BARD

Generador de Historias

PROCESAMIENTO DE LENGUAJE NATURAL - 2022

Eugenia Sol Piñeiro

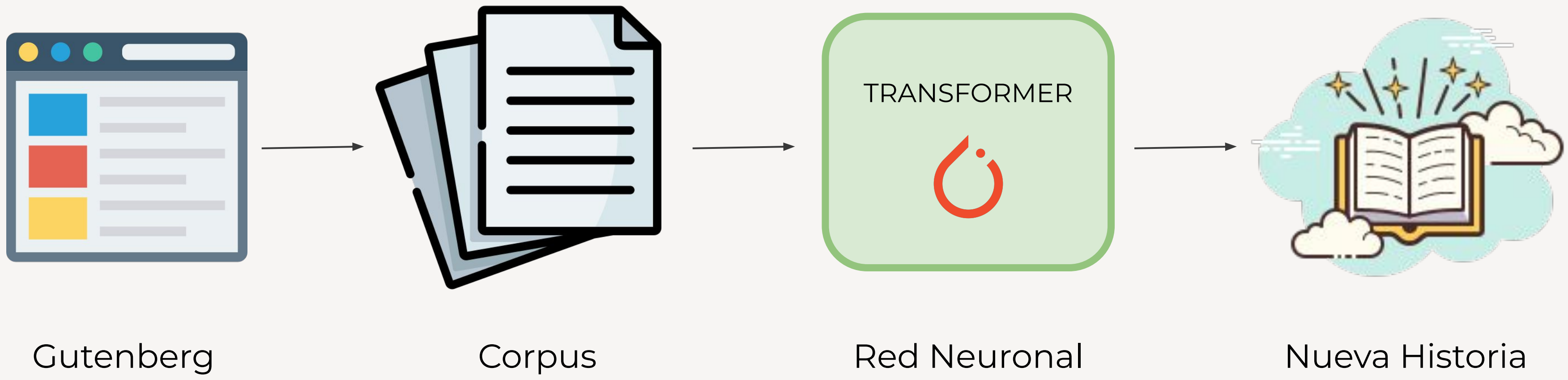
TABLA DE CONTENIDOS

01

INTRODUCCIÓN

Generador de Historias

GENERADOR DE HISTORIAS



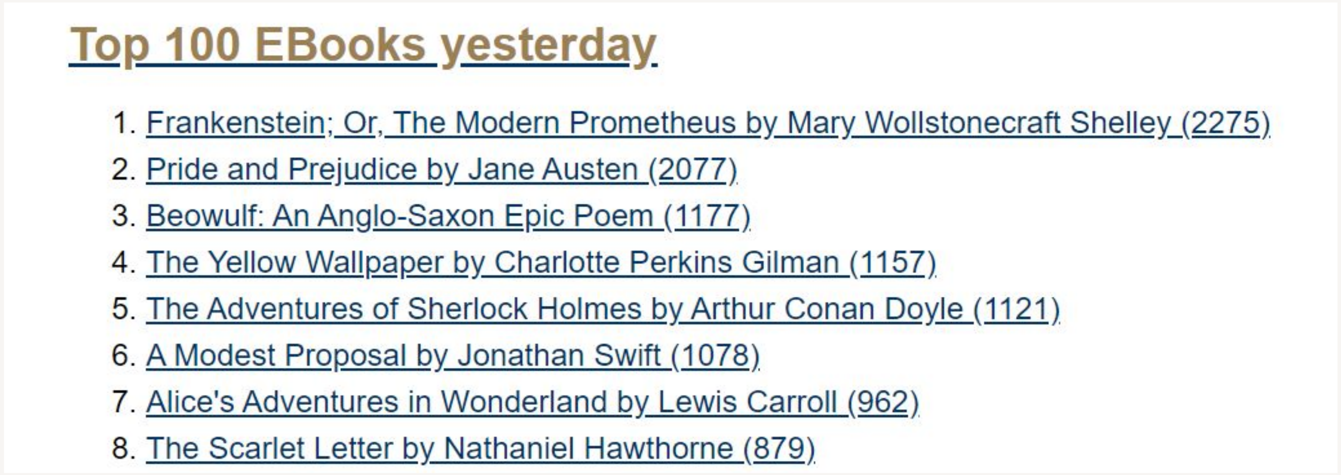
02

CORPUS

Obtención de los textos

WEB SCRAPING

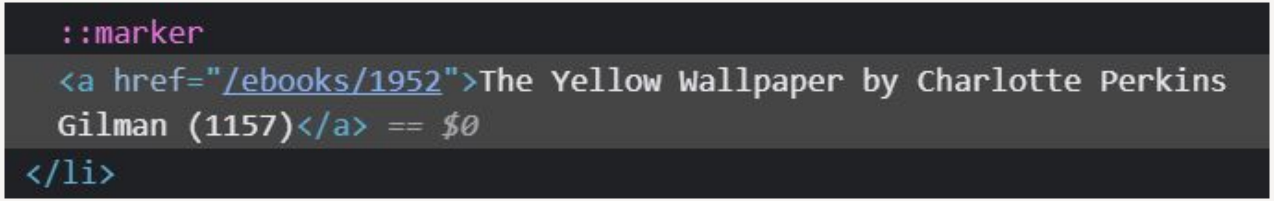
De la página de **Gutenberg** se obtuvieron los **libros más populares**:



Top 100 EBooks yesterday

1. Frankenstein; Or, The Modern Prometheus by Mary Wollstonecraft Shelley (2275)
2. Pride and Prejudice by Jane Austen (2077)
3. Beowulf: An Anglo-Saxon Epic Poem (1177)
4. The Yellow Wallpaper by Charlotte Perkins Gilman (1157)
5. The Adventures of Sherlock Holmes by Arthur Conan Doyle (1121)
6. A Modest Proposal by Jonathan Swift (1078)
7. Alice's Adventures in Wonderland by Lewis Carroll (962)
8. The Scarlet Letter by Nathaniel Hawthorne (879)

WEB SCRAPING



```
  ::marker
  <a href="/ebooks/1952">The Yellow Wallpaper by Charlotte Perkins
  Gilman (1157)</a> == $0
</li>
```

```
get_popular_books("https://www.gutenberg.org/browse/scores/top")  # Obtener HTML

atags = soup.find_all('a')                                   # Obtener todas las tags
tags = [tag.get('href') for tag in atags]
ebook_regex = re.compile(‘/ebooks/’ + '[0-9]+')             # Tag = /ebooks/number


# Returns ebook numbers
text = strip_headers(load_etext(ebook_number)).strip() # Gutenberg API
```

WEB SCRAPING

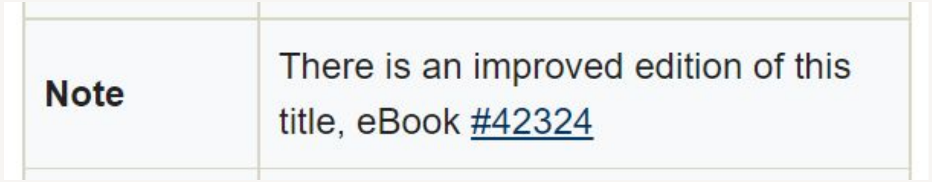
Problema encontrado: Algunos `ebook_number` están deprecados

Solucion:

Si el `ebook_number` está deprecado

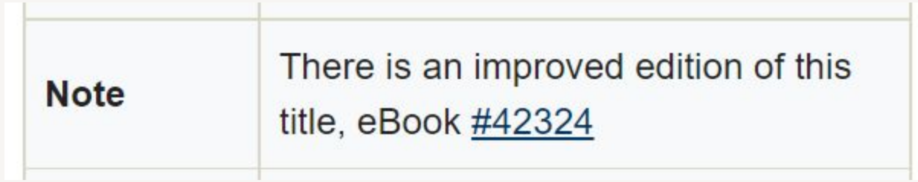
Buscar versión actualizada

Guardar texto

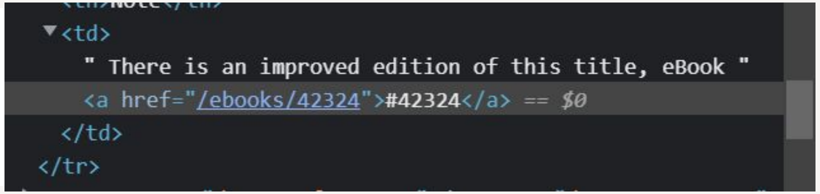


Note	There is an improved edition of this title, eBook #42324

WEB SCRAPING



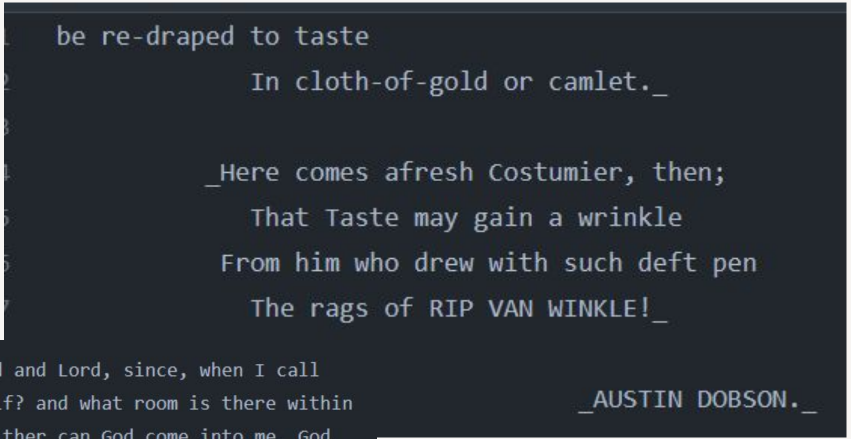
Note	There is an improved edition of this title, eBook #42324



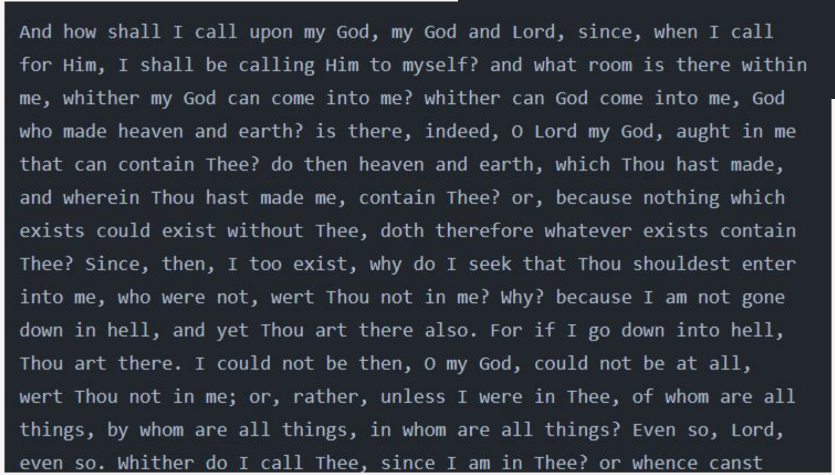
```
▼<td>
   " There is an improved edition of this title, eBook "
   <a href="/ebooks/42324">#42324</a> == $0
 </td>
</tr>
```

```
get_improved_edition(ebook_number):

   page = 'https://www.gutenberg.org/ebooks/' + str(ebook_number)
   r = requests.get(page.replace("\n", ""))        # Nuevo request
   soup = BeautifulSoup(r.content, 'html.parser')  # Obtener HTML

# Conseguir version mejorada
```

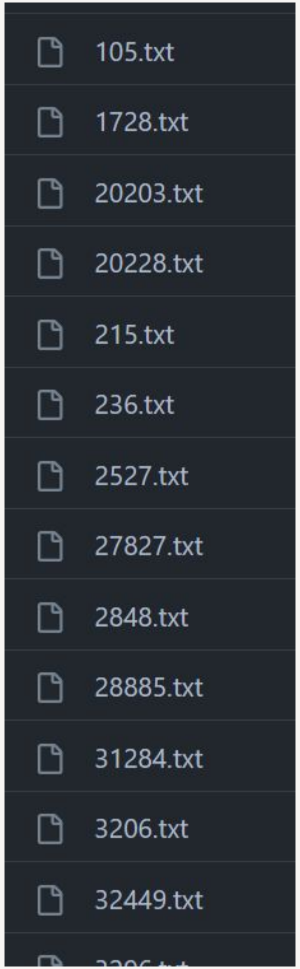
CORPUS



```
be re-draped to taste
             In cloth-of-gold or camlet._

          _Here comes afresh Costumier, then;
             That Taste may gain a wrinkle
           From him who drew with such deft pen
             The rags of RIP VAN WINKLE!_

                                   _AUSTIN DOBSON._
```

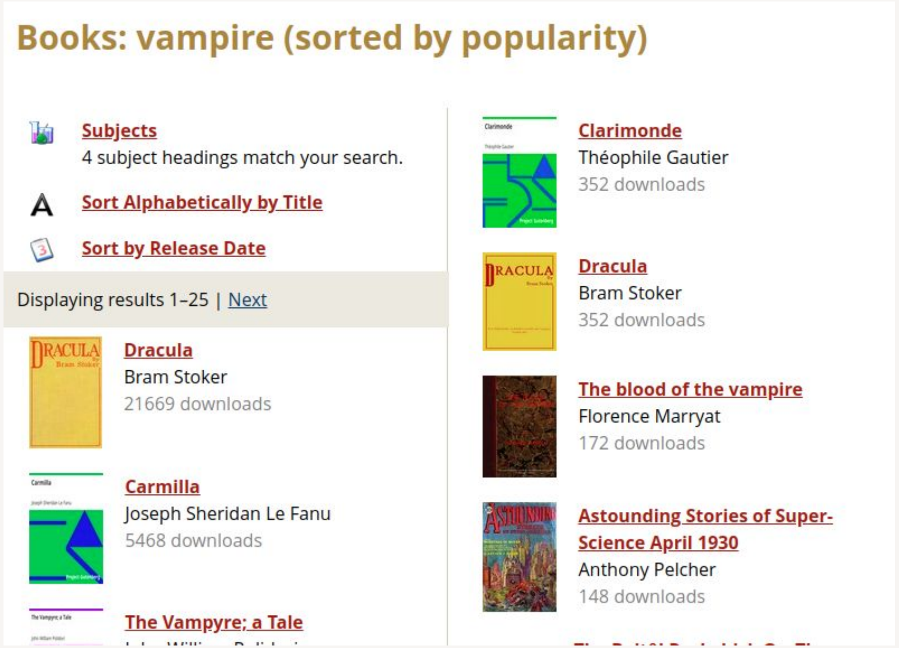


```
And how shall I call upon my God, my God and Lord, since, when I call
for Him, I shall be calling Him to myself? and what room is there within
me, whither my God can come into me? whither can God come into me, God
who made heaven and earth? is there, indeed, O Lord my God, aught in me
that can contain Thee? do then heaven and earth, which Thou hast made,
and wherein Thou hast made me, contain Thee? or, because nothing which
exists could exist without Thee, doth therefore whatever exists contain
Thee? Since, then, I too exist, why do I seek that Thou shouldest enter
into me, who were not, wert Thou not in me? Why? because I am not gone
down in hell, and yet Thou art there also. For if I go down into hell,
Thou art there. I could not be then, O my God, could not be at all,
wert Thou not in me; or, rather, unless I were in Thee, of whom are all
things, by whom are all things, in whom are all things? Even so, Lord,
even so. Whither do I call Thee, since I am in Thee? or whence canst
```



```
105.txt
1728.txt
20203.txt
20228.txt
215.txt
236.txt
2527.txt
27827.txt
2848.txt
28885.txt
31284.txt
3206.txt
32449.txt
```

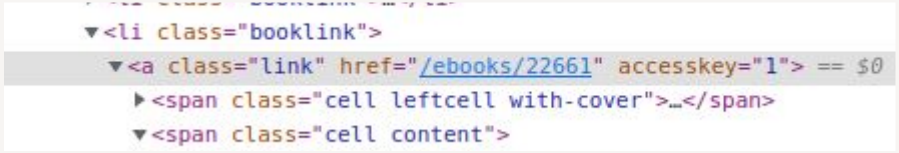
CORPUS ESPECÍFICO



Books: vampire (sorted by popularity)

Subjects
4 subject headings match your search.
Sort Alphabetically by Title
Sort by Release Date
Displaying results 1–25 | Next

Dracula Bram Stoker 21669 downloads	Clarimonde Théophile Gautier 352 downloads
Carmilla Joseph Sheridan Le Fanu 5468 downloads	Dracula Bram Stoker 352 downloads
The Vampyre; a Tale	The blood of the vampire Florence Marryat 172 downloads
	Astounding Stories of Super-Science April 1930 Anthony Pelcher 148 downloads



```
▼<li class="booklink">
  ▼<a class="link" href="/ebooks/22661" accesskey="1"> == $0
    ▶<span class="cell leftcell with-cover">…</span>
    ▼<span class="cell content">
```

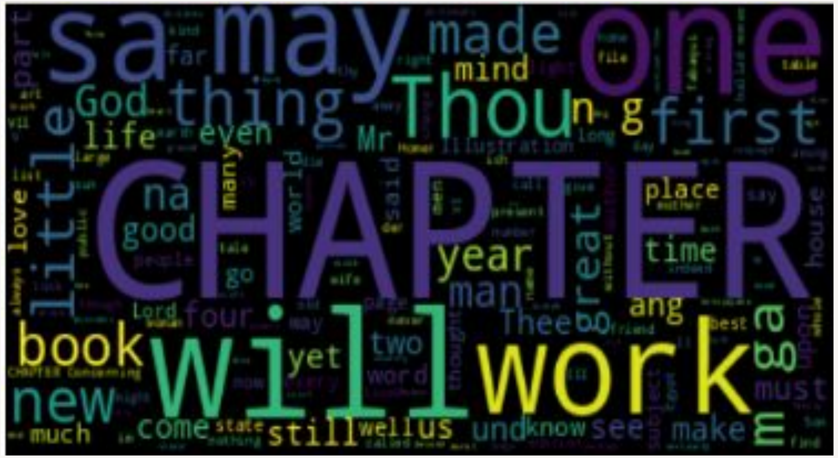
```
build_specific_corpus(query)

page = f'https://https://www.gutenberg.org/ebooks/search/?query={query}&submit_search=Go%21&start_index={len_first_page}'
```

03

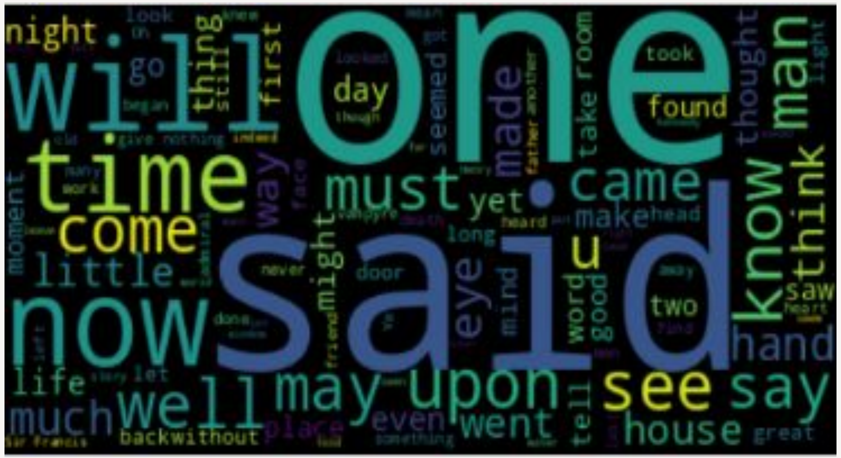
ANÁLISIS

WORD CLOUD

POPULAR



VAMPIRES



Palabras muy genéricas

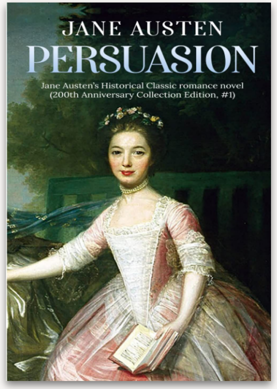
TFIDF

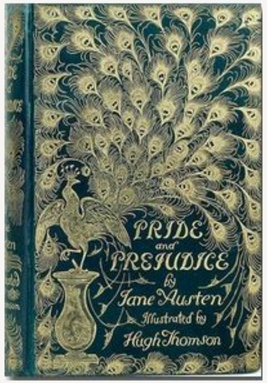
```
tfidf = TfidfVectorizer(
   ngram_range=[1, 1],     # Solo unigramas
   max_df=0.8,             # Document Frequency > 0.8
   min_df=0.1,             # Document Frequency < 0.1
   max_features=None,      # Se consideran todas las features
   analyzer='word',        # Necesario para remover stopwords
   stop_words=stopwords.words('english') # Lista de stopwords
)
```

[Sckit.learn TFIDF - Documentación]

TFIDF

Palabra	TF-IDF
little	0.31
thou	0.23
thee	0.23
boston	0.22
vanity	0.10


JANE AUSTEN
PERSUASION
Jane Austen's Historical Classic romance novel (200th Anniversary Collection Edition, #1)

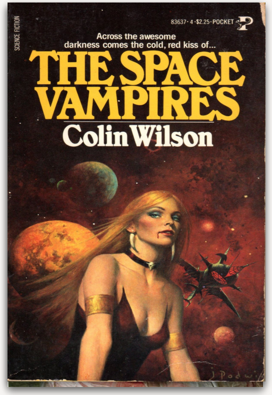

PRIDE and PREJUDICE by Jane Austen Illustrated by Hugh Thomson

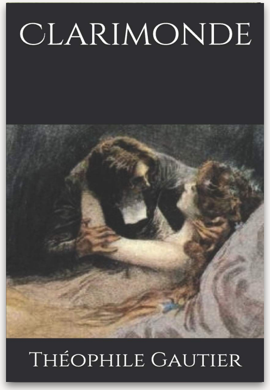
Vocabulario: 801 palabras

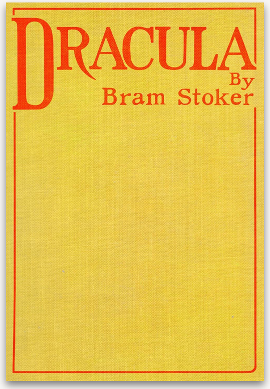
[**'lady'** 'laid' 'land' 'language'
 'languages' ' ... **'letters'** 'library' 'life'
'literary' 'literature' 'little' 'live' 'lived' 'lively'
... 'london' **'love'** 'mad'... **'man'** 'manner'
'married' 'morals' '**'servants'** 'poem' 'poet'
'poetry' 'pride' ''**principles'** 'read' 'reader'
'ridiculous' 'right' 'romantic' **'woman'**
'women' **'writer'** 'writing' 'writings'
'written' 'wrote' ...]

TFIDF

Palabra	TF-IDF
Ernest	0.804
boy	0.176
jack	0.174
nervous	0.063
shakespeare	0.061
souls	0.038
mysterious	0.038
vampyre	0.009







Vocabulario: 13946 palabras

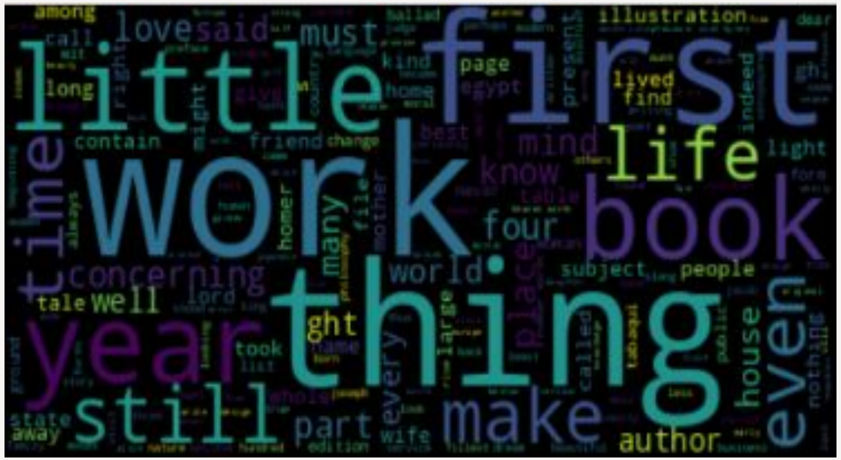
[**abominable** abominations **accident** accidental accidentally accidents **twilight** twinkle twinkled twisted tyranny tyrant ugliness **ugly** ultimate **vampires** vampirism vampyre vampyres violence **violent** violently wisdom wise wizard wizards ...]

WORD CLOUD

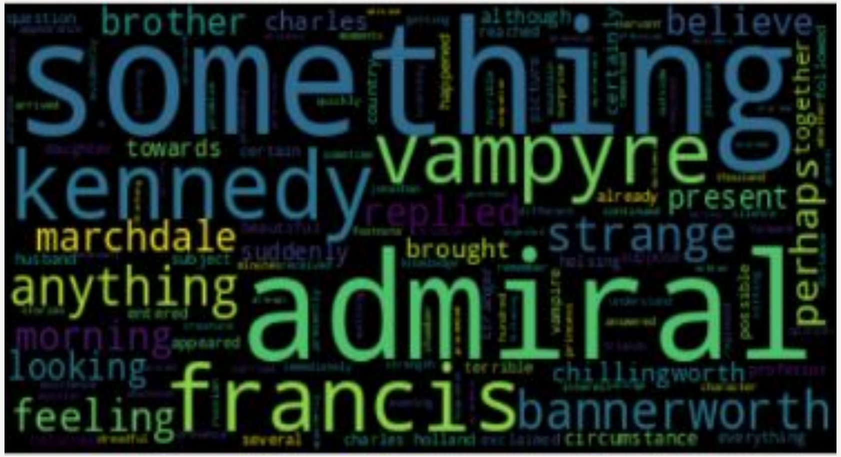
Limpieza:

- Shortwords
- Stopwords
- Generic words

POPULAR



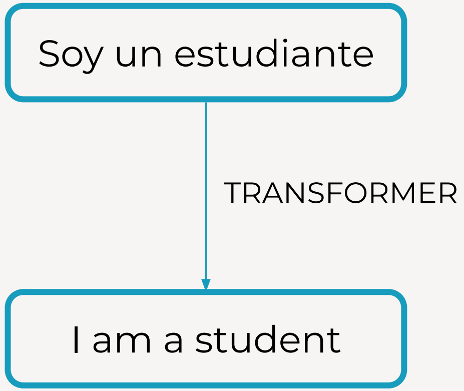
VAMPIRES



04

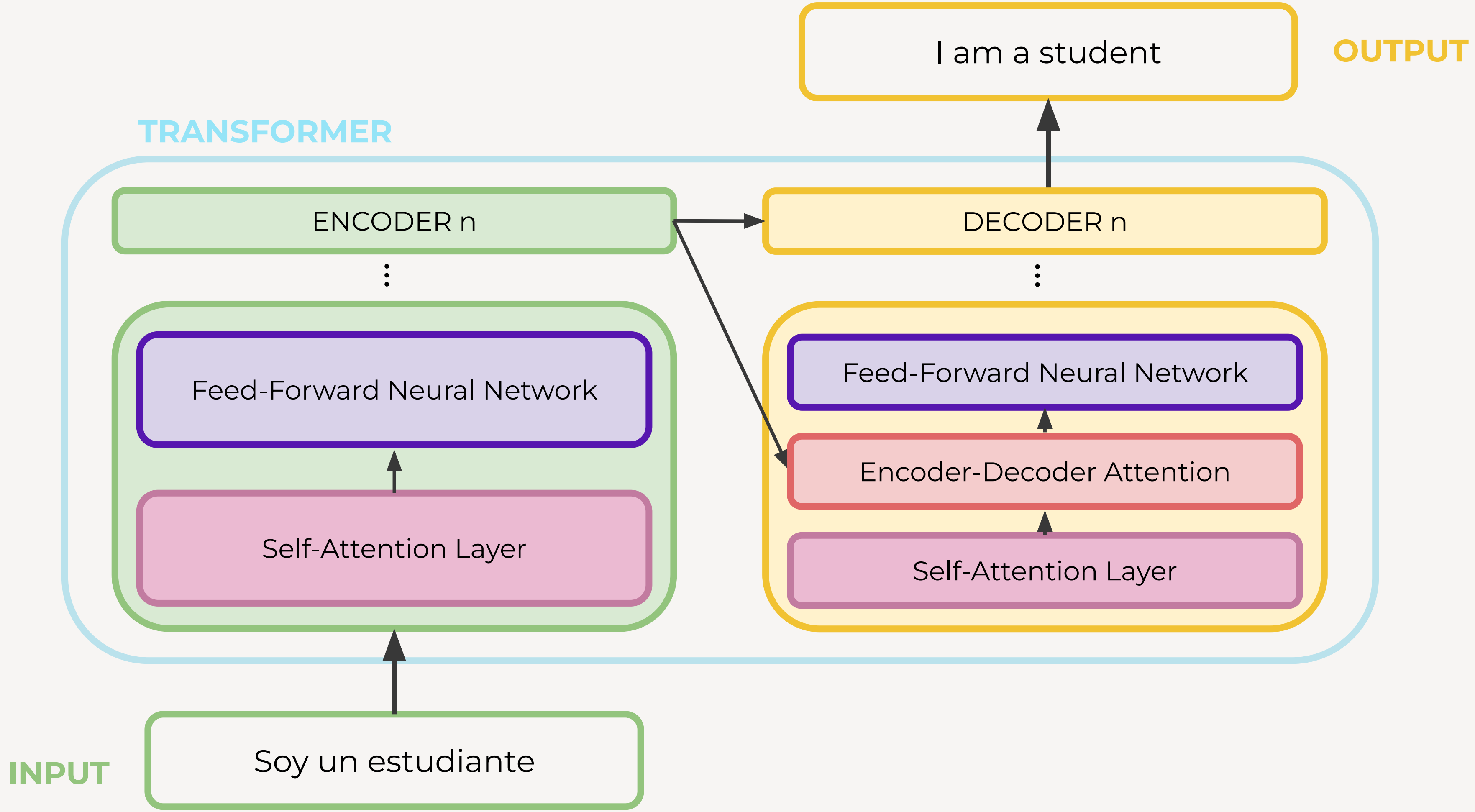
TRANSFORMERS

Breve Introducción:
función, arquitectura y aplicaciones



¿QUÉ ES UN TRANSFORMER?

Red neuronal utilizada para transformar una secuencia en otra basada en un mecanismo de **atención**.


I am a student
OUTPUT
TRANSFORMER
ENCODER n
DECODER n
Feed-Forward Neural Network
Self-Attention Layer
Feed-Forward Neural Network
Encoder-Decoder Attention
Self-Attention Layer
INPUT
Soy un estudiante

SELF-ATTENTION LAYER

Ayudar al Encoder a **mirar otras palabras** en la secuencia de input mientras analiza una palabra en particular.

En el Decoder solo puede mirar palabras previas en la secuencia de output.

INPUT

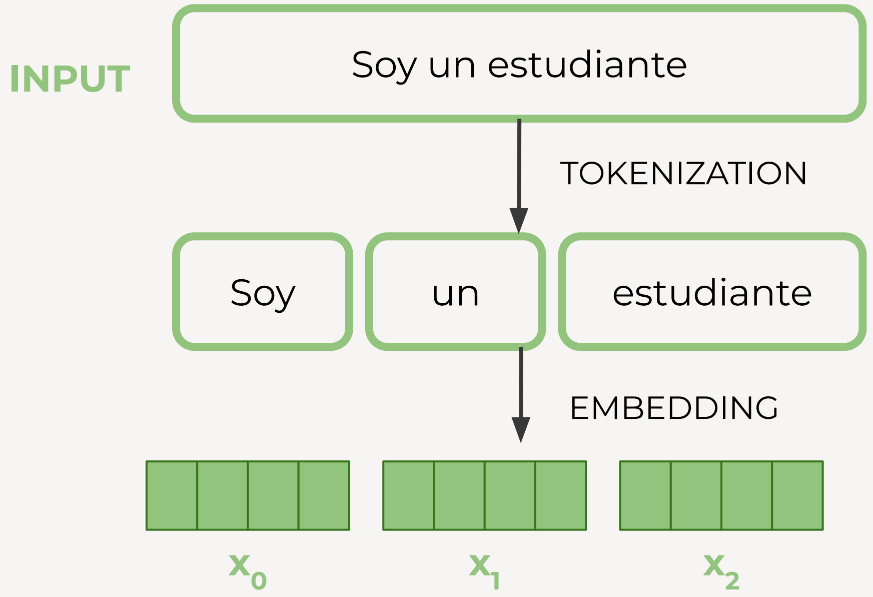
The animal didn’t cross the street because **it** was tired

¿La palabra *“it”* se refiere al animal o a la calle ?

EMBEDDING

- Cada token es representado en un vector de números reales de dimensión 512.
  A esto se lo llama ***embedding*** (algoritmos: *word2vec*)

- El primer encoder recibe como input el *embedding*.



TOKENS

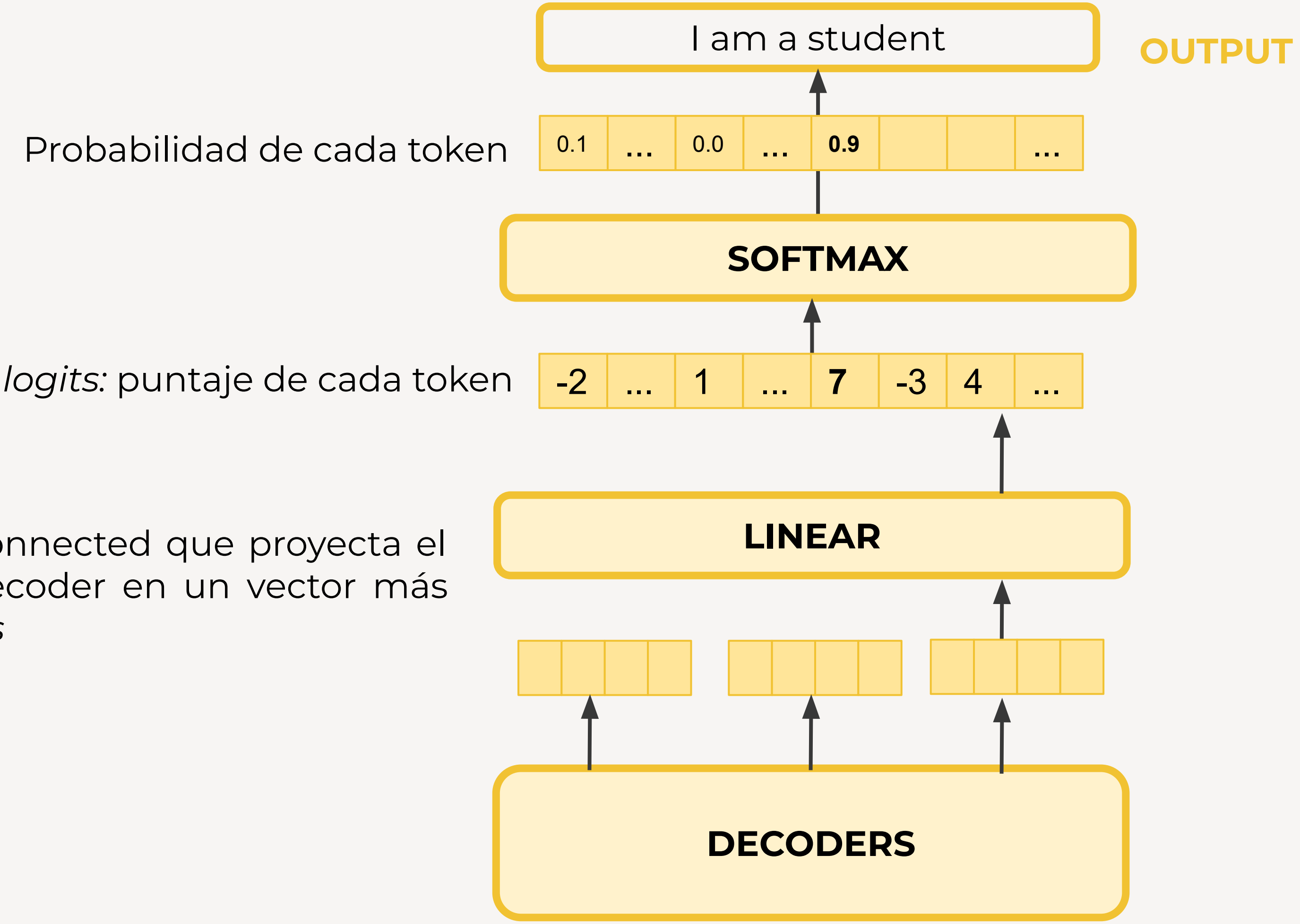
```
She had, however, one very intimate friend, a
sensible, deserving woman, who had been brought, by strong attachment
to herself, to settle close by her, in the village of Kellynch; and on
her kindness and advice, Lady Elliot mainly relied for the best help
and maintenance of the good principles an
```

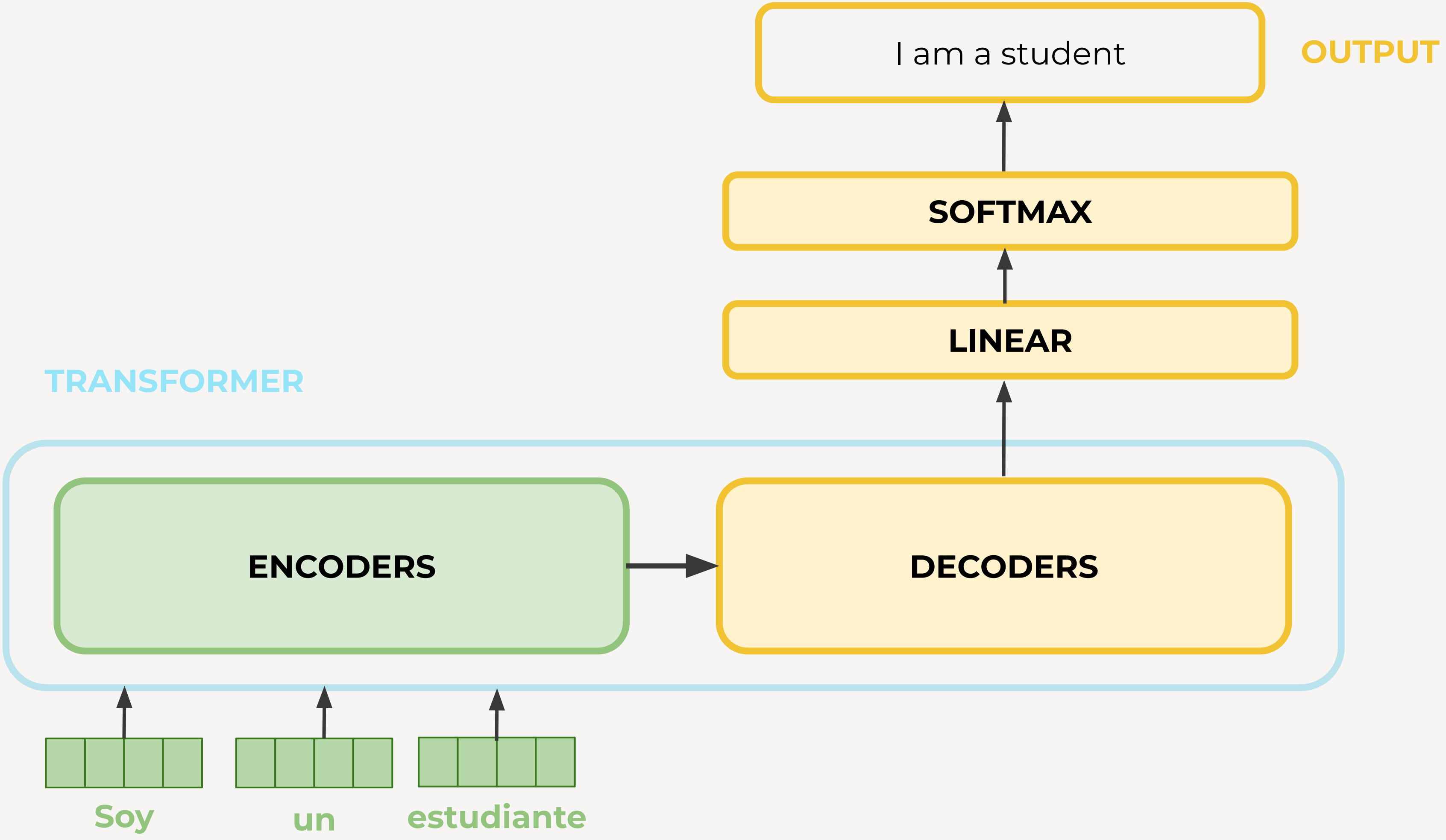
```
tokens = nltk.word_tokenize(text) # Tokenization


 [... 'Lady', 'Elliot', 'mainly', 'relied', 'for', 'the', 'best', 'help', 'and',
'maintenance', 'of', 'the', 'good', 'principles'... ]
```

WORD2VEC

```
model = Word2Vec(                    # Embeddings
   sentences=[lines],
   min_count=1,
   sg=1,
   window=7
)
word2vec(text=f.read(), key='lady')


[('to', 0.541), ('the', 0.529), ('her', 0.524), ('had', 0.516), … ]
# Remove stopwords
[('deserving', 0.286), ('borough', 0.274), ('the', 0.274), ('society', 0.272),
('dignity', 0.258), ('object', 0.240) … ]
```

LINEAR
Red neuronal fully-connected que proyecta el output del último decoder en un vector más grande llamado *logits*




I am a student
OUTPUT
SOFTMAX
LINEAR
TRANSFORMER
ENCODERS
DECODERS
Soy
un
estudiante

APLICACIONES

- Comprensión, generación y traducción de textos: OpenAI Language Model
- Predecir la siguiente palabra de una oración: IntelliSense in Visual Studio Code
- Juegos de Estrategia Real-Time: AlphaStar - StarCraft II
- Detección de Anomalías: Spacecraft Anomaly Detection
- Reconocimiento de Imágenes: Patches Are All You Need?

  An Image is Worth 16x16 Words
- Sentiment Analysis: GPT3 - OpeanAI

05

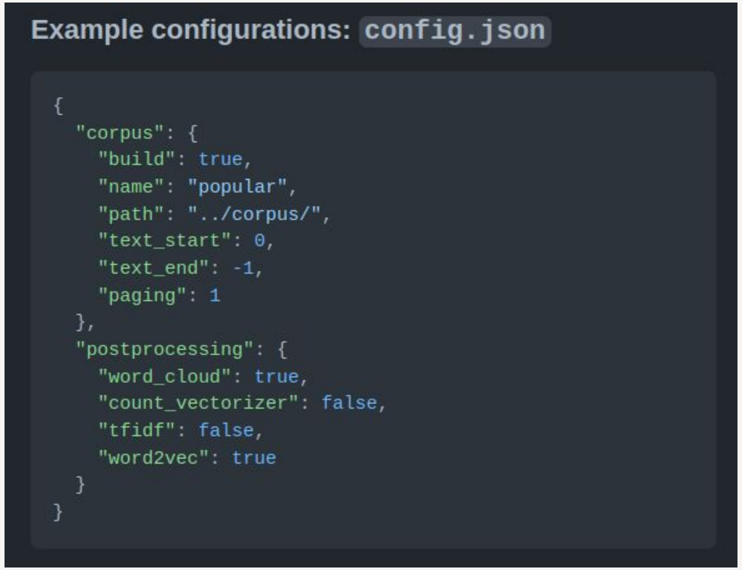
FINAL

Próximos pasos

EXTENSIÓN PARA EL FINAL

- Se puede **personalizar** el corpus y especificar la red entrenándola únicamente con una categoría de libros deseada.

- Una vez obtenidos los **resultados** del Transformer:
    - **Evaluación:**
      Accuracy (train: originales, test: generados)
      Clustering (textos "similares")
    - **Mejores resultados** :
      Variar la longitud de los textos del corpus
      Entrenar con más textos
      Hacer una limpieza de palabras


Example configurations: `config.json`

```
{
  "corpus": {
    "build": true,
    "name": "popular",
    "path": "../corpus/",
    "text_start": 0,
    "text_end": -1,
    "paging": 1
  },
  "postprocessing": {
    "word_cloud": true,
    "count_vectorizer": false,
    "tfidf": false,
    "word2vec": true
  }
}
```

Repositorio en Github

06

BIBLIOGRAFÍA

¿Dónde puedo profundizar?

BIBLIOGRAFÍA

Attention Is All You Need - Paper

GPT-2 - GPT-3 - BERT

Inside Machine Learning

Illustrated Transformer

Positional Encoding - Visualization

Word Embedding - Visualization

Simple Transformer in Python - Github

Interactive Transformer

¡ GRACIAS !