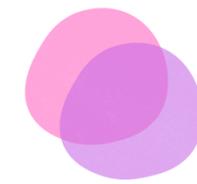




USANDO PYTHON Y NLP
PARA APRENDER
LENGUAJES



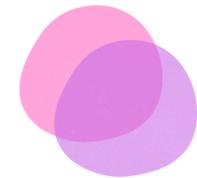
A HABLAR



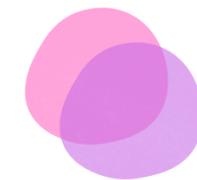
Historia



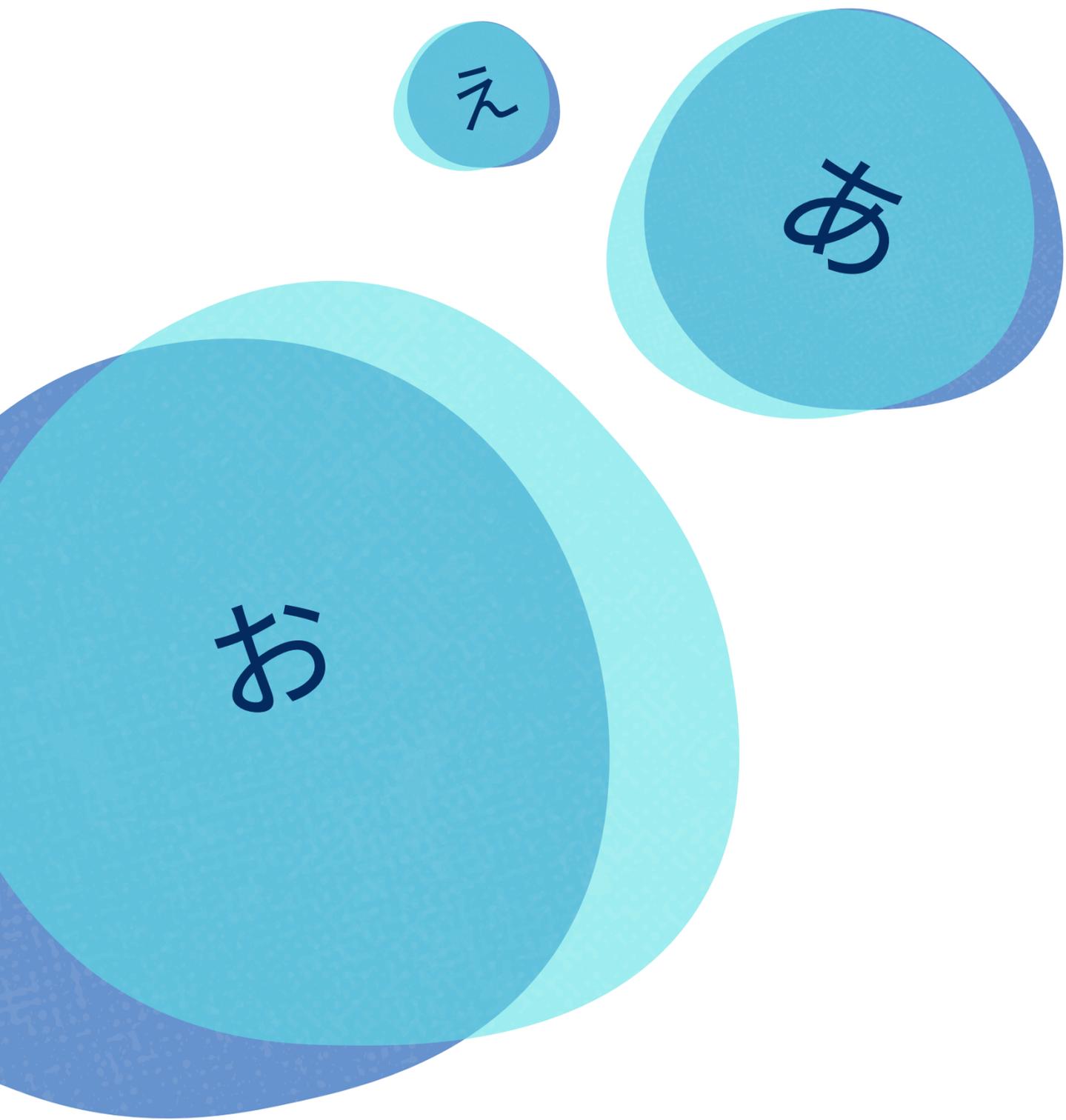
NLP y NLU



Spacy y Stanza



Proyecto



HISTORIA

Como un chiste llega
demasiado lejos

POR QUE
JAPONES
Simple, sonaba
cool. (mala
decisión)

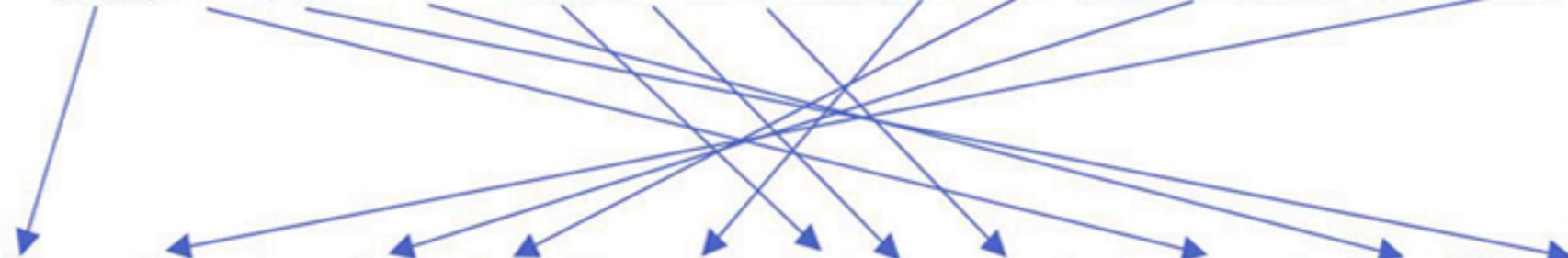


Empezar con
duolingo



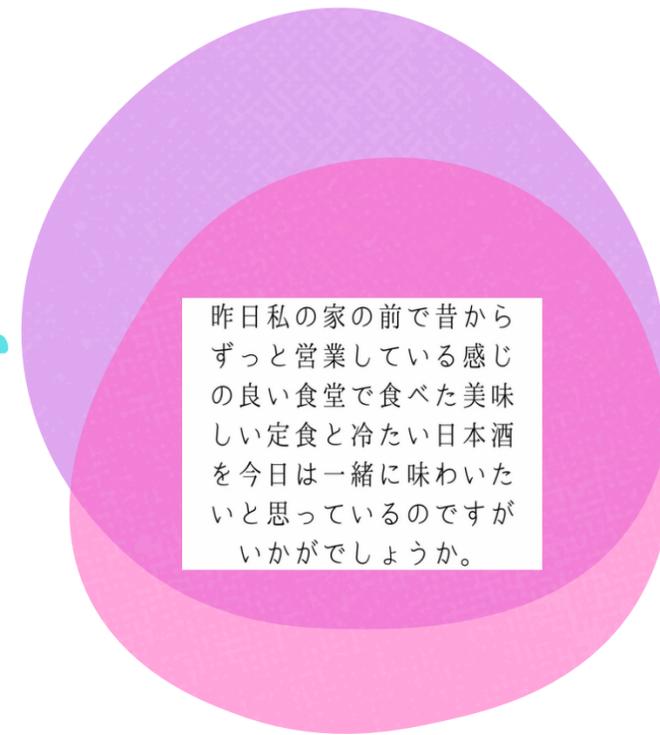
Japanese: (私は) 先月日本の本社で佐藤さんが出席した会議について、何の報告も受けていません。

English: I haven't heard anything about the meeting Sato-san attended last month at the HQ in Japan.





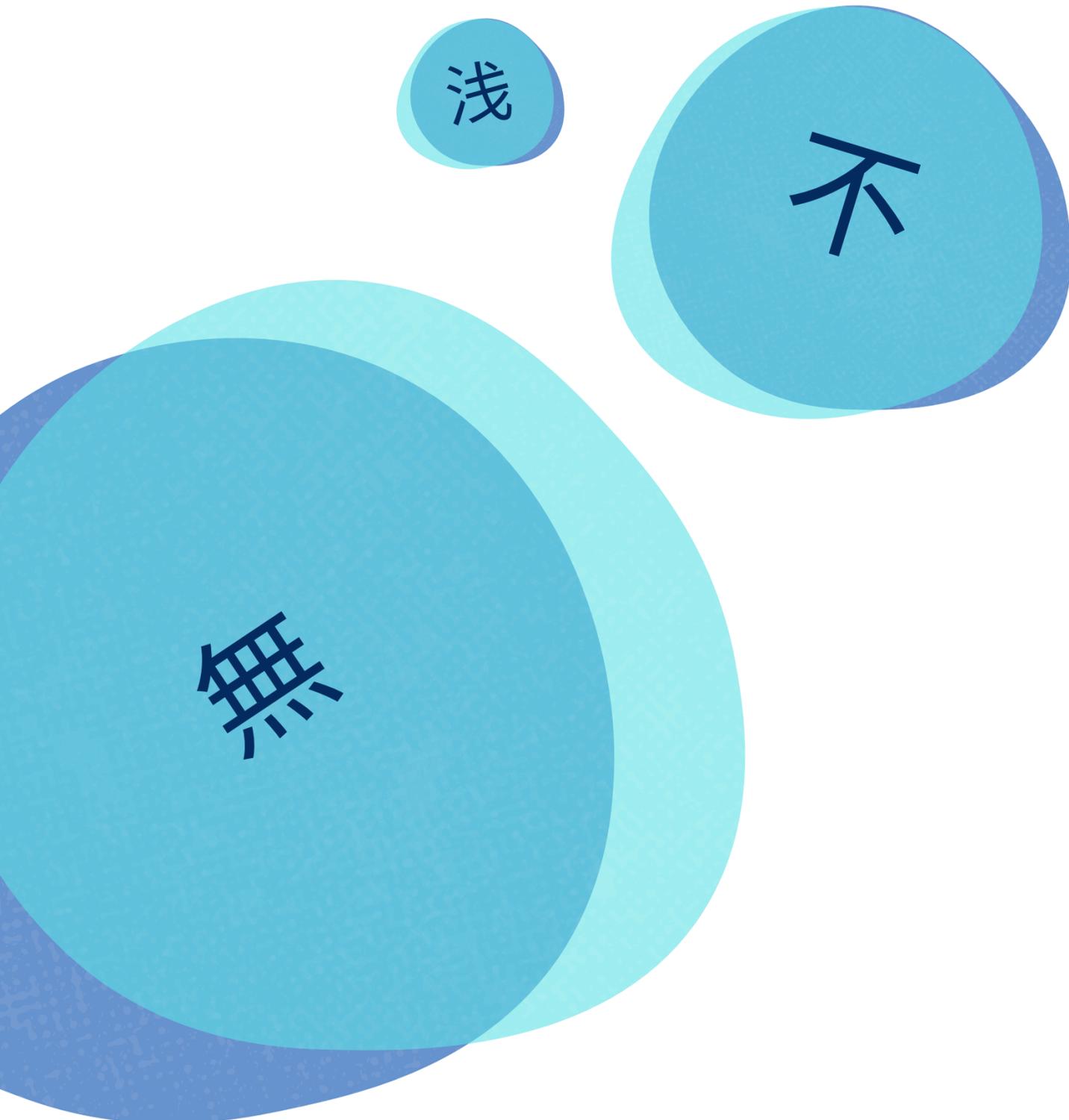
Irse a Japón a estudiar



Sufrir con los sistemas de escritura



Pasar horas estudiando para aun asi sufrir leyendo



浅

不

無

NLP Y NLU

Los fundamentos.

NLP

El campo de la AI
dedicado a procesar,
generar y analizar
lenguaje humano.



NLU

La rama de NLP que se enfoca en entender el lenguaje natural, no solo procesar.



UPOS

Etiquetas universales
para las categorías
gramaticales de las
palabras



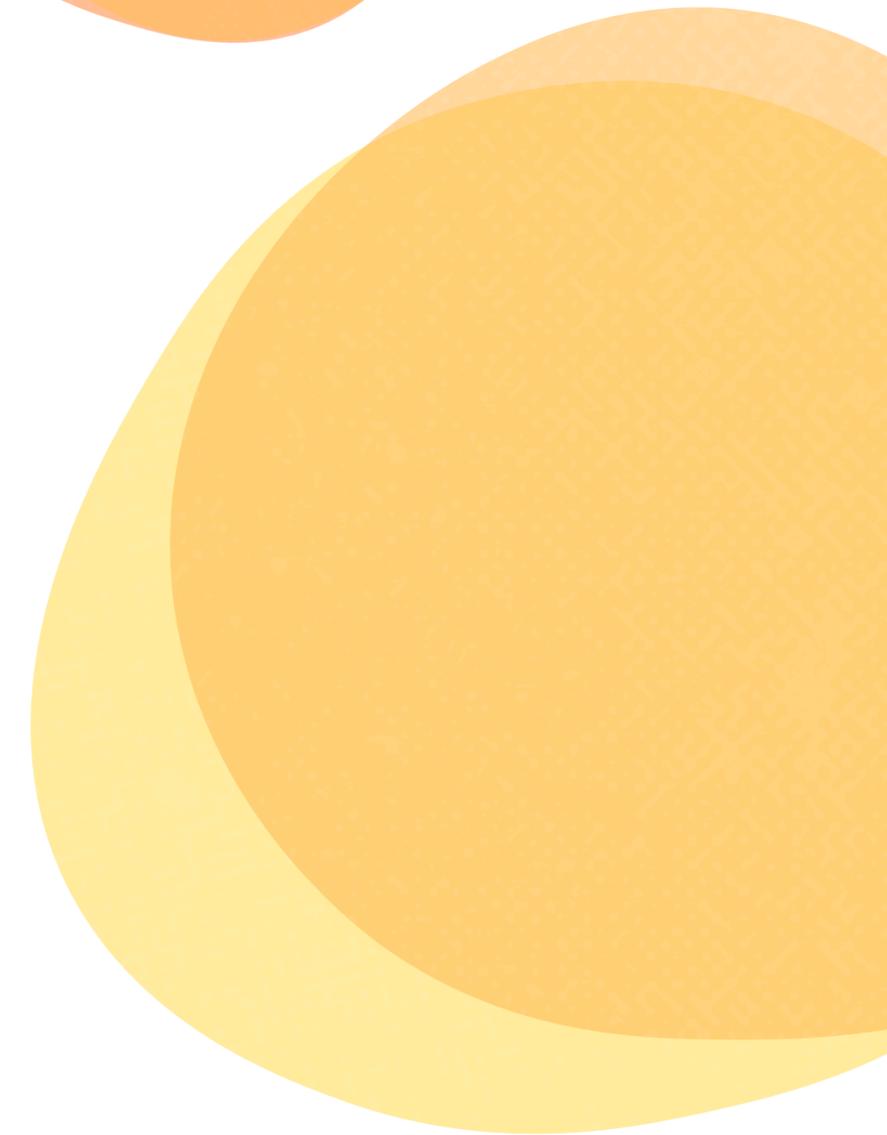
LEMMA

La forma base de una
palabra.



TAG

Elementos del habla
específicos del
lenguaje



DEPENDENCIA

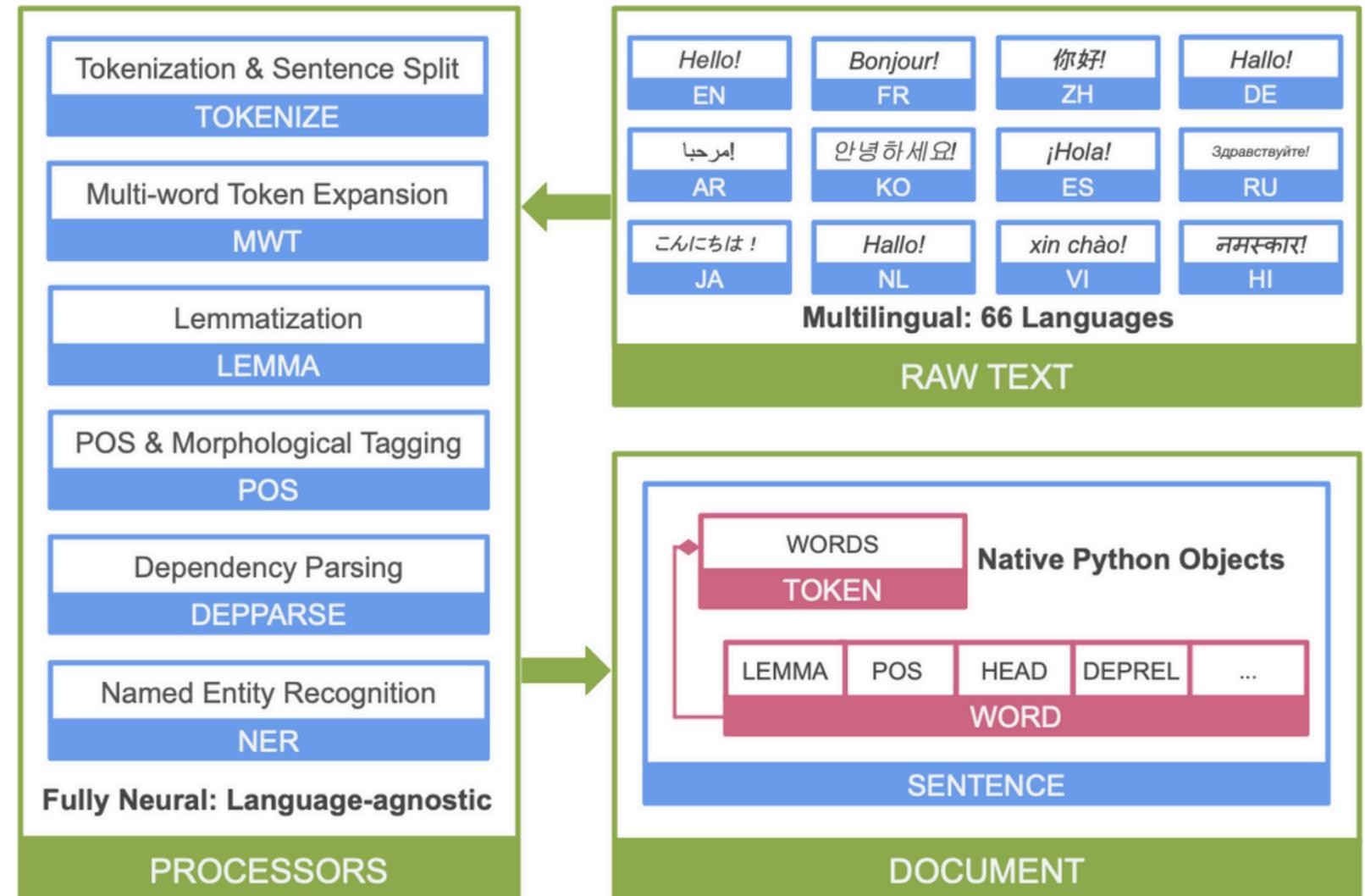
Relacion entre los diferentes tokens, es decir la cabeza del token

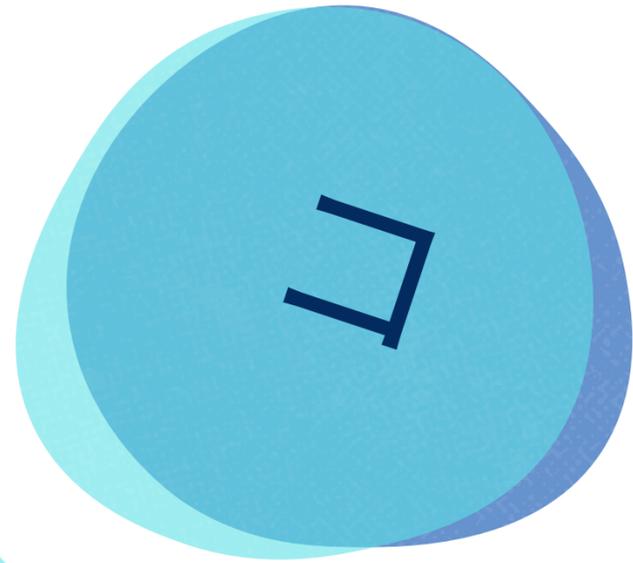


ENT_TYPE
Tipo de entidades,
como nombres de
ciudades, etc.



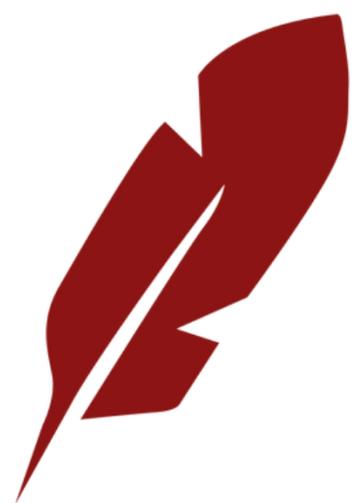
Una Pipeline tipica de NLP





SPACY Y STANZA

La magia en días
aburridos



Stanza



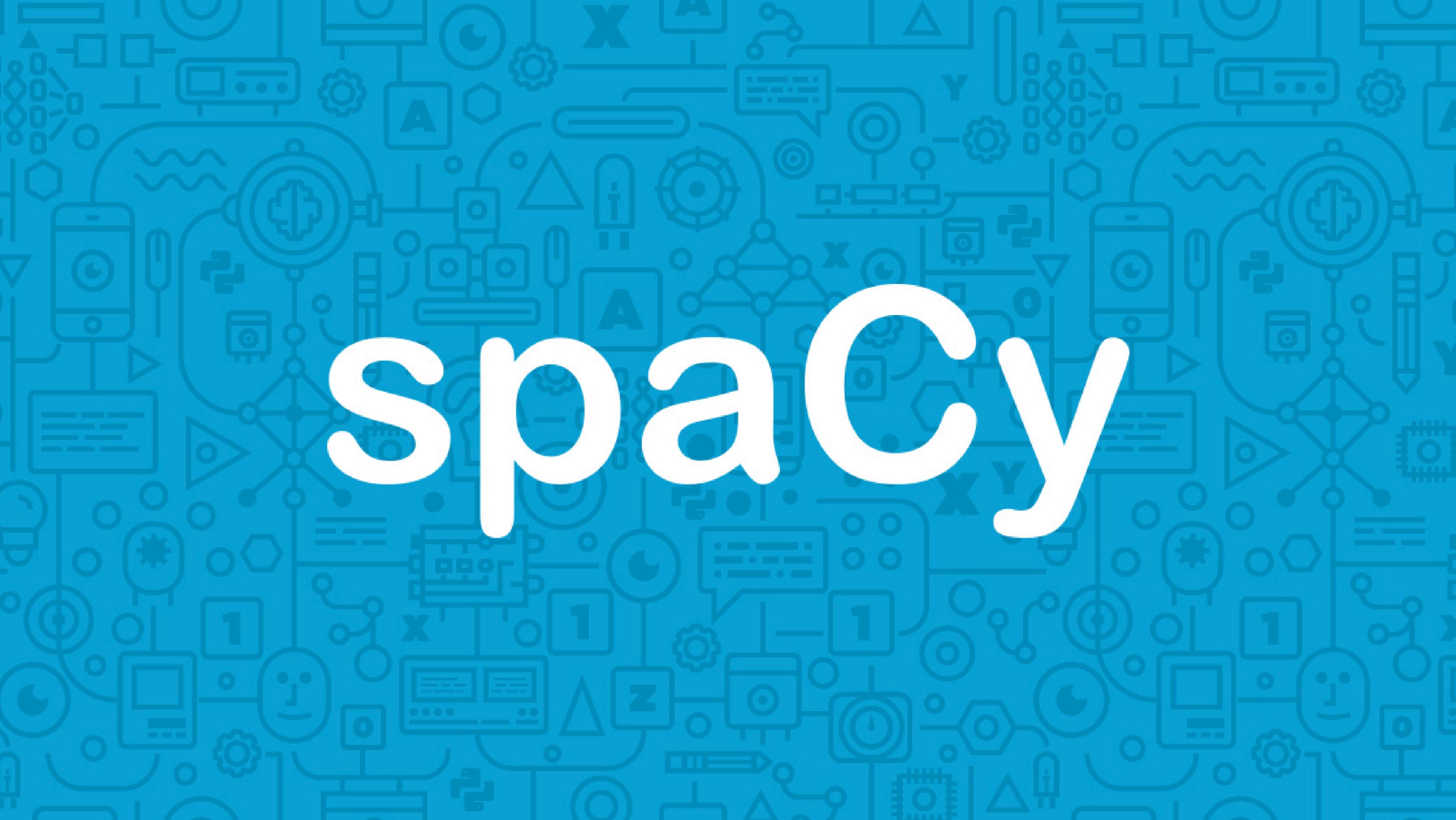
¿QUE ES?

Un modulo de python
para analisis del
lenguaje natural.

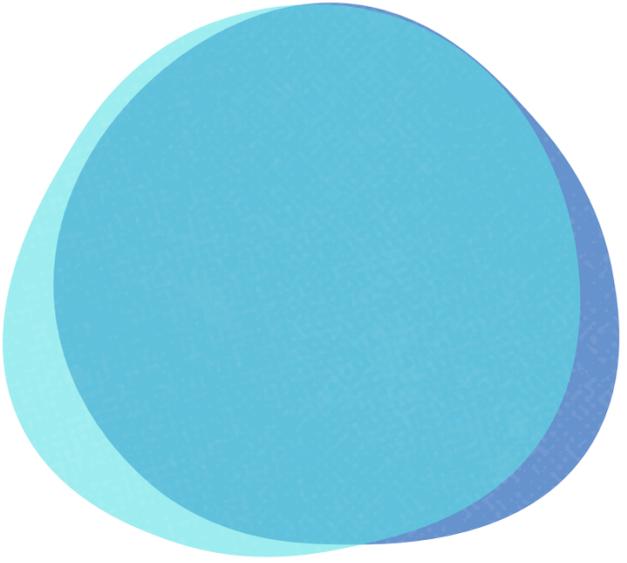
DEMO

stanza



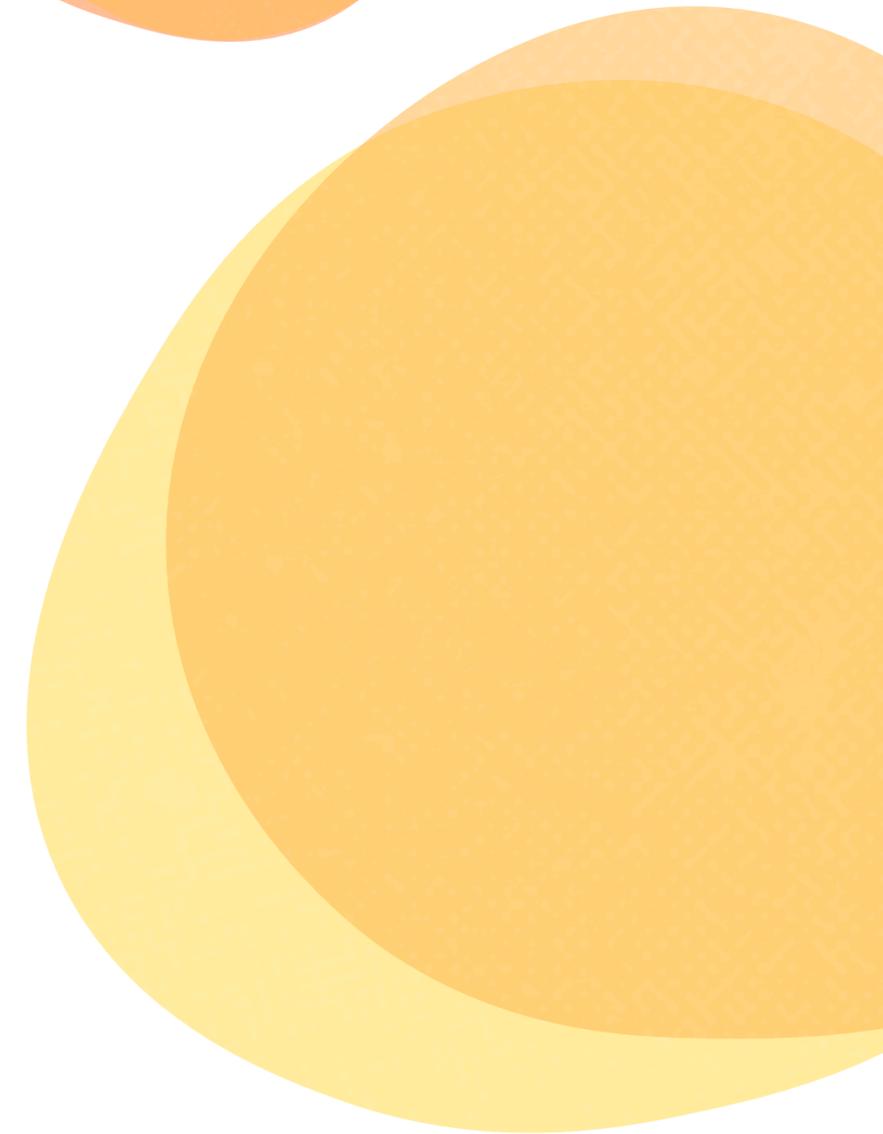
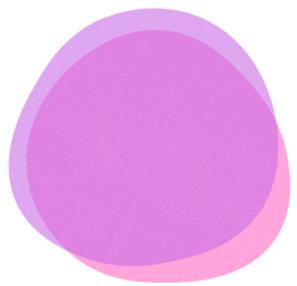
The background is a solid blue color with a repeating pattern of white line-art icons. These icons include various symbols such as gears, lightbulbs, speech bubbles, circuit boards, and abstract shapes, all connected by thin lines, suggesting a network or a complex system. The overall aesthetic is clean and modern, typical of a tech or science-themed graphic.

spacy



POR QUE

Libreria para NLP
avanzado que se
puede usar en
producción.



The slide features several decorative circles. On the left, there is a large blue circle and a smaller purple circle. On the right, there are three orange circles of varying sizes, with the largest one at the bottom right. The text is centered on the slide.

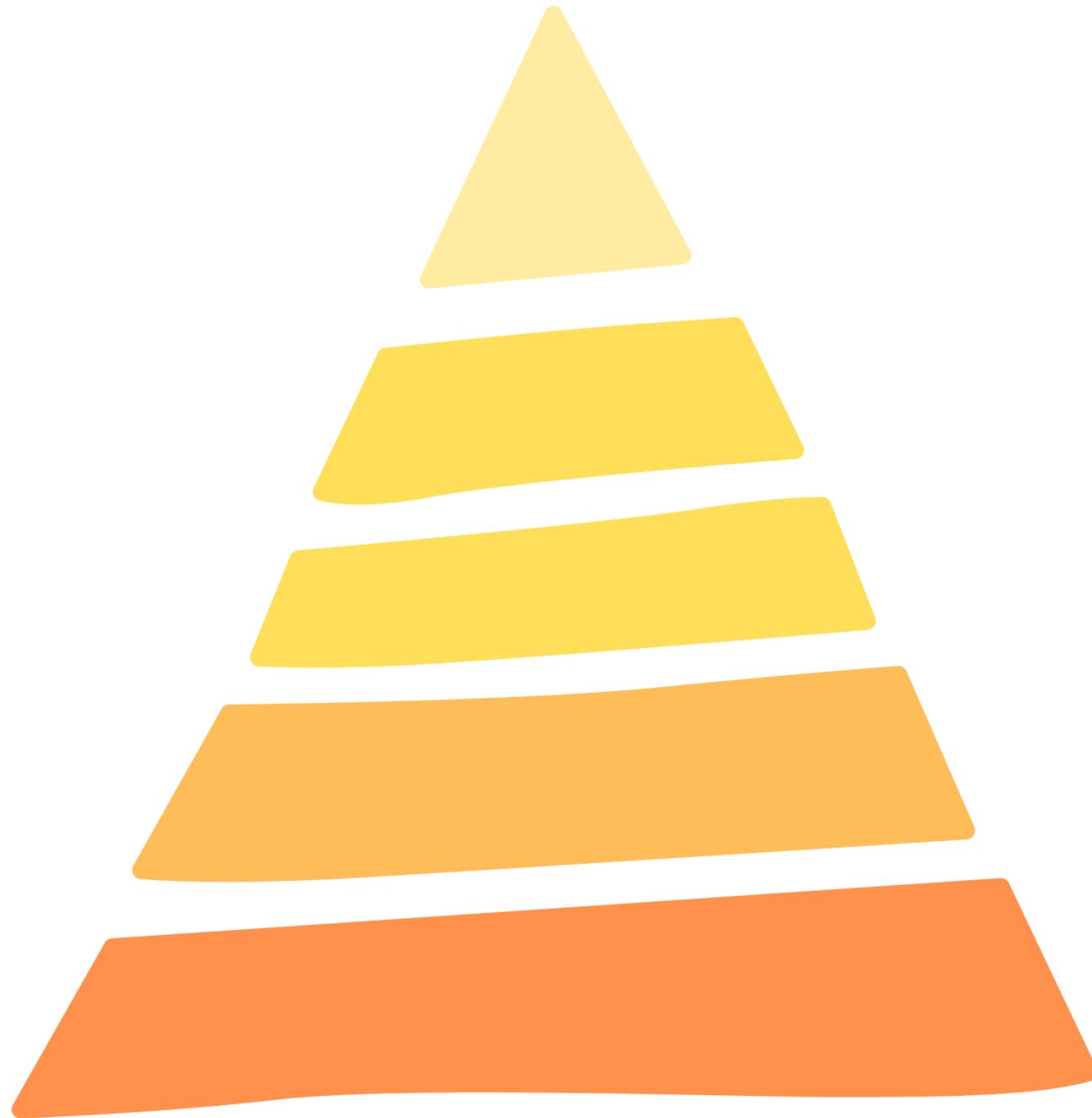
QUE ESTA COOL

Con un código bastante simple se puede aplicar todo lo que hemos visto.



FEATURES

- Amplio soporte de lenguajes
- Pipelines de NLP pre-entrendas
- Buena velocidad
- Soporte para modelos propios en TF y PyTorch



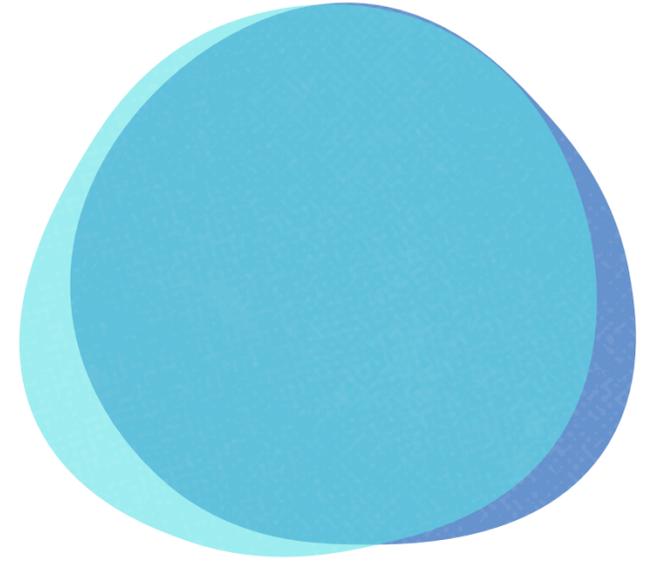
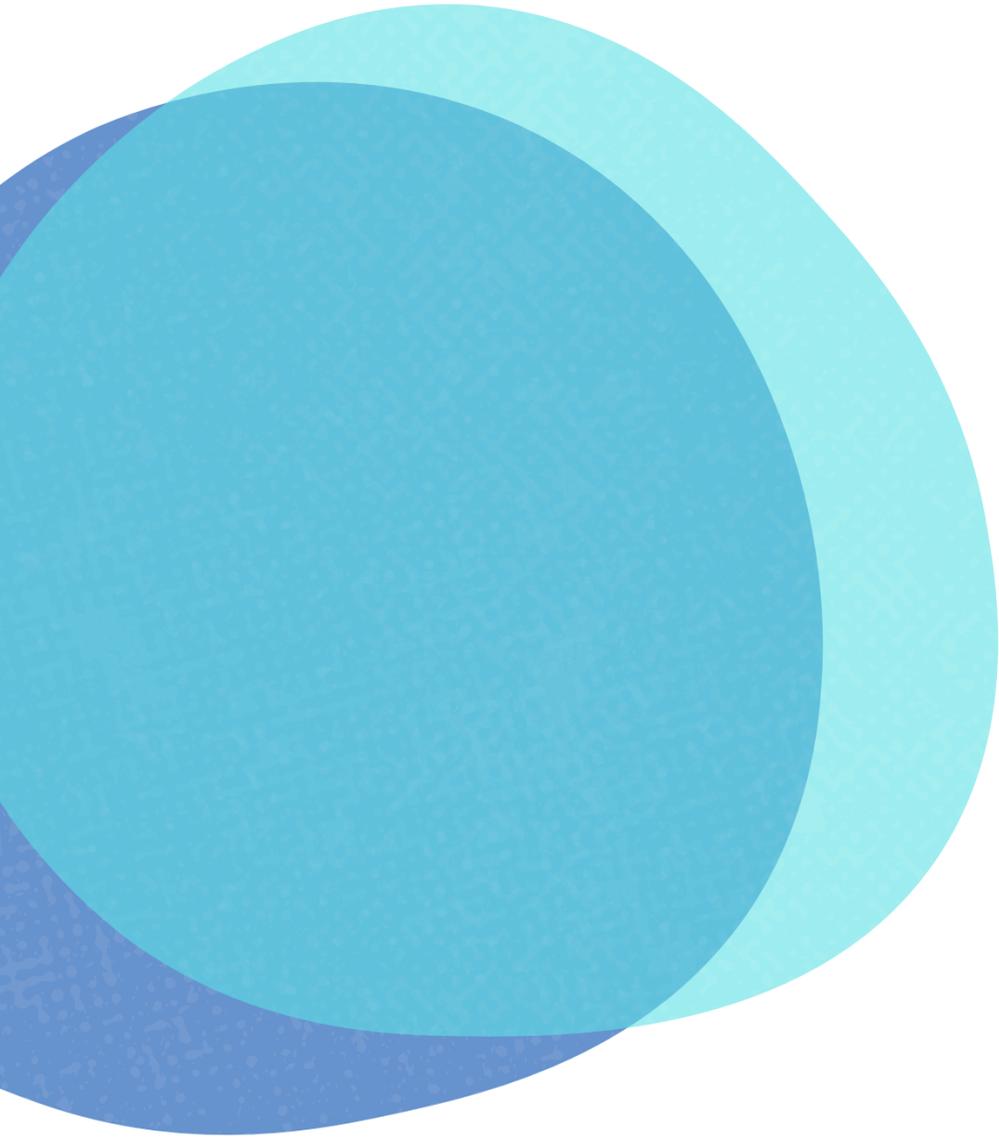
VENTAJAS

- Facil de usar como principiante
- Bueno para construir una aplicación entera.
- Eficiente en GPU y CPU
- Probar diferentes NN facilmente



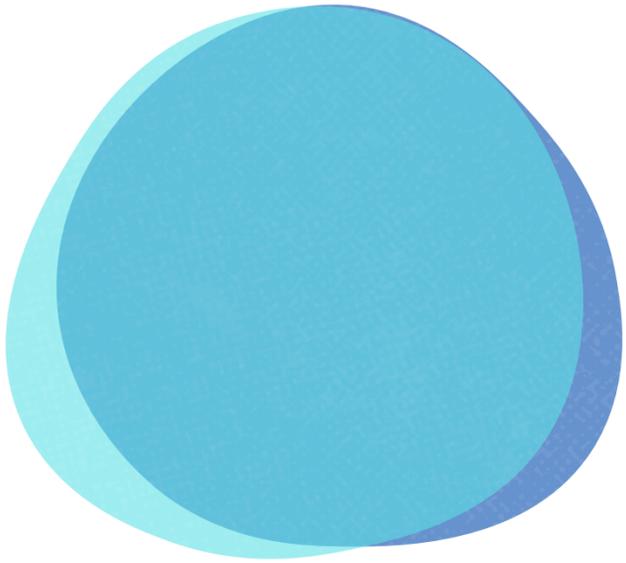
DESVENTAJAS

- No tiene funciones generativas
- No esta orientada a hacer research



CODE SAMPLE





```
8 class NLPService: 2 usages  Miguel Necochea +1*
9 def __init__(self, spacy_model: str = "ja_ginza_bert_large"):  Miguel Necochea *
10     spacy.prefer_gpu()
11     self.nlp = spacy.load(spacy_model)
12     self.fallback = kakasi()
13     self.japanese_detector = JapaneseDetector()
14
15 def _convert_reading(self, surface: str) -> str:  Miguel Necochea
16     fb = self.fallback.convert(surface)
17     return ''.join(item["kana"] for item in fb) if fb else None
18
19 def _token_to_dict(self, token) -> Dict: 1 usage  Miguel Necochea *
20     surface = token.text
21     reading = token.morph.get("Reading")
22
23     if reading:
24         reading = reading[0]
25
26     is_japanese = self.japanese_detector.is_japanese(surface)
27
28     return {
29         "surface": surface,
30         "reading": reading,
31         "lemma": token.lemma_,
32         "pos": token.pos_,
33         "tag": token.tag_,
34         "dep": token.dep_,
35         "head": token.head.text,
36         "morph": token.morph.to_dict(),
37         "offset": token.idx,
38         "ent_iob": token.ent_iob_,
39         "ent_type": token.ent_type_,
40         "is_japanese": is_japanese
41     }
42
43 def tokenize_batch(self, texts: List[str]) -> List[List[Dict]]: 1 usage  Miguel Necochea +1
44     docs = self.nlp.pipe(texts, batch_size=max(1, len(texts)))
45     return [
46         [self._token_to_dict(tok) for tok in doc]
47         for doc in docs
48     ]
49
```





陽

變

何

PROYECTO

¿Bueno y todo esto pa'
que?



Pensemos en Syntax Highlighting

```

class NLPService:
    def __init__(self, spacy_model: str = "ja_ginza_bert_large"):
        spacy.prefer_gpu()
        self.nlp = spacy.load(spacy_model)
        self.fallback = kakasi()
        self.japanese_detector = JapaneseDetector()

    def _convert_reading(self, surface: str) -> str:
        fb = self.fallback.convert(surface)
        return ''.join(item["kana"] for item in fb) if fb else None

    def _token_to_dict(self, token) -> Dict:
        surface = token.text
        reading = token.morph.get("Reading")

        if reading:
            reading = reading[0]

        is_japanese = self.japanese_detector.is_japanese(surface)

        return {
            "surface": surface,
            "reading": reading,
            "lemma": token.lemma_,
            "pos": token.pos_,
            "tag": token.tag_,
            "dep": token.dep_,
            "head": token.head.text,
            "morph": token.morph.to_dict(),
            "offset": token.idx,
            "ent_iob": token.ent_iob_,
            "ent_type": token.ent_type_,
            "is_japanese": is_japanese
        }

    def tokenize_batch(self, texts: List[str]) -> List[List[Dict]]:
        docs = self.nlp.pipe(texts, batch_size=max(1, len(texts)))
        return [
            [self._token_to_dict(tok) for tok in doc]
            for doc in docs
        ]

```

```

8 class NLPService: 2 usages Miguel Necochea +1*
9     def __init__(self, spacy_model: str = "ja_ginza_bert_large"): Miguel Necochea
10         spacy.prefer_gpu()
11         self.nlp = spacy.load(spacy_model)
12         self.fallback = kakasi()
13         self.japanese_detector = JapaneseDetector()
14
15     def _convert_reading(self, surface: str) -> str: Miguel Necochea
16         fb = self.fallback.convert(surface)
17         return ''.join(item["kana"] for item in fb) if fb else None
18
19     def _token_to_dict(self, token) -> Dict: 1 usage Miguel Necochea *
20         surface = token.text
21         reading = token.morph.get("Reading")
22
23         if reading:
24             reading = reading[0]
25
26         is_japanese = self.japanese_detector.is_japanese(surface)
27
28         return {
29             "surface": surface,
30             "reading": reading,
31             "lemma": token.lemma_,
32             "pos": token.pos_,
33             "tag": token.tag_,
34             "dep": token.dep_,
35             "head": token.head.text,
36             "morph": token.morph.to_dict(),
37             "offset": token.idx,
38             "ent_iob": token.ent_iob_,
39             "ent_type": token.ent_type_,
40             "is_japanese": is_japanese
41         }
42
43     def tokenize_batch(self, texts: List[str]) -> List[List[Dict]]: 1
44         docs = self.nlp.pipe(texts, batch_size=max(1, len(texts)))
45         return [
46             [self._token_to_dict(tok) for tok in doc]
47             for doc in docs
48         ]

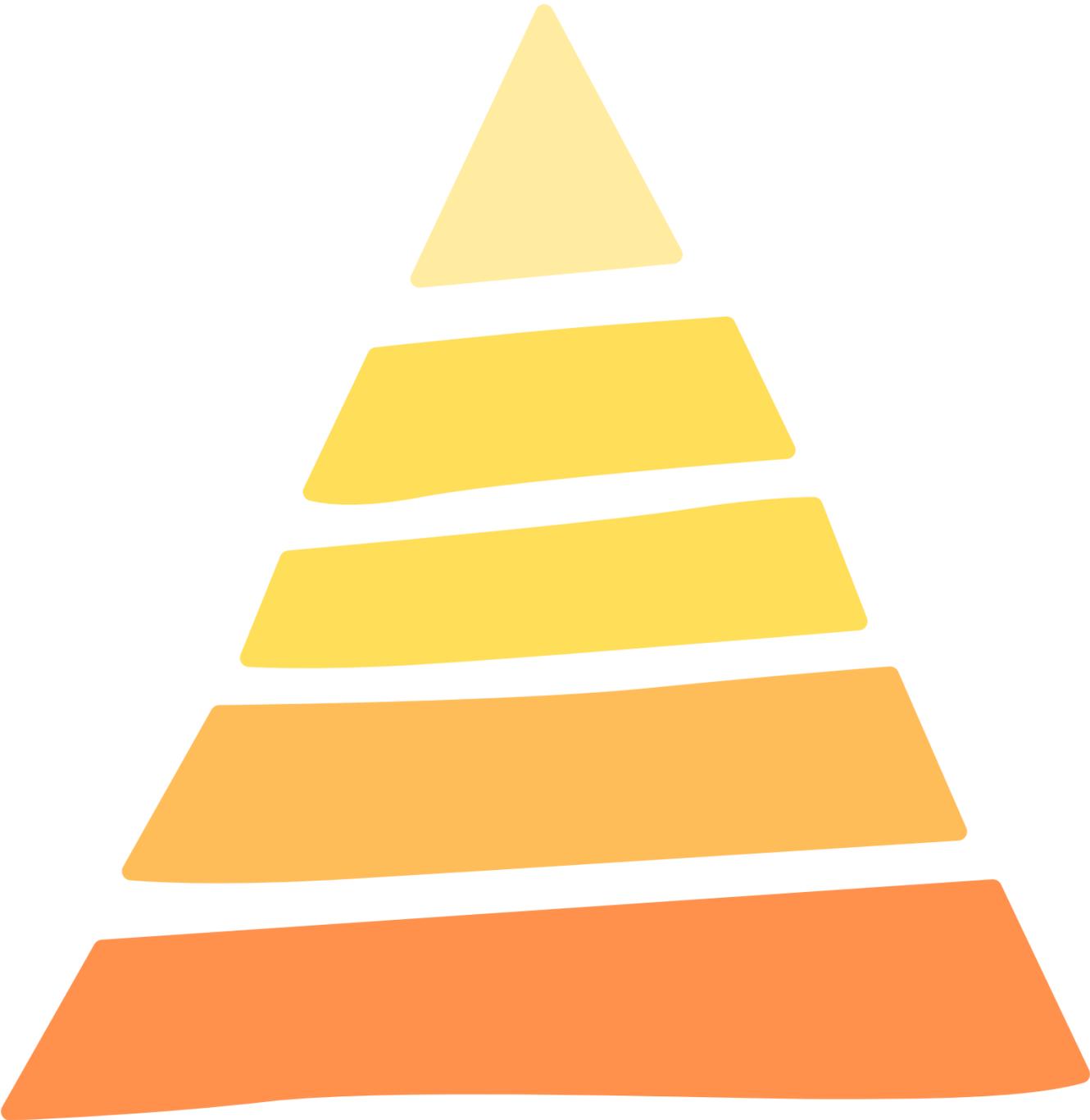
```



El color lo hace
mas fácil (casi
siempre)

Y asi nacio Complexa





¿QUE ES?

Complexa es una extension de Chrome que hace un análisis morfológico del texto en japones de paginas web para ayudar en el aprendizaje del japonés.

その分厚く肥ったルールブックはお前自前の妄想

その恩着せがましいお説教は鏡に向かってどうぞ

その無様に狂ったルールブックをすぐに他人に強要

その矛盾だらけの不純な行動基準 手放せ

その^{ぶあつ}分厚く^{ふと}肥ったルールブックはお^{まえ}前^{まえ}自前の^{もうそう}妄想

その^{おんき}恩着せがましいお説^{せっきょう}教は鏡^{かがみ}に向^むかってどうぞ

その^{ぶざま}無様に^{くる}狂ったルールブックを^{たにん}すぐに他人に^{きょうよう}強要

その^{むじゆん}矛盾だらけの^{ふじゆん}不純な^{こうどうきじゆん}行動基準^{てばな}手放せ



Y la información
de cada token

Surface	不純
Reading	ふじゅん
Lemma	不純
Part of Speech	ADJ
Tag	名詞-普通名詞-形状詞可能
Dependency	acl
Head	行動基準
Offset	8
IOB	Data not found.
Entity	Data not found.
Morph Features	Data not found.

Exclude word

Look meaning

その矛盾だらけの不純な行動基準見直せ

Surface	矛盾
Reading	むじゅん
Lemma	矛盾
Part of Speech	NOUN
Tag	名詞-普通名詞-サ変可能
Dependency	nmod
Head	行動基準
Offset	2
IOB	Data not found.
Entity	Data not found.
Morph Features	Data not found.

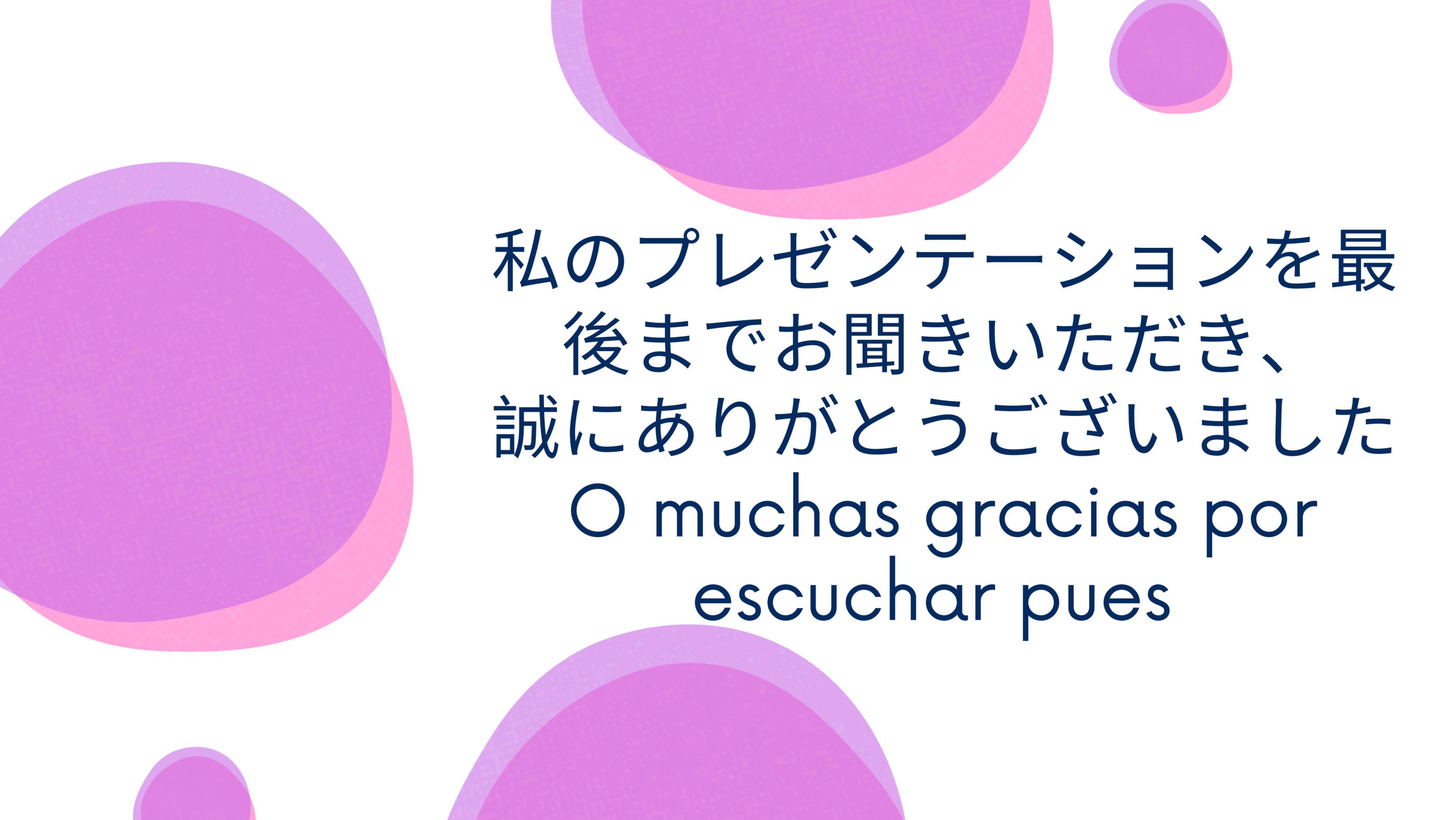
Exclude word

Look meaning

その矛盾だらけの不純な行



Live Demo!



私のプレゼンテーションを最
後までお聞きいただき、
誠にありがとうございました

○ muchas gracias por
escuchar pues



Q&A