Verificación de autoría de discurso cruzado basada en grafos

Jorge Alfonso Martínez Galicia

Contenido

- Introducción
- Objetivo
- Pregunta de investigación
- Corpus
- Metodología
- Resultados
- Conclusión

Introducción

- Análisis de autoría.
- Retos en la análisis de autoría.
- Enfoques tradicionales.
- Representación de textos como grafos.
- Verificación de autoría de discurso cruzado basada en grafos.

Objetivo

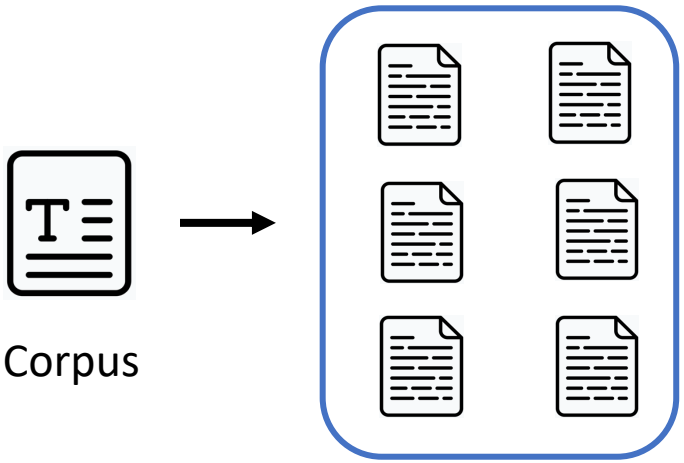
Identificar si dos documentos fueron escritos por el mismo autor, representando cada texto como un grafo utilizando como modelo una arquitectura de red siamesa la cuál se propone en artículo [2], este enfoque será entrenado y evaluado con el corpus del PAN@CLEF 2022 desafío de verificación de autoría [3].

Pregunta de investigación

Dados dos textos pertenecientes a diferentes tipos de discurso, ¿Es posible determinar si fueron escritos por el mismo autor?

Corpus PAN@CLEF 2022

Parejas de textos (problemas)

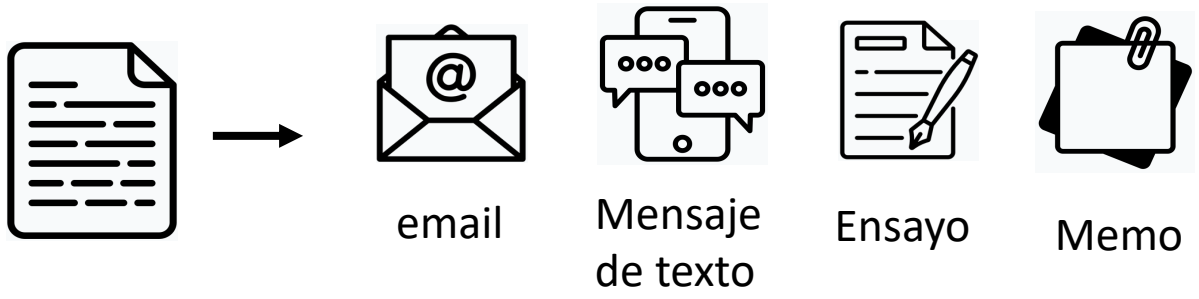


Corpus

Metadatos

- id de los autores
- id del problema
- Tipo de discurso
- Verdad fundamental (Truth)

Tipos de discurso (TD)



Dado que la longitud del texto de los mensajes de correo electrónico y de texto puede ser muy pequeña, cada texto perteneciente a estos TD es una concatenación de diferentes mensajes. Se usa la etiqueta <nuevo> para indicar los límites de los mensajes.

Corpus PAN@CLEF 2022

- Todos los autores tienen una edad similar (18-22) y son hablantes nativos de inglés.
- El tema de las muestras de texto no está restringido.
- El nivel de formalidad puede variar dentro de un determinado tipo de discurso.
  - Por ejemplo, los mensajes de texto pueden estar dirigidos a familiares o conocidos no familiares.

Corpus PAN@CLEF 2022

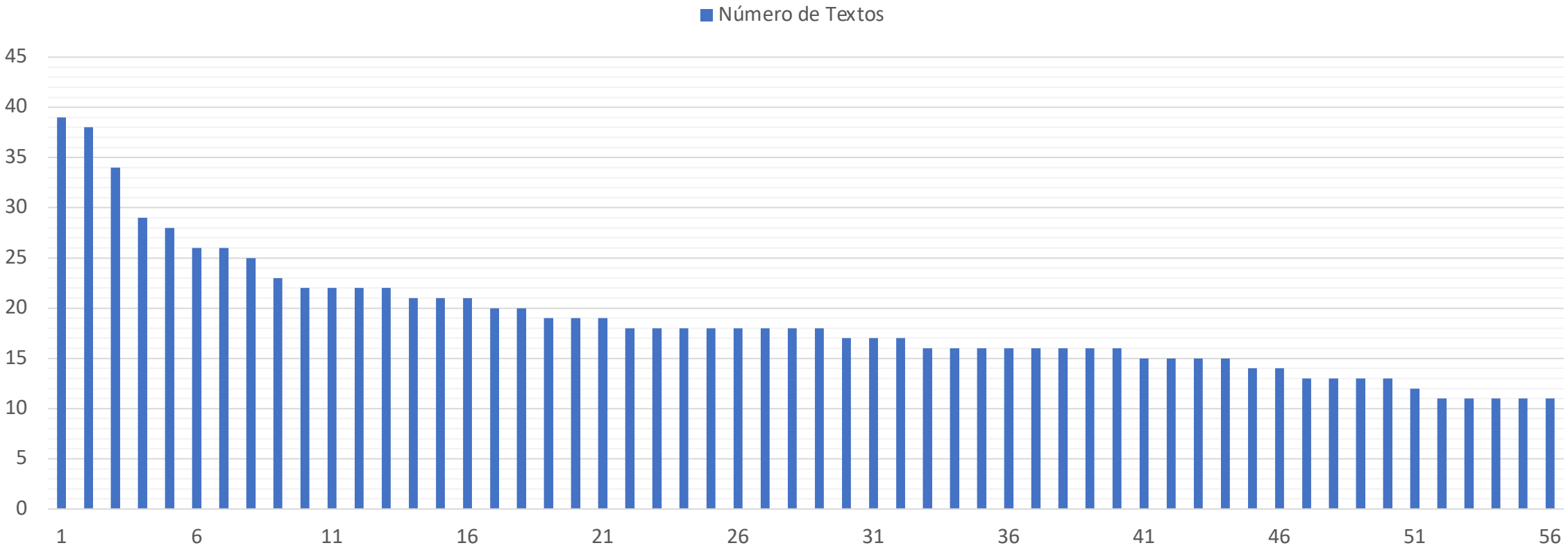
¿Cuándo un problema es positivo?

Cuando ambos textos del problema fueron escritos por el mismo autor.

Corpus PAN@CLEF 2022

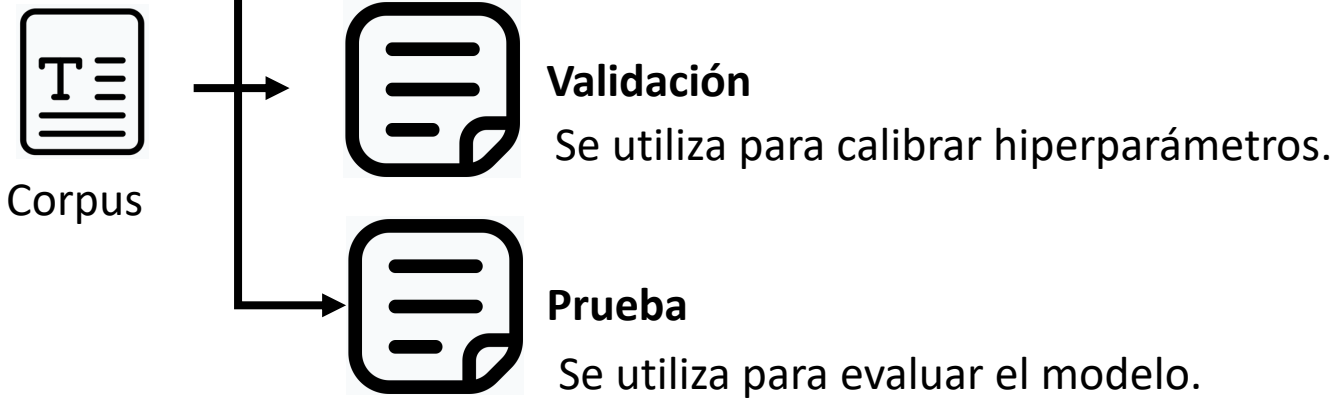
- Textos totales: 1,046
- Autores: 56
- Parejas de textos (problemas): 12,264
- Problemas positivos: 6132

Número de textos por autor



División del corpus

- Se divide el corpus en tres partes: entrenamiento, validación y prueba.



Características que deben cumplir las particiones:

- Deben ser ajenas en autores, es decir, no deben compartir textos de un mismo autor.
- Las particiones deben de estar calibradas en cuanto al número de problemas positivos y negativos.

División del corpus

- Textos totales: 1,046
- Autores: 56
- Parejas de textos (problemas): 12,264
- Problemas positivos: 6132

La problemática que se encuentra al tratar de realizar las particiones, es que no se pueden generar particiones ajenas en autor, esto debido a:

- Los autores están relacionados entre si por algún problema.
- El número de problemas es demasiado, considerando la cantidad de textos y autores con los que se cuentan.

División del corpus

Solución propuesta a la problemática de las particiones.

- Las particiones se realizan sobre la lista de autores, tomando aquellos autores que cuentan con mas textos para la partición de entrenamiento.

Esta solución resulta efectiva para cumplir con el requerimiento de autor ajeno pero no con el de particiones balanceadas.

División del corpus

Para solventar el problema de particiones no balanceadas, se generan problemas nuevos.

Aspectos a considerar para generar nuevos problemas:

- No pueden existir parejas de textos del mismo tipo de discurso.
- Se obtienen las parejas de tipos de discurso utilizando un clasificador bayesiano, para identificar que pareja le corresponde a cierto tipo de discurso con base a la probabilidad conjunta de las parejas del corpus.
- Parejas de TD obtenidas:
  - Email, mensaje de texto
  - Memo, email
  - Ensayo, email

División del corpus

- Para el conjunto de entrenamiento se generan 9,843 nuevos problemas de los cuales 2,397 son positivos.
- Para el conjunto de validación se generan 467 nuevos problemas de los cuales 103 son positivos.
- Para el conjunto de prueba se generan 659 nuevos problemas de los cuales 146 son positivos.

Particiones resultantes

Partición	Total de problemas	Positivos
Entrenamiento	15,732	7866
Validación	754	377
Prueba	1070	532

Modelado de textos como gráficos

Esta modelado se realizó con basa en la propuesta señalada en [2]. Este tipo de representación de texto intenta capturar la relación entre las palabras y las etiquetas POS.

A continuación se muestra el proceso para llevar acabo esta representación, utilizando una oración extraía del corpus.

<nl>I am a Second year <course> student at <university>. I am interested.

Modelado de textos como gráficos

El texto es preprocesado realizando los siguientes pasos:

1. Se sustituyen los caracteres non-ASCII por su equivalente en ASCII.
2. Se tokeniza el texto, tomando las etiquetas <ln>, <Universe>, <new>, etc. que aparecen en el corpus como tokens individuales.
3. Se obtienen el POS.
4. Se pasa todo el texto a minúsculas.

Cabe destacar que no se remueven signos de puntuación.

Modelado de textos como gráficos

Las etiquetas POS utilizadas son PENN-Treebank [5], adicional a esto se agregan dos etiquetas mas $PUNCT para todos los signos de puntuación y $OTHER para cualquier otra palabra que el modelo de NLTK no identifique.

Después del proceso de tokenización y de obtener el POS, el resultado es el siguiente:

```
[('<nl>','NN'), ('i','PRP'), ('am','VBP'), ('a','DT'), ('second','JJ'),
('year','NN'), ('<course>','FW'), ('student','NN'), ('at','IN'),
('<university>','NNP'), ('.','$PUNCT'), ('i','PRP'), ('am','VBP'),
('interested','JJ'), ('.','$PUNCT')]
```

Modelado de textos como gráficos

El grafo de coocurrencia se construye considerando que dos palabras coexisten si aparecen una al lado de la otra en el texto.

El grafo esta definido como un par ordenado:

$$G = (V, E)$$

Donde:

$V =$ *conjunto de vértices compuesto por tuplas (palabra,POS).*
$E = conjunto\ de\ aristas$.

Modelado de textos como gráficos

Para poder cambiar la estructura del grafo así como la información extraída del texto se define un conjunto de etiquetas POS denotadas como REDUCE_LABELS.

Con todo lo anterior definido se construye el grafo con la siguiente definición.

$$M[i] = \begin{cases} (palabra, POS)\ si\ POS\ \notin REDUCE\_LABELS \\ (POS, POS)\ si\ POS\ \in REDUCE\_LABELS \end{cases}$$

Donde:
$M = lista\ de\ tuplas\ enmascaradas$.

Modelado de textos como gráficos

Con esta construcción se identifican todas las tuplas con una etiqueta específica en el conjunto REDUCE_LABELS como un solo nodo.

En los experimentos realizados al modelo se evalúan grafos generados con diferentes conjuntos de REDUCE_LABELS.

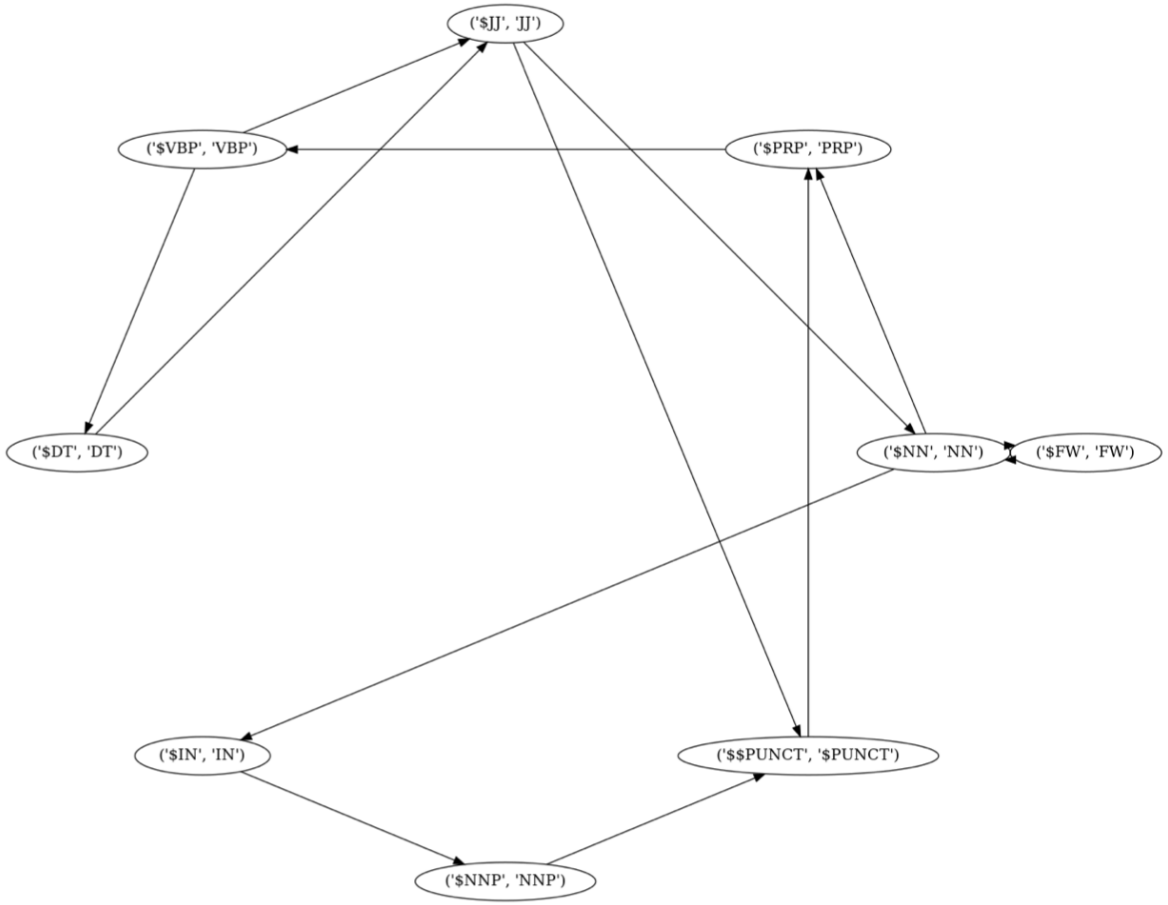
- Grafo short: grafo generado usando todo el conjunto de posibles etiquetas POS como REDUCE_LABELS.
- Grafo full: grafo generado usando REDUCE_LABELS igual a vacío.

Modelado de textos como gráficos

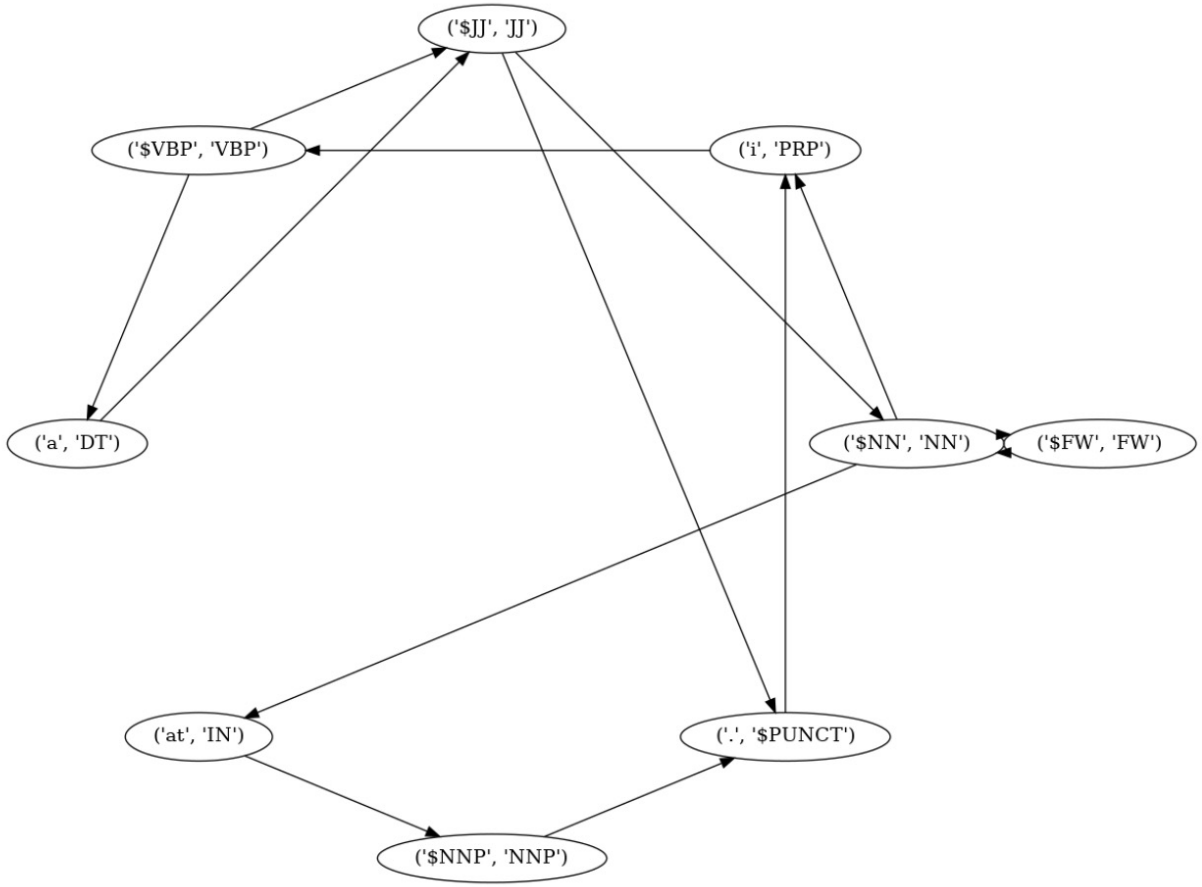
- Grafo med: grafo generado usando el siguiente conjunto como REDUCE_LABELS, que corresponden a adjetivos, sustantivos, adverbios, verbos, etc.:

```
REDUCE_LABELS = [    'JJ',     'JJR',    'JJS',              #Adjectives
                     'NN',     'NNS',    'NNP',   'NNPS',    #Nouns
                     'RB',     'RBR',    'RBS',              #Adverbs
                     'VB',     'VBD',    'VBG',              #Verbs
                     'VBN',    'VBP',    'VBZ',              #Verbs
                     'CD',                                   #Cardinal numbers
                     'FW',                                   #Foreign words
                     'LS',                                   #List item marker
                     'SYM',                                  #Symbols
                     '$OTHER',                               # Others]
```

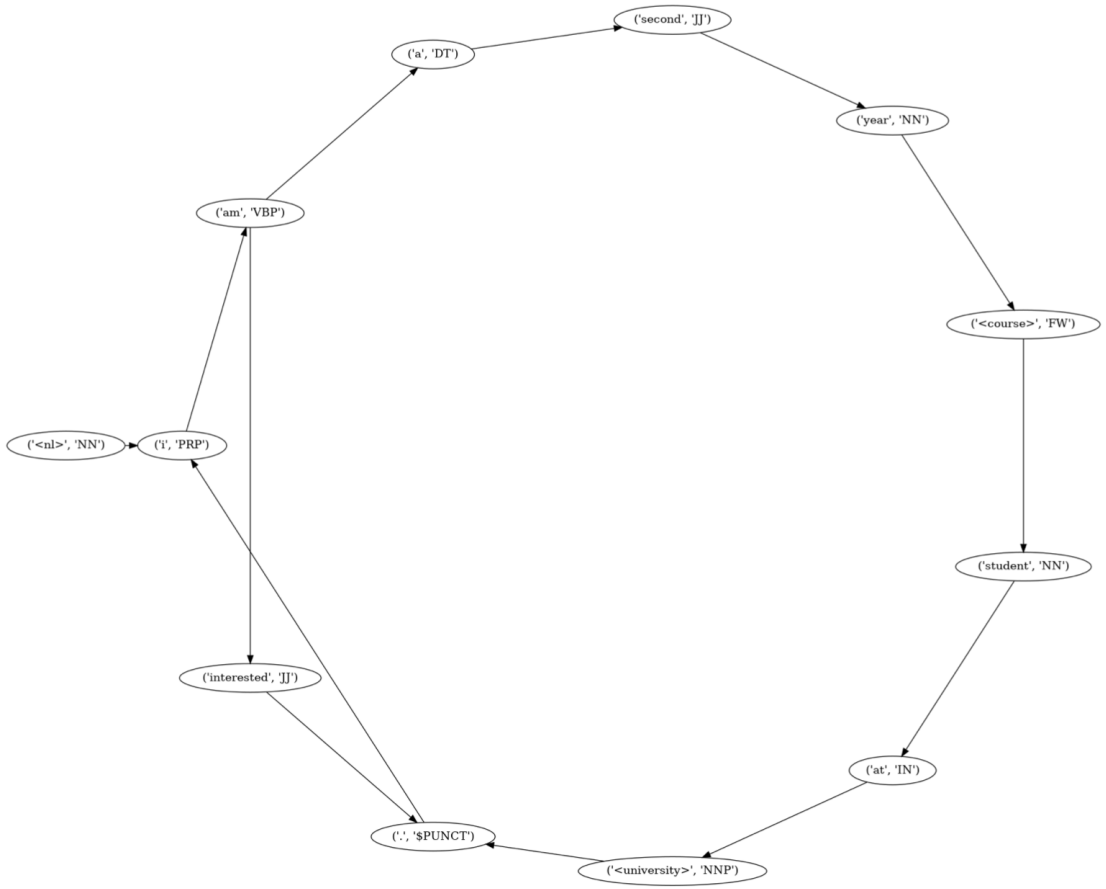
Grafo short



Grafo med

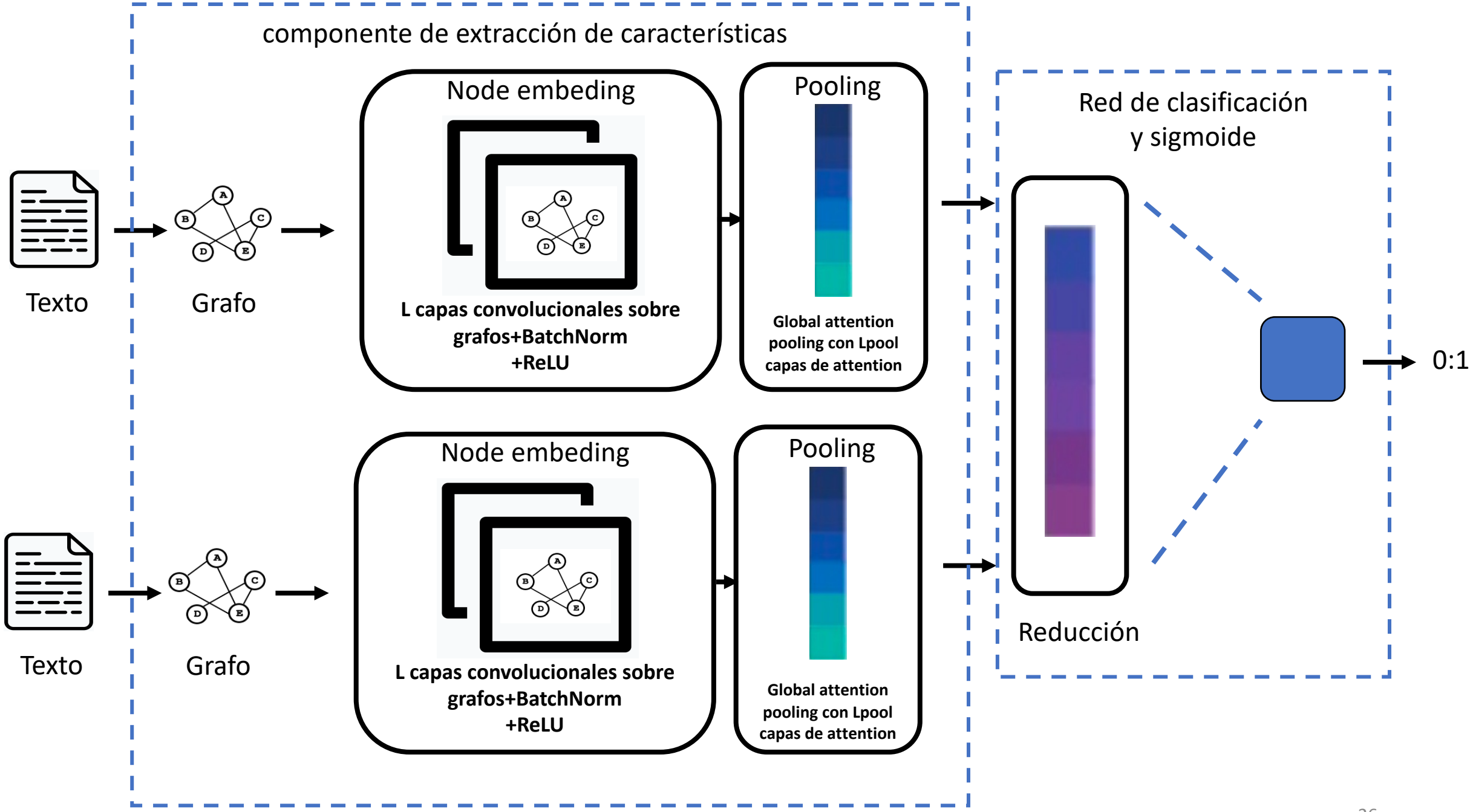


Grafo full


('second', 'JJ')
('a', 'DT')
('year', 'NN')
('am', 'VBP')
('<course>', 'FW')
('<nl>', 'NN')
('i', 'PRP')
('student', 'NN')
('interested', 'JJ')
('at', 'IN')
('.', '$PUNCT')
('<university>', 'NNP')

Red siamesa basada en grafos

Se utiliza el modelo de red siamesa que se propone en [2], esta construida por dos componentes idénticos de extracción de características con pesos compartidos, un paso de reducción y una red de clasificación.


componente de extracción de características
Texto
Grafo
Node embeding
L capas convolucionales sobre grafos+BatchNorm +ReLU
Pooling
Global attention pooling con Lpool capas de attention
Texto
Grafo
Node embeding
L capas convolucionales sobre grafos+BatchNorm +ReLU
Pooling
Global attention pooling con Lpool capas de attention
Red de clasificación y sigmoide
Reducción
0:1

Experimentos

Se realizaron una serie de experimentos, variando los siguientes hiperparámetros:

- Tipo de gráfica (short, med o full)
- Número de capas convolucionales sobre grafos
- Tipo de capa de convolución sobre grafo
  - LeConv
  - GraphConv
  - TAGConv
- Número de capas de la red neuronal de la etapa de pooling.
- Número de capas de la red de clasificación.
- Función de pérdida.

Resultados

En la siguiente tabla se muestra el desempeño de las diferentes configuraciones de grafo.

Cada fila muestra el promedio de las cinco medidas de evaluación del modelo (AUC, F1, c@1, F_0.5u, Brier).

	Partición de prueba
Grafo short	**60.9**
Grafo med	56.36
Grafo full	59.98

Resultados

Medidas de evaluación en particiones de validación y prueba del modelo con tipo de gráfica short.

	Partición de validación	**Partición de prueba**
Promedio	0.614623	0.602744
AUC	0.596329	0.537275
F1	0.666667	0.665493
C@1	0.505474	0.511568
F_0.5u	0.557613	0.559668
Brier	0.747029	0.739715

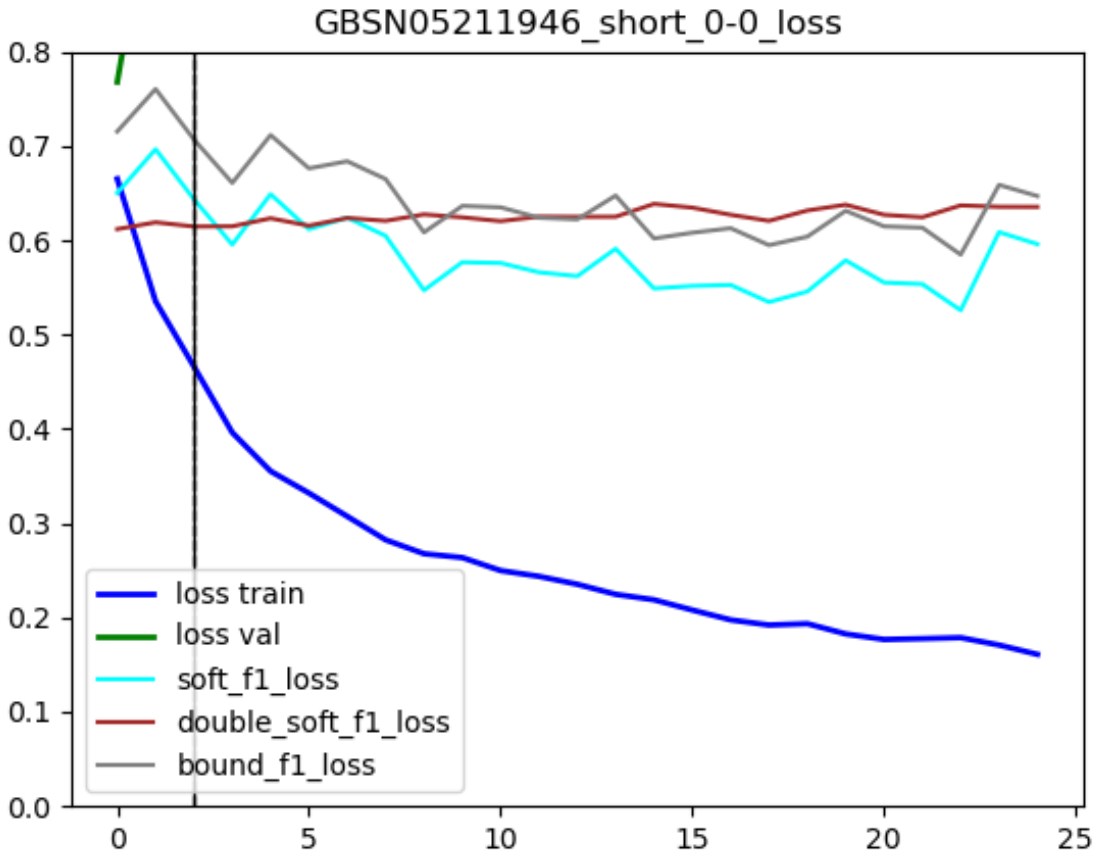
Resultados

Hiperparámetros utilizados para la red siamesa basada en grafos

Hiperparámetro	Valor definido
Tipo de gráfica	Short
Número de capas convolucionales sobre grafos	2
Tipo de capa de convolución sobre grafo	LeConv
Número de capas de la red neuronal del pooling	2
Número de capas de la red de clasificación	2
Función de pérdida	soft_f1

Resultados

Comportamiento de funciones de perdida en validación y prueba



Conclusiones

- Sobre ajuste del modelo en las primeras configuraciones.
- Ajuste de hiperparámetros y mejoramiento del rendimiento.
- Pruebas adicionales.
- Mejoras al modelo.

Referencias

1. Mihalcea, R.; Radev, D. *Graph-Based Natural Language Processing and Information Retrieval*; MIT Press: Cambridge, NY, USA, 2011.
2. Embarcadero, Ruiz D., Gómez, Adorno H., Embarcadero, Ruiz A., and Sierra, G. (2022) Graph based Siamese network for authorship verification, *Mathematics*, *10*(2), 277.
3. M. Kestemont, E. Stamatatos, E. Manjavacas, J. Bevendorff, M. Potthast, and B. Stein, Overview of the Cross-Domain Authorship Verification Task at PAN 2021. in: [4].
4. M. Kestemont, E. Manjavacas, I. Markov, J. Bevendorff, M. Wiegmann, E. Stamatatos, B. Stein, and M.Potthast, Overview of the Cross-Domain Authorship Verification Task at PAN 2021.
5. Marcus, M. *Building a Large Annotated Corpus of English: The Penn Treebank*; Technical Report; Defense Technical Information Center: Fort Belvoir, VA, USA, 1993; doi:10.21236/ADA273556.