MODELOS DE TRADUCCIÓN DE ESTADOS FINITOS

A. Pérez[(1)]*, *F. Casacuberta*[(2)], *M. I. Torres*[(1)], *V. Guijarrubia*[(1)], *R. Justo*[(1)], *E. Alonso*[(1)], *J. M. Alcaide*[(1)]

[(1)] Dep. Electricidad y Electrónica
Facultad de Ciencia y Tecnología
Universidad del País Vasco
manes@we.lc.ehu.es
[(2)] Dep. Sistemas Informáticos y Computación
Facultad de Informática
Universidad Politécnica de Valencia
fcn@dsic.upv.es

RESUMEN

El sistema **sysid = upv-ehu sfst** consiste en un transductor estocástico de estados finitos con suavizado por back-off. El suavizado permite modelar los n-gramas no-vistos. Adicionalmente, incluye un tratamiento para palabras desconocidas que se desarrolla en la fase de búsqueda. Este mecanismo considera que la palabra desconocida es equivalente a alguna de las palabras vistas, pero se queda sin definir hasta finalizar el análisis de la cadena de entrada.

1. DESCRIPCIÓN DEL SISTEMA

El sistema que aquí presentamos (**sysid = upv-ehu sfst**), es un sistema de traducción *basado en corpus*, que a diferencia de los sistemas *basados en reglas*, se dota de inteligencia artificial por inducción, y no en base a una serie de reglas. Se trata de una aproximación empírica donde se cuenta simplemente con un conjunto de ejemplos en la lengua origen y su traducción en la lengua destino que servirán para elaborar una representación del conocimiento sobre el problema de traducción. La gran gran ventaja de la aproximación inductiva frente a la deductiva, es que existe un método para adquirir el conocimiento que es independiente de las lenguas que intervengan. Con lo cual, es posible emplear, con mayor o menor éxito, la misma técnica para cualquier lengua de entrada o de salida.

Entre los sistemas basados en corpus, el que nos ocupa consiste en una *aproximación paramétrica*, que en lugar de memorizar los ejemplos disponibles (como lo hacen las *memorias de traducción*), los emplean para construir un modelo matemático que se caracteriza por un conjunto de parámetros. El valor de estos parámetros se estima a partir de las muestras dadas, o pares de traducción en nuestro caso.

*Este trabajo ha sido parcialmente financiado por la Universidad del País Vasco, 9/UPV 00224.310-15900/2004, y el proyecto CICYT, TIN2005-08660-C04-03.

Los modelos de IBM [1] establecen las bases de la aproximación estadística que posteriormente han derivado en múltiples extensiones aún más ambiciosas. A grandes rasgos, el objetivo de estos modelos consiste en encontrar la cadena $\mathbf{t}$ de la lengua destino que mejor se adapte a una cadena dada en la lengua origen $\mathbf{s}$ como describe la ecuación (1). Esta expresión se conoce como *ecuación fundamental de la traducción automática* y está desglosada en dos factores: el *modelo de traducción* desde la lengua destino a la lengua origen ($P(\mathbf{s}|\mathbf{t})$) y el *modelo de lenguaje de la lengua destino* ($P(\mathbf{t})$). Este último se encarga de disponer una estructura sintáctica oportuna a la secuencia de palabras que ofrece el modelo de traducción.

$$\widehat{\mathbf{t}} = \arg\max_{\mathbf{t}} P(\mathbf{t}|\mathbf{s}) = \arg\max_{\mathbf{t}} P(\mathbf{s}|\mathbf{t})P(\mathbf{t}) \quad (1)$$

Alternativamente, en este trabajo se consideran los modelos de traducción de estados finitos: los *transductor estocástico* [2, 3]. En esta aproximación, el mismo modelo incluye tanto la componente de transferencia de una lengua a la otra, como una estructura sintáctica, con lo cual no se hace uso de un modelo de lenguaje auxiliar. Estos modelos, resuelven de manera distinta el problema de la traducción, y como objetivo buscan, no la salida más verosímil dada la entrada, sino, aquella cadena que maximice la probabilidad conjunta de la entrada y la salida. Tanto la definición del modelo como los métodos de síntesis y decodificación se describen en detalle en las siguientes secciones.

1.1. Definición

Un *transductor estocástico de estados finitos* (TEEF) es una tupla $\mathcal{T} = \langle \Sigma, \Delta, Q, q_0, R, F, P \rangle$, donde:

- Σ es un conjunto finito de símbolos de entrada (vocabulario en la lengua origen);
- Δ es un conjunto finito de símbolos de salida (vocabulario en la lengua destino);

Q es un conjunto finito de estados (nodos);

$q_0 \in Q$ es el estado inicial;

$R \subseteq Q \times \Sigma \times \Delta^* \times Q$ es un conjunto de transiciones (aristas) del tipo $(q, s, \tilde{t}, q')$, que representa una transición desde el estado q hasta el estado q', consumiendo el símbolo de entrada s y produciendo la secuencia de slida $\tilde{t}$;

$P : R \rightarrow [0, 1]$ es la distribución de probabilidad de transición;

$F : Q \rightarrow [0, 1]$ es la distribución de probabilidad de estados terminales;

Las distribuciones de probabilidad están sujetas a:

$$\forall q \in Q \quad F(q) + \sum_{\forall s, \tilde{t}, q'} P(q, s, \tilde{t}, q') = 1 \qquad (2)$$

2. DISEÑO: INFERENCIA DEL MODELO

Tanto la topología como las distribuciones de probabilidad de los TEEF se pueden inferir automáticamente a partir de un corpus bilingüe mediante algoritmos como OSTIA [4], OMEGA[5] o GIATI [6, 7], siendo este último el empleado en este trabajo.

Dado un corpus bilingüe, es decir, un conjunto de ejemplos del tipo $(\mathbf{s}, \mathbf{t}) \in \Sigma^* \times \Delta^*$, donde $\mathbf{s}$ denota una cadena en la lengua origen y $\mathbf{t}$ su traducción en la lengua destino, el algoritmo GIATI se puede resumir como sigue:

1. Empleando una *función de etiquetado* ($\mathcal{L}$), cada par de cadenas se transforma en una sola cadena de un *vocabulario extendido* ($\Gamma \subseteq \Sigma \times \Delta^*$). La transformación $\mathcal{L}$ busca la segmentación monótona óptima para cada par de cadenas. A cada palabra de la cadena de entrada le asocia una secuencia de palabras (cero o más) de la cadena de la lengua destino según indique la segmentación elegida. Por lo tanto, cada *símbolo extendido* consta de un símbolo en la lengua origen junto con cero o más símbolos de la lengua destino.

2. Una vez que el conjunto bilingüe de ejemplos se ha convertido en un conjunto de cadenas monolingües ($\mathbf{z} \in \Gamma^*$), se puede inferir un modelo de lenguaje (de la lengua extendida). En este trabajo se hace uso de un modelo de lenguaje k-explorable en sentido estricto [8], generando su correspondiente autómata estocástico de estados finitos. En particular, se utiliza un autómata suavizado por *back-off* con descuento de *Witten-Bell* [9].

3. El símbolo extendido asociado a cada transición del autómata se transforma, mediante la *función inversa de etiquetado* ($\mathcal{L}^{-1}$), en una palabra de entrada y una secuencia de salida, $(w/\tilde{p}) \in \Sigma \times \Delta^*$, dando lugar al TEEF deseado.

En este punto, se puede observar que un estado del TEEF representa, conjuntamente, un prefijo en una cadena de entrada y otro en la lengua de salida. Así mismo, el TEEF se puede entender como un modelo de lenguaje de la lengua de entrada y otro de la lengua de salida integrados en la misma máquina de estados finitos.

Debido a la técnica de inferencia descrita en el paso 2, el TEEF es determinista respecto de los símbolos extendidos, es decir, desde un estado, con un símbolo extendido se accede a un único estado destino. Sin embargo, es no-determinista respecto de los símbolos de entrada, es decir, desde un estado dado, con el mísmo símbolo de entrada se puede acceder a múltiples estados destino, emitiendo distintas secuencias en la lengua destino.

Por otra parte, hay que aclarar, que el conjunto de símbolos de entrada y el de salida (Σ, Δ) queda definidos por aquellas palabras que aparezcan en las muestras de la lengua origen y destino respectivamente. Hay que definir un mecanismo adicional (que se describirá en la sección 3.1) para tratar todo símbolo fuera de este conjunto, es decir, las palabras fuera del vocabulario (OOV). En este aspecto, los modelos basados en conocimiento lingüístico [10] ofrecen mecanismos de análisis morfológico que ayudan a tratar sin restricciones cualquier símbolo de entrada.

Las técnicas de suavizado aportan al modelo capacidad de generalización, ya que incorpora un mecanismo para tratar los eventos (n-gramas) no vistos en la fase de entrenamiento. A pesar de que el suavizado de autómatas es un problema resuelto con éxito, el suavizado de transductores es aún un problema abierto, aunque existen propuestas en esta línea [11]. En nuestro trabajo, el suavizado afecta sólo a la lengua de entrada, ya que desde un estado se admite cualquier palabra de entrada, pero no está contemplada la posibilidad de emitir cualquier secuencia de salida. Es decir, dados un prefijo (n-grama) en la lengua origen y otro en la lengua destino, se contempla la posibilidad de observar cualquier palabra de entrada tras la secuencia dada, pero las secuencias de salida están restringidas.

3. ANÁLISIS: ALGORITMO DE BÚSQUEDA

Dada una cadena de entrada ($\mathbf{s}$), un TEEF ($\mathcal{T}$) ofrece como salida aquella cadena que maximice la probabilidad conjunta:

$$\widehat{\mathbf{t}} = \arg \max_{\forall \mathbf{t} \in \Delta^*} P(\mathbf{s}, \mathbf{t}) = \arg \max_{\mathbf{t}} \sum_{\forall d(\mathbf{s}, \mathbf{t})} P(d(\mathbf{s}, \mathbf{t})) \quad (3)$$

La probabilidad asociada al par $(\mathbf{s}, \mathbf{t})$ es la suma de las probabilidades de todos los posibles caminos (d) compatibles con ese par. La solución de la ecuación (3) es un problema de elevada complejidad computacional [12], no obstante, se puede resolver de forma eficiente con la siguiente aproximación:

$$\sum_{\forall d(\mathbf{s}, \mathbf{t})} P(d(\mathbf{s}, \mathbf{t})) \approx \max_{d(\mathbf{s}, \mathbf{t})} P(d(\mathbf{s}, \mathbf{t})) \equiv P_{\mathcal{T}}(\mathbf{s}, \mathbf{t}) \quad (4)$$

Dando lugar a la expresión (5), que se puede implementar mediante el *algoritmo de Viterbi*, el cual ofrece la secuencia óptima de estados del TEEF por los cuales se consume la cadena de entrada y se emite la cadena formada por cada una de las secuencias de salida a través del camino seleccionado.

$$\widehat{\mathbf{t}} \approx \arg \max_{\forall \mathbf{t} \in \Delta^*} P_{\mathcal{T}}(\mathbf{s}, \mathbf{t}) \quad (5)$$

3.1. Tratamiento de palabras desconocidas

En esta sección se proponen distintas alternativas para tratar de traducir una cadena que incluye palabras desconocidas. Se considera una palabra desconocida cualquier símbolo que no se encuentre en el vocabulario definido en la etapa de entrenamiento ($\forall w \notin \Sigma$).

En líneas generales, un conjunto de entrenamiento significativo recoge todas las palabras de la tarea, salvo quizá nombres propios, dígitos, siglas, etc. No obstante, hay tareas en las que las muestras de entrenamiento son escasas y es necesario disponer de algún mecanismo para tratar las palabras desconocidas, como las que aquí se han estudiado:

1. Asumiendo que el corpus tiene una cobertura suficiente como para que las palabras fuera del vocabulario sean nombres propios, u otro tipo de singularidades que se transcribirían de igual modo en cualquier lengua, se puede dar como salida la propia palabra de entrada. Es decir, la traducción de una palabra no-vista será la misma palabra.

2. Dados los pares de frases, se puede estimar un diccionario (estadístico). Con los pares de traducción que ofrece un diccionario (palabra en la lengua origen y su traducción en la lengua destino), se puede inferir un modelo de traducción a nivel de carácter. Este modelo serviría para traducir, carácter a carácter, las palabras no vistas. En general, para lenguas muy distantes entre sí, se prevé una dispersión demasiado alta como para que el modelo funcione con éxito, pero sí podría funcionar para lenguas cercanas entre sí. Se ha probado esta técnica en la tarea evaluada, ofreciendo como ejemplo las siguiente traducciones para palabras desconocidas: (política, POLíTICA), (visitará, VI FUTURO Tú NECESITAR), (residencia, ACREDITAR DOMICILIO).

3. Se puede tratar la palabra del vocabulario como si fuera una palabra indefinida del vocabulario. Es decir, como si fuera un sinónimo de alguna de las palabras del vocabulario, pero sin saber con certeza de cuál se trata. Adicionalmente, se puede definir una métrica para modelar esta incertidumbre, de modo que a pesar de que no sepamos de qué palabra se trata, podamos primar unas sobre otras, aún sin desestimar ninguna. Una métrica que se podría emplear para ponderar la similitud de la palabra desconocida a cada una de las palabras del vocabulario sería la distancia de Levenshtein a nivel de carácter. Un caso particular de esta propuesta, consiste, simplemente, en considerar que la palabra desconocida se puede reemplazar por la palabra más cercana del vocabulario, sin explorar ninguna otra.

Con las dos primeras aproximaciones, la traducción de una palabra desconocida puede que sea también desconocida en la lengua destino. Es decir, las dos primeras aproximaciones pueden generar símbolos fuera del vocabulario que se ha definido en el entrenamiento para la lengua destino ($\forall v \notin \Delta$). La tercera, sin embargo, garantiza que el símbolo de salida será una palabra vista.

En el problema que nos ocupa, donde la lengua destino son signos, que aunque vienen representados de forma textual bien podrían haberse representado mediante etiquetas numéricas, las dos primeras aproximaciones no parecen convenientes. Por otra parte, la lengua de entrada está codificada en carácteres en minúscula y la de salida en mayúscula, con algunas excepciones sencillas de describir y que permitirían emplear la primera aproximación. No obstante, en este trabajo hemos preferido considerar el símbolo como un ente atómico, que podría venir codificado mediante una etiqueta numérica, y no como una secuencia de carácteres, lo cual nos hace desestimar las dos primeras aproximaciones. Desafortunadamente, la tercera aproximación es más conservadora, y no generará nuevas palabras en la lengua de salida.

Una vez seleccionada la tercera alternativa, hay que establecer la métrica deseada. Por simplicidad, en este trabajo no se ha incluido métrica alguna que favorezca a unas palabras sobre otras. Es decir, se ha considerado que la verosimilitud de todas las palabras del vocabulario es la misma. Este tratamiento para palabras desconocidas es muy sencillo llevarlo a la práctica en modelos de estados finitos. El análisis sería como sigue: desde un estado dado, ante una palabra desconocida se accede a todos los estados a los que se podría llegar desde ese estado con cada una de las palabras del vocabulario. Es decir, se consideran todas las hipótesis compatibles con la historia representada por el estado.

4. CONDICIONES DE LA EVALUACIÓN

La tarea consiste en traducción de texto en castellano a lengua de signos española, dentro de un dominio restringido. Son frases pronunciadas por un funcionario que atiende a las personas que desean sacar o renovar el DNI o el pasaporte.

Las características principales del conjunto de entrenamiento se presentan en la tabla 1. En promedio, el ratio de repetición de cada palabra en la lengua de entrada es de $5{,}0$ y el $10\,\%$ son *singletons* (palabras que sólo aparecen una vez en las muestras). Estos dos datos indican que habrá un elevado número de eventos no-vistos.

En cuanto a las características del conjunto de evaluación en la lengua de entrada (texto) y la salida ofrecida por el sistema, se muestran en la tabla 2. Cabe destacar el elevado número de palabras desconocidas al que hay que enfrentarse.

A pesar de que el conjunto de referencia no está disponible para evaluar el modelo presentado, se puede hacer una estimación de su capacidad por la técnica de *leaving-one-out*. Para esta estimación, se han mezclado los pares de entrenamiento de acuerdo a una semilla aleatoria. Y se han diseñado tres experimentos independientes tomando distintas particiones de 140 pares para entrenamiento y 60 para evaluación. Estas tres particiones guardan aproximadamente la misma proporción entre entrenamiento y test que el conjunto de evaluación ($81/200 \approx 60/140$). En cada uno de los tres experimentos así diseñados se dispone de la referencia, así pues se puede hacer una predicción de las prestaciones, que en este caso son $BLEU \approx 59$.

		Texto	**Signos**
Aprendizaje	Pares de frases	200	
	Pares distintos	190	
	Frases distintas	189	183
	Palabras	2240	2075
	Vocabulario	443	260
	Longitud media	11.2	10.3

Tabla 1. Características del corpus en castellano en su representación textual y en su representación por signos.

		Texto	**Signos**
Test	Pares de frases	81	
	Palabras	963	898
	Palabras desconocidas	59	0
	Longitud media	11.8	11.0
	PP (3grams)	19.1	10.8

Tabla 2. Características del conjunto de evaluación y la salida ofrecida por el sistema

5. BIBLIOGRAFÍA

[1] Peter F. Brown, J. Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, F. Jelinek, John D. Lafferty, R. L. Mercer, y P. S. Roossin, "A statistical approach to machine translation," *Computational Linguistics*, vol. 16, no. 2, pp. 79–85, June 1990.

[2] E. Vidal, F. Thollard, F. Casacuberta C. de la Higuera, y R. Carrasco, "Probabilistic finite-state machines - part I," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 7, pp. 1013–1025, 2005.

[3] E. Vidal, F. Thollard, F. Casacuberta C. de la Higuera, y R. Carrasco, "Probabilistic finite-state machines - part II," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 7, pp. 1025–1039, 2005.

[4] J. Oncina, P. García, y E. Vidal, "Learning subsequential transducers for pattern recognition interpretation tasks," *IEEE Trans. on PAMI*, vol. 15, no. 5, pp. 448–458, 1993.

[5] J. M. Vilar, Enrique Vidal, y J. C. Amengual, "Learning extended finite state models for language translation," in *Proc. of the 12th European Conf. on Artificial Intelligence*, Budapest, Hungary, Aug. 1996, pp. 92–96.

[6] Enrique Vidal, "Finite-state speech-to-speech translation," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Munich, Germany, Apr. 1997, vol. 1, pp. 111–114.

[7] F. Casacuberta y E. Vidal, "Machine translation with inferred stochastic finite-state transducers," *Computational Linguistics*, vol. 30, no. 2, pp. 205–225, 2004.

[8] I. Torres y A. Varona, "k-tss language models in speech recognition systems," *Computer Speech and Language*, vol. 15, no. 2, pp. 127–149, 2001.

[9] A. Varona y I. Torres, "Using Smoothed k-TLSS Language Models in Continous Speech Recognition," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1999, vol. II, pp. 729–732.

[10] Antonio M. Corbí-Bellot, Mikel L. Forcada, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Gema Ramírez-Sánchez, Felipe Sánchez-Martínez, Iñaki Alegria, Aingeru Mayor, y Kepa Sarasola, "An open-source shallow-transfer machine translation engine for the romance languages of spain," in *Proceedings of the Tenth Conference of the European Associtation for Machine Translation*, Budapest, Hungary, May 2005, pp. 79–86.

[11] D. Llorens, *Suavizado de autómatas y traductores finitos estocásticos*, Ph.D. thesis, Universidad Politécnica de Valencia, 2000.

[12] F. Casacuberta, "Inference of finite-state transducers by using regular grammars and morphisms," in *Grammatical Inference: Algorithms and Applications*, A.L. Oliveira, Ed., vol. 1891 of *Lecture Notes in Computer Science*, pp. 1–14. Springer-Verlag, 2000, 5th International Colloquium Grammatical Inference -ICGI2000-. Lisboa. Portugal. Septiembre.