TEMA 7 – HOJA DE EJERCICIOS I

Se plantean diferentes ejercicios para el Reconocimiento de Entidades Nombradas (NER por sus siglas en inglés).

Ejercicio 1. Reconocimiento de entidades nombradas con NLTK y SpaCy

Tanto NLTK como SpaCy incluyen modelos y herramientas preentrenadas para el reconocimiento de entidades nombradas, siendo capaces de identificar categorías de entidades comunes como personas, localizaciones u organizaciones. En este ejercicio se plantea usar dichas librerías para detectar algunos de los tipos de entidades nombradas más comunes.

De este modo, dada la frase de entrada "Barack Obama was the 44th president of the United States" se pide lo siguiente:

a) Utiliza NLTK para detectar las entidades nombradas contenidas en dicha frase. Para ello, tokeniza el texto en palabras, añádele las etiquetas de categoría gramatical y proporciona las tuplas resultantes de *(palabra, categoría gramatical)* a la función nltk.ne_chunk. Nota que dicha función construye un objeto nltk.Tree que agrupa la secuencia de tokens que forman las entidades nombradas, por lo que se deberá recorrer dicho árbol para devolver las entidades nombradas. Observa y analiza las diferencias entre pasarle binary=False y binary=True a la función nltk.ne_chunk.
b) Emplea SpaCy para reconocer las entidades nombradas de la frase anterior. Con dicho fin, procesa la frase de entrada con el modelo de inglés "en_core_web_sm" y extrae las entidades nombradas con la tupla de objetos *Span* doc.ents, la cual recoge todas las entidades nombradas del componente de NER del pipeline de SpaCy.
c) Tradicionalmente, los modelos preentrenados de SpaCy se han basado en arquitecturas de redes convolucionales. Sin embargo, con el surgimiento de los Transformers, SpaCy ha desarrollado nuevas versiones de sus modelos utilizando este tipo de arquitectura de redes neuronales. Dada la frase de entrada anteriormente analizada, detecta nuevamente las entidades nombradas en la misma con el modelo de SpaCy "en_core_web_trf", el cual ha sido entrenado a partir del modelo de Transformers RoBERTa.
d) Detecta nuevamente con NLTK y SpaCy las entidades nombradas en el conjunto de frases mostrado a continuación. Compara la detección de entidades realizada por los modelos de inglés de SpaCy "en_core_web_sm" y "en_core_web_trf". ¿Observas diferencias en las entidades reconocidas por estos modelos? ¿A qué podrían deberse?

```
sentences = [
    "Apple is looking at buying U.K. startup for $1 billion",
    "The Eiffel Tower is located in Paris, France",
    "Cristiano Ronaldo scores winner in Champions League final"
]
```

e) Los modelos anteriormente vistos proporcionan un reconocimiento de entidades nombradas de uso general. Sin embargo, en ciertos dominios especializados como el biomédico, es fundamental detectar entidades propias de dichos ámbitos. Dados los modelos de reconocimiento de entidades nombradas de SciSpaCy, y en particular el modelo "en_ner_jnlpba_md", detecta las entidades nombradas de la frase "The p53 protein regulates the expression of CDKN1A mRNA in HeLa cells.". ¿Qué entidades biomédicas se han detectado? ¿Qué etiquetas tienen asignadas dichas entidades reconocidas?

Referencias:

- https://www.nltk.org/api/nltk.tokenize.word_tokenize.html
- https://www.nltk.org/api/nltk.tag.pos_tag.html
- https://www.nltk.org/api/nltk.chunk.ne_chunk.html
- https://www.nltk.org/api/nltk.tree.tree.html
- https://spacy.io/api/doc#ents
- https://allenai.github.io/scispacy/

Ejercicio 2. Fine-tuning de modelos de NER preentrenados

Aunque SpaCy proporciona modelos preentrenados para detectar ciertas entidades nombradas, a veces es necesario adaptar y realizar un fine-tuning de estos modelos para distintos casos de uso. Por ello, en este ejercicio se propone ajustar un modelo preentrenado de SpaCy con datos de entrenamiento etiquetados para el reconocimiento de entidades nombradas.

Con dicho fin, dado el siguiente texto correspondiente a un artículo de una noticia:

text = """En medio de una cuidada puesta en escena digna del fin de una guerra, y ante más de 50 cámaras de medios, algunos venidos de fuera de España para la ocasión, Carmen Cervera, el Ministro de Cultura Miquel Iceta, en representación del Gobierno del Estado español, y Borja Thyssen-Bornemisza han procedido esta mañana a la dilatada firma del acuerdo que permite la permanencia en España del Mata Mua y una parte importante de la colección de la baronesa Thyssen-Bornemisza. El cuadro más famoso de pintor postimpresionista francés Paul Gauguin ya cuelga en las salas del museo madrileño. Ha sido una larga travesía en las que se han desarrollado intensísimas negociaciones, retiradas de las partes, principios de acuerdo y retrasos desde hace casi una década. "Ha sido necesario mucho esfuerzo de las dos partes, pero hoy al fin es una hermosa realidad", ha declarado una satisfecha la baronesa. "Hasta hace pocos días no estaba seguro que el cuadro estuviese aquí entre nosotros. Hoy tenemos el honor y el privilegio de disfrutar en España de esta y otras pinturas que componen una colección de

las más importantes del mundo. Esta firma es un final feliz", concluyó el ministro de Cultura. El contrato de arrendamiento asegura la permanencia del cuadro en España, junto con 320 obras pertenecientes a la colección Carmen Thyseen-Bornemisza, una cuarta parte menos de la garantía actual, durante 15 años. A cambio, la baronesa recibirá 6,5 millones de euros anuales en calidad de préstamo. Transcurrido ese tiempo, y pagado el importe total, 97,5 millones de euros, el Estado podrá optar a la compra del cuadro, descontando del precio final lo pagado. Esto puede suponer reeditar en el futuro los problemas hoy finiquitados."""

Hay que realizar los siguientes pasos:

a. Carga el modelo de español "es_core_news_sm" y observa la detección de entidades nombradas que realiza este modelo sobre el anterior artículo.
b. A continuación, dada la siguiente lista de tuplas de entrenamiento de entidades nombradas, las cuales contienen diferentes frases y las posiciones y categoría de las entidades nombradas presentes en las mismas:

```
training = [
    ("Acuerdo entre el Gobierno del Estado Español y el Gobierno de la República Federal de Alemania
    para la aplicación del Convenio de 20 de abril de 1966 sobre Seguro de Desempleo.", [(17, 44, "ORG"),
    (50, 94, "ORG"), (156, 175, "MISC")]),
    ("Esta mañana, el primer secretario del PSC, Miquel Iceta, ha tomado posesión del cargo de ministro
    de Cultura y Deportes.", [(16, 33, "PER"), (38, 41, "ORG"), (43, 55, "PER"), (89, 119, "PER")]),
    ("España el país que más gusta a los franceses.", [(0, 6, "LOC")])
]
```

Transforma cada tupla en un objeto de tipo spacy.training.Example para su posterior uso durante el ajuste (fine-tuning) del modelo cargado de SpaCy. Puedes realizar la transformación de cada tupla mediante el siguiente código de ejemplo:

```
example = Example.from_dict(
    nlp.make_doc(raw_text),
    {"entities": entity_offsets}
)
```

c. Realiza un fine-tuning del modelo de SpaCy en español cargado usando la anterior lista de ejemplos de entrenamiento. Para ello, en primer lugar, identifica los componentes del pipeline de Spacy que no se desean reentrenar mediante el código propuesto a continuación:

```
disabled_pipes = [pipe for pipe in nlp.pipe_names if pipe != "ner"]
```

Tras ello, procede a reentrenar el componente de NER del pipeline cargado mediante el siguiente código de ejemplo:

```
optimizer = nlp.create_optimizer()
with nlp.disable_pipes(*disabled_pipes):
    for _ in range(25):
            random.shuffle(training)
            for example in examples:
                    nlp.update([example], sgd=optimizer)
```

Detecta de nuevo las entidades nombradas en el artículo analizado con el modelo ya reentrenado. ¿Han cambiado las entidades detectadas? ¿Ha variado su clasificación?

d. El conjunto de entrenamiento proporcionado en el apartado b) contenía las mismas categorías de entidades nombradas que el modelo de español cargado. Sin embargo, es habitual que se requiera reconocer también nuevas categorías de entidades nombradas. Por ello, dado el siguiente nuevo conjunto de entrenamiento, el cual introduce entidades nombradas de tipo "JOB":

```
training = [
    ("Acuerdo entre el Gobierno del Estado Español y el Gobierno de la República Federal de Alemania
    para la aplicación del Convenio de 20 de abril de 1966 sobre Seguro de Desempleo.", [(17, 44, "ORG"),
    (50, 94, "ORG"), (156, 175, "MISC")]),
    ("Esta mañana, el primer secretario del PSC, Miquel Iceta, ha tomado posesión del cargo de ministro
    de Cultura y Deportes.", [(16, 33, "JOB"), (38, 41, "ORG"), (43, 55, "PER"), (89, 119, "JOB")]),
    ("España el país que más gusta a los franceses.", [(0, 6, "LOC")])
]
```

Añade la nueva etiqueta "JOB" al pipeline de NER. Puedes hacerlo mediante el siguiente código propuesto:

```
ner = nlp.get_pipe('ner')

print("Default labels", ner.labels)
ner.add_label('JOB')
print("New labels", ner.labels)
```

Tras ello, repite el proceso de reentrenamiento del modelo de español con los nuevos ejemplos de entrenamiento. ¿Hay diferencias en las entidades nombradas detectadas y sus categorías?

Referencias:

- https://spacy.io/api/example
- https://spacy.io/usage/training/#api-train
- https://spacy.io/api/entityrecognizer#add_label