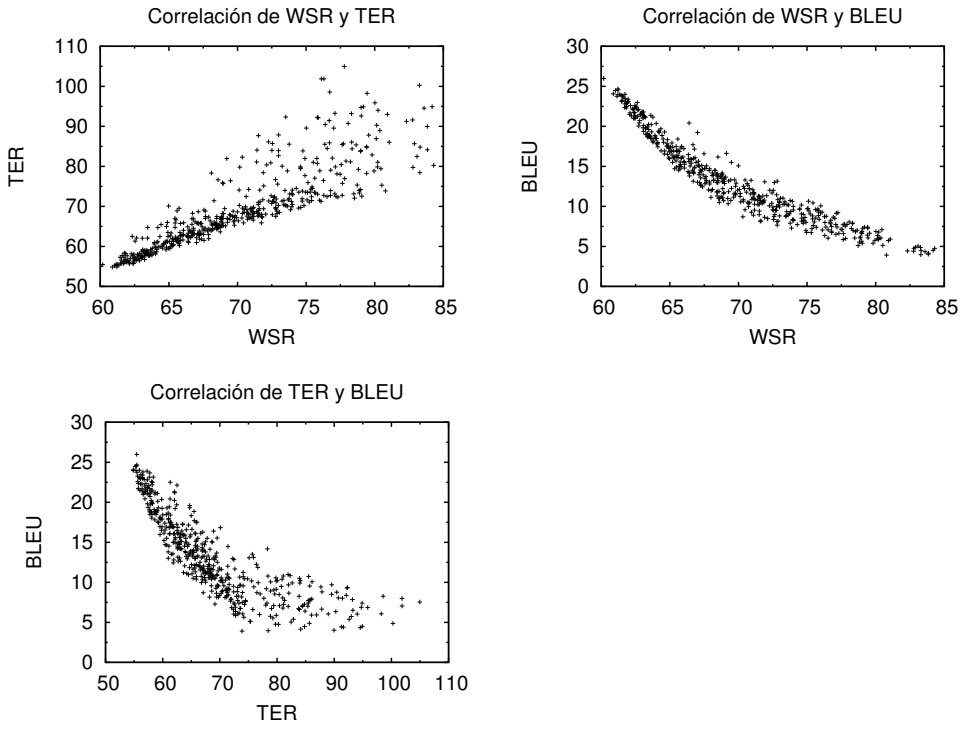

Figura 3.11: Correlación de los valores de WSR, TER y BLEU obtenidos al traducir el conjunto de prueba NC11 utilizando 501 conjuntos de pesos semi-aleatorios $\boldsymbol{\lambda}$, incluyendo el conjunto de pesos dado por MERT.

3.5. Conclusiones

En este capítulo se han podido ver los resultados de las tres estrategias de adaptación online planteadas, así como los valores de correlación para las métricas de calidad WSR, TER y BLEU en un entorno no adaptativo.

La primera de las estrategias emplea el algoritmo DRR definido para un escenario de post-edición para intentar minimizar la métrica de calidad TER e indirectamente optimizar el WSR. Para ello se asume que el WSR está fuertemente correlacionado con el TER. Para validar tal asunción se empleó un sistema básico de IMT entrenado sin modelo de reordenamiento.

Puesto que esta aproximación únicamente logró minimizar el TER no se obtuvo los resultados esperados. Por lo tanto se intentaron optimizar otras métricas de calidad mediante el algoritmo DRR, de forma que la correlación de estas con el WSR fuera mayor y por tanto se obtuvieran mejores resultados de WSR. A pesar de no conseguir lograr el objetivo, se pudo comprobar el funcionamiento de la estrategia de una forma sencilla y analizar los resultados en función del valor de los parámetros utilizados, como es el caso del tamaño de la lista de N-best. Gracias a ello se pudo comprobar mediante esta serie de experimentos iniciales, los cuales emplean un vector de características de 8 componentes, que el tamaño más apropiado para la lista de N-best era de 5.000 hipótesis por oración, ya que más allá de este valor los resultados mejoraban sólo ligeramente a costa de un mayor coste computacional y temporal.

Para intentar mejorar los resultados mediante esta estrategia de adaptación, se entreno un sistema esta vez con modelo de reordenamiento. Para estos experimentos se empleó como métrica a optimizar por el algoritmo DRR el TER como originalmente se había intentado, ya que con ninguna de las anteriores se había logrado mejorar el WSR. De nuevo, a pesar de no lograrse mejorar el WSR con éxito, se consiguió mejorar el TER. Además, se realizó el mismo experimento pero esta vez para calcular el TER a partir de las mejores hipótesis que el algoritmo DRR ha dado a partir de la lista de N-best y no a partir de las mejores hipótesis extraídas de los grafos de palabras como se realiza en el resto de experimentos. A pesar de que se logró como mejor resultado el mismo valor en ambos sistemas, el valor de TER al calcularlo sobre las mejores hipótesis de la lista de N-best siempre mejoraba al del sistema de referencia independientemente del valor del parámetro α, mientras que al calcularlo a partir de los grafos de palabras, las mejoras en TER dependen completamente del valor de α. Esto se debe a que el TER en un grafo de palabras no depende exclusivamente del mejor camino de cada uno de los grafos, es decir, de las mejores hipótesis sino que depende de todo el grafo de palabras. Por contra, el calculo del TER sobre las hipótesis de la lista de N-best depende únicamente de dichas hipótesis.

Vistos los resultados de la estrategia anterior se planteó esta vez la posibilidad de minimizar de forma directa el WSR en lugar de hacerlo indirectamente a partir de la minimización de otra métrica como se había realizado anteriormente. Para ello se definió la estrategia a la que denominamos *Primera aproximación*, de modo que se pudiera realizar un primer acercamiento a este cambio de paradigma. Esta estrategia únicamente mejoró de forma despreciable el WSR dado por el sistema de referencia para un valor de α muy pequeño, en donde la adaptación online estaba prácticamente desac-

tivada. Esto puede deberse principalmente a la simplicidad del paso de actualización de los parámetros del modelo log-lineal que plantea esta estrategia.

Por ello, se intentó combinar lo mejor de las dos estrategias anteriores, es decir, abordar el problema de minimizar el WSR mediante una minimización directa de este y emplear un algoritmo de optimización de $\boldsymbol{\lambda}$ más potente que el definido en la estrategia *Primera aproximación*, esperando mejorar de esta forma los resultados. Por lo tanto se definió una nueva estrategia de adaptación online, la cual se basa en una reformulación del algoritmo DRR, esta vez definido para IMT. Mediante esta estrategia se consiguió mejorar aproximadamente 3 décimas el WSR del sistema de referencia. Estos resultados pueden considerarse muy alentadores, ya que a pesar de no ser una mejora muy notable sí abre un nuevo frente de investigación en el campo de la IMT.

Por otra parte, se muestra los valores de correlación de WSR y TER, WSR y BLEU, y por último TER y BLEU para un sistema IMT en el que no se emplea adaptación online. Aquí, se puede ver que estos resultados muestran una correlación alta entre dichas métrica. A pesar de ello, hay que tener en cuenta, que estos resultados han sido obtenidos empleando 501 $\boldsymbol{\lambda}$ seleccionados de forma semi-aleatoria en un espacio de búsqueda restringido. Por ello, es posible que aunque estos experimentos muestren una fuerte correlación entre las métricas de calidad, los resultados en donde se emplea adaptación online, y el espacio de búsqueda no esta restringido demuestren que en determinados problemas reales de adaptación online esta fuerte correlación no tiene por qué existir.

Capítulo 4

Conclusiones y trabajo futuro

En este trabajo fin de máster se ha analizado la aplicabilidad de los algoritmos Driscriminative Ridge Regression (DRR) vistos en la sección 2.2.1 y la estrategia de adaptación a la cual hemos denominado *Primera aproximación* dentro de un entorno IMT simulado para actualizar los pesos del modelo log-lineal de un sistema de traducción automática estadística dentro del estado del arte. Este sistema SMT es el encargado de generar las distintas traducciones que propondrá el sistema IMT una vez este haya adaptado los pesos del modelo log-lineal.

En los experimentos reportados se utilizó la versión ya conocida del algoritmo DRR definido para un escenario de post-edición que tan buenos resultados dio en él. A pesar de esto, esta definición falla dentro de un escenario IMT no ofreciendo resultados esperados aún con los múltiples intentos por encontrar una métrica de calidad que albergara una fuerte correlación con la empleada en IMT, el WSR. Debido a esto, se plantea una nueva definición del algoritmo DRR para un escenario IMT, en donde esta vez el método de reranking no reordena las diferentes hipótesis para una oración de entrada $\mathbf{x}$, sino que intentará reordenar los distintos grafos de palabras pertenecientes a una misma oración. Esta reordenación se realiza empleando como representación de la calidad de cada uno de los grafos de palabras su mejor camino, ya que para la estrategia definida los grafos de palabras son la base del sistema IMT. Bajo esta novedosa definición, el algoritmo DRR ofrece resultados muy alentadores, a pesar de no llegar a mostrar mejoras tan notables como las vistas en un escenario de post-edición en [Martínez-Gómez et al., 2012] bajo su definición original, sí abre el camino a futuras investigaciones bajo esta linea.

Por otra parte, los resultados experimentales realizados mediante la estrategia *Primera aproximación* no son tan favorables como los de la estrategia anterior, ya que quizás se trate de un algoritmo demasiado sencillo para abordar un problema tan complejo como adaptar los pesos del modelo log-lineal en un escenario IMT.

Como trabajo futuro, nos gustaría estudiar el uso de la nueva definición del algoritmo DRR empleando un $\mathbf{\Lambda}$ más grande, es decir, con mayor número de conjuntos de pesos, y que a su vez estos fueran extraídos de forma semi-aleatoria a partir de

técnicas de muestreo mas complejas. Una posibilidad de interés sería generar $\mathbf{\Lambda}$ mediante *Markov chain Monte Carlo* [Bishop, 2007] o a través del algoritmo Downhill simplex [Nelder and Mead, 1965]. Además bajo un mayor número de conjuntos de pesos creados mediante alguna de estas técnicas es posible obtener grafos de palabras de mayor calidad que los obtenidos hasta el momento, lo que podría provocar que los resultados de el algoritmo DRR mejoraran notablemente debido a encontrar conjuntos de pesos que se ajustasen mejor a un conjunto de test específico.

Para concluir, cabe destacar que este trabajo fin de máster ha dado lugar a una publicación en una conferencia internacional [López-Salcedo et al., 2012].

Apéndice

En este apéndice puede verse el cuadro 4.1 con los resultados obtenido al emplear la estrategia de adaptación online descrita en la sección 2.2.1.1 mediante un sistema IMT preliminar entrenado sin modelo de reordenamiento. En él se pueden ver las distintas métricas empleadas para intentar minimizar el WSR de forma indirecta asumiendo una gran correlación de este con ellas.

Cuadro 4.1: Resultados del sistema SMT inicial para algoritmo DRR definido bajo un escenario de post-edición utilizando diversas métricas para 1.000 frases del conjunto de test del corpus News Comentary 2011. k indica miles de elementos. N-best indica el tamaño de la lista de N-best empleada y BP indica la brevity-penalty.

Métrica	**Núm. N-best**	**TER**	**WSR**	**BLEU**	**BP**
referencia	-	68.7828	73.1460	15.0902	0.9928
TER	5k	66.1873	74.1791	12.8300	0.8282
$\frac{\text{TER}}{\text{BP}}$	5k	67.7835	73.4043	14.6796	0.9643
$\frac{\text{TER}}{\text{BP}}$	10k	67.4408	73.3824	14.7419	0.9580
$\frac{\text{TER}}{\text{BP}} - \text{BLEU}$	5k	69.0546	73.4130	14.8098	0.9897
TER-BLEU	5k	66.3790	74.4243	12.5417	0.8253
WER-BLEU	5k	66.9693	74.9059	11.5550	0.7893
$\frac{\text{WER}}{\text{BP}} - \text{BLEU}$	5k	68.4936	73.3692	14.6957	0.9816
$\frac{\text{WER+TER}}{\text{BP}} - \text{BLEU}$	5k	70.3730	73.5225	14.5545	1
WER	5k	66.5805	74.5950	12.0398	0.7998
$\frac{\text{WER}}{\text{BP}}$	5k	68.5455	73.2817	14.8703	0.9835
$\text{WER} + \frac{\text{TER}}{\text{BP}}$	5k	66.0746	73.7063	13.8931	0.8850
$\text{WER} + \frac{\text{TER}}{\text{BP}} - \text{BLEU}$	5k	66.3707	73.8377	13.5659	0.8824
$\frac{\text{WER+TER}}{\text{BP}}$	5k	69.9087	73.3892	14.9372	0.9976

Lista de símbolos y abreviaciones

Abreviación	Descripción	Definición
MT	Machine Translation	página 1
SMT	Statistical Machine Translation	página 1
$\mathbf{x}$	oración en el idioma origen	página 3
$\mathbf{y}$	oración en el idioma destino	página 3
$\hat{\mathbf{y}}$	mejor traducción según el modelo	página 3
$\mathbf{a}$	conjunto de variables de alineamiento ocultas	página 3
$\mathcal{A}(\mathbf{x}, \mathbf{y})$	conjunto de posibles alineamientos entre $\mathbf{x}$ y $\mathbf{y}$	página 3
M	número de modelos en el modelo log-lineal	página 3
$h_m(\mathbf{x}, \mathbf{y})$	función de puntuación	página 3
λ_m	pesos de $h_m(\mathbf{x}, \mathbf{y})$	página 3
$g(\mathbf{x}, \mathbf{y})$	puntuación de la hipótesis	página 3
$\mathbf{h}(\cdot\|\cdot)$	vector de características	página 3
$\boldsymbol{\lambda}$	vector de pesos	página 3
PB	phrase-based	página 6
$\tilde{x}_k$	segmento en el idioma origen	página 6
$\tilde{y}_k$	segmento en el idioma destino	página 6
K	número de segmentos en el modelo de PB	página 6
I	longitud de la oración de entrada $\mathbf{x}$	página 6
J	longitud de la oración de salida $\mathbf{y}$	página 6
$C(\tilde{x})$	número de segmentos de $\tilde{x}$	página 7
$C(\tilde{x}, \tilde{y})$	número de veces que $\tilde{x}$ e $\tilde{y}$ se extraen en el corpus	página 7
MERT	Minimum Error Rate Training	página 8

Abreviación	Descripción	Definición
CAT	Computer Assisted Translation	página 10
IMT	Interactive Machine Translation	página 12
$\mathbf{p}$	prefijo de la oración	página 12
s_l	sufijo de la oración erróneo	página 12
k	palabra introducida a continuación del prefijo aceptado	página 12
$\hat{s}_h$	sufijo de la oración propuesto	página 12
$\mathbf{y}^\tau$	oración de referencia	página 21
λ_m^t	pesos para $h_m(\mathbf{x}, \mathbf{y})$ en el instante t	página 21
$\boldsymbol{\lambda}^t$	vector de pesos en el instante t	página 21
$\mathbf{y}^*$	mejor hipótesis del sistema	página 21
$\mu(\cdot)$	medida de calidad	página 21
$l(\mathbf{y})$	diferencia en la medida de la calidad	página 21
$\phi(\mathbf{y})$	diferencia en la puntuación	página 21
$(\mathbf{x}_t, \mathbf{y}_t^\tau)$	oración bilingüe observada en el instante t	página 24
$\check{\boldsymbol{\lambda}}_t$	término de actualización en el instante t	página 24
α	ratio de aprendizaje	página 24
$\|\|\cdot\|\|^2$	norma Euclídea	página 25
$\mathbf{H_x}$	matriz de características	página 25
$\mathbf{H_x^*}$	matriz de las mejores características	página 25
$\mathbf{Rx}$	matriz de la resta de puntuaciones	página 25
$\mathbf{l_x}$	vector columna con la diferencia de las medidas de calidad	página 25
$\mathbf{I}$	matriz identidad	página 25
β	termino de regularización	página 25
$W_{\boldsymbol{\lambda}^n}(\mathbf{x})$	grafo de palabras asociado a la oración $\mathbf{x}$ y al peso $\boldsymbol{\lambda}$	página 26
$\boldsymbol{\lambda}^n$	conjunto de pesos n-esimo extraído de forma semi-aleatoria	página 26
$\boldsymbol{\lambda}^N$	último conjunto de pesos extraído de forma semi-aleatoria	página 26
$\boldsymbol{\lambda}^*$	mejor conjunto de pesos extraído de forma semi-aleatoria	página 26
$\boldsymbol{\Lambda}$	súper conjunto de conjuntos de pesos semi-aleatorios	página 26
l_y	vector con la diferencia de las medidas de calidad de $\boldsymbol{W_\lambda(x)}$	página 26
$\mathbf{h}_{\boldsymbol{\lambda}^N}$	características obtenidas a partir de $\boldsymbol{\lambda}^N$	página 26
$\mathbf{h}_{\boldsymbol{\lambda}^*}$	características obtenidas a partir de $\boldsymbol{\lambda}^*$	página 26
Moses	decodificador	página 33
σ^2	varianza	página 33
μ	media	página 33
BLEU	Billingual Evaluation Understudy, medida de precisión en SMT	página 35
TER	Translation Edit Rate, medida de error en SMT	página 35
WER	Word Error Rate, medida de error en SMT	página 35
WSR	Word Stroke Ratio, medida de error en IMT	página 35
ρ_{C_1,C_2}	coeficiente de correlación de Pearson entre C_1 y C_2	página 57

Índice de figuras

Índice de cuadros

Bibliografía

[Arnold., 2003] Arnold., D. (2003). Computers and translation: A translator's guide. pages 119–142.

[Barrachina et al., 2009] Barrachina, S., Bender, O., Casacuberta, F., Civera, J., Cubel, E., Khadivi, S., Lagarda, A., Ney, H., Tomás, J., Vidal, E., and Vilar, J.-M. (2009). Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28.

[Berger et al., 1996] Berger, A., Brown, P. F., Pietra, S. A., Pietra, V. J., Kehler, A. S., and Mercer, R. L. (1996). Language translation apparatus and method of using Context-Based translation models. *United States Patent*, Patent Number 5,510,981.

[Bertoldi and Federico, 2009] Bertoldi, N. and Federico, M. (2009). Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the 4th EACL Workshop on Statistical Machine Translation*.

[Bishop, 2007] Bishop, C. M. (2007). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1 edition.

[Brown et al., 1993] Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. (1993). The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311.

[Callison-Burch et al., 2004] Callison-Burch, C., Bannard, C., and Schroeder, J. (2004). Improved statistical translation through editing. In *European Association for Machine Translation*.

[Callison-Burch et al., 2007] Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., and Schroeder, J. (2007). (meta-) evaluation of machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 136–158, Prague, Czech Republic. Association for Computational Linguistics.

[Callison-Burch et al., 2011] Callison-Burch, C., Koehn, P., Monz, C., and Zaidan, O. F., editors (2011). *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Edinburgh, Scotland.

[Casacuberta et al., 2009] Casacuberta, F., Civera, J., Cubel, E., Lagarda, A. L., Lapalme, G., Macklovitch, E., and Vidal, E. (2009). Human interaction for high-quality machine translation. *Commun. ACM*, 52(10):135–138.

[Casacuberta and Vidal, 2007] Casacuberta, F. and Vidal, E. (2007). Learning finite-state models for machine translation. *Machine Learning*, 66:69–91. 10.1007/s10994-006-9612-9.

[Chomsky, 1956] Chomsky, N. (1956). Three models for the description of language. *IRI Transactions on Information Theory*, 2(3):113–124.

[Civera and Juan, 2007] Civera, J. and Juan, A. (2007). Domain adaptation in statistical machine translation with mixture modelling. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Clarkson and Robinson, 1997] Clarkson, P. and Robinson, A. J. (1997). Language model adaptation using mixtures and an exponentially decaying cache. In *Proceedings of ICASSP-97*, pages 799–802.

[Crammer et al., 2006] Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., and Singer, Y. (2006). Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585.

[Denkowski and Lavie, 2010] Denkowski, M. and Lavie, A. (2010). Choosing the right evaluation for machine translation: an examination of annotator and automatic metric performance on human judgment tasks. In *Proccedings in the 9th Conference of the Association for Machine Translation in the Americas*.

[España-Bonet and Marquez, 2010] España-Bonet, C. and Marquez, L. (2010). Robust estimation of feature weights in statistical machine translation. In *14th Annual Conference of the European Association for Machine Translation*. EAMT.

[Fordyce, 2007] Fordyce, C. S. (2007). *Overview of the IWSLT 2007 Evaluation Campaign*, volume 11, pages 1–12.

[Foster, 2002] Foster, G. (2002). *Prediction for Translators*. PhD thesis, Université de Montréal.

[Foster et al., 2002] Foster, G., Langlais, P., and Lapalme, G. (2002). User-friendly text prediction for translators. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 148–155, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Gabriel Reverberi,] Gabriel Reverberi, Sandor Szedmak, N. C.-B. Deliverable of package 4: Online learning algorithms for computer-assisted translation.

[Gascó et al., 2012] Gascó, G., Rocha, M.-A., Sanchis-Trilles, G., Andrés-Ferrer, J., and Casacuberta, F. (2012). Does more data always yield better translations? In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 152–161, Avignon, France. Association for Computational Linguistics.

[Hutchins, 1999] Hutchins, J. (1999). Retrospect and prospect in computer-based translation. In *Proceedings of MT Summit VII MT in the great translation era*, pages 30–44.

[K. Papineni and Zhu, 2002] K. Papineni, S. Roukos, T. W. and Zhu, W. (2002). Bleu: A method for automaticevaluation of machine translation. In *Association for Computational Linguistics*.

[Kay, 1997] Kay, M. (1997). It's still the proper place. *Machine Translation*, 12(1/2):35–38.

[Kneser and Ney, 1995] Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pages 181–184.

[Koehn, 2004] Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP'04*, pages 388–395.

[Koehn, 2005] Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit, 2005*, pages 79–86.

[Koehn et al., 2005] Koehn, P., Axelrod, A., Mayne, A. B., Callison-Burch, C., Osborne, M., and Talbot, D. (2005). Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation. In *Proceedings of the international workshop on Spoken Language Translation, 2005*.

[Koehn and Monz, 2006] Koehn, P. and Monz, C. (2006). Manual and automatic evaluation of machine translation between european languages. In *Proceedings of the Workshop on Statistical Machine Translation*, StatMT '06, pages 102–121, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Koehn et al., 2003] Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 48–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Koehn and Schroeder, 2007] Koehn, P. and Schroeder, J. (2007). Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 224–227, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Koehn et al., 2007] Koehn et al., P. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the ACL Demo and Poster Sessions, 2007*, pages 177–180.

[Kuhn and De Mori, 1990] Kuhn, R. and De Mori, R. (1990). A cache-based natural language model for speech recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(6):570–583.

[Langlais et al., 2004] Langlais, P., Lapalme, G., and Loranger, M. (2004). Transtype: Development-evaluation cycles to boost translator's productivity. *Machine Translation (Special Issue on Embedded Machine Translation Systems*, 17(17):77–98.

[Lewis, 2009] Lewis, M. P., editor (2009). *Ethnologue: Languages of the World.*

[López-Salcedo et al., 2012] López-Salcedo, F. J., Sanchis-Trilles, G., and Casacuberta, F. (2012). Online learning of log-linear weights in interactive machine translation. In *Proceeding of iberSPEECH, 2012.*

[Marcu and Wong, 2002] Marcu, D. and Wong, W. (2002). A phrase-based, joint probability model for statistical machine translation. In *Proceedings of EMNLP*, pages 133–139.

[Martínez-Gómez et al., 2012] Martínez-Gómez, P., Sanchis-Trilles, G., and Casacuberta, F. (2012). Online adaptation strategies for statistical machine translation in post-editing scenarios. *Pattern Recognition*, page In press.

[Martínez-Gomez, 2010] Martínez-Gomez, P. (2010). Online learning via dynamic reranking for computer assisted translation. Master's thesis, Universidad Politécnica de Valencia, Valencia, Spain.

[Moore and Lewis, 2010] Moore, R. C. and Lewis, W. (2010). Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 220–224, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Nagao, 1984] Nagao, M. (1984). A framework of a mechanical translation between japanese and english by analogy principle. In *Proceedings of the international NATO symposium on Artificial and human intelligence*, pages 173–180, New York, NY, USA. Elsevier North-Holland, Inc.

[Nelder and Mead, 1965] Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7:308–313.

[Nepveu et al., 2004] Nepveu, L., Lapalme, G., Québec, M., and Foster, G. (2004). Adaptive language and translation models for interactive machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[Och, 2003] Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Och and Ney, 2002] Och, F. J. and Ney, H. (2002). Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 295–302, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Och and Ney, 2004] Och, F. J. and Ney, H. (2004). The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449.

[Och et al., 1999] Och, F. J., Tillmann, C., Ney, H., and Informatik, L. F. (1999). Improved alignment models for statistical machine translation. In *University of Maryland, College Park, MD*, pages 20–28.

[Och et al., 2003] Och, F. J., Zens, R., and Ney, H. (2003). Efficient search for interactive statistical machine translation. In *In EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 387–393.

[Ortiz et al., 2003] Ortiz, D., Varea, I., and Casacuberta, F. (2003). An empirical comparison of stack-based decoding algorithms for statistical machine translation. In Perales, F., Campilho, A., de la Blanca, N., and Sanfeliu, A., editors, *Pattern Recognition and Image Analysis*, volume 2652 of *Lecture Notes in Computer Science*, pages 654–663. Springer Berlin / Heidelberg. 10.1007/978-3-540-44871-6_76.

[Ortiz-Martínez et al., 2010] Ortiz-Martínez, D., García-Varea, I., and Casacuberta, F. (2010). Online learning for interactive statistical machine translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 546–554, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th annual conference of the Association for Computational Linguistics, 2002*, pages 311–318.

[Papineni et al., 1998] Papineni, K. A., Roukos, S., and Ward, R. T. (1998). Maximum likelihood and discriminative training of direct translation models. In *Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 189–192, Seattle, Washington, USA.

[Paula Estrella et al., 2004] Paula Estrella, A. P.-B., , and King, M. (2004). A new method for the study of correlations between mt evaluation metrics and some surprising results. In *In 11th International Conference on Theoretical and Methodological Issues in Machine Translation,2004*.

[Powell, 1964] Powell, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, 7(2):155–162.

[Pym, 1990] Pym, P. J. (1990). Pre-editing and the use of simplified writing for mt: an engineer[U+02BC]s experience of operating an mt system. *Translating and the Computer*, 10(November 1988):80–96.

[Sánchez and Benedí, 2006] Sánchez, J. A. and Benedí, J. M. (2006). Stochastic inversion transduction grammars for obtaining word phrases for phrase-based statistical machine translation. In *Proceedings of the Workshop on Statistical Machine Translation*, StatMT '06, pages 130–133, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Sanchis-Trilles and Casacuberta, 2010] Sanchis-Trilles, G. and Casacuberta, F. (2010). Log-linear weight optimisation via bayesian adaptation in statistical machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1077–1085, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Sanchis-Trilles et al., 2009] Sanchis-Trilles, G., Cettolo, M., Bertoldi, N., and Federico, M. (2009). Online language model adaptation for spoken dialog translation. In *International Workshop on Spoken Language Translation*, pages 160–167.

[Schwenk and Senellart, 2009] Schwenk, H. and Senellart, J. (2009). Translation model adaptation for an arabic/french news translation system by lightly-supervised training. In *MT Summit*.

[Shah et al., 2010] Shah, K., Barrault, L., and Schwenk, H. (2010). Translation model adaptation by resampling. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, WMT '10, pages 392–399, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Shannon, 1948] Shannon, C. E. (1948). A mathematical theory of communication. *Bell Sys. Tech. J.*, 27:379–423, 623–656.

[Snover et al., 2006] Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proc. of the 7th biennial conference of the Association for Machine Translation in the Americas, 2006*, pages 223–231.

[Stauffer and Grimson, 2000] Stauffer, C. and Grimson, W. E. L. (2000). Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:747–757.

[Stolcke, 2002] Stolcke, A. (2002). SRILM – an extensible language modeling toolkit. In *Proc. of the 7th international conference on Spoken Language Processing, 2002*, pages 901–904.

[Tiedemann, 2010] Tiedemann, J. (2010). To cache or not to cache?: experiments with adaptive models in statistical machine translation. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, WMT '10, pages 189–194, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Tomás and Casacuberta, 2001] Tomás, J. and Casacuberta, F. (2001). Monotone statistical translation using word groups.

[Toselli et al., 2011] Toselli, A. H., Vidal, E., and Casacuberta, F., editors (2011). *Multimodal Interactive Pattern Recognition and Applications*. Springer, 1st edition edition. http://www.springer.com/computer/hci/book/978-0-85729-478-4.

[Ueffing et al., 2002] Ueffing, N., Och, F. J., and Ney, H. (2002). Generation of word graphs in statistical machine translation. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 156–163, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Vauquois, 1968] Vauquois, B. (1968). A survey of formal grammars and algorithms for recognition and transformation in mechanical translation. In *IFIP Congress (2)'68*, pages 1114–1122.

[Vidal et al., 2007] Vidal, E., Rodríguez, L., Casacuberta, F., and García-Varea, I. (2007). Interactive pattern recognition. In *MLMI*, pages 60–71.

[Watanabe et al., 2003] Watanabe, T., Sumita, E., and Okuno, H. G. (2003). Chunk-based statistical translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 303–310, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Weaver, 1955] Weaver, W. (1949/1955). Translation. In Locke, W. N. and Boothe, A. D., editors, *Machine Translation of Languages*, pages 15–23. MIT Press, Cambridge, MA. Reprinted from a memorandum written by Weaver in 1949.

[Zens and Ney, 2004] Zens, R. and Ney, H. (2004). Improvements in phrase-based statistical machine translation. In *Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics Annual Meeting*, pages 257–264, Boston, MA.

[Zens et al., 2002] Zens, R., Och, F. J., and Ney, H. (2002). Phrase-based statistical machine translation. In *German Conf. on Artificial Intelligence*, pages 18–32, Aachen, Germany.

[Zhao et al., 2004] Zhao, B., Eck, M., and Vogel, S. (2004). Language model adaptation for statistical machine translation with structured query models. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.