Traducción estadística basada en frases: MOSES

Jaime Ferrando Huertas

Enero 2021

Contents

1 Introducción **2**

2 Ejercicios **2**
2.1 Experimento inicial 2
2.2 Ejercicio 1 2
2.3 Ejercicio 2 2
2.4 Ejercicio 3 3
2.5 Ejercicio 4 3
2.6 Ejercicio 5 4
2.7 Ejercicio 6 4

3 Conclusión **4**

1 Introducción

El objetivo de esta práctica es evaluar el toolkit MOSES para entrenamiento de modelos de traducción automática. Se ha entrenado un modelo básico de traducción español-ingles según las instrucciones del boletín de prácticas y se han realizado distintos experimentos basados en este experimento inicial para los ejercicios opcionales. Para evaluar nuestros modelos hemos hecho uso de la métrica BLEU [4] y hemos usado el corpus de EuTrans.

2 Ejercicios

2.1 Experimento inicial

Nuestro primer experimento lo realizamos siguiendo las instrucciones del boletín de prácticas. Primero entrenamos un modelo de lenguaje del idioma de salida, ingles. Entrenamos este modelo con la herramienta SRILM [5] construyendo un modelo basada en tri-gramas con suavizado de Kneser-Ney [2]. Una vez contamos con nuestro modelo de lenguaje entrenamos el modelo de traducción inicial con MOSES, este modelo se usará ahora para realizar el ajuste de pesos sobre el corpus de desarrollo junto con la técnica MERT [3] con un máximo de iteraciones de 5. Una vez entrenado el modelo y ajustados sus pesos evaluamos el BLEU en el conjunto de test y obtenemos un 91.73.

A partir de ahora crearemos distintos experimentos con el fin de aprender sobre MOSES y como distintos parámetros afectan a nuestros modelos, los compararemos siempre con el resto de experimentos mediante el BLEU.

2.2 Ejercicio 1

En el primer ejercicio creamos un modelo sin ajuste de pesos log lineal, es decir sin el ajuste de pesos con la técnica MERT. Obtenemos los siguientes resultados:

Experimento	MERT Iter	N-Gramas	Suavizado	Monótono	BLEU
Sin ajustar pesos	-	3	Kneser-Ney	NO	88.42

Podemos ver que los resultados obtenidos son bastante bajos en comparación a nuestro experimento inicial. En el siguiente experimento trataremos el ajuste de pesos y trataremos de buscar el número de iteraciones óptimo.

2.3 Ejercicio 2

En este apartado queremos estudiar como afecta el proceso de ajuste de pesos al modelo entrenado. Hemos visto un gran decremento en el BLEU cuando no es usado en el ejercicio 1 por lo que ahora es mas importante aun optimizar este paso. Para ello vamos a estudiar el número de iteraciones del MERT y como óptimas se traduce en el BLEU. Obtenemos los siguientes resultados.

Experimento	MERT Iter	N-Gramas	Suavizado	Monótono	BLEU
MERT-5	5	3	Kneser-Ney	NO	91.73
MERT-7	7	3	Kneser-Ney	NO	91.71
MERT-10	10	3	Kneser-Ney	NO	**91.90**
MERT-15	15	3	Kneser-Ney	NO	91.81

Table 1: Evolución del BLEU dadas iteraciones de MERT

Vemos una gran mejoría en el BLEU en todos nuestros experimentos, esto era algo de esperar dado el experimento inicial y el ejercicio uno. En concreto vemos como 10 iteraciones son el valor óptimo para el ajuste de pesos en nuestro caso de uso.

2.4 Ejercicio 3

En este ejercicio vamos a estudiar como afecta el número de n-gramas con el que entrenamos nuestro modelo del lenguaje a nuestro modelo final de traducción. Hemos variado los n-gramas sobre la base del experimento inicial (originalmente entrenado con 3 tri-gramas) con 5 iteraciones de MERT para el ajuste de pesos log-lineal. Obtenemos los siguientes resultados

Experimento	MERT Iter	N-Gramas	Suavizado	Monótono	BLEU
N-GRAMAS 2	5	2	Kneser-Ney	NO	89.13
N-GRAMAS 3	5	3	Kneser-Ney	NO	**91.73**
N-GRAMAS 4	5	4	Kneser-Ney	NO	91.42
N-GRAMAS 5	5	5	Kneser-Ney	NO	90.79

Table 2: Evolución del BLEU dado n-gramas del modelo de lenguaje

Podemos ver como los tri-gramas son la mejor elección para nuestro modelo de lenguaje y que este juega un papel muy importante en el sistema final, llegando a reducir el BLEU por debajo de 90 en el caso de bi-gramas. El corpus que manejamos ahora mismo no es muy grande y podría ser que esto perjudicara a modelos de ngramas más complejos. Pensamos que esta es la razón por la que los tri-gramas funcionan mejor que los n-gramas 4,5 y sería digno de evaluar con corpus más grandes.

2.5 Ejercicio 4

En este ejercicio se nos pide evaluar MIRA [1] como substituto a MERT para el calculo de ajuste de pesos del modelo log-lineal. Para ello debemos hacer uso de la herramienta implementada en Moses $k - bestbatchMIRATuning$ con la instrucción $-batch - mira$, un algoritmo de entrenamiento online que usa las k mejores listas para aproximar el espacio de búsqueda usado por el decoder del modelo.

Con este experimento obtenemos los un BLEU de 91.11 que es **menor** al 91.73 obtenido por MERT con modelo de lenguaje de tri-gramas. Sin embargo

el tiempo de entrenamiento para este experimento es muchísimo menor por lo que es una opción a evaluar si el tiempo de entrenamiento es un factor decisivo en nuestros modelos.

2.6 Ejercicio 5

En este ejercicio evaluamos distintos tipos de suavizado en nuestro modelo de lenguaje y como afectan al BLEU de nuestro modelo de traducción final. En el modelo de lenguaje que hemos usado hasta ahora hemos usado Kneser, ahora probaremos distintos métodos de suavizado como Witten-Bell y Good-Turing. Obtenemos los siguientes resultados:

Experimento	MERT Iter	N-Gramas	Suavizado	Monótono	BLEU
Witten-bell	5	3	Witten-Bell	NO	**91.97**
Kneser-Ney	5	3	Kneser-Ney	NO	91.73
Good-Turing	5	3	Good-Turing	NO	90.22

Table 3: Evolución del BLEU para distintos métodos de suavizado

El método de suavizado de Witten-Bell obtiene mejores resultados, acercándose casi a los 92 puntos.

2.7 Ejercicio 6

En este último ejercicio probamos MOSES monótono, en este experimento modificamos el método para realizar los alineamientos. La diferencia entre monótono y no monótono es que el monótono no permite reordenar los bloques en el paso del alineamiento mientras que el no monótono sí. Obtenemos un BLEU de 89.98 que es menor al obtenido en el experimento inicial 91.73.

3 Conclusión

Gracias a esta práctica hemos hecho uso de MOSES para realizar distintos experimentos que nos han permitido entender mejor distintos aspectos de la traducción estadística a la vez de como estos modelos son afectados por sus parámetros. Se puede encontrar ahora una tabla con todos los experimentos realizados.

Experimento	MERT Iter	N-Gramas	Suavizado	Monótono	BLEU
Experimento inicial	5	3	Kneser-Ney	NO	91.73
Sin ajustar pesos	-	3	Kneser-Ney	NO	88.42
MERT-5	5	3	Kneser-Ney	NO	91.73
MERT-7	7	3	Kneser-Ney	NO	91.71
MERT-10	10	3	Kneser-Ney	NO	91.90
MERT-15	15	3	Kneser-Ney	NO	91.81
N-GRAMAS 2	5	2	Kneser-Ney	NO	89.13
N-GRAMAS 3	5	3	Kneser-Ney	NO	91.73
N-GRAMAS 4	5	4	Kneser-Ney	NO	91.42
N-GRAMAS 5	5	5	Kneser-Ney	NO	90.79
MIRA	MIRA	3	Kneser-Ney	NO	91.11
Witten-bell	5	3	Witten-Bell	NO	**91.97**
Kneser-Ney	5	3	Kneser-Ney	NO	91.73
Good-Turing	5	3	Good-Turing	NO	90.22
Monótono	5	3	Good-Turing	SI	89.98

Table 4: Experimentos realizados

References

[1] Koby Crammer. Ultraconservative online algorithms for multiclass problems. 07 2001.

[2] Reinhard Kneser and H. Ney. Improved backing-off for m-gram language modeling. *1995 International Conference on Acoustics, Speech, and Signal Processing*, 1:181–184 vol.1, 1995.

[3] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan, July 2003. Association for Computational Linguistics.

[4] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[5] Andreas Stolcke. Srilm — an extensible language modeling toolkit. *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)*, 2, 07 2004.