Lenguaje natural en la Inteligencia Artificial (PLN)

Definición

También conocido como el procesamiento del lenguaje natural es el campo de conocimiento de la Inteligencia Artificial que se ocupa de la investigar la manera de comunicar las máquinas con las personas mediante el uso de lenguas naturales, como el español, el inglés o el chino.

Las lenguas humanas pueden expresarse por escrito (texto), oralmente (voz) y también mediante signos. Naturalmente, el PLN está más avanzado en el tratamiento de textos, donde hay muchos más datos y son más fáciles de conseguir en formato electrónico.

Los audios, aunque estén en formato digital, hay que procesarlos para transcribirlos en letras o caracteres y, a partir de ahí, entender la pregunta. El proceso de respuesta es el inverso: primero se elabora la oración y luego se “sintetiza la voz”.

El PLN combina la lingüística computacional (modelado del lenguaje humano basado en reglas) con modelos estadísticos, de machine learning y de deep learning. Juntas, estas tecnologías permiten que las computadoras procesen el lenguaje humano en forma de texto o datos de voz y "comprendan" su significado completo, con la intención y el sentimiento de la persona que habla o escribe.

Modelos para procesamiento del lenguaje natural

Tratar computacionalmente una lengua implica un proceso de **modelización matemática**. Los ordenadores sólo entienden de bytes y dígitos y los informáticos codifican los programas empleando lenguajes de programación como C, Python o Java. Existen dos aproximaciones generales al problema de la modelización lingüística:

Modelos Lógicos: gramáticas

Los lingüistas escriben reglas de reconocimiento de patrones estructurales, empleando un formalismo gramatical concreto. Estas reglas, en combinación con la información almacenada en diccionarios computacionales, definen los patrones que hay que reconocer para resolver la tarea (buscar información, traducir, etc.).

Estos modelos lógicos pretenden reflejar la estructura lógica del lenguaje y surgen a partir de las teorías de N. Chomsky en los años 50.

Modelos probabilísticos del lenguaje natural: basados en datos

La aproximación es a la inversa: los lingüistas recogen colecciones de ejemplos y datos (corpus) y a partir de ellos se calculan las frecuencias de diferentes unidades lingüísticas (letras, palabras, oraciones) y su probabilidad de aparecer en un contexto determinado. Calculando esta probabilidad, se puede predecir cuál será la siguiente unidad en un contexto dado, sin necesidad de recurrir a reglas gramaticales explícitas.

Es el paradigma de “aprendizaje automático” que se ha impuesto en las últimas décadas en Inteligencia Artificial: los algoritmos infieren las posibles respuestas a partir de los datos observados anteriormente en el corpus.

Componentes del procesamiento del lenguaje natural

A continuación, vemos algunos de los componentes del procesamiento del lenguaje natural. No todos los análisis que se describen se aplican en cualquier tarea de PLN, sino que depende del objetivo de la aplicación.

- Análisis morfológico o léxico. Consiste en el análisis interno de las palabras que forman oraciones para extraer lemas, rasgos flexivos, unidades léxicas compuestas. Es esencial para la información básica: categoría sintáctica y significado léxico.
- Análisis sintáctico. Consiste en el análisis de la estructura de las oraciones de acuerdo con el modelo gramatical empleado (lógico o estadístico).
- Análisis semántico. Proporciona la interpretación de las oraciones, una vez eliminadas las ambigüedades morfosintácticas.

- Análisis pragmático. Incorpora el análisis del contexto de uso a la interpretación final. Aquí se incluye el tratamiento del lenguaje figurado (metáfora e ironía) como el conocimiento del mundo específico necesario para entender un texto especializado.

Un análisis morfológico, sintáctico, semántico o pragmático se aplicará dependiendo del objetivo de la aplicación. Por ejemplo, un conversor de texto a voz no necesita el análisis semántico o pragmático. Pero un sistema conversacional requiere información muy detallada del contexto y del dominio temático.

Herramientas y enfoques de PLN

Python y el kit de herramientas de lenguaje natural (NLTK)
El lenguaje de programación **Python** proporciona una amplia variedad de herramientas y bibliotecas para abordar tareas específicas del PLN. Muchas de estas se encuentran en Natural Language Toolkit, o NLTK, una colección de código abierto de bibliotecas, programas y recursos educativos para crear programas de PLN.

El NLTK incluye bibliotecas para muchas de las tareas de PLN enumeradas anteriormente, además de bibliotecas para subtareas, como análisis sintáctico de oraciones, segmentación de palabras, derivación y lematización (métodos para recortar palabras a sus raíces) y tokenización (para dividir frases, oraciones, párrafos y pasajes en fichas que ayudan a la computadora a comprender mejor el texto). También incluye bibliotecas para implementar capacidades como el razonamiento semántico, la capacidad de llegar a conclusiones lógicas basadas en hechos extraídos del texto.

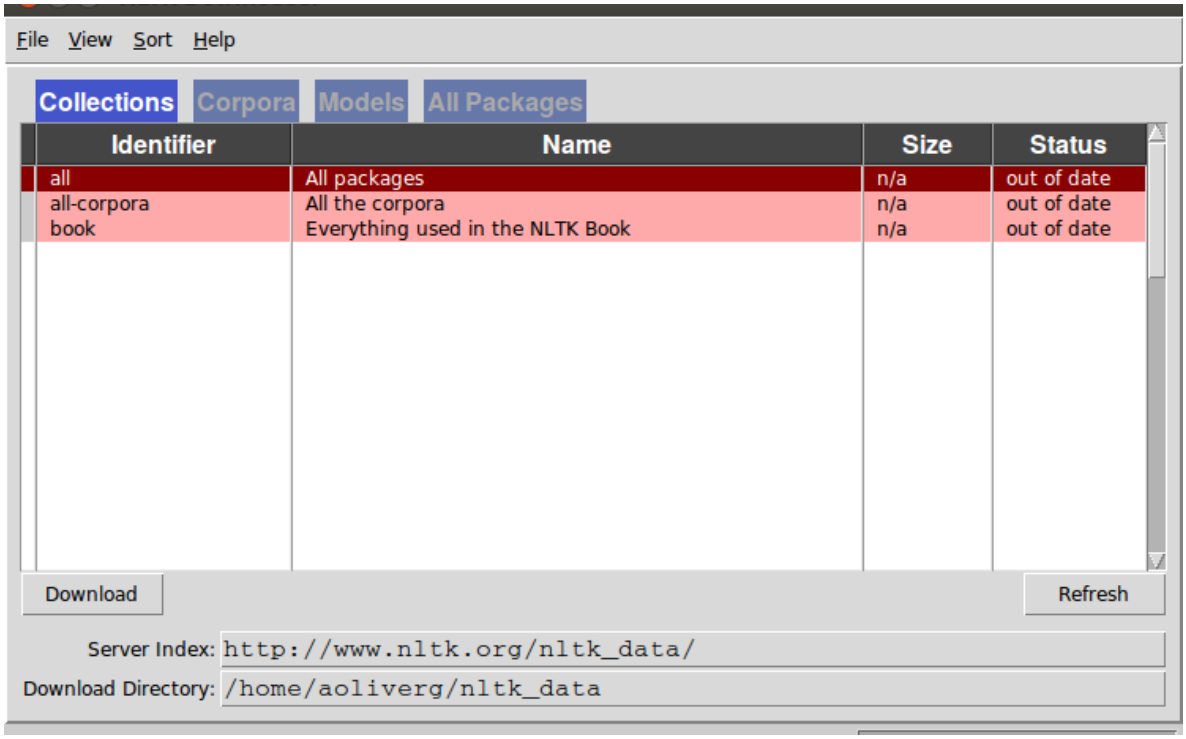
Instalación de los datos lingüísticos de NLTK

NLTK proporciona muchos datos lingüísticos: listas de palabras, corpus, modelos de lenguaje, etc. Se puede ver una lista completa y actualizada de los datos lingüísticos proporcionados con el NLTK en http://www.nltk.org/nltk_data/.

Para instalar los datos abrimos un intérprete interactivo de Python y escribimos:

```
import nltk
nltk.download()
```

Aparecerá una ventana cómo la siguiente:



Aquí podemos seleccionar **All y Download**. De este modo descargamos todos los datos disponibles. En esta sección presentamos unos breves ejemplos, que ejecutaremos desde el intérprete interactivo, y nos servirán para verificar la instalación de NLTK y los datos, y ver algunas funcionalidades.

Ejemplo de tokenización:

```
>>> import nltk

>>> texto="This is a sentence. This is another sentence."

>>> nltk.tokenize.word_tokenize(texto)

['This', 'is', 'a', 'sentence', '.', 'This', 'is', 'another', 'sentence', '.']
```

Un ejemplo de etiquetado morfosintáctico

```
>>> tokenized=nltk.word_tokenize(texto)
>>> nltk.pos_tag(tokenized)
[('This', 'DT'), ('is', 'VBZ'), ('a', 'DT'), ('sentence', 'NN'),
('.', '.'), ('This', 'DT'), ('is', 'VBZ'), ('another', 'RP'),
('sentence', 'NN'), ('.', '.')]
```

Un ejemplo de acceso a los datos del NLTK, en este caso a un corpus etiquetado del catalán.

```
>>> from nltk.corpus import cess_cat
>>> cess_cat.words()
['El', 'Tribunal_Suprem', '-Fpa-', 'TS', '-Fpt-', 'ha', ...]
>>> cess_cat.tagged_words()
[('El', 'da0ms0'), ('Tribunal_Suprem', 'np0000o'), ...]
```