



# Preparación de la Información

## Información en la web

**CYBER WOW** Hasta **60%** **¡ÚLTIMAS HORAS!** Quedan: **00 07 26 21**

Plaza Vea Real Plaza Oechsle Vivanda Inkafarma Mifarma Club Pro Nuestro Blog Venta empresa

**PROMART** HOME CENTER ¿Qué estás buscando? Ubicación Chiclayo, Chiclayo Hola, inicia sesión! Mi cuenta y pedidos 0

Todas las categorías CYBER WOW ¡ÚLTIMAS UNIDADES! ¡CUPONAZOS! TODO PARA EL FRÍO -60%! ¡RENOVEA TU CELU! LIQUIDACIÓN

Promart > Muebles > Dormitorio > Roperos

**Ropero Siena 6 Puertas 2 Cajones MDP Amarula Doripel**  
DORIPEL | SKU: 156444 Compartir

**OFERTA CYBER WOW** Quedan: **07 : 27 : 20**

Tarjeta oh! **S/ 299** -33%  
Precio online **S/ 319** -28%  
Precio lista **S/ 449**

Calcula tus cuotas con Tarjeta oh!

¡Arma tu combo y ahorra! Ver Promo

1 **Agregar**

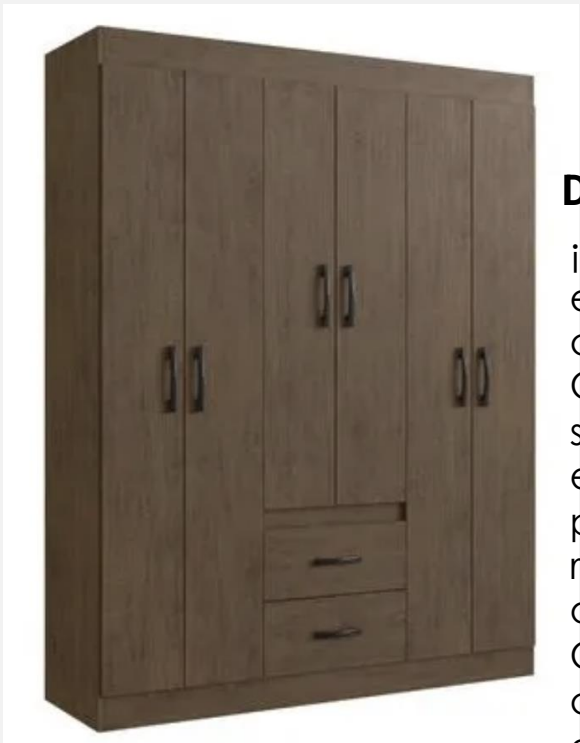
**Métodos de entrega**

<b>Despacho programado</b> Desde el 18 de julio a partir de S/25 <a href="#">Calcular &gt;</a>	<b>Despacho express</b> No disponible <a href="#">Ver distritos &gt;</a>	<b>Retiro en tienda y otros puntos</b> Desde el 18 de julio Gratis <a href="#">Ver tiendas &gt;</a>
--	--	---



# Datos

## SKU: 156444 - ROPERO SIENA 6 PUERTAS 2 CAJONES AMARULA



### DESCRIPCION EMOCIONAL

¡Organiza tu espacio con estilo y practicidad gracias al Ropero Siena 6 Puertas 2 Cajones MDP Amarula! Con su diseño espacioso y elegante, es la solución perfecta para mantener tu ropa y accesorios ordenados y al alcance. Con seis puertas y dos amplios cajones, tendrás espacio de sobra para guardar todo lo que necesitas...

A-Z DES_DIV	A-Z COD_AREA	A-Z DES_AREA	A-Z COD_DPTO	A-Z DES_DPTO	A-Z COD_LIN	A-Z DES_LIN	A-Z COD_STL
HOGAR Y DECO	A13	MUEBLES	D030	DORMITORIOS	L0126	CLOSETS RTA	156444

### FICHA TECNICA

Garantía	6 meses	Material	MDP
Altura Del Producto	183.00 cm	Número de espejos	0
Ancho Del Producto	156.00 cm	Cuenta con espejo	No
Profundidad Del Producto	45.00 cm	Marca	Doripel
Color principal	Marrón	Número de Cajones	2
Ancho (Rango)	140 - 159 cm	Número de puertas	6
Material predominante de empaque primario	Cartón	Peso Del Producto	68.00 kg
Peso en gramos del empaque primario	1000 g	Tipo de puerta	Batiente
Ultimo stock	Sí	Número de repisas	2
Modelo	Amarula	Número de piezas	49
Tipo de Producto	Ropero		
¿Dónde usarlo?	Hogar		
Incluye	Rieles metalicos		
Observaciones adicionales	Las fotografías y descripción del producto son referenciales.		
Recomendaciones De Uso	Para habitación		



# Datos

Descripcion.csv  
Ficha.csv  
Imágenes.csv  
Precios.csv  
Producto.csv

A1	A	B	C	D	E	F	G	H	I	J
1	PRD_LVL_CHILD	PRD_LVL_NUMBER	PRD_FULL_NAME	PRD_LVL_ID	COD_DIV	DES_DIV	COD_AREA	DES_AREA	COD_DPTO	DES_DPTO
2	12635	10236	OVALIN CIRCULAR P/SOBREP ORANGE VIDRIOTRASPARENTE	1	T03	ACABADOS	A11	BAVëOS	D083	SANITARIOS
3	12636	10237	OVALIN CIRCULAR P/SOBREP ORANGE VIDRIO PAVONADO	1	T03	ACABADOS	A11	BAVëOS	D083	SANITARIOS
4	12694	10298	LLAVE DUCHA ORANGE FROST S/SALIDA	1	T03	ACABADOS	A11	BAVëOS	D082	GRIFERIA DE BAVëO
5	12476	10077	SILLA ESTAMBUL FIJA H-602F NEGRA	1	T04	HOGAR Y DECO	A13	MUEBLES	D031	OFICINAS
6	12575	10177	MESA DE ESTUDIO ESPAVëA CASTAVëO	1	T04	HOGAR Y DECO	A13	MUEBLES	D031	OFICINAS
7	12622	10223	PAPELERA ORANGE 3LT C/PEDAL ACE/INOX CR	1	T03	ACABADOS	A11	BAVëOS	D085	ORGANIZACION DE BAVëO
8	12623	10224	PAPELERA ORANGE 5LT C/PEDAL ACE/INOX CR	1	T03	ACABADOS	A11	BAVëOS	D085	ORGANIZACION DE BAVëO
9	12625	10226	REPISA BAVëO ORANGE MULTIFUNCIONAL CR	1	T03	ACABADOS	A11	BAVëOS	D085	ORGANIZACION DE BAVëO
10	12806	10410	BATERIA 7LBI / 7PLC.12V. CAPSA	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
11	12807	10411	BATERIA 5U1R / 5PLC.12V. CAPSA	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
12	12818	10422	BATERIA 13API.13 PLC.12V	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
13	12820	10424	BATERIA CAPSA 11APCGI/11PLC.12V	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
14	12822	10426	BATERIA 13APCGI / 13PLC.12V. CAPSA	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
15	12824	10428	BATERIA 15APCGI / 15PLC.12V. CAPSA	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
16	12825	10429	BATERIA CAPSA 1135/11PLC.12V	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
17	12830	10434	BATERIA 11WI / 11PLC.12V. CAPSA	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
18	12835	10439	BATERIA CAPSA 15MB/15PLC.12V	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
19	12836	10440	BATERIA 15MBI / 15PLC.12V. CAPSA	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
20	12837	10441	BATERIA CAPSA 1731T/17PLC.12V	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
21	12838	10442	BATERIA CAPSA 20B/20PLC.12V	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
22	12839	10443	BATERIA CAPSA 20 4D/20PLC.12V	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
23	12843	10447	DESHUMEDECEDOR BOLA SECA CIA	1	T04	HOGAR Y DECO	A16	LIMPIEZA	D015	LIMPIADORES Y DESINFECTANTES
24	12844	10448	DESHUMEDECEDOR BOLA SECA RECAMBIO 400 GR X 3	1	T04	HOGAR Y DECO	A16	LIMPIEZA	D015	LIMPIADORES Y DESINFECTANTES
25	12849	10453	DESHUM DUO MINI BOLA SECA BOUQUET	1	T04	HOGAR Y DECO	A16	LIMPIEZA	D015	LIMPIADORES Y DESINFECTANTES
26	26428	24062	INGLETEADORA GCM 10 BOSCH	1	T02	FERRETERIA	A04	HERRAMIENTAS	D019	HERRAMIENTAS ELECTRICAS ESTACIONARIAS
27	26429	24063	TALADRO PERCUTOR 1/2 GSB 20-2RE BOSCH	1	T02	FERRETERIA	A04	HERRAMIENTAS	D018	HERRAMIENTAS ELECTRICAS PORTATILES
28	14669	12271	CUBRE ZAPATOS CELESTE (BOLSA X 20 PZS)	1	T02	FERRETERIA	A04	HERRAMIENTAS	D023	SEGURIDAD INDUSTRIAL
29	12538	10140	GATA LAGARTO 2 TN	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
30	12540	10142	GATA DE BOTELLA 2 TN	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ
31	12541	10143	GATA DE BOTELLA 4 TN	1	T02	FERRETERIA	A04	HERRAMIENTAS	D155	AUTOMOTRIZ



# Datos

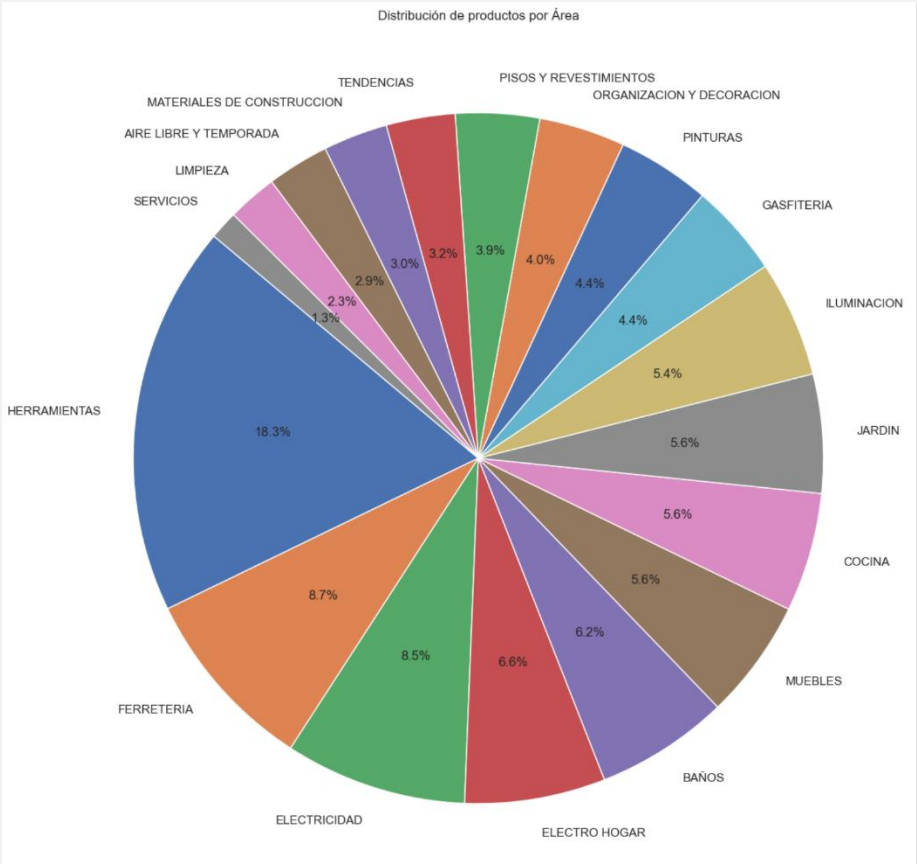
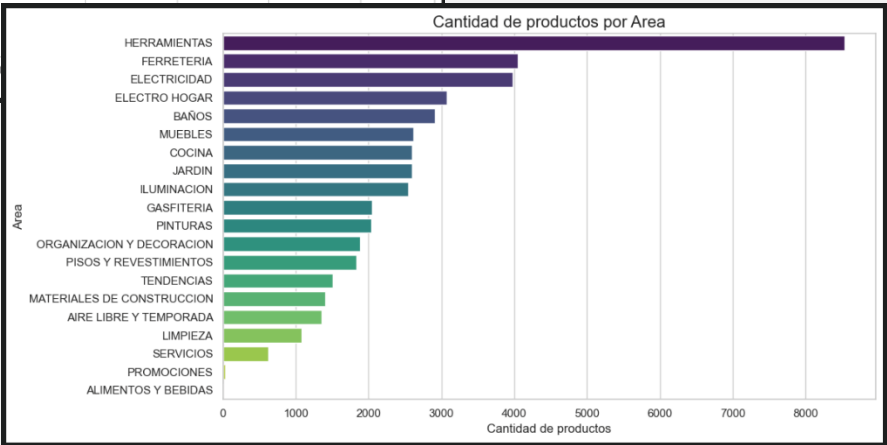
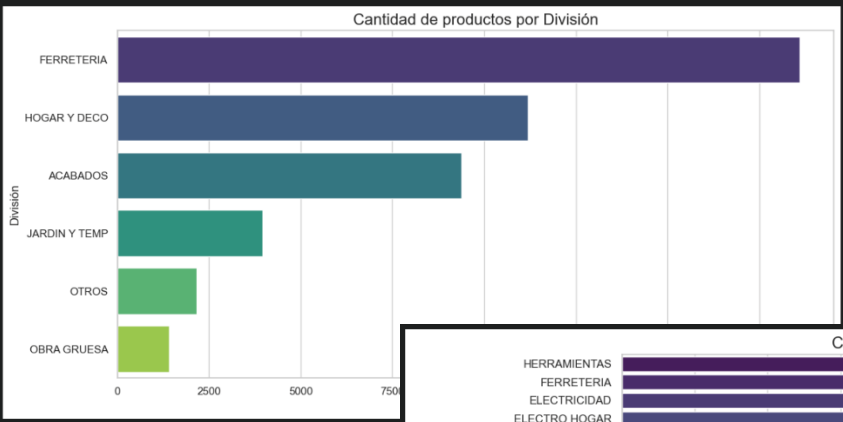
- \_ Eliminar columnas que no tiene información relevante
- \_ Eliminar valores NaN
- \_ Eliminar saltos de línea
- \_ Eliminar caracteres especiales
- \_ Eliminar etiquetas html

```
def formatearDescripcion(text,marca):  
    replacements = [  
        (' ' + marca, ''), # quitamos la marca en la descripcion  
        (' /', ''),  
        (' p/', ' para '),  
        (' c/', ' con '),  
        (' ace/', ' acero '),  
        (' s/', ' sin '),  
        (' d/', ' d '),  
        (' v/', ' vaso de '),  
        ('sobrep ', 'sobreponer '),  
        ('+', ' '),  
        (' l/', ' luz ')  
    ]  
  
    for old, new in replacements:  
        text = text.replace(old, new)  
  
    return text
```



# Datos

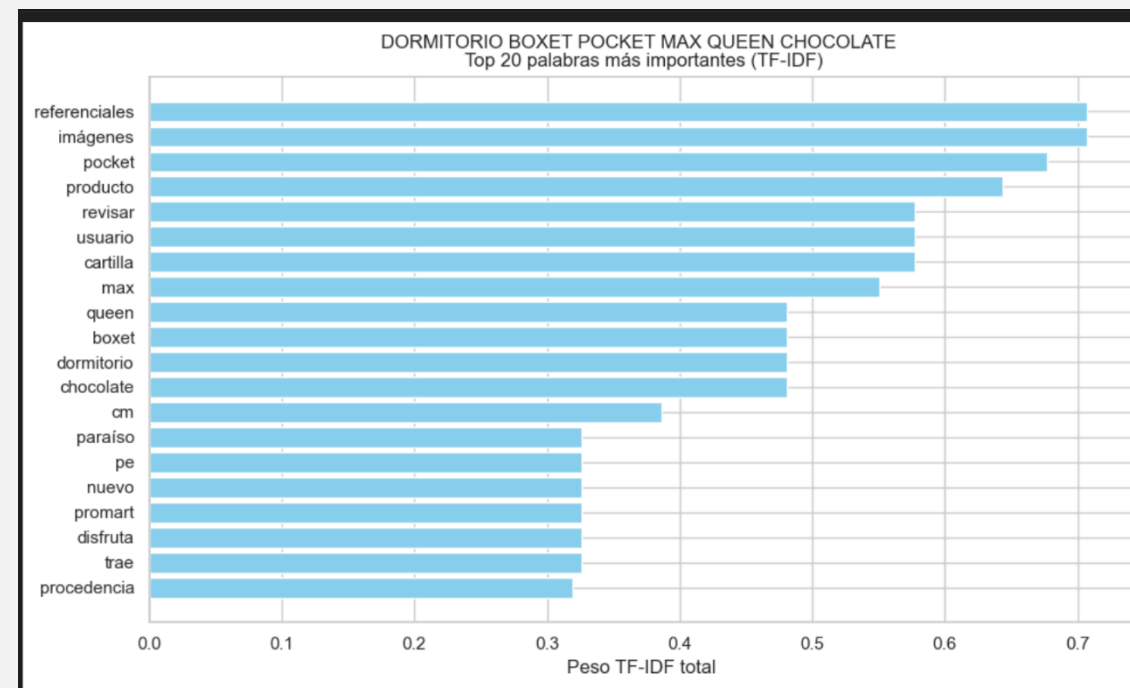
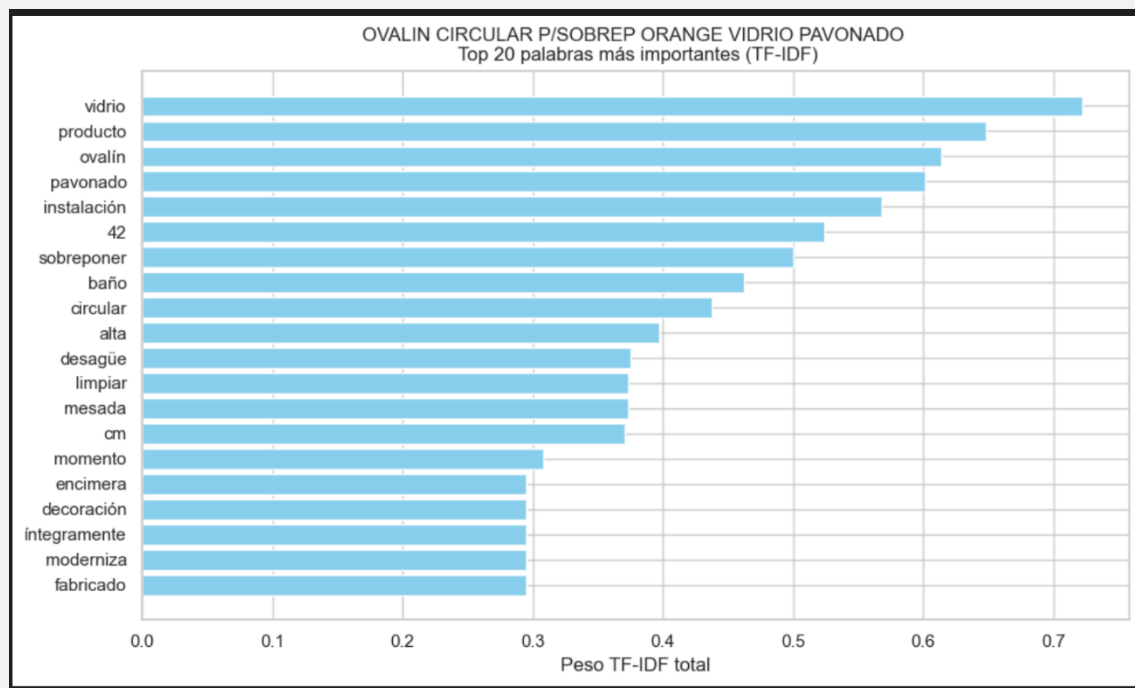
Generamos algunas graficas:





# Datos

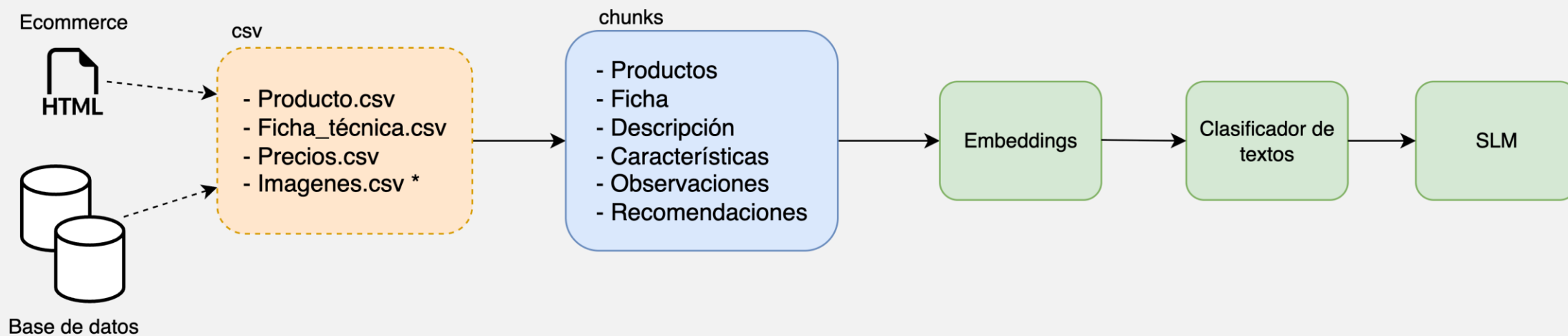
Generamos algunas graficas:





# Datos

Flujo de trabajo:







# Preprocesamiento

Generando Chunks:

SKU	DESCRIPCION	DES_DIV	DES_AREA	DES_DPTO	DES_LIN	DES_MARCA	DES_PROCE	URL	PRECIO
10236	OVALIN CIRCULAR P/SOBREP ORANGE VIDRIOTRANSPAR...	ACABADOS	BAÑOS	SANITARIOS	OVALINES	ORANGE	IMPORTADO	https://promart.vteximg.com.br/arquivos/ids/38...	79.9
10237	OVALIN CIRCULAR P/SOBREP ORANGE VIDRIO PAVONADO	ACABADOS	BAÑOS	SANITARIOS	OVALINES	ORANGE	IMPORTADO	https://promart.vteximg.com.br/arquivos/ids/50...	89.9
10298	LLAVE DUCHA ORANGE FROST S/SALIDA	ACABADOS	BAÑOS	GRIFERIA DE BAÑO	LLAVE DUCHA	ORANGE	IMPORTADO	https://promart.vteximg.com.br/arquivos/ids/51...	80.0
10077	SILLA ESTAMBUL FIJA H-602F NEGRA	HOGAR Y DECO	MUEBLES	OFICINAS	SILLAS FIJAS	ORANGE	IMPORTADO	https://promart.vteximg.com.br/arquivos/ids/38...	109.0
10177	MESA DE ESTUDIO ESPAÑA CASTAÑO	HOGAR Y DECO	MUEBLES	OFICINAS	ESCRITORIOS RTA	POLITORNO	IMPORTADO	https://promart.vteximg.com.br/arquivos/ids/59...	575.0

```
def formatoChunk(p):  
    text = formatearDescripcion(p['DESCRIPCION'].lower(),p['DES_MARCA'].lower())  
    if p['DES_MARCA'] != '':  
        text = text + ' de la marca '+p['DES_MARCA'].lower()  
  
    text = text + f" del area de {p['DES_AREA'].lower()} para los {p['DES_DPTO'].lower()} de la linea {p['DES_LIN'].lower()}"  
  
    if p['DES_PROCE'] != '':  
        text = text + ' de procedencia '+p['DES_PROCE'].lower()  
  
    text = text + f" al precio de {str(p['PRECIO'])} soles"  
  
    return limpiarTexto(text)
```

ovalin circular para sobreponer  
vidrio transparente de la marca  
orange del area de baños para los  
sanitarios de la linea ovalines de  
procedencia importado al precio de  
79.9 soles





# Preprocesamiento

Generando Chunks:

CHUNK_PRODUCTO	CHUNK_FICHA	CHUNK_DESCRIPCION	CHUNK_CARACTERISTICAS	CHUNK_OBSERVACIONES	CHUNK_RECOMENDACIONES
ovalin circular para sobreponer vidriotranspar...	acabado brillante, tamaño del producto moderad...	Moderniza la decoración de tu baño con este ov...	Ovalín para sobreponer en mesada de baño hecho...	Se recomienda complementar la instalación del ...	Al momento de instalar no usar cemento ni arci...
ovalin circular para sobreponer vidrio pavonad...	acabado pavonado, tamaño del producto moderado...	Moderniza la decoración de tu baño con este ov...	Ovalín para sobreponer en mesada de baño hecho...	Se recomienda complementar la instalación del ...	Al momento de instalar no usar cemento ni arci...
llave ducha frost sin salida de la marca orang...	acabado cromado, altura del producto 6.00 cm, ...	Dale un toque elegante a la grifería de tu duc...	Llave para ducha convencional de cuerpo interi...	Previo a la compra, verificar si se puede adap...	Maniobrar con cuidado para evitar el deterioro...
silla estambul fija h- 602f negra de la marca o...	altura del producto 79.00 cm, ancho del produc...	p¡Dale un toque moderno y funcional a tus espa...	Silla ideal para oficina, dormitorio o cualqui...	Silla fija, tiempo de vida estimado 2 años. Vi...	Estas sillas pueden ser limpiadas con un limpi...
mesa de estudio españa castaño de la marca pol...	acabado satinado, altura del producto 76.00 cm...	h2Diseño elegante y funcional con la Mesa de e...	Estructura de MDP, base del mismo material de ...	Mueble Fijo, no es necesario adosarlo a la par...	Se debe realizar la limpieza de sus muebles co...

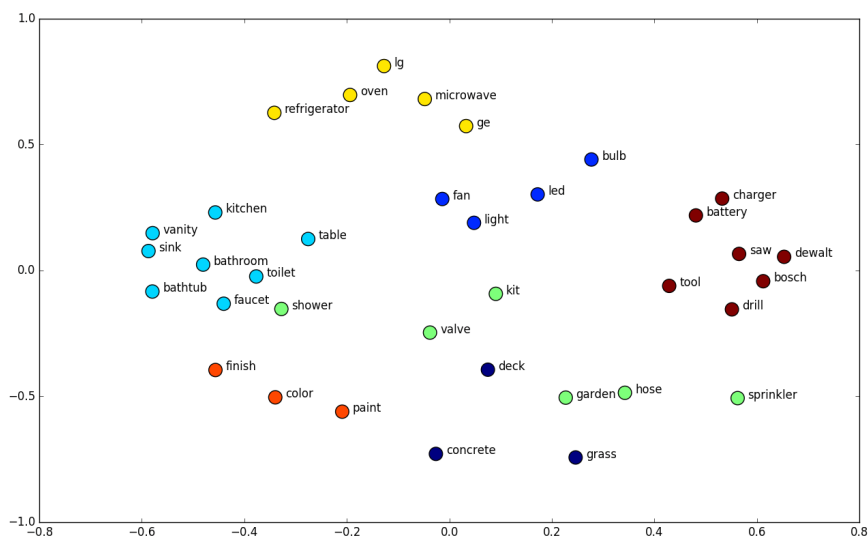


# Preprocesamiento

Generando embeddings:

```
from sentence_transformers import SentenceTransformer
import faiss
import numpy as np

# Cargar modelo localmente
embedder = SentenceTransformer("modelos/all-MiniLM-L6-v2")
```



```
# Convertir textos a embeddings
corpus_embeddings = embedder.encode(df['CHUNK_PRODUCTO'].tolist(), convert_to_numpy=True)
np.save("embeddings/CHUNK_PRODUCTO.npy", corpus_embeddings)

corpus_embeddings = embedder.encode(df['CHUNK_FICHA'].tolist(), convert_to_numpy=True)
np.save("embeddings/CHUNK_FICHA.npy", corpus_embeddings)

corpus_embeddings = embedder.encode(df['CHUNK_DESCRIPCION'].tolist(), convert_to_numpy=True)
np.save("embeddings/CHUNK_DESCRIPCION.npy", corpus_embeddings)

corpus_embeddings = embedder.encode(df['CHUNK_CARACTERISTICAS'].tolist(), convert_to_numpy=True)
np.save("embeddings/CHUNK_CARACTERISTICAS.npy", corpus_embeddings)

corpus_embeddings = embedder.encode(df['CHUNK_OBSERVACIONES'].tolist(), convert_to_numpy=True)
np.save("embeddings/CHUNK_OBSERVACIONES.npy", corpus_embeddings)

corpus_embeddings = embedder.encode(df['CHUNK_RECOMENDACIONES'].tolist(), convert_to_numpy=True)
np.save("embeddings/CHUNK_RECOMENDACIONES.npy", corpus_embeddings)
```



## Probando la distancia vectorial:

```
import faiss
import numpy as np
```

```
embeddings = np.load("embeddings/CHUNK_PRODUCTO.npy")
embeddings = np.concatenate((embeddings,np.load("embeddings/CHUNK_FICHA.npy")))
embeddings = np.concatenate((embeddings,np.load("embeddings/CHUNK_DESCRIPCION.npy")))
embeddings = np.concatenate((embeddings,np.load("embeddings/CHUNK_CARACTERISTICAS.npy")))
embeddings = np.concatenate((embeddings,np.load("embeddings/CHUNK_OBSERVACIONES.npy")))
embeddings = np.concatenate((embeddings,np.load("embeddings/CHUNK_RECOMENDACIONES.npy")))
```

```
# Crear índice FAISS
index = faiss.IndexFlatL2(embeddings.shape[1])
index.add(embeddings)
```

```
def responder_pregunta_rag(pregunta, k=10):
    # Embed la pregunta
    pregunta_emb = embedder.encode([pregunta], convert_to_numpy=True)

    # Buscar los k textos más cercanos
    distancias, indices = index.search(pregunta_emb, k)
    print(distancias)
    print(indices)
```

```
responder_pregunta_rag('necesito una refrigeradora')
```

```
[[0.47783175 0.48447216 0.48447216 0.48447216 0.48447216 0.48447216
 0.48447216 0.48447216 0.48447216 0.5248814 ]]
[[176543 224699 224700 224701 224702 224703 224704 227696 227697 227731]]
```

SKU	156424
DESCRIPCION	OSTER FRIGOBAR 122LTS OS-PMB129BB NEGRO
DES_DIV	HOGAR Y DECO
DES_AREA	ELECTRO HOGAR
DES_DPTO	LINEA BLANCA
DES_LIN	REFRIGERADORAS
DES_MARCA	OSTER
DES_PROCE	NACIONAL
URL	NaN
PRECIO	999.0
CHUNK_PRODUCTO	oster frigobar 122lts os-pmb129bb negro de la ...
CHUNK_FICHA	acabado liso, altura del producto 84.00 cm, an...
CHUNK_DESCRIPCION	Compartimiento refrigerado. Control de tempera...
CHUNK_CARACTERISTICAS	Compartimiento refrigerado. Patas ajustables. ...
CHUNK_OBSERVACIONES	Ideal para espacios pequeños del hogar u ofici...
CHUNK_RECOMENDACIONES	Colocar en superficies totalmente planas.

Name: 37073, dtype: object



# Enrutando el contexto

Usando spaCy:

	CHUNK_PRODUCTO	PREGUNTA	ES_ABIERTA	RESPUESTA
0	sikaflex 221 blanco x 600 ml de la marca sika ...	¿Qué beneficios ofrece el producto sikaflex 22...	1	El producto sikaflex 221 blanco x 600 ml de la...
1	maceta cuadr piedra 30x30x30cm con sist. riego...	¿Por qué elegir el producto maceta cuadr piedr...	1	El producto maceta cuadr piedra 30x30x30cm con...
2	cuchilla plastica econo de la marca home tools...	¿Se puede usar el producto cuchilla plastica e...	0	Sí, en la mayoría de los casos.
3	coladera redonda abs con rejilla cromada abs d...	¿Qué características destacan del producto col...	1	El producto coladera redonda abs con rejilla c...
4	guirnalda de pompones blancos 1.8m c.natural d...	¿Se puede usar el producto guirnalda de pompon...	0	Depende del modelo.

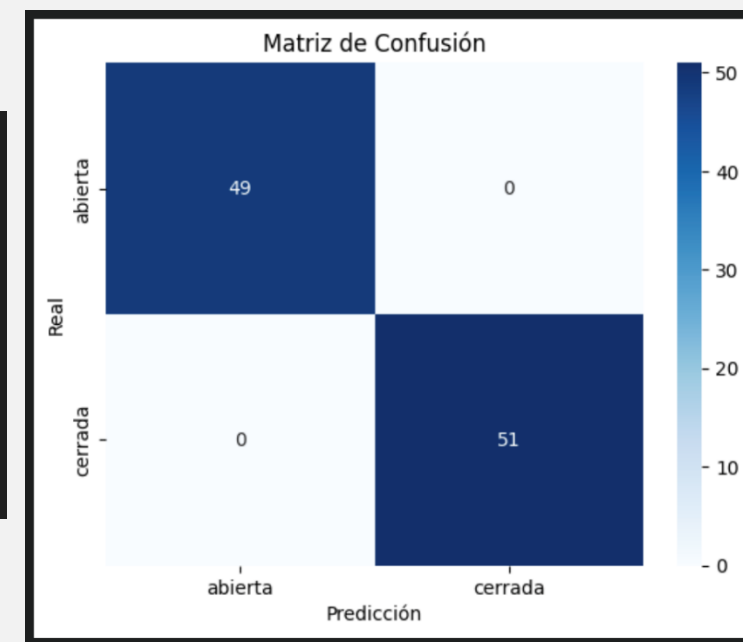
```
import spacy
from spacy.training.example import Example
from spacy.util import minibatch, compounding
import random

# Preparar datos para entrenamiento
train_data = []
for _, row in df_train.iterrows():
    texto = row["PREGUNTA"]
    es_abierto = row["ES_ABIERTA"]
    labels = {
        "cats": {
            "abierto": float(es_abierto == 1),
            "cerrada": float(es_abierto == 0)
        }
    }
    train_data.append((texto, labels))

# Crear modelo en blanco (sin necesidad de conexión)
#nlp = spacy.blank("es")
nlp = spacy.load("es_core_news_sm", disable=["lemmatizer"])
```

```
--- Reporte de clasificación ---
```

	precision	recall	f1-score	support
abierto	1.00	1.00	1.00	49
cerrada	1.00	1.00	1.00	51
accuracy			1.00	100
macro avg	1.00	1.00	1.00	100
weighted avg	1.00	1.00	1.00	100





# Enrutando el contexto

Usando spaCy:

```
# Patrones de consulta general de producto
producto_patterns = [
    [{"LOWER": {"IN": ["tienen", "hay", "venden", "disponible"]}, {"POS": "NOUN"}],
    [{"LOWER": "busco"}, {"POS": "DET", "OP": "?"}, {"POS": "NOUN"}],
    [{"LOWER": {"IN": ["necesito", "quisiera"]}, {"POS": "DET", "OP": "?"}, {"POS": "NOUN"}],
    [{"LOWER": {"IN": ["tienen", "hay"]}, {"LOWER": {"IN": ["descuento", "promocion", "oferta"]}}],
    [{"LOWER": "este"}, {"LOWER": "producto"}, {"LOWER": "en"}, {"LOWER": {"IN": ["oferta", "descuento"]}}],
    [{"LOWER": "está"}, {"LOWER": "en"}, {"LOWER": {"IN": ["oferta", "promocion", "rebaja"]}}],
    [{"LOWER": "precio"}, {"LOWER": "rebajado"}]
]

# Patrones de ficha técnica
ficha_patterns = [
    [{"LOWER": {"IN": ["peso", "material", "garantia", "dimensiones", "voltaje", "capacidad", "watts"]}}],
    [{"LOWER": {"IN": ["es"]}, {"LOWER": {"IN": ["resistente", "impermeable", "inoxidable"]}}],
    [{"LOWER": "de"}, {"LOWER": "que"}, {"LOWER": "material"}, {"LOWER": "es"}]
]

descripcion_patterns = [
    [{"LOWER": "que"}, {"LOWER": "es"}],
    [{"LOWER": "para"}, {"LOWER": "que"}, {"LOWER": "sirve"}],
    [{"LOWER": "cual"}, {"LOWER": "es"}, {"LOWER": "su"}, {"LOWER": "funcion"}],
    [{"LOWER": "como"}, {"LOWER": "funciona"}],
    [{"LOWER": "que"}, {"LOWER": "hace"}],
    [{"LOWER": {"IN": ["describe", "describirme", "puedes", "podrías"]}, {"LOWER": {"IN": ["esto", "este", "producto", "articulo", "objeto"]}}],
    [{"LOWER": "que"}, {"LOWER": "características"}, {"LOWER": "tiene"}],
]
```

¿Tienen sillas plegables? → ['CONSULTA\_PRODUCTO']  
Busco una lámpara de escritorio. → ['CONSULTA\_PRODUCTO']  
¿Cual es el peso del taladro? → ['FICHA\_TECNICA']  
¿Es resistente al agua? → ['FICHA\_TECNICA']  
¿Tienen este producto en descuento? → ['CONSULTA\_PRODUCTO']  
¿que voltaje tiene el microondas? → ['FICHA\_TECNICA']  
Para que sirve una cocina? → ['DESCRIPCION']  
venden camas? → ['CONSULTA\_PRODUCTO']  
que cama me recomiendan? → ['RECOMENDACION']



# Utilizando el SLM

Usamos PHI 4 Mini:

```
# Cargar tokenizer y modelo de lenguaje localmente
from transformers import AutoTokenizer, AutoModelForCausalLM
import torch
torch.set_num_threads(8)

tokenizer = AutoTokenizer.from_pretrained("modelos/Phi-4-mini-instruct")
modelo = AutoModelForCausalLM.from_pretrained("modelos/Phi-4-mini-instruct", torch_dtype=torch.bfloat16).to("cpu")
```

✓ 39.7s

```
def modelScore(pregunta):
    contexto = "ovalin circular para sobreponer vidrio transparente de la marca orange del area de baños para los sanitarios"
    prompt = f"""
    CONTEXTO:
    {contexto}
    PREGUNTA: {pregunta}
    RESPUESTA: """
    inputs = tokenizer(prompt, return_tensors="pt").to(modelo.device)
    outputs = modelo.generate(**inputs, min_new_tokens=10, max_new_tokens=25, do_sample=True, top_p=0.5, temperature=0.1)
    return tokenizer.decode(outputs[0], skip_special_tokens=True)
```

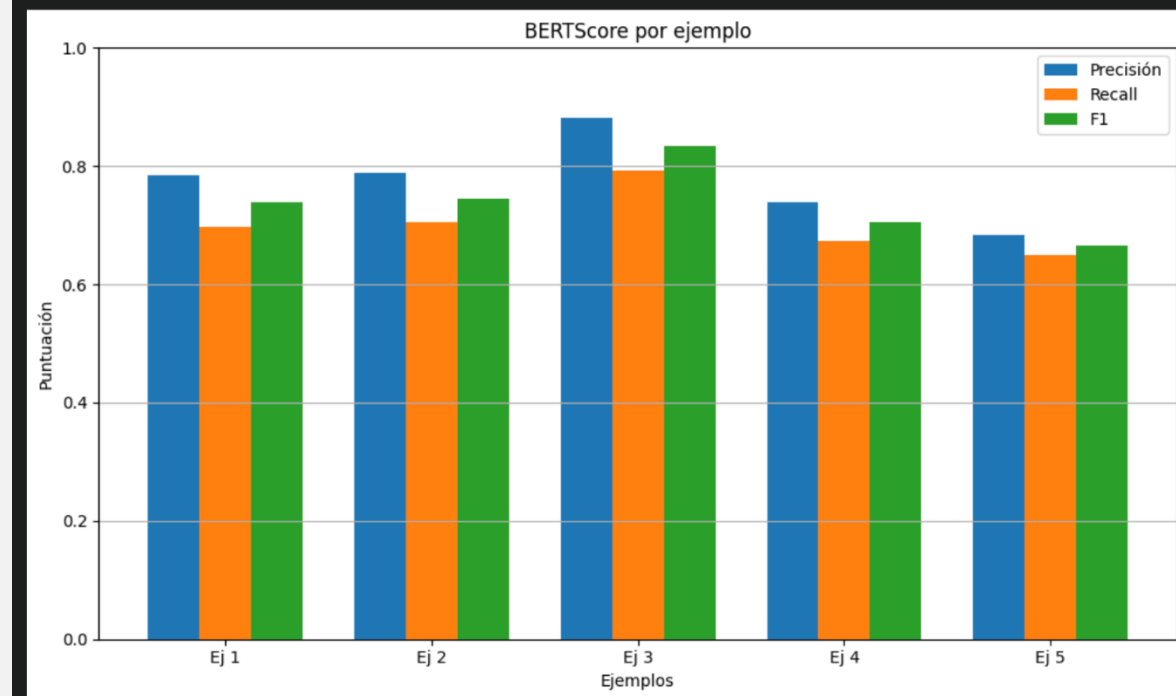


# Algunas métricas

Usamos BERT SCORE:

```
# Cual es la procedencia del ovalin?  
# Cual es el precio de ovalin?  
# el ovalin es circular o cuadrado?  
# de que marca es el ovalin?  
# de que area es el ovalin?  
  
# Candidatos y referencias  
candidatos = [  
    "es importando",  
    "es 79.9 soles",  
    "Es circular",  
    "Es Orange",  
    "Del area de baños"  
]  
  
referencias = [  
    "La procedencia del ovalin es de procedencia importado",  
    "El precio de ovalin es de 79.9 soles.",  
    "el ovalin es circular",  
    "el ovalin es de la marca orange",  
    "el ovalin es un tipo de vidrio de la marca orange, utilizado"
```

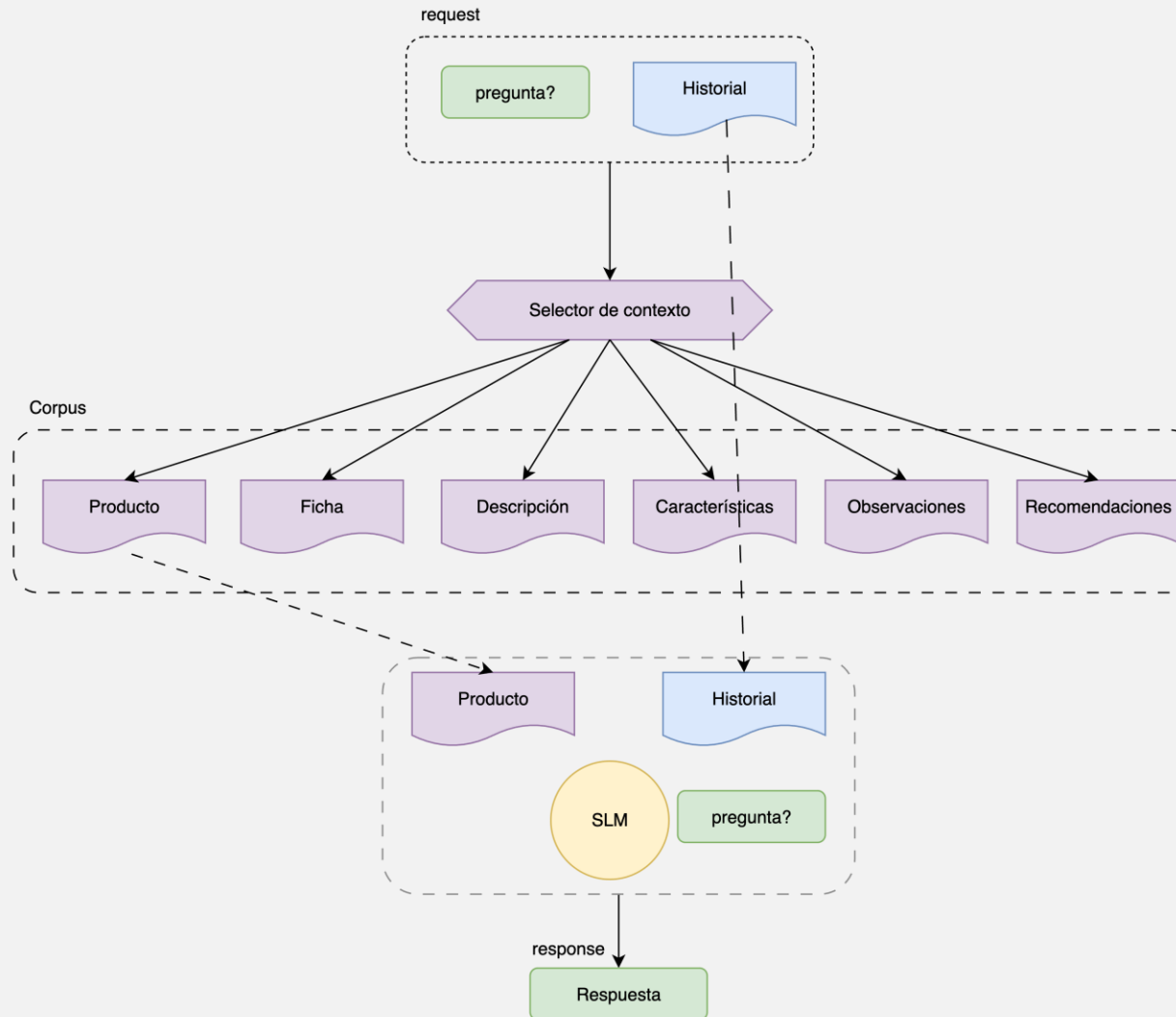
```
calculating scores...  
computing bert embedding.  
100%|██████████| 1/1 [00:00<00:00, 1.47it/s]  
computing greedy matching.  
100%|██████████| 1/1 [00:00<00:00, 562.09it/s]  
done in 0.68 seconds, 7.31 sentences/sec
```







# Integrando los modelos





# Integrando los modelos

```
def promartBoot(texto, mensajes, k):
    # -----
    pregunta_emb = embedder.encode([texto], convert_to_numpy=True)
    # Buscar los k textos más cercanos
    distancias, indices = index.search(pregunta_emb, k)
    contexto = "\n".join([getChunk(i) for i in indices[0]])
    mensajes.append({"role": "system", "content": "CONTEXTO: "+contexto})
    #print(contexto)
    print('-----')
    # -----

    mensajes.append({"role": "user", "content": texto})

    pipe = pipeline("text-generation", model=modelo, tokenizer=tokenizer)

    generation_args = {
        "max_new_tokens": 500,
        "return_full_text": False,
        "temperature": 0.0,
        "do_sample": False,
    }

    output = pipe(mensajes, **generation_args)
    return output[0]['generated_text']
```

```
promartBoot('necesito un ovalin circular mimbell', [], 3)
```

Device set to use cuda:0

The following generation flags are not valid and may be ignored: ['temperature']. Set `TRANSFORMERS\_VERBOSITY=info` for more details.

-----  
'Parece que estás buscando un ovalin circular mimbell. Si estás interesado en comprar uno, aquí tienes una sugerencia basada en tus preferencias:\n\n\*\*Ovalin circular mimbell de la marca Italgrif del área de baños para los sanitarios'



# Integrando los modelos

```
from fastapi import FastAPI
from fastapi.middleware.cors import CORSMiddleware
from pydantic import BaseModel
import nest_asyncio
import uvicorn
from pyngrok import ngrok
```

```
@app.post("/clasificar")
def clasificar(body: Body):
    texto = body.pregunta
    k = body.k
    history = body.history
    return {"respuesta": promartBoot(texto, history, k)}
```

The screenshot shows a web browser at `localhost/promart/#` displaying a chat interface. The header is red with the text "UNI - MAESTRIA IA". On the left, there is a profile for "Andree Ochoa" (Administrator) and a large watermark of the UNI logo. The main chat area shows a conversation with "Promart Boot" (Activo). The chat history includes:

- Promart Boot: "Bienvenido, en que puedo ayudarte?" (03:54:36)
- User: "quiero un ovalin para mi baño, cuales tienen?" (03:54:54)
- Promart Boot: "el ovalin para sobreponer divani blanco de la marca trebol del area de baños para los sanitarios de la linea ovalines de procedencia nacional al precio de 179.9 soles" (03:55:36)
- User: "tienen alguna coladera redonda?" (03:56:54)
- Promart Boot: "no, no tenemos coladeras redondas, solo coladeras de varios tamaños y formas." (03:57:37)

At the bottom is a text input field labeled "Escribe ...". On the right side, there are two JSON blocks:

**Request:**

```
{
  "pregunta": "tienen alguna coladera redonda?"
}
```

**Response:**

```
{
  "respuesta": "no, no tenemos coladeras redondas, solo coladeras de varios tamaños y formas."
}
```



*Gracias*