UNIVERSIDAD COMPLUTENSE MADRID

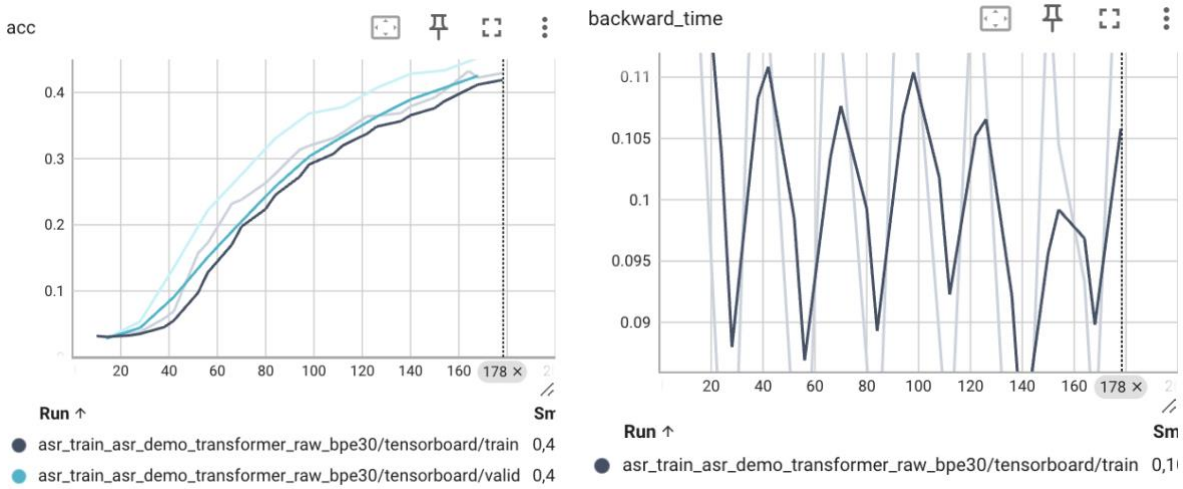
Procesamiento del habla
Federico Ortega Riba
fedorteg@ucm.es

Informe práctica final

1. Ejecución con Demo_transformer

Para empezar, hemos seguido las instrucciones para ejecutar el código usando train_asr_demo_transformer, para lo que se ha tenido que especificar en cada una de las celdas a partir del paso 10. Este *notebook* se puede consultar como *Recipe_Tutorial_Demo_transformer.ipynb*, anexo a la carpeta del informe.

Las estadísticas que nos proporciona *tensorboard* son las siguientes:



En cuanto al CER/WER, se presentan estos resultados:

Resultados del conjunto de datos de prueba

WER (Tasa de Error de Palabras)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	773	71.9	20.4	7.6	1.8	29.9	72.3

- La **WER** del conjunto de prueba es 29,9%, lo que significa que casi el 30% de las palabras son incorrectas (substituidas, eliminadas o insertadas).
- La **tasa de palabras correctas (Corr)** es del 71.9%.
- **Sub** (substituciones) supone un 20,4%, **Del** (eliminaciones) 7,6%, e **Ins** (inserciones) un 1,8%.

CER (Tasa de Error de Caracteres)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	2565	86.0	5.0	9.0	1.2	15.2	72.3

- La **CER** del conjunto de prueba es 15,2%, lo que significa que el 15,2% de los caracteres son incorrectos.
- La **tasa de caracteres correctos (Corr)** es del 86%.

Resultados del conjunto de datos de validación

WER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	591	64.8	25.2	10.0	3.7	38.9	75.0

- La **WER** del conjunto de validación es 38,9%, casi un 10% mayor que en el conjunto de prueba.
- La **tasa de palabras correctas (Corr)** es del 64,8%.

CER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	1915	79.6	7.2	13.2	1.8	22.2	75.0

- La **CER** del conjunto de validación es 22,2%.
- La **tasa de caracteres correctos (Corr)** es del 79,6%.

Análisis general

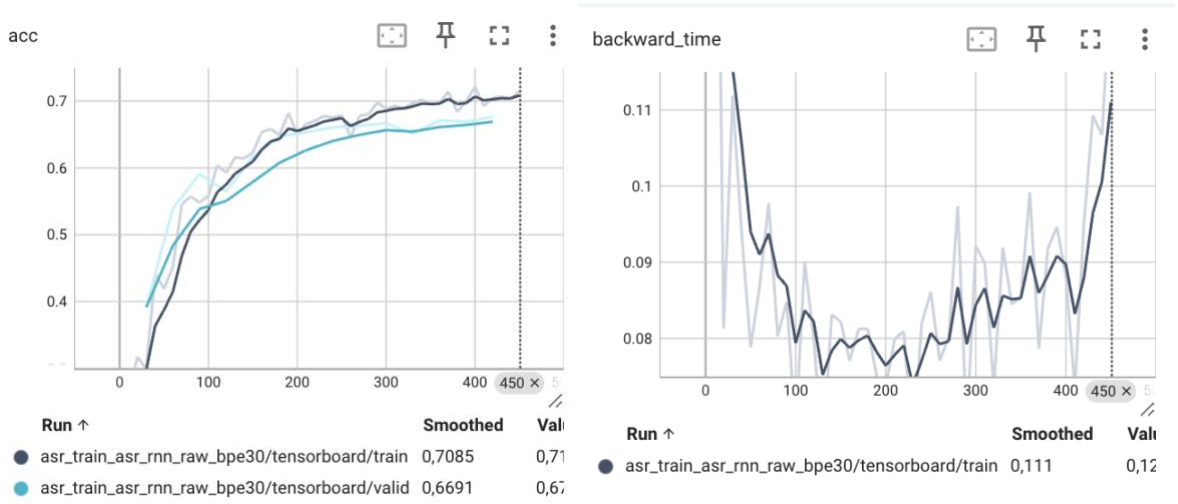
1. **Comparación entre prueba y validación**:
    - La WER es más baja en el conjunto de prueba (29,9%) comparado con el conjunto de validación (38,9%). Esto sugiere que el modelo podría estar mejor optimizado para el conjunto de prueba o que hay una mayor dificultad en los datos de validación.
2. **Tasas de Error**:

- La mayor tasa de sustituciones en WER y CER indica que el modelo tiene problemas para identificar las palabras correctas, reemplazándolas con otras incorrectas.
- La tasa de eliminación es notablemente alta, especialmente en el conjunto de validación, indicando que el modelo omite palabras frecuentemente.

2. Ejecución con RNN

Para empezar, hemos seguido las instrucciones para ejecutar el código usando train_asr_rnn, para lo que se ha tenido que especificar en cada una de las celdas a partir del paso 10. Este *notebook* se puede consultar como *Recipe_Tutorial_rnn.ipynb*, anexo a la carpeta del informe.

Las estadísticas que nos proporciona *tensorboard* son las siguientes:



En cuanto al CER/WER, se presentan estos resultados:

Resultados del conjunto de datos de prueba

WER (Tasa de Error de Palabras)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	773	39.6	46.4	14.0	10.6	71.0	86.2

CER (Tasa de Error de Caracteres)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	2565	60.4	19.0	20.6	3.8	43.4	86.2

Federico Ortega Riba
fedorteg@ucm.es

Resultados del Conjunto de Datos de Validación

WER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	591	34.5	49.1	16.4	10.8	76.3	90.0

CER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	1915	56.1	21.7	22.2	4.8	48.7	90.0

1. **Comparación entre prueba y validación**:
    - La WER es más baja en el conjunto de prueba (71,0%) comparado con el conjunto de validación (76,3%). Esto sugiere que el modelo podría estar mejor optimizado para el conjunto de prueba o que hay una mayor dificultad en los datos de validación.
    - La CER sigue una tendencia similar, siendo más baja en el conjunto de prueba.
2. **Tasas de Error**:
    - La mayor tasa de sustituciones en WER y CER indica que el modelo tiene problemas para identificar las palabras correctas, reemplazándolas con otras incorrectas.
    - La tasa de eliminación es notablemente alta, especialmente en el conjunto de validación, indicando que el modelo omite palabras frecuentemente.

3. **Cambio de parámetros: 3 ejemplos**

    a. Ejemplo 1

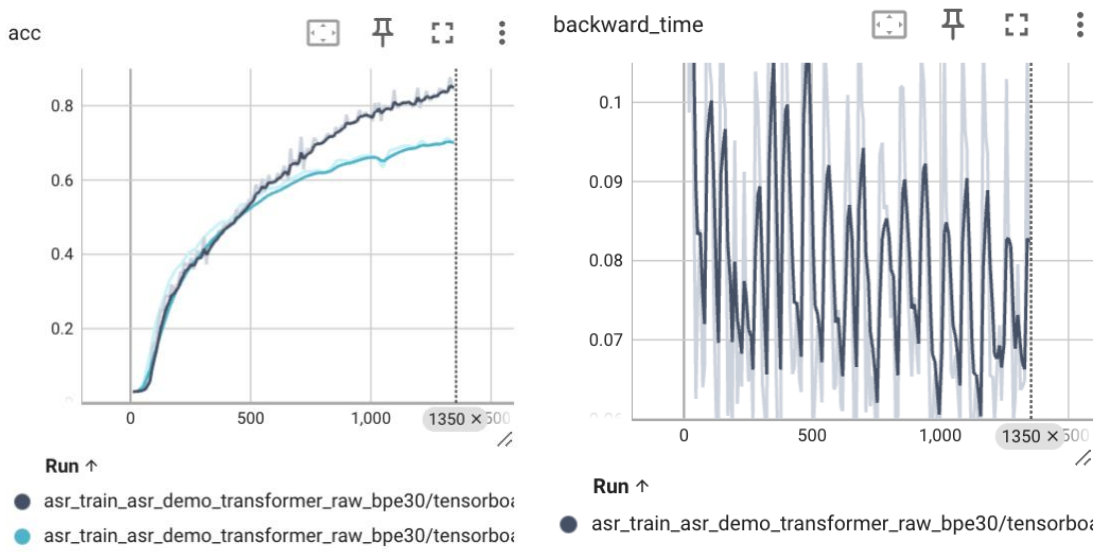
Veamos ahora cada uno de los ejemplos. Los parámetros que se han cambiado para todos los ejemplos son los de los archivos de entrenamiento (*train_asr_demo_transformer*) y el archivo *ash.sh*. En el primer caso son los que aparecen resaltados en negrita:

- Asr.sh 1
    - use_lm=false

- word_vocab_size=**7000** # Size of word vocabulary.
- min_wav_duration=**0.3** # Minimum duration in second.
- max_wav_duration=**25** # Maximum duration in second.

- train_asr_demo_transformer_1
  - batch_size = **32**
  - accum_grad: **2** # gradient accumulation steps
  - max_epoch: **50**
  - patience: **10**
  - keep_nbest_models: **5**
  - num_workers: **4**

Se ha optado en este caso por disminuir la lista de entrenamiento del vocabulario y las epochs, además de añadir un patience. Los resultados de *tensorboard* son:



b. Ejemplo 2

En el segundo caso son los que aparecen resaltados en negrita:

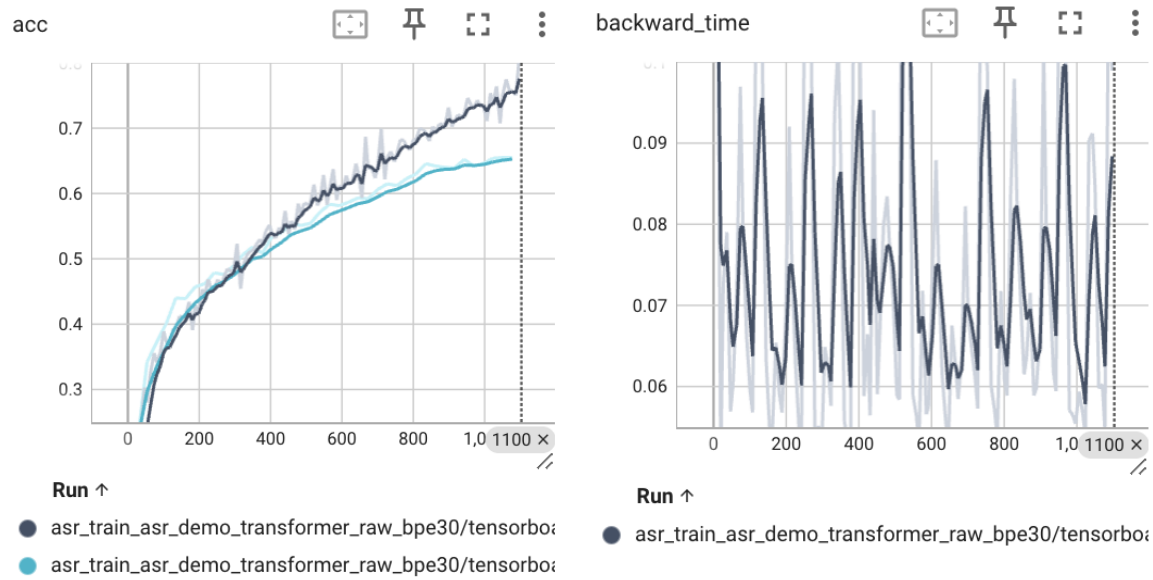
- asr.sh
  - use_lm=false
  - word_vocab_size=**15000** # Size of word vocabulary.
  - min_wav_duration=**0.2** # Minimum duration in second.
  - max_wav_duration=**30**
  - asr_speech_fold_length=**1000** # fold_length for speech data during ASR training.

  - asr_text_fold_length=**200** # fold_length for text data during ASR training.
  - lm_fold_length=**200**
  - optim: **adamw** # Cambiar a **AdamW** para una mejor regularización
  - optim_conf:
  - lr: 0.0003 # Reducir la tasa de aprendizaje
  - scheduler: **reduceonplateau** # Cambiar a **ReduceLROnPlateau** para ajustar la tasa de aprendizaje
  - scheduler_conf:
  - factor: 0.5
  - patience: 5
  - min_lr: 1e-6

- train_asr_demo_transformer_
  - batch_size = **32**
  - accum_grad: **2** # gradient accumulation steps
  - max_epoch: **70**
  - patience: **10**
  - keep_nbest_models: **8**

En este caso, se ha optado por modificar muchos más parámetros: aumentar el número de listas de entrenamiento y vocabulario, la longitud que pueden tener los audios y cambiar el optimizador. En la fase comparación, veremos que estos cambios no han supuesto una mejora, sino todo lo contrario.

Estas son las gráficas correspondientes de *tensorboard*:



Federico Ortega Riba
fedorteg@ucm.es

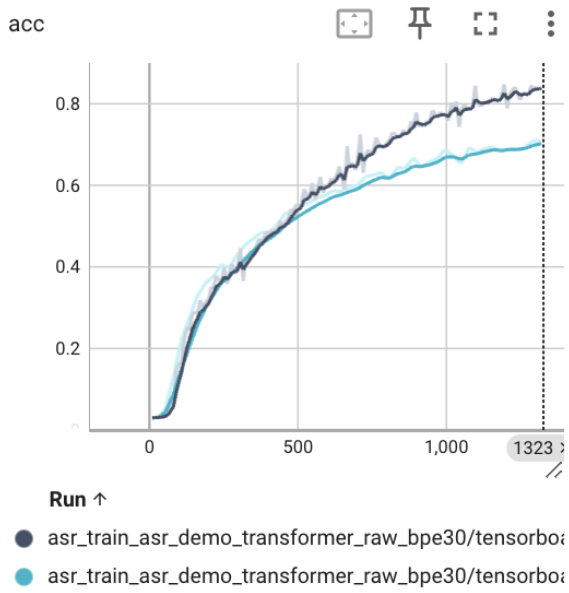
c. Ejemplo 3

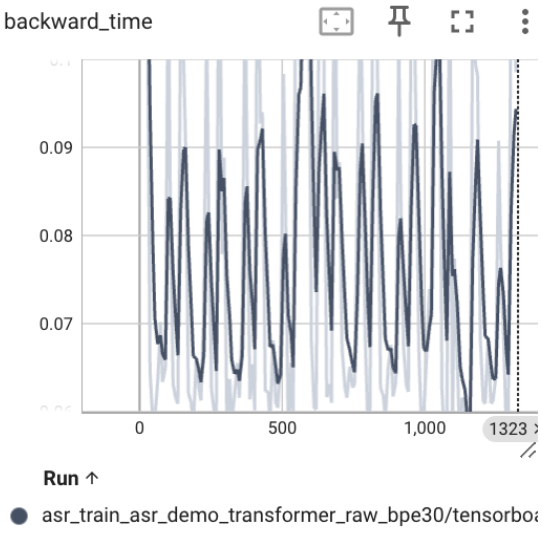
En el tercer caso son los que aparecen resaltados en negrita:

- asr.sh
  - use_lm=false
  - word_vocab_size=**7000** # Size of word vocabulary.
  - min_wav_duration=**0.1** # Minimum duration in second.
  - max_wav_duration=**15**

- train_asr_demo_transformer
  - batch_size = **32**
  - accum_grad: **1** # gradient accumulation steps
  - max_epoch: **50**
  - patience: **10**

Las gráficas correspondientes son:





4. Comparación de resultados tras el cambio de parámetros

Primer modelo

WER (Tasa de Error de Palabras)

Federico Ortega Riba
fedorteg@ucm.es

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	773	68.6	23.5	7.9	2.6	34.0	72.3
decode_asr_lm_lm_train_bpe30_valid.loss.ave_asr_model_valid.acc.ave/test	130	773	0.0	0.0	100.0	0.0	100.0	100.0

CER (Tasa de Error de Caracteres)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	2565	83.9	5.9	10.2	1.9	18.0	72.3
decode_asr_lm_lm_train_bpe30_valid.loss.ave_asr_model_valid.acc.ave/test	130	2565	0.0	0.0	100.0	0.0	100.0	100.0

Resultados del conjunto de datos de validación

WER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	591	61.6	30.5	8.0	3.2	41.6	74.0

CER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	1915	76.9	8.7	14.4	1.8	25.0	74.0

Segundo modelo

WER (Tasa de Error de Palabras)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	773	0.0	0.0	100.0	0.0	100.0	100.0

CER (Tasa de Error de Caracteres)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	2565	0.0	0.0	100.0	0.0	100.0	100.0

Resultados del conjunto de datos de validación

WER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	591	0.0	0.0	100.0	0.0	100.0	100.0

CER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	1915	0.0	0.0	100.0	0.0	100.0	100.0

Tercer modelo

WER (Tasa de Error de Palabras)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	773	67.0	24.5	8.5	1.6	34.5	73.1

CER (Tasa de Error de Caracteres)

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
decode_asr_asr_model_valid.acc.ave/test	130	2565	84.7	6.5	8.8	2.5	17.9	73.1

Resultados del conjunto de datos de validación

WER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	591	60.9	30.6	8.5	4.7	43.8	74.0

CER

dataset	Snt	Wrd	Corr	Sub	Del	Ins	Err	S.Err
org/train_dev	100	1915	78.2	9.6	12.2	2.6	24.4	74.0

Análisis comparativo

1. **Rendimiento general**:

- **Primer modelo**: El primer modelo presenta un rendimiento razonable con una WER de 34% y una CER de 18% en el conjunto de prueba. Sin embargo, los resultados de WER y CER son significativamente peores en el caso de un modelo específico (decode_asr_lm_lm_train_bpe30_valid.loss.ave_asr_model_valid.acc.ave) que tiene un 100% de error en ambos métricas, indicando un fallo en el modelo.
- **Segundo modelo**: El segundo modelo muestra un rendimiento extremadamente deficiente con un 100% de error en todas las métricas, lo que sugiere un fallo completo en el entrenamiento o la implementación del modelo. Esto se puede deber a los cambios de parámetros tan grandes que hemos realizado. Lo tomaremos como ejemplo negativo de cambios de parámetros.
- **Tercer modelo**: El tercer modelo es el más consistente, con una WER de 34,5% y una CER de 17,9% en el conjunto de prueba, siendo ligeramente mejor que el primer modelo.

2. **Comparación de validación**:
   - Los resultados de validación siguen una tendencia similar a los de prueba. El tercer modelo tiene una WER y CER ligeramente mejores que el primer modelo, mientras que el segundo modelo tiene un rendimiento pésimo.
3. **Estabilidad del modelo**:
   - El tercer modelo parece ser el más estable y confiable, con los mejores resultados generales y consistencia entre las métricas de prueba y validación.
   - El primer modelo tiene buenos resultados, pero muestra fallos significativos en ciertas configuraciones.
   - El segundo modelo no es viable debido a su rendimiento erróneo.