Experimentos de identificación de autores

Daniel Embarcadero

Contenido

- Modelo de verificación de autoría:
  - Verificación de autoría
  - Dataset
  - Modelo propuesto
  - Entrenamiento
  - Ajuste de threshold
  - Evaluación
- Pregunta práctica:
  - Planteamiento
  - Documentos
  - Experimento
  - Resultados

Modelo de verificación de autoría

Primera sección

Verificación de autoría

Definición de la tarea: Dados dos textos determinar si ambos fueron escritos por el mismo autor.

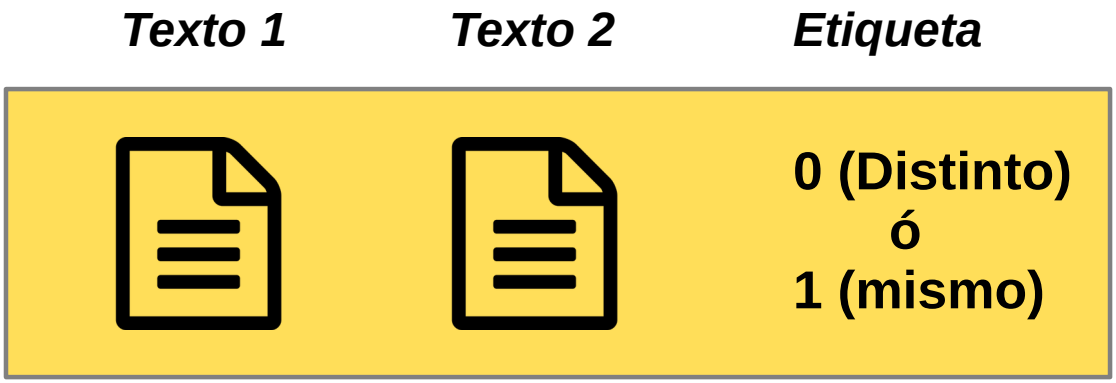
En nuestro modelo:

- Se reciben dos textos y se devuelve un número entre 0 y 1.
- Los puntajes representan:
  - < 0.5 : Probablemente distinto autor
  - = 0.5 : No concluyente
  - > 0.5 : Probablemente mismo autor

Dataset

- Para entrenar el modelo, necesitamos parejas de textos etiquetados a forma de conocer si ambos textos fueron o no escritos por el mismo autor.

Una instancia:



- Utilizamos los conjunto de datos de entrenamiento provistos por PAN para la tarea de Verificación de Autoría del 2021.
- Son dos conjuntos independientes de datos a los que llamaremos “small” y “large” de acuerdo a su tamaño.

Características de los conjuntos de datos:

	Problems	Texts	Authors	Topics
Small	52,601	93,662	52,654	1,600
Large	275,565	494,226	278,162	1,600

- *Problems:* Total de parejas de textos en cada conjunto
- *Texts:* Número de textos distintos que conforman las parejas.
- *Authors:* Número de autores distintos que escriben alguno de los textos.
- *Topics:* Número de temas distintos que tratan los textos.

Con el fin de desarrollar y evaluar el modelo, definimos tres particiones en cada conjunto de datos.

Llamaremos a estas particiones entrenamiento, validación y prueba. Cada una tiene 80%, 10% y 10% de las instancias del conjunto total, respectivamente.

Las particiones se construyen de forma que sean ajenas por autores; esto es, el autor de un texto en cierta partición, no puede ser autor de ningún texto en otra partición.

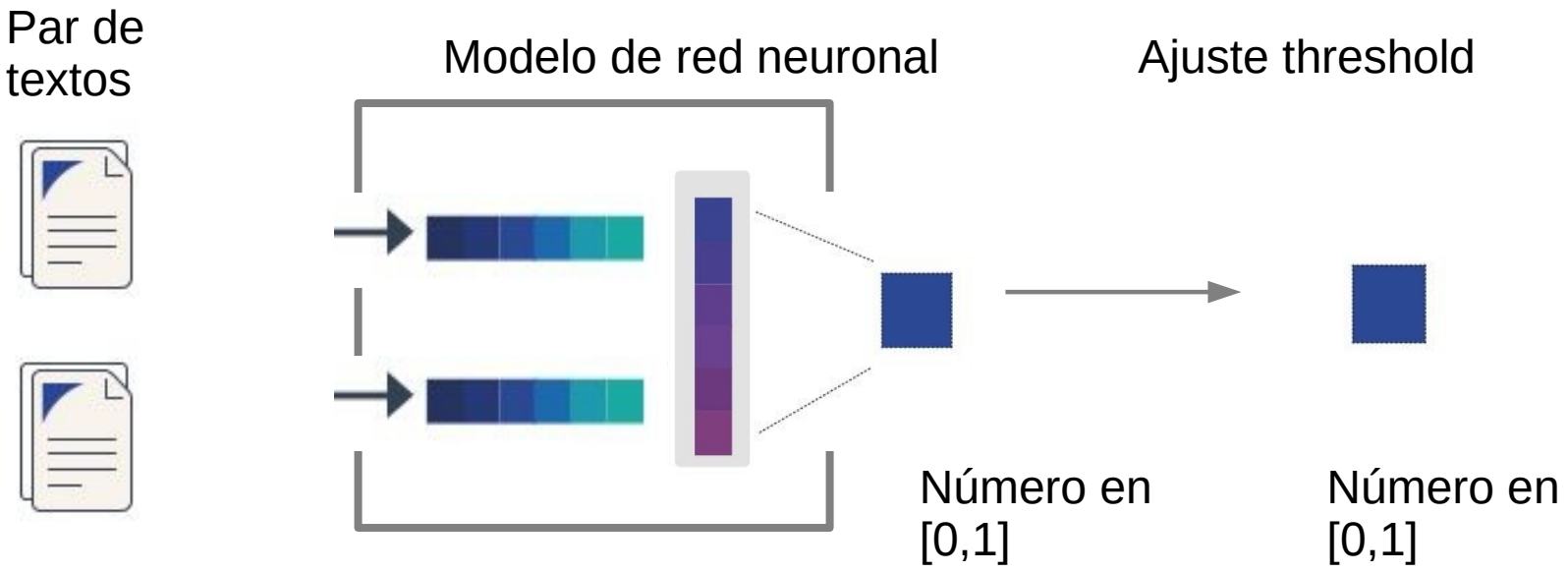
	Small dataset		Large dataset	
	Total	Positive	Total	Positive
Train split	42,077	22,560	220,438	120,201
Val split	5,259	2,636	27,554	13,783
Test split	5,259	2,633	27,554	13,777

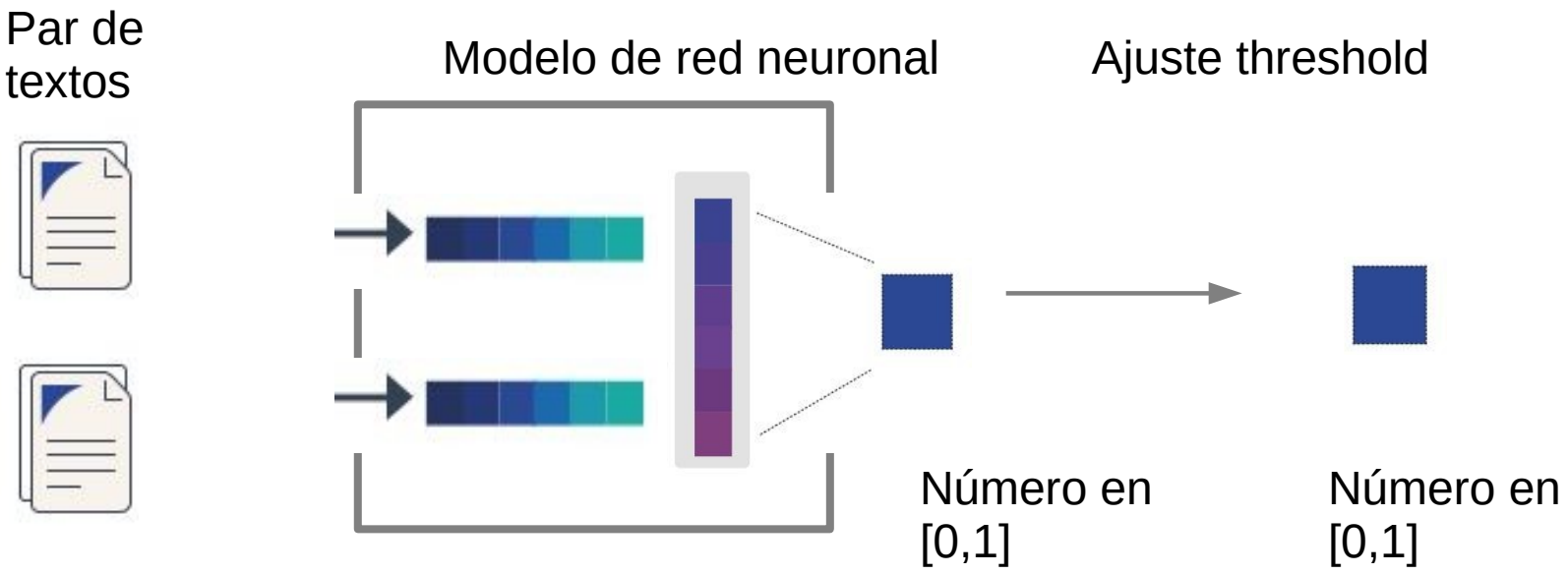
Total: Total de pares de textos en cada partición
Positive: Total de pares de textos con etiquetas “Mismo autor”

Modelo propuesto

Nuestro modelo recibe un par de textos y devuelve un número entre 0 y 1.

El modelo se compone de dos partes:
- Modelo de red neuronal siamesa.
- Ajuste de threshold



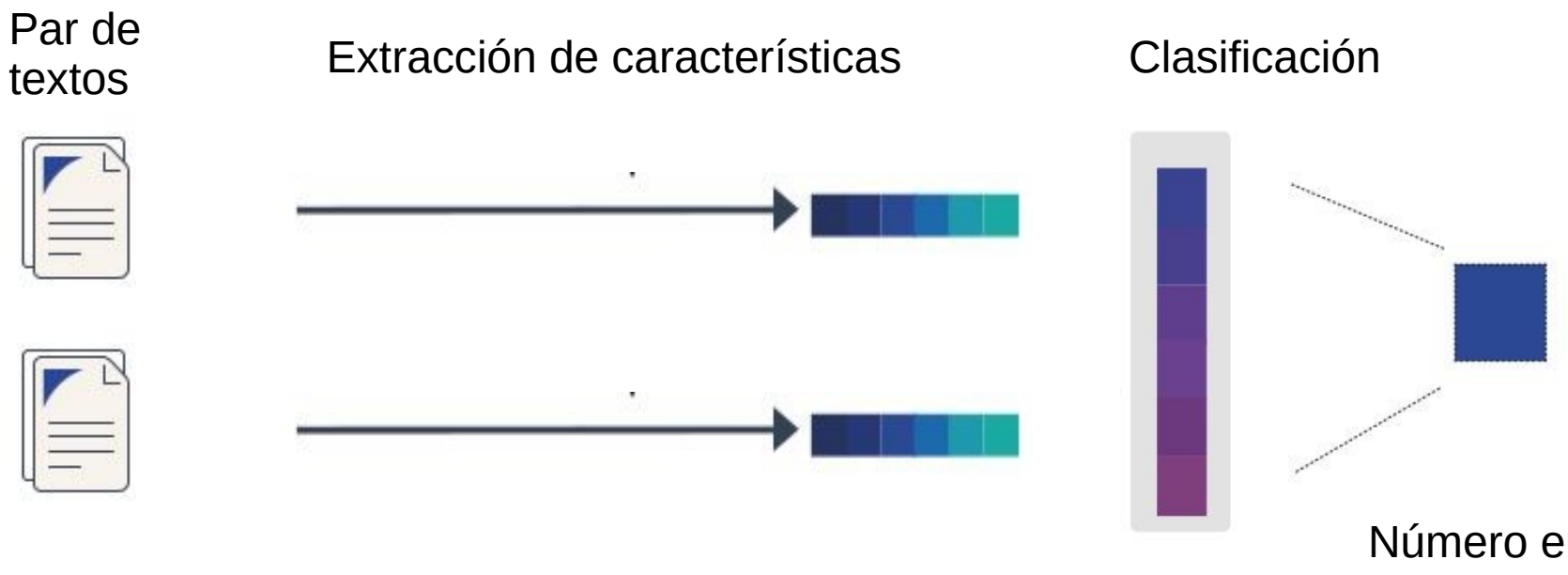


El ajuste de threshold se realizó con la intención de que el modelo final asignara el puntaje 0.5 a algunas instancias. Con esto buscamos que los pares de textos para los que fuera dificil decidir si eran de un mismo autor o no, obtuvieran valor de 0.5. Más adelante se detallará el ajuste realizado.

Modelo de red neuronal siamesa

Este modelo recibe un par de textos y devuelve un número entre 0 y 1.

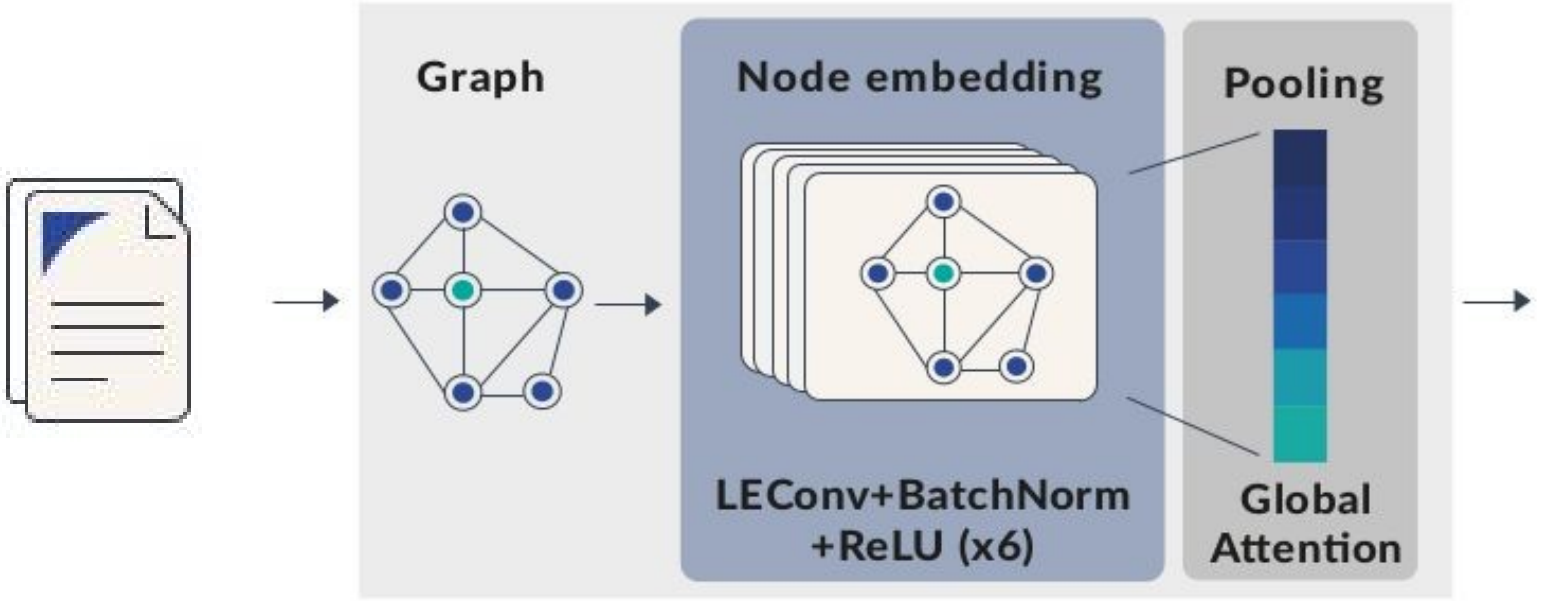
Se compone de dos subredes para extracción de características con pesos compartidos y una red de clasificación.



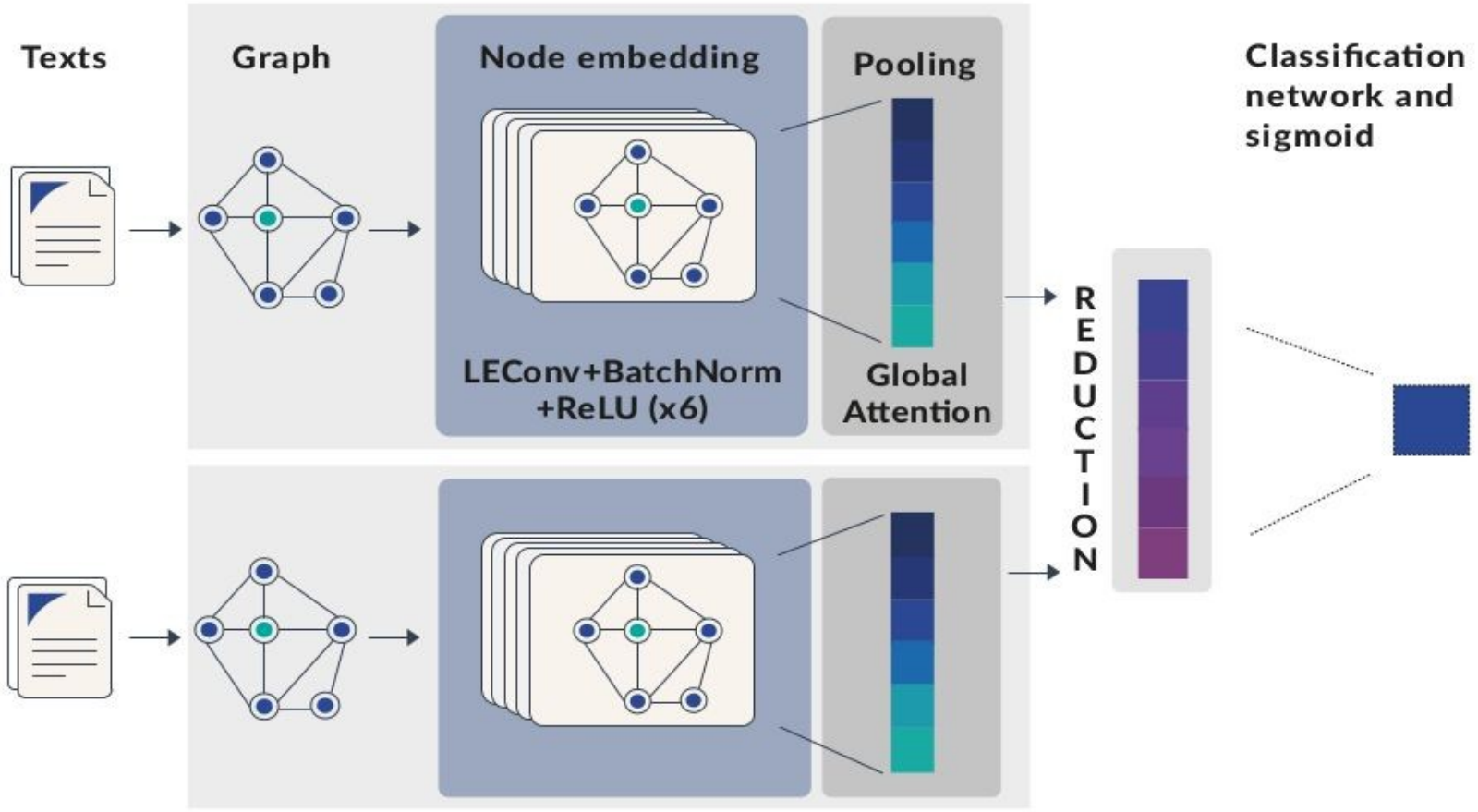
Para la extracción de características:

- Modelamos el texto como grafo
- Utilizamos capas conocidas como redes convolucionales sobre grafos

Esto se realiza sobre cada uno de los textos de forma independiente.

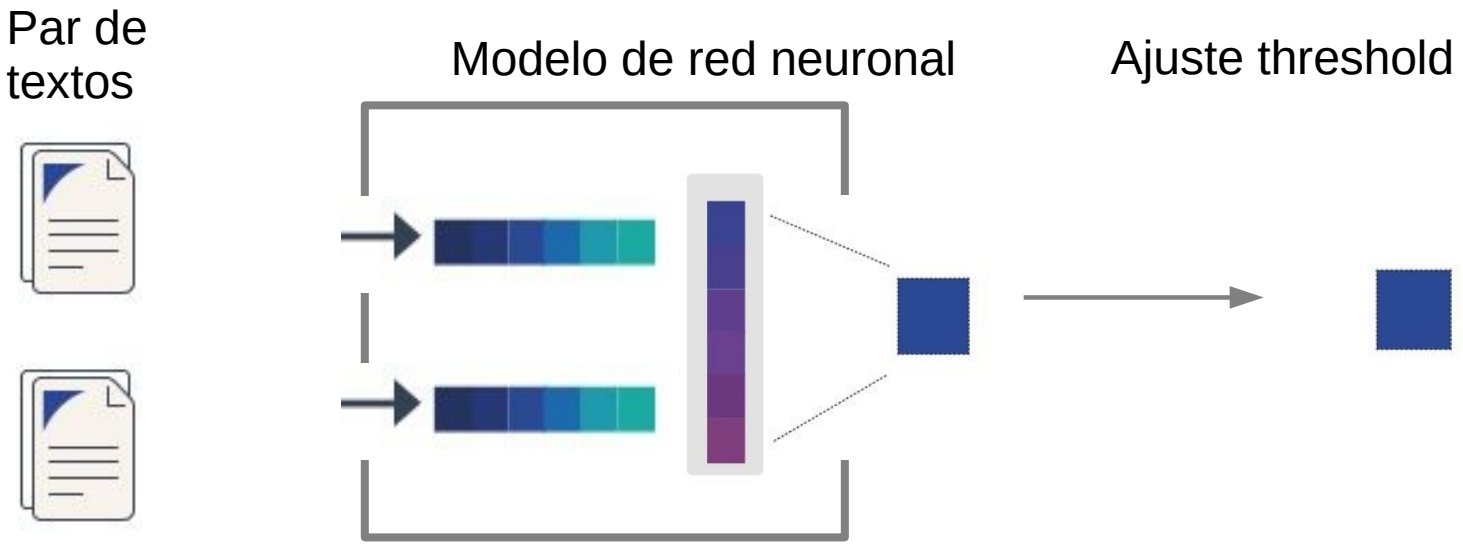


La arquitectura del modelo de red neuronal siamesa es:



Entrenamiento

- Recordemos que nuestro modelo se compone de dos partes.



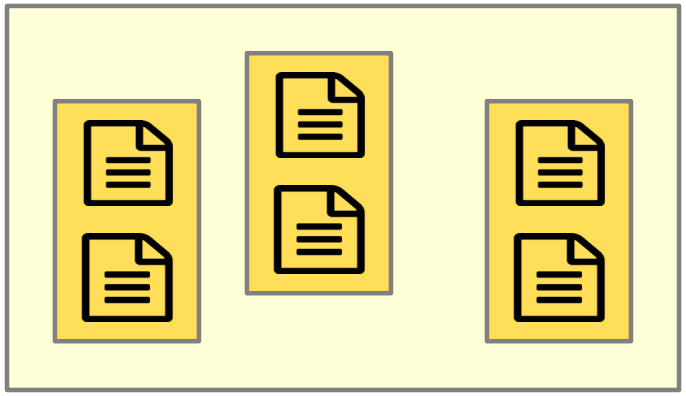
Primero se entrena el modelo de red neuronal y después se definen los valores m y th para el ajuste de threshold.

Entrenamiento del modelo de red neuronal

- Nuestro modelo tiene parámetros entrenables en sus capas.
- Estos parámetros se ajustan en función de las instancias (pares de textos y etiqueta) que el modelo analiza.
- Cuando el modelo recibe una instancia:
  - En base al par de textos, obtiene un valor entre 0 y 1.
  - Compara este valor con la etiqueta de esta instancia y actualiza sus parámetros. Al actualizar los parámetros busca que la próxima vez que vea esta instancia, su predicción se acerque más a la etiqueta.

- El entrenamiento se realiza a lo largo de épocas. Una época es una iteración. En esta iteración el modelo ve todas y cada una de las parejas de textos en la partición de entrenamiento.

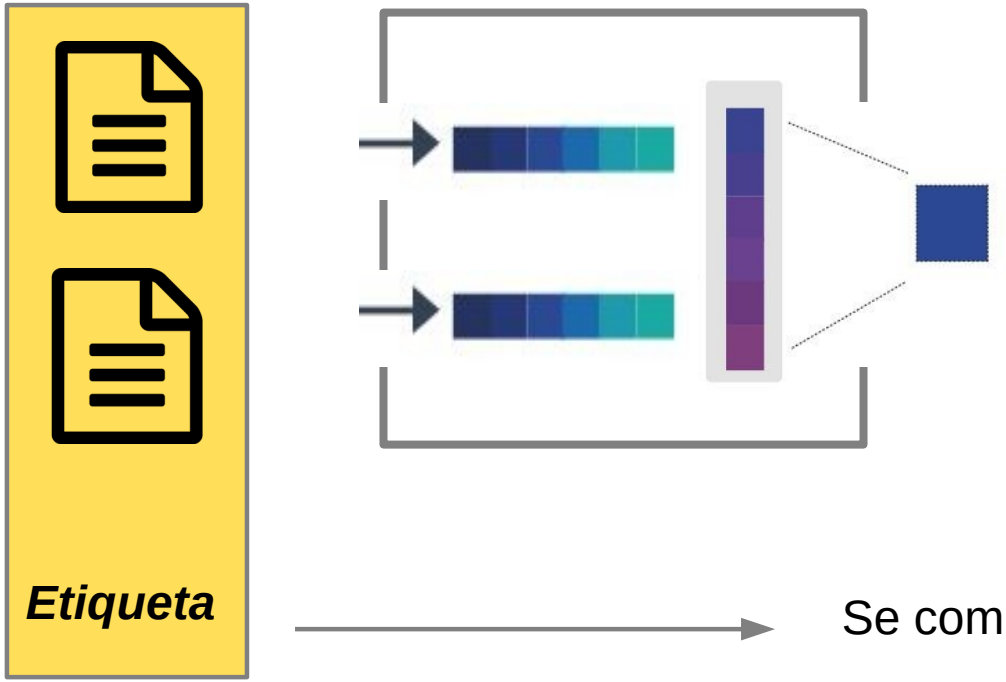
Conjunto de entrenamiento



Una época concluye cuando el modelo considera todas las instancias en el conjunto de entrenamiento.

Modelo

Predicción



x

Se comparan

Se repite este procedimiento por varias épocas.

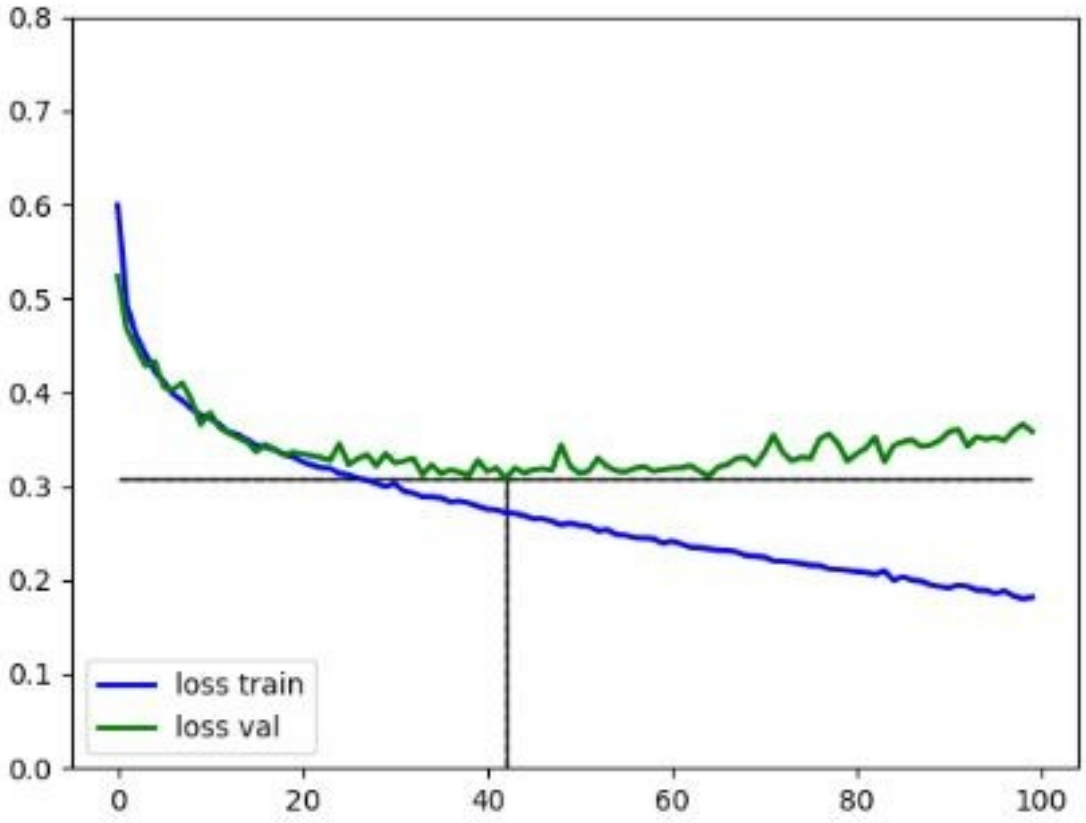
Al final de cada época cuantificamos qué tan bien nuestro modelo esta prediciendo sobre las instancias de la partición de validación (instancias que el modelo no usó para ajustar sus parámetros).

Para cuantificar esto:

- El modelo no actualiza en ninguna forma sus parámetros, estos quedan congelados.
- Para cada instancia en el conjunto de validación: Predecimos un número entre 0 y 1 utilizando nuestro modelo sobre el par de textos.
- Con las predicciónes y las etiquetas de todas las instancias calculamos una función de pérdida.

Utilizamos la función de pérdida Binary Cross Entropy (BCE por sus siglas).

Este es un ejemplo de como varía la función de pérdida conforme avanzan las épocas.



Función de pérdida BCE

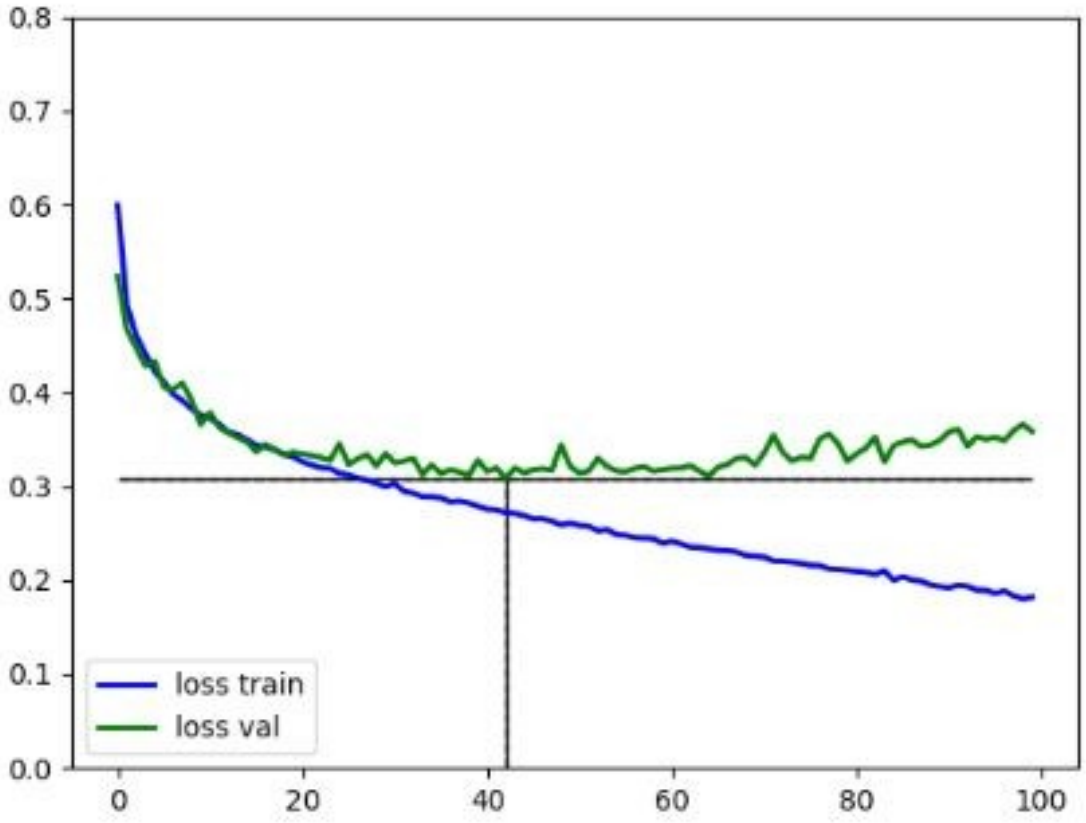
El eje X representa el estado (valor de los parámetros) del modelo en cada época, para las primeras 100 épocas.

El eje Y representa el valor de la función de pérdida BCE.

En verde se grafica el resultado de evaluar el conjunto de validación.

En azul (solo referencia) se grafica el resultado de evaluar el conjunto de entrenamiento.

Este es un ejemplo de como varía la función de pérdida conforme avanzan las épocas.



Función de pérdida BCE

Valores menores en la función de pérdida significan un modelo con predicciónes más precisas.

De entre todos los posibles estados (valor de parámetros) elegimos el estado en la época que obtuvo la menor pérdida cuando evaluamos el conjunto de validación.

Así definimos los parámetros de nuestro modelo entrenado.

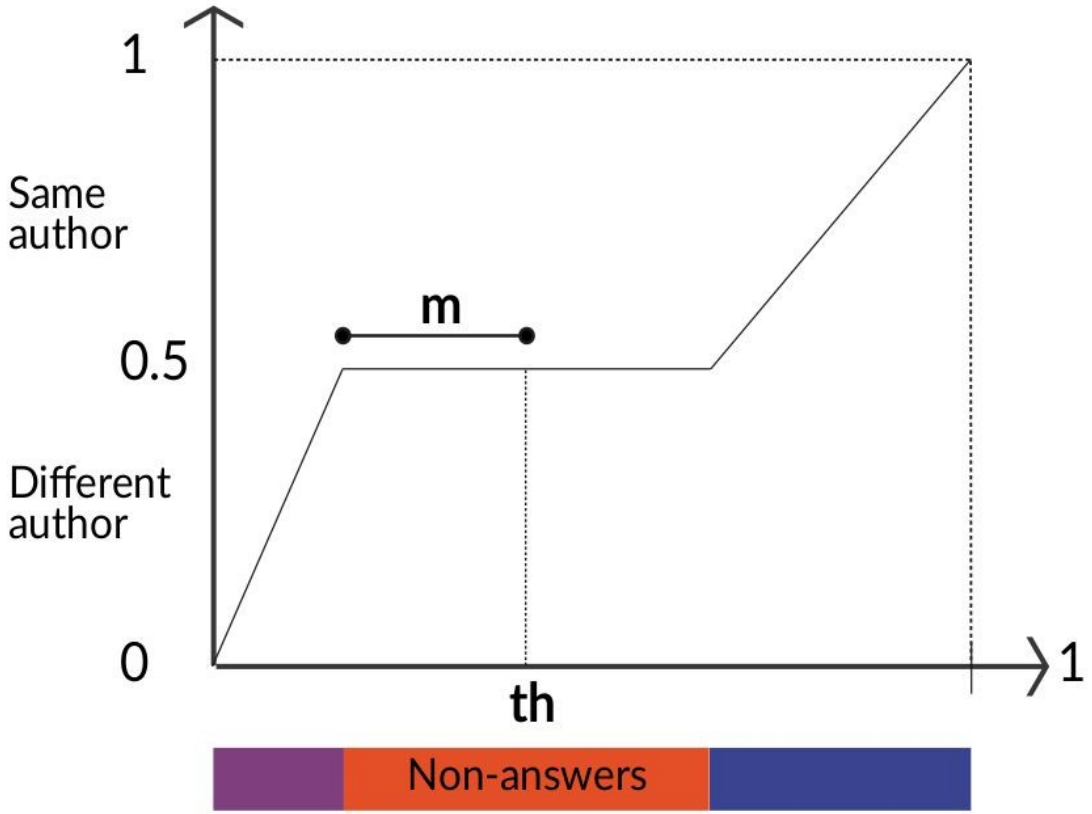
Ajuste de threshold

El ajuste de threshold se realizó con la intención de que el modelo final asignará el puntaje 0.5 a algunas instancias. De esta forma, los pares de textos para los que fuera difícil decidir si eran de un mismo autor o no, obtienen valor de 0.5.

Algunas métricas utilizadas para evaluar la tarea de Verificación de Autoría consideran que es mejor clasificar a una instancia como “no concluyente” que asignarle la etiqueta equivocada.

Este ajuste se propuso debido a que, en la tarea de Verificación de Autoría para el PAN 2021, algunas de las métricas utilizadas consideraban lo anterior.

Para el ajuste de threshold, el modelo final considera el valor obtenido por el modelo de red neuronal siamesa y le aplica la siguiente función:



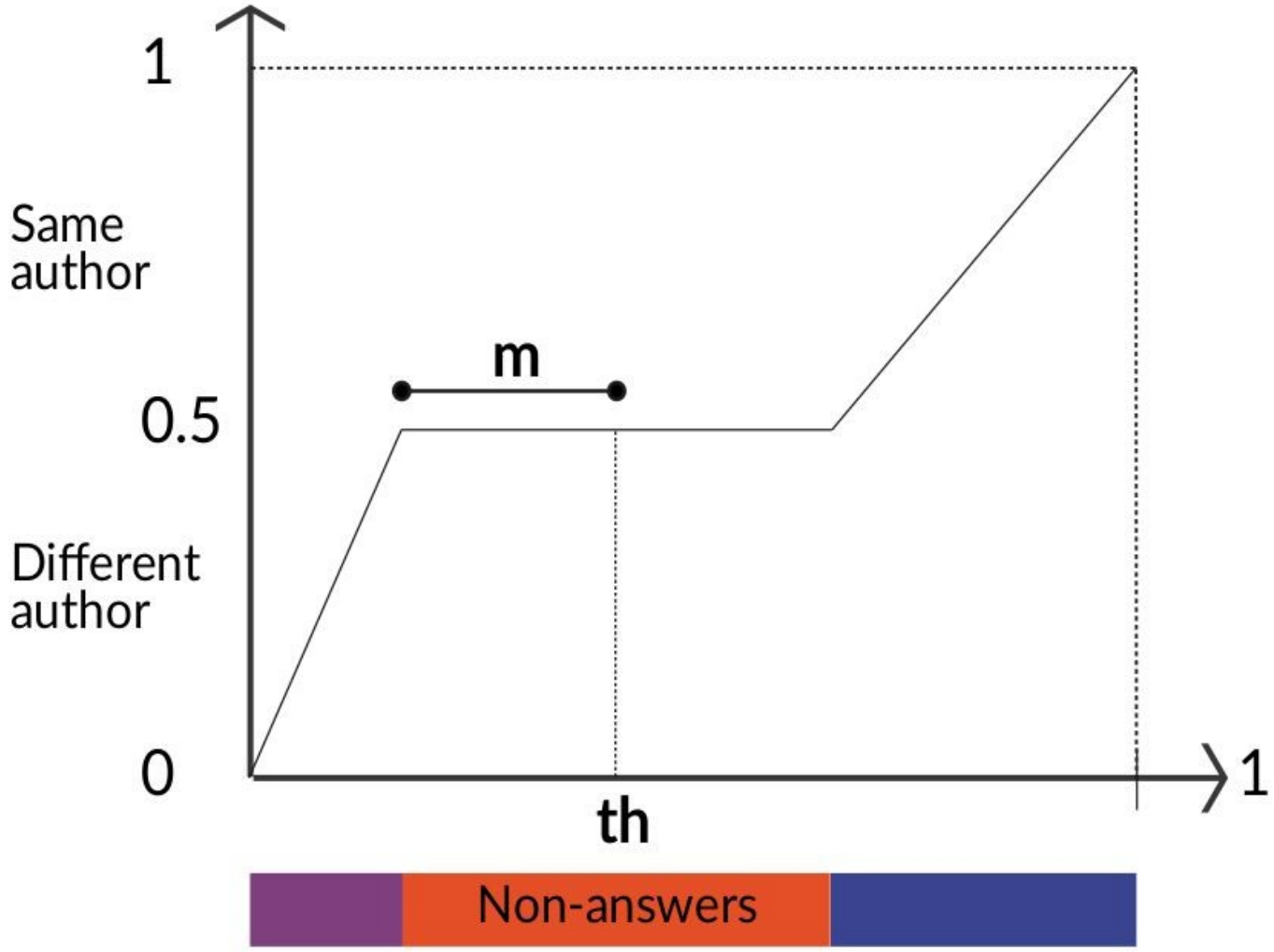
m: Margen
th: threshold

Esta es una función de [0 ,1] en [0,1] no decreciente.
Transforma:

- El intrevalo [0, th-m] en el intervalo [0 , 0.5].
- El intervalo [th-m, th+m] en 0.5.
- El intervalo [th+m, 1] en [0.5, 1].

Para nuestro ajuste necesitamos definir los valores:

- m: margen
- th: threshold



Comenzamos con un modelo de red neuronal siamesa entrenado, ninguno de los parámetros de este se va a modificar.

Consideramos las posibles parejas de valores (th, m) cuando th y m varían de la siguiente manera:

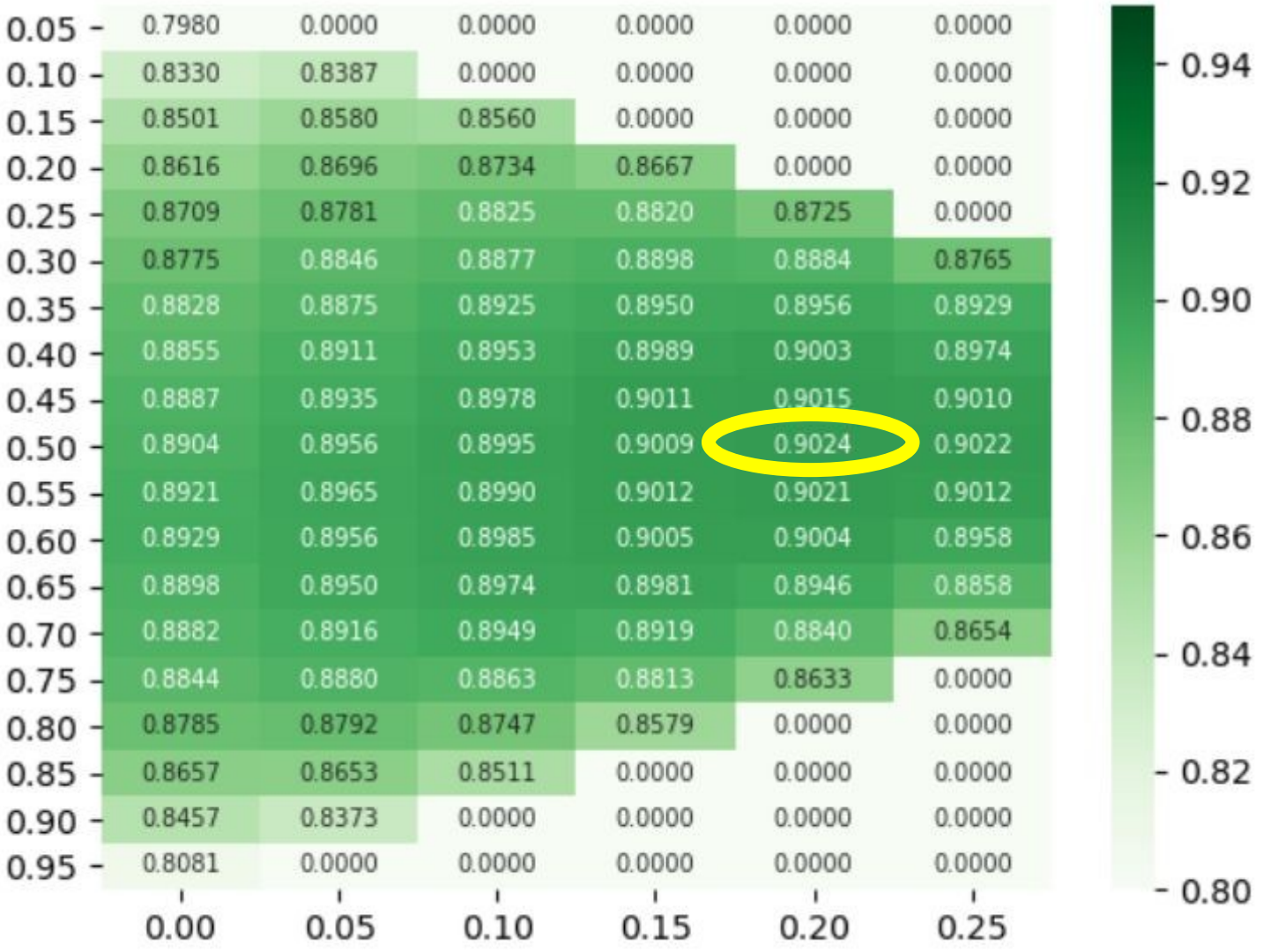
- m en {.05, .10, .15, .20, ... , .95}
- th en {0, 0.05, 0.10, ... , 0.25, 0.30}

Para todas las instancias en el conjunto de **validación,** obtenemos las predicciones usando el modelo entrenado.

Para cada pareja (th, m) se ajustan las predicciónes. Usando las predicciones ajustadas, se calculan las métricas. Elegimos th y m como la pareja que obtuvo las mejores métricas con esta transformación.

Las métricas utilizadas fueron: F1, AUC-ROC, F0.5u, C@1 y Brier score.

Promedio de las métricas para cada pareja (m, th). Elegimos la pareja con puntaje más alto.



Eje X: valores de m (margen)

Eje Y: valores de th (threshold)

Evaluación

Para evaluar, consideramos el modelo de red neuronal siamesa entrenado y la pareja (th, m) con mejor puntaje. Es decir, el modelo de red neuronal con los parámetros tal cual estaban en la mejor época.

Ahora queremos cuantificar que tan bien se desempeña el modelo. Para esto utilizamos las instancias del conjunto de prueba (las cuales no se han usado ni para ajustar parámetros, ni para seleccionar la mejor época, ni para encontrar la pareja (th, m)).

Para cuantificar esto:

- El modelo no actualiza en ninguna forma sus parámetros, estos ya están definidos. Tampoco se cambia la pareja (th, m)
- Para cada instancia en el conjunto de prueba: Predecimos un número entre 0 y 1 utilizando nuestro modelo y el ajuste sobre el par de textos.
- Con las predicciónes y las etiquetas de todas las instancias calculamos métricas adecuadas al problema.

Las métricas utilizadas fueron: F1, AUC-ROC, F0.5u, C@1 y Brier score

Pregunta práctica

Segunda sección

Planteamiento

- Contamos con seis documentos en formato .doc
- Suponemos que cada documento fue escrito por un único autor.

- Preguntas:
  - ¿Podemos decir cuántos autores hay en este conjunto de documentos?
  - ¿Podemos decir cuáles textos corresponden a cada autor?

Documentos

Nombre	Palabras
EE1	2187
EE2	1649
EE3	575
EE6	1565
EE7	889
EE8	527

- Todos los textos en inglés.

Experimento

Contamos con dos modelos distintos e independientes entrenados cada uno en el conjunto de datos “small” y “large” respectivamente. Por simplicidad los llamaremos modelo “small” y modelo “large”.

Por modelo nos referimos al modelo de red neuronal entrenado con el ajuste de threshold también ya definido. Los modelos ya tienen sus parámetros definidos, no se actualizarán dichos parámetros.

Recordemos que un modelo:

- Recibe dos textos y devuelve un número entre 0 y 1. Este número ya considera el ajuste de threshold.
- Los puntajes devueltos representan:
  - < 0.5 : Probablemente distinto autor
  - = 0.5 : No concluyente. El modelo considera este par un problema difícil de decidir.
  - > 0.5 : Probablemente mismo autor

Recordemos también que el modelo fue desarrollado para devolver un valor de 0.5 en aquellos pares de textos que considerará difíciles de clasificar.

Nuestro modelo fue desarrollado para recibir dos textos y devolver un valor en función de estos. Para este nuevo problema, necesitamos hacer una clasificación en un conjunto de textos con más de dos elementos.

Decidimos utilizar los valores devueltos para cada pareja de textos para intentar resolver la pregunta. Hay que considerar que estos valores no se desarrollaron para responder este tipo de problema.

Para este experimento, se obtuvieron los valores para las parejas de textos de la siguiente forma:

- Se leen los textos y se ordenan de acuerdo a su nombre.
- Se consideran las parejas de textos distintos. Solo se considera la primera de las parejas (a, b) y (b, a).

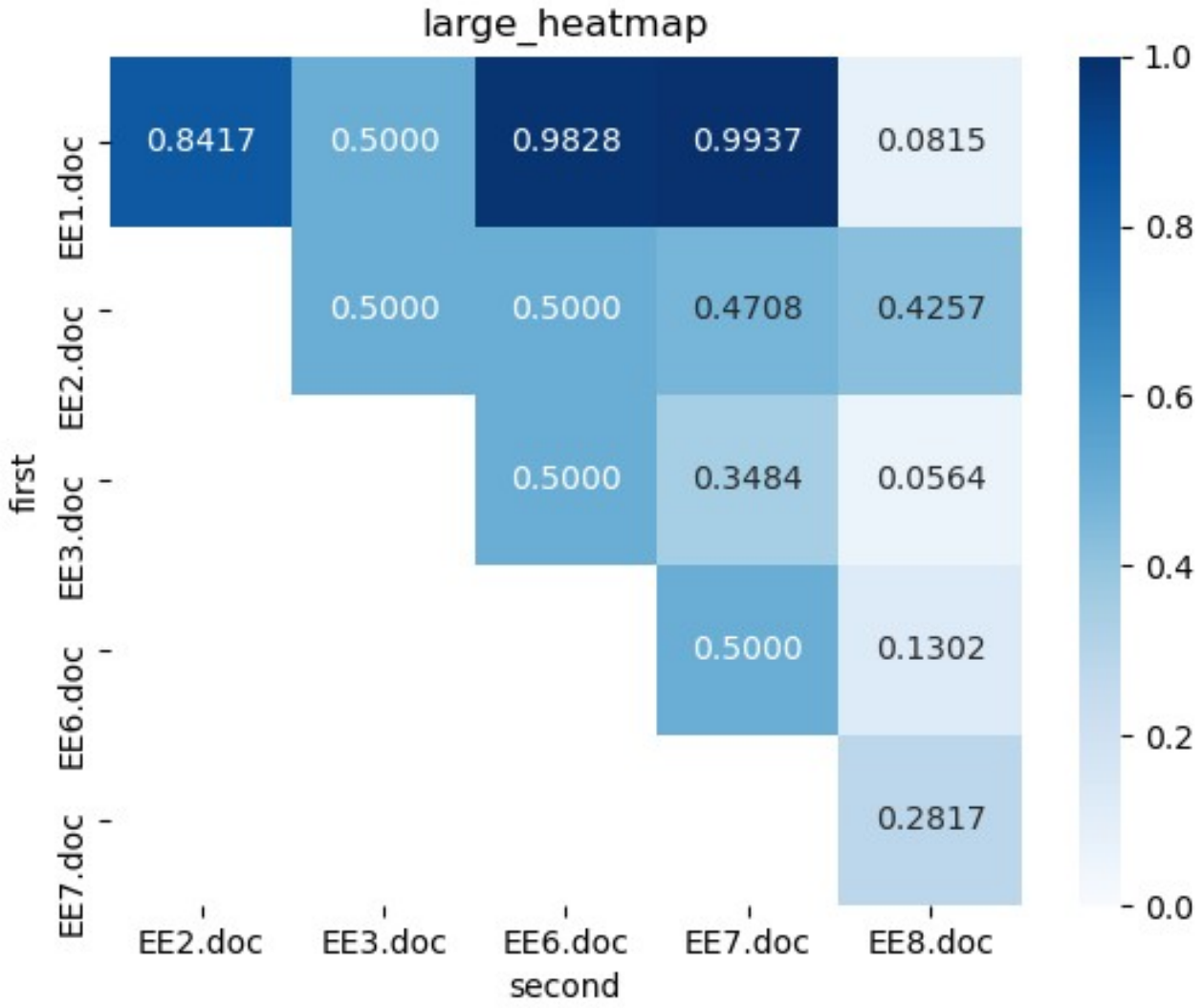
Para cada pareja se obtiene la predicción con el modelo.

Resultados

Cada celda muestra el puntaje de una pareja al evaluarla en el modelo entrenado en el conjunto “large” con el ajuste de threshold.

Cada celda representa la pareja formada por el documento en su celda y el documento en su columna.

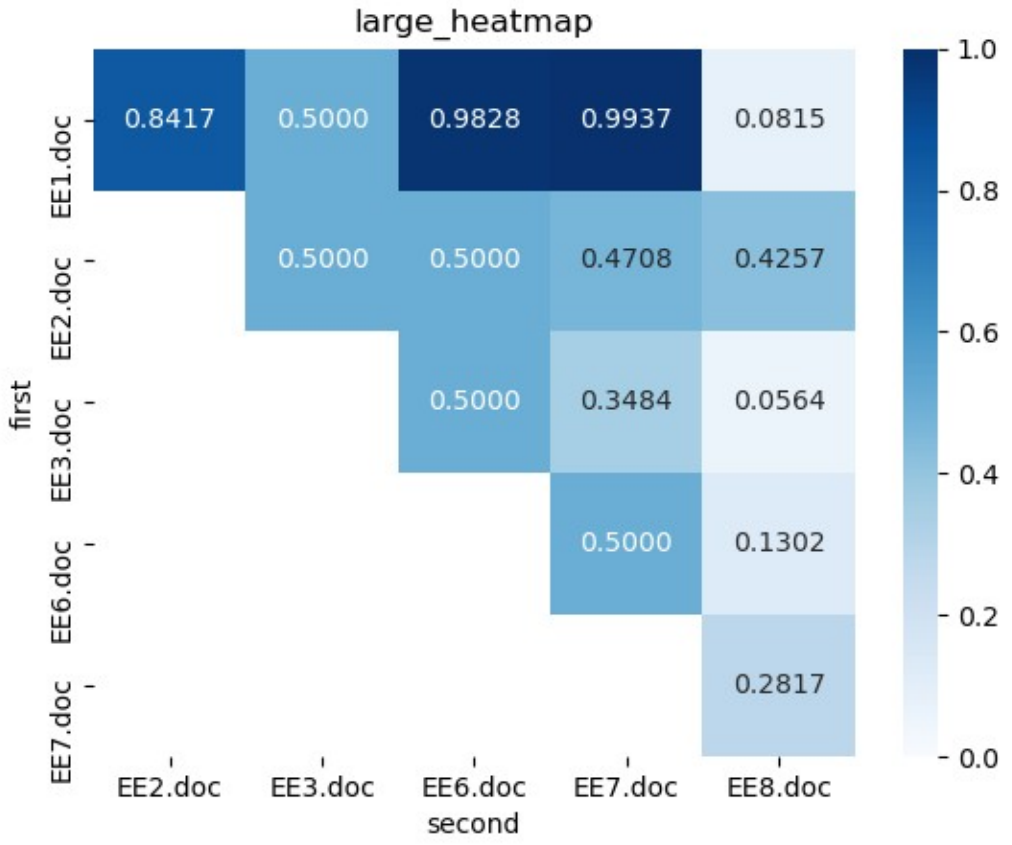
Celdas con colores claros representan valores bajos y celdas con colores oscuros representan valores altos.



Nuestro modelo nos devuelve un valor por cada pareja. Para tratar de responder cuantos autores escribieron estos textos, tenemos que clasificar los textos.

Vamos a clasificar los textos a y b como de un mismo autor si la pareja (a, b) tuvo puntaje mayor a 0.5.

En este caso vemos que las parejas (EE1, EE2), (EE1, EE6) y (EE1, EE7) obtienen puntaje mayor a 0.5. Por esta razón diremos que los textos EE1, EE2, EE6 y EE7 fueron escritos por un mismo autor.



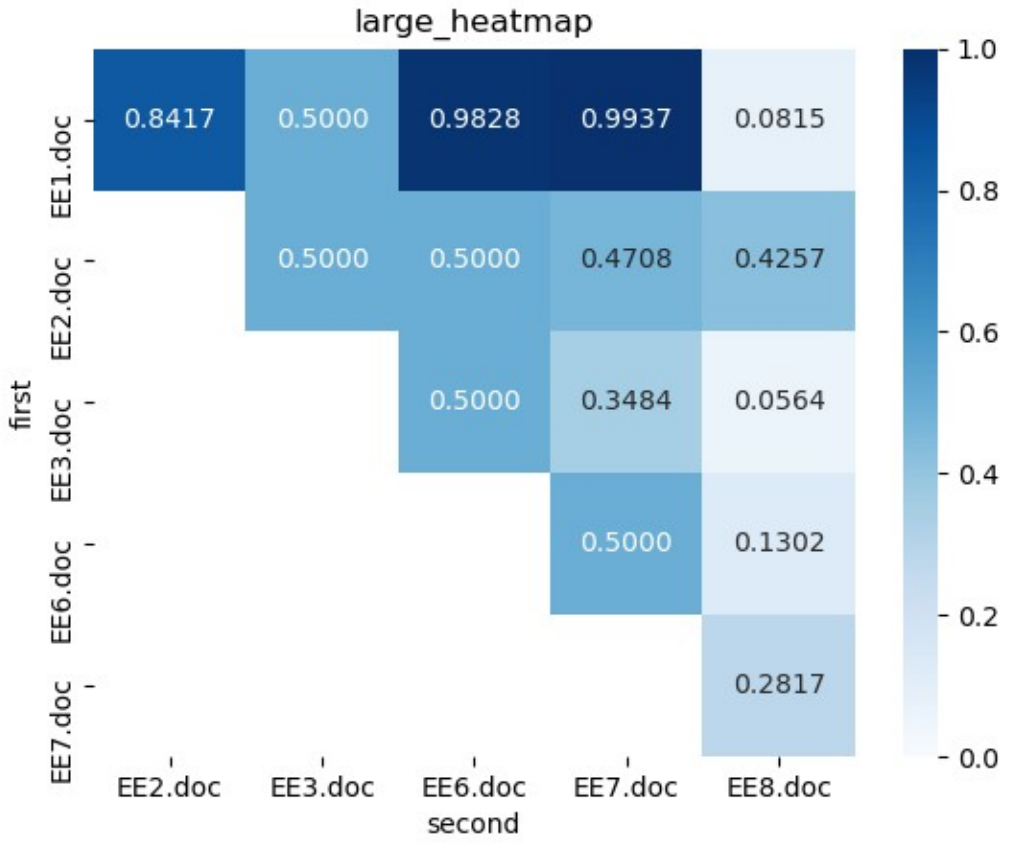
Cualquier pareja que tiene al texto EE8 obtiene puntajes menores a 0.5. Por esta razón diremos que este texto no comparte autor con ningún otro.

Para el texto EE2, vemos que no hay parejas mayores a 0.5 que nos permitan concluir que este texto fue escrito por el mismo autor que otro. Consideraremos que EE2 no comparte autor con otro texto.

Nuestros textos quedan clasificados:

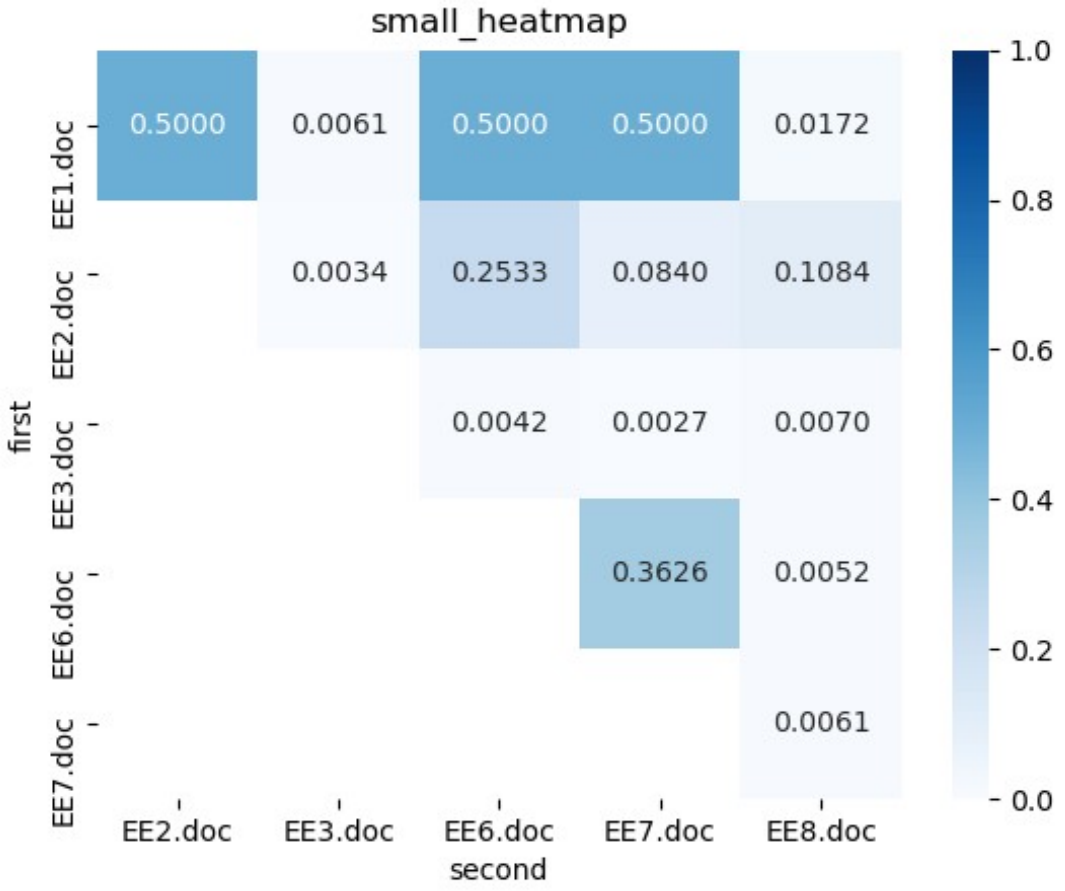
EE1: Autor 1
EE2: Autor 1
EE3: Autor 2
EE6: Autor 1
EE7: Autor 1
EE8: Autor 3

De acuerdo a lo observado hay 3 autores representados en estos textos.



El modelo “large” fue el modelo que obtuvo los mejores resultados para la tarea de Verificación de Autoría. Se presentan los resultados del modelo “small” en la tabla a fin de conocer los puntajes que da este modelo en este nuevo problema.

Con estos puntajes, no podemos concluir que ninguna de las parejas haya sido escrita por un mismo autor. En este caso, no nos es posible responder a las preguntas planteadas en el problema.



Experimento de verificación de autores

¡Gracias!