Applied NLP for Information Retrieval

PRI 23/24 · Information Processing and Retrieval
M.EIC · Master in Informatics Engineering and Computation

Sérgio Nunes
Dept. Informatics Engineering
FEUP · U.Porto

Based on ???

Outline

- Introduction to NLP
- NLP Fundamentals
- Text Representation
- NLP Tools and Resources
- Practical Examples

Introduction to Natural Language Processing

NLP in Information Retrieval

- Natural Language Processing (NLP) is a field of artificial intelligence that focuses on the interaction between humans and computers through natural language.
- In the context of Information Retrieval (IR), NLP plays a key role in improving the effectiveness of search and retrieval systems.
  - Extraction of structured information from unstructured text, making it easier to index and search.
  - Help in understanding user queries and documents, improving relevance ranking.
  - Extraction of entities, relations, and sentiment from text, enabling richer retrieval experiences.

Key NLP Concepts for Information Retrieval

- **Tokenization**
  - Tokenization is the process of breaking text into individual words or tokens.
- **Part-of-Speech (POS) Tagging**
  - POS tagging assigns grammatical categories (e.g., noun, verb, adjective) to each word in a sentence.
- **Named Entity Recognition (NER)**
  - NER identifies and classifies entities such as names of people, places, organizations, and dates in text.
- **Syntactic Analysis**
  - Syntactic analysis involves parsing sentences to understand their grammatical structure.

NLP Terminology

- **Stemming and Lemmatization**
  - Stemming reduces words to their root form (e.g., "jumping" to "jump"), aiding in query expansion and document matching.
  - Lemmatization is a similar process that reduces words to their base or dictionary form (e.g., "better" to "good").
- **Semantic Analysis**
  - Semantic analysis explores the meaning of words and phrases, enabling the understanding of context.
  - It's essential for query understanding, disambiguation, and identifying synonyms.
- **Sentiment Analysis**
  - Sentiment analysis determines the sentiment (positive, negative, neutral) expressed in text.
  - In IR, sentiment analysis can be used for personalized content recommendation and sentiment-based filtering.
- **Information Extraction**
  - Information extraction involves identifying structured information (e.g., dates, events, relationships) within unstructured text.
  - It's valuable for creating structured databases from textual sources.

Levels of NLP Text Processing

- **Raw Text**: The initial unprocessed textual data, such as documents, articles, and web pages.
- **Tokenization**: The first level, responsible for segmenting raw text into individual tokens or words, facilitating further analysis.
- **Syntax Analysis**: At this stage, NLP parses sentences to grasp their grammatical structure, including parts of speech and sentence structure.
- **Semantic Analysis**: Going beyond syntax, this level explores the meaning of words and phrases in context, addressing synonyms, word sense disambiguation, and contextual nuances.
- **Named Entity Recognition (NER)**: NER identifies and categorizes entities like people, locations, organizations, and dates within text, enhancing information extraction.
- **Sentiment Analysis**: Sentiment analysis determines the emotional tone expressed in text, valuable for understanding attitudes and opinions.
- **Information Extraction**: NLP extracts structured information, such as dates, events, and relationships, from unstructured text, facilitating database creation.
- **Semantics and Meaning**: This level assembles all extracted information to comprehend the overall meaning of the text. It is crucial for tasks like query understanding and relevance ranking in Information Retrieval.

Levels of NLP Text Processing

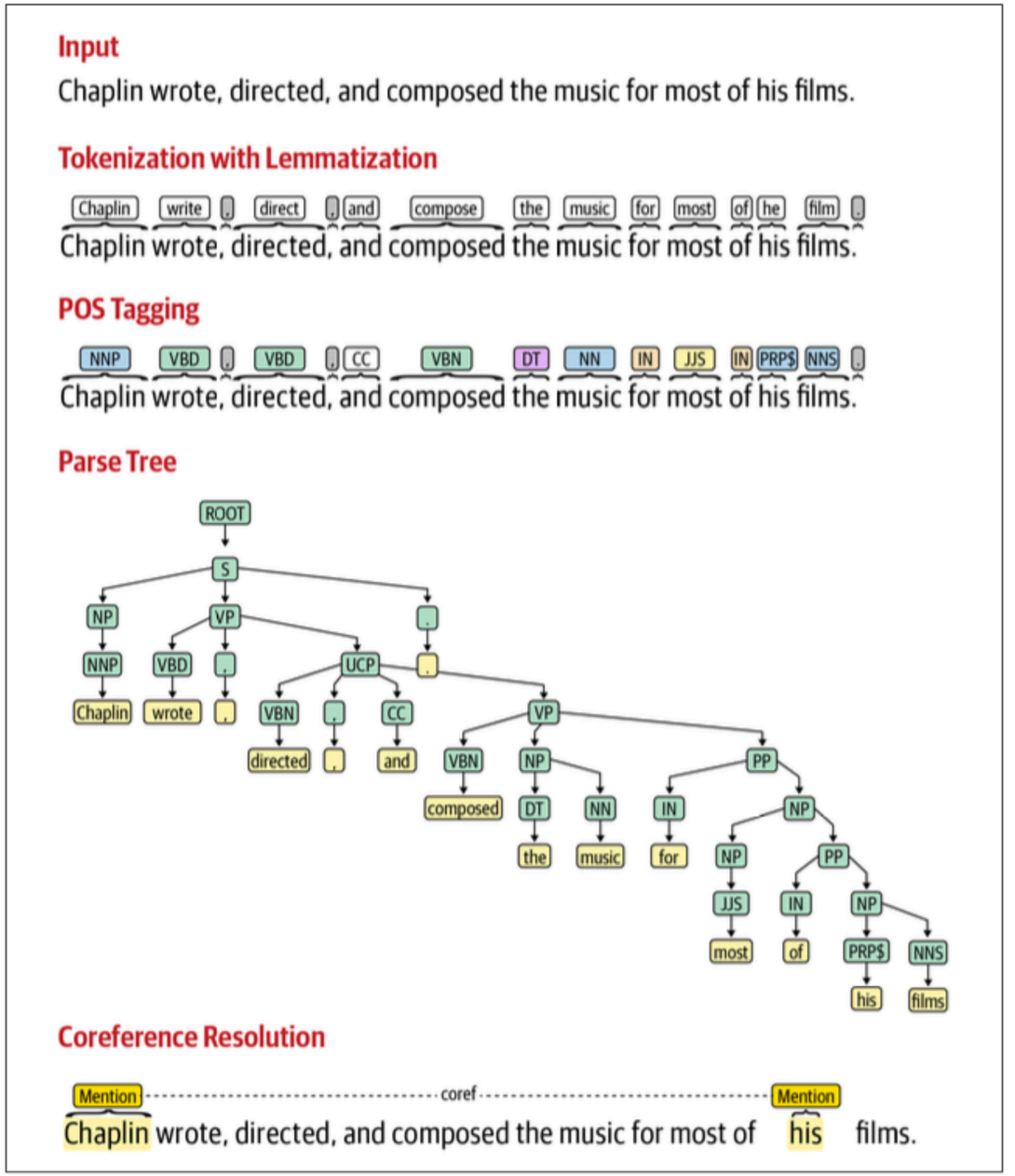


Figure 2-10. Output from different stages of NLP pipeline processing

NLP Fundamentals

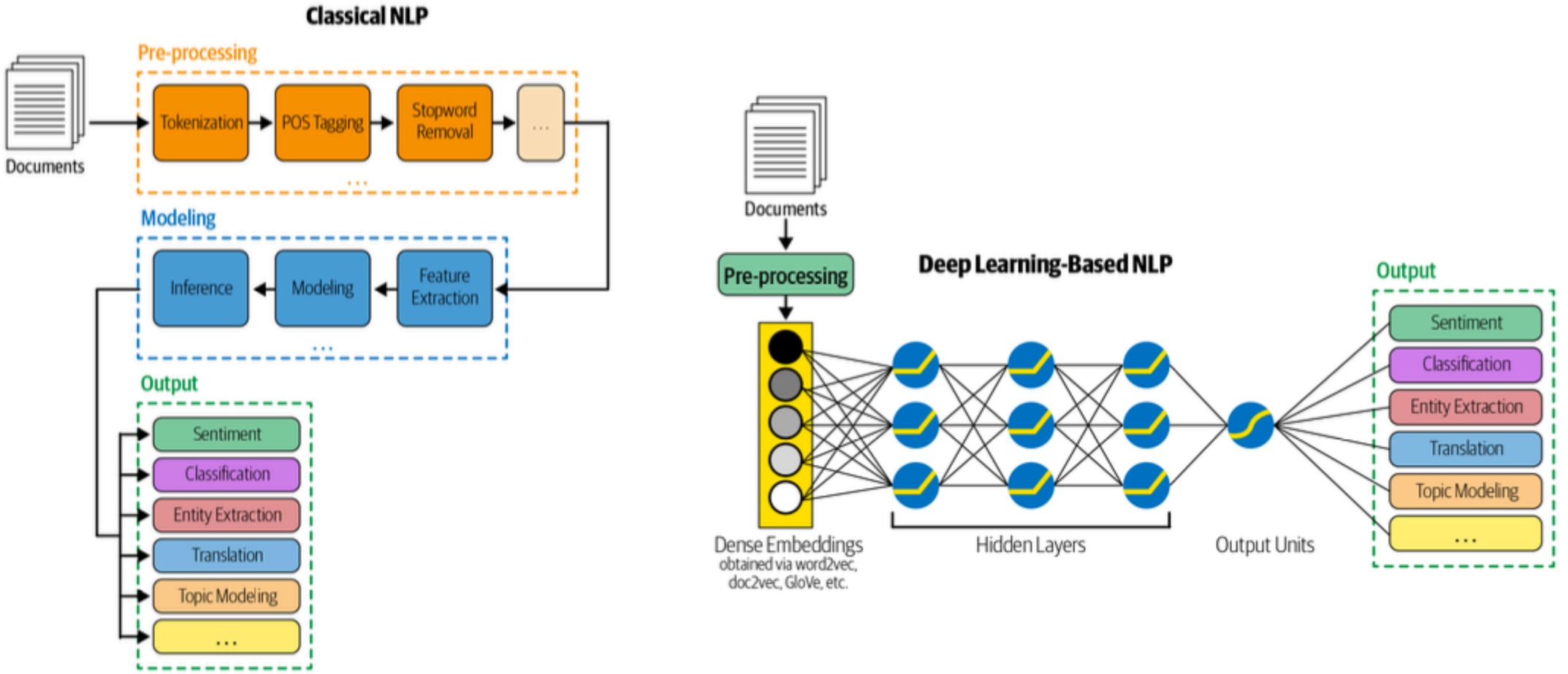
NLP Approaches: Rule-Based and Statistical

- **Rule-Based Approaches**
  - Rule-based NLP relies on predefined rules and patterns to analyze and manipulate text.
  - These rules are designed by linguists or experts.
  - Example: Email Address Extraction
    - Rule: Identify sequences of characters with "@" and "." to capture email addresses.
  - This approach is precise but may struggle with complex or evolving language patterns.
  - Is also costly and hard to scale.
- **Statistical Approaches**
  - Statistical models use probabilistic techniques to analyze language based on the statistical properties of large corpora of text.
  - Example: Language Model for Text Prediction
    - Model: Predict the next word in a sentence based on the probability distribution of word sequences.
  - Statistical approaches excel in tasks like speech recognition and machine translation.

NLP Approaches: Machine Learning and Deep Learning

- **Machine Learning Techniques**
  - Machine learning algorithms, both supervised and unsupervised, are applied to analyze and interpret text.
  - Example: Sentiment Analysis Using a Decision Tree
  - Model: Train a decision tree classifier on a labeled dataset to predict sentiment (positive, negative, neutral) in text.
  - Machine learning is effective for tasks like text classification and part-of-speech tagging.
- **Deep Learning**
  - Deep learning models, including neural networks, have transformed NLP by capturing complex language patterns.
  - Example: Machine Translation with a Transformer Model
  - Model: Implement a Transformer-based model to translate text from one language to another.
  - Deep learning excels in machine translation, language modeling, and sentiment analysis.

Classical NLP and Deep Leaning-Based NLP



Images from: Practical Natural Language Processing, O'Reilly (2020).

NLP Tasks

- The field NLP is organized into well-defined tasks, each addressing specific challenges in language understanding and processing. Some of the most prominent include:
- **Text Classification**: Assigning predefined categories or labels to text, often used for sentiment analysis, topic categorization, and spam detection.
- **Machine Translation**: Translating text from one language to another, a fundamental task in multilingual communication.
- **Speech Recognition**: Converting spoken language into written text, enabling voice assistants and transcription services.
- **Question Answering (QA)**: Automatically generating concise and accurate answers to user questions based on a given text corpus.
- **Information Extraction (IE)**: Identifying structured information (e.g., events, relationships) within unstructured text.
- **Summarization**: Creating concise summaries of long documents or articles, aiding in content digestion.
- **Language Generation**: Automatically generating human-like text, used in chatbots, content generation, and more.

NLP Tasks

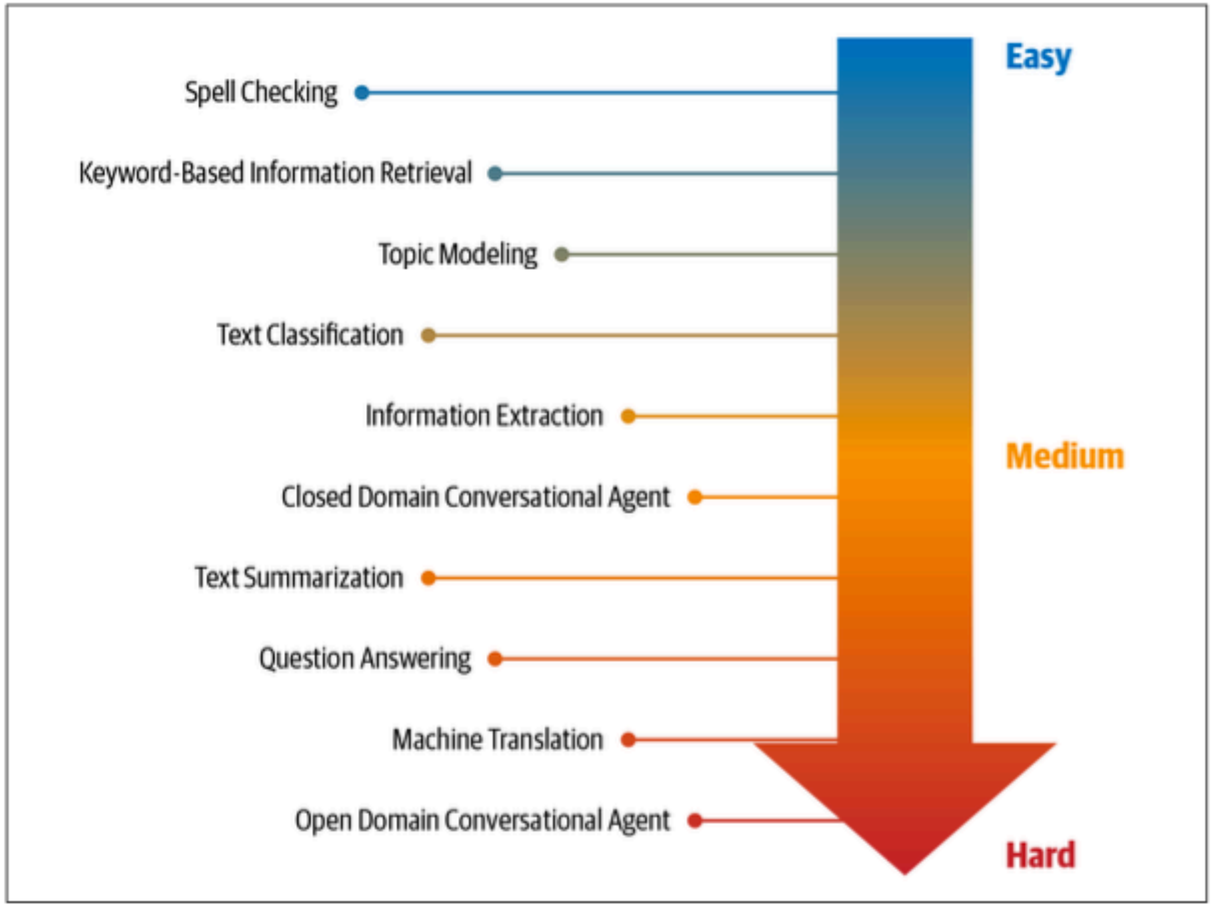


Figure 1-2. NLP tasks organized according to their relative difficulty

Image from: Practical Natural Language Processing, O'Reilly (2020).

Key NLP Tasks for Information Retrieval

- In the context of Information Retrieval, specific NLP tasks are particularly relevant for optimizing search and retrieval.
- **Tokenization**: The process of dividing text into tokens or words, which is fundamental for indexing and query processing in IR systems.
- **Stemming and Lemmatization**: Techniques that reduce words to their base forms, aiding in query expansion and document matching.
- **Part-of-Speech (POS) Tagging**: Assigning grammatical categories to words in a sentence, which helps in query parsing and understanding.
- **Named Entity Recognition (NER)**: Identifying and categorizing entities like people, locations, organizations, and dates within text, contributing to better information extraction.
- **Relation Extraction**: Determining relationships between entities mentioned in text, allowing for more precise retrieval.
- **Sentiment Analysis**: Analyzing the emotional tone expressed in text, which can be used to tailor search results based on sentiment.

NLP is Challenging

- NLP is a challenging problem domain.
- Language ambiguity
- Common knowledge
- Creativity
- Diversity across languages

Text Representation

Text Representation

- Text representation is a necessary first-step in an NLP process.

- **Frequency-Based Representations:**
  - Bag of Words (BoW)
  - Term Frequency-Inverse Document Frequency (TF-IDF)
  - N-grams

- **Semantic Representations**
  - Word Embeddings
  - Document Embeddings
  - Topic Models

- **Structure-Based Representations**
  - Syntax-Based Representations
  - Graph-Based Representations

Bag-of-Words (BoW)

- BoW represents text as an unordered collection of words, disregarding grammar and word order. BoW simplifies text into term frequency counts for each document.
  - “‘Good dog, good dog!’, said the quick brown fox."
  - BoW representation:
  - { "good": 2, "dog": 2, "said": 1, "the": 1, "quick": 1, "brown": 1, "fox": 1 }

- Advantages: simplicity, efficiency, and word independence.
- Disadvantages: loss of context, semantic gaps, and sparsity.

Term Frequency-Inverse Document Frequency (TF-IDF)

- The TF-IDF representation assigns a weight to each term in a document based on its frequency in the document and rarity across the entire corpus.
  - "Good dog, good dog!', said the quick brown fox."
  - TF-IDF representation for the document:
  - { "Good": 0.2, "dog": 0.3, "said": 0.15, "quick": 0.25, "brown": 0.3, "fox": 0.3 }

- Advantages: capture term importance, discriminates between term, and flexibility.
- Disadvantages: loss of context, semantic gaps, no word order, and sparsity.

N-grams

- N-grams represent text by considering sequences of n contiguous words.
- A “bag of n-grams”.
  - "'Good dog, good dog!’, said the quick brown fox."
  - N-grams representation with n=2 (bigrams):
  - { "'Good dog,", "dog, good", "good dog!’,", "dog!’, said", "said the", "the quick", "quick brown", "brown fox." }
- Advantages: contextual information, phrase detection, and improved semantics.
- Disadvantages: increased dimensionality, and size sensitivity.

Vector-Based

- Word representation
  - Map each word into a vector with the size of the vocabulary (i.e. one-hot encoding).
  - "'Good dog, good dog!', said the quick brown fox."
  - Vocabulary: [brown, dog, fox, good, quick, said, the]
    - good: [0, 0, 0, 1, 0, 0, 0]; dog [0, 1, 0, 0, 0, 0, 0]

- Document representation
  - Using word frequencies: [1, 2, 1, 2, 1, 1, 1]
  - Using normalized unitary vectors (later topic).

Word Embeddings

- Word Embeddings represent words as dense vectors in a continuous space, capturing semantic relationships.
- Document Embeddings represent entire documents as a single dense vector.

- Advantages:
  - Semantic Similarity: capture semantic relationships, enabling measurement of word similarity and analogy tasks (e.g., "king" - "man" + "woman" = "queen").
  - Reduced Dimensionality: typically have lower dimensionality compared to one-hot encodings, making them computationally more efficient.
- Challenges:
  - Data Intensity: training high-quality word embeddings requires large text corpora and computational resources.
  - Interpretable Features: interpreting the components of word embeddings can be challenging, as they are learned automatically.

NLP Tools and Resources

NLP Libraries and Tools

- **NLP libraries** provide pre-built functions and models for common NLP tasks.
  - NLTK, spaCy, Stanford NLP, Gensim, Hugging Face Transformers.

- **Pre-trained models** are neural network-based models that have learned from vast text corpora and can be fine-tuned for specific NLP tasks.
  - BERT, GPT-3, Word2Vec, FastText.

- **Pre-trained word embeddings** capture semantic relationships between words and are used as feature vectors in NLP tasks.
  - Word2Vec embeddings, GloVe embeddings.

Application Examples

NLP Pipeline

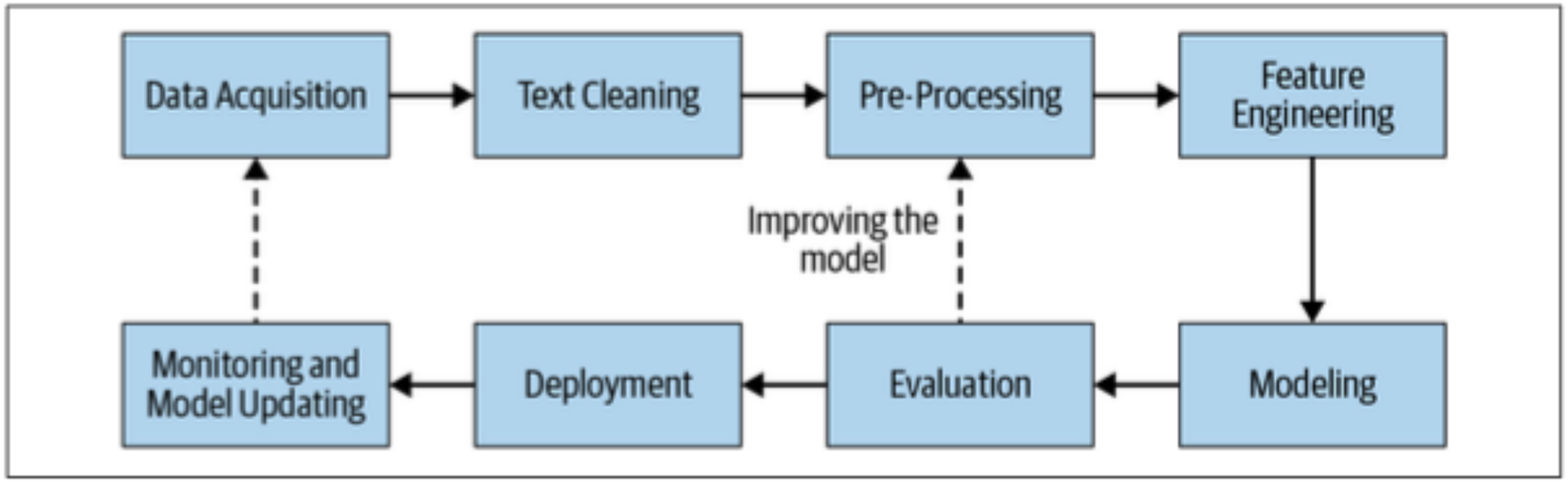


Figure 2-1. Generic NLP pipeline

Image from: Practical Natural Language Processing, O'Reilly (2020).

Pre-Processing Steps

- Preliminaries
  - Sentence segmentation and word tokenization.
- Frequent steps
  - Stop word removal, stemming and lemmatization, removing digits/punctuation, lowercasing, etc.
- Other steps
  - Normalization, language detection, code mixing, transliteration, etc.
- Advanced processing
  - POS tagging, parsing, coreference resolution, etc.

From: Practical Natural Language Processing, O'Reilly (2020).

Sentence and Word Segmentation

```
from nltk.tokenize import sent_tokenize, word_tokenize

# From Wikipédia.

mytext = "Processamento de
computação, inteligência ar
geração e compreensão autom
geração de língua natural c
computadores em linguagem c
de língua natural convertem
mais formais, mais facilmen
desafios do PLN são compree
extraiam sentido de linguag

my_sentences = sent_tokeniz

for sentence in my_sentence
    print("sentence: " + se
    print(word_tokenize(sen
```

```
sentence: Processamento de língua natural (PLN) é uma subárea da ciência da
computação, inteligência artificial e da linguística que estuda os problemas da
geração e compreensão automática de línguas humanas naturais.

['Processamento', 'de', 'língua', 'natural', '(', 'PLN', ')', 'é', 'uma', 'subárea',
'da', 'ciência', 'da', 'computação', ',', 'inteligência', 'artificial', 'e', 'da',
'linguística', 'que', 'estuda', 'os', 'problemas', 'da', 'geração', 'e',
'compreensão', 'automática', 'de', 'línguas', 'humanas', 'naturais', '.']

sentence: Sistemas de geração de língua natural convertem informação de bancos de
dados de computadores em linguagem compreensível ao ser humano e sistemas de
compreensão de língua natural convertem ocorrências de linguagem humana em
representações mais formais, mais facilmente manipuláveis por programas de
computador.

[‘Sistemas', 'de', 'geração', 'de', 'língua', 'natural', 'convertem', 'informação',
'de', 'bancos', 'de', 'dados', 'de', 'computadores', 'em', 'linguagem',
'compreensível', 'ao', 'ser', 'humano', 'e', 'sistemas', 'de', 'compreensão', 'de',
'língua', 'natural', 'convertem', 'ocorrências', 'de', 'linguagem', 'humana', 'em',
'representações', 'mais', 'formais', ',', 'mais', 'facilmente', 'manipuláveis',
'por', 'programas', 'de', 'computador', '.']
```

Stop Word Removal

```
from nltk.tokenize import sent_tokenize, word_tokenize
import nltk
nltk.download('punkt')
nltk.download('stopwords')

# From Wikipédia.

mytext = “…"

my_sentences = sent_tokenize(myte

stopwords = nltk.corpus.stopwords
print(stopwords[:10])

for sentence in my_sentences:
    print("sentence: " + sentence

    words = word_tokenize(sentenc

    # Print all words
    print("All tokens:", words)

     # Filter out stopwords and print non-stopwords
    non_stopwords = [word for word in words if word.lower() not in stopwords]
    print("Non-stopwords:", non_stopwords)
```

```
['a', 'à', 'ao', 'aos', 'aquela', 'aquelas', 'aquele', 'aqueles', 'aquilo', ‘as']

sentence: Processamento de língua natural (PLN) é uma subárea da ciência da
computação, inteligência artificial e da linguística que estuda os problemas da
geração e compreensão automática de línguas humanas naturais.

All tokens: ['Processamento', 'de', 'língua', 'natural', '(', 'PLN', ')', 'é',
'uma', 'subárea', 'da', 'ciência', 'da', 'computação', ',', 'inteligência',
'artificial', 'e', 'da', 'linguística', 'que', 'estuda', 'os', 'problemas', 'da',
'geração', 'e', 'compreensão', 'automática', 'de', 'línguas', 'humanas', 'naturais',
'.']

Non-stopwords: ['Processamento', 'língua', 'natural', '(', 'PLN', ')', 'subárea',
'ciência', 'computação', ',', 'inteligência', 'artificial', 'linguística', 'estuda',
'problemas', 'geração', 'compreensão', 'automática', 'línguas', 'humanas',
'naturais', '.']
```

Stemming and Lemmatization

- **Stemming** refers to the process of removing suffixes and reducing a word to some base form such that all different variants of that word can be represented by the same form.

- **Lemmatization** is the process of mapping all the different forms of a word to its base word, or lemma.

```
import nltk
nltk.download('wordnet')

from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()

print(lemmatizer.lemmatize("better", pos="a")) #a is for adjective

>>> good
```

```
import spacy
sp = spacy.load('en_core_web_sm')

token = sp(u'better')
for word in token:
    print(word.text, word.lemma_)

>>> better well
```

Entity Recognition

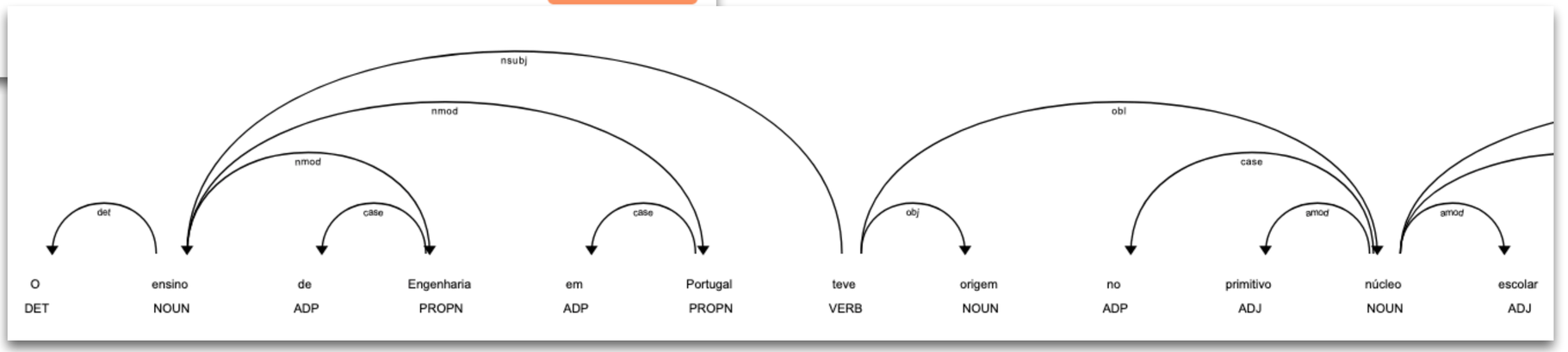
```python
import spacy

# Load the Portuguese NLP model
nlp = spacy.load("pt_core_news_sm")

# Input text in Portuguese
text = "O ensino de Engenharia em Portugal teve origem no primitiv
da Aula Náutica, decretado em 30 de julho de 1765. A cidade do Por
importante centro de navegação e comércio, tendo nascido rapidamen
mercantil nos seus habitantes. Mas vivia-se, nessa altura, uma sit
preocupante: o comércio estava a ser prejudicado pelos corsários,
nas praias do Norte de África, assaltavam os navios carregados de
Para resolver o problema, os Homens de Negócio da Praça do Porto p
licença para construir à sua custa duas fragatas de 24 a 30 peças
esquadras que da cidade saíssem para os portos da América. Com o i
construção, verificou-se a necessidade de apresentar pessoas capaz
manobrar as referidas naus. E daí surge, na cidade do Porto, a Aula de Náutica."

# Process the text using spaCy
doc = nlp(text)

# Iterate through the entities identified by spaCy
for ent in doc.ents:
    print(f"Entidade: {ent.text}, Entidade: {ent.label_}")
```

```
Entidade: Engenharia, Rótulo: LOC
Entidade: Portugal, Rótulo: LOC
Entidade: Aula Náutica, Rótulo: MISC
Entidade: cidade do Porto, Rótulo: LOC
Entidade: Norte, Rótulo: LOC
Entidade: África, Rótulo: LOC
Entidade: Homens de Negócio da Praça do Porto, Rótulo: MISC
Entidade: Rei, Rótulo: PER
Entidade: América, Rótulo: LOC
Entidade: cidade do, Rótulo: LOC
Entidade: Aula de Náutica, Rótulo: MISC
```

Text Visualization

- Explore spaCy's 'displaCy' module.

O ensino de Engenharia LOC em Portugal LOC teve origem no primitivo núcleo escolar da Aula Náutica MISC , decretado em 30 de julho de 1765. A cidade do Porto LOC era um importante centro de navegação e comércio, tendo nascido rapidamente o espírito mercantil nos seus habitantes. Mas vivia-se, nessa altura, uma situação preocupante: o comércio estava a ser prejudicado pelos corsários, que, escondidos nas praias do Norte LOC de África LOC , assaltavam os navios carregados de mercadorias. Para resolver o problema, os Homens de Negócio da Praça do Porto MISC pediram ao Rei PER licença para construir à sua custa duas fragatas de 24 a 30 peças para proteger as esquadras que da cidade saíssem para os portos da América LOC . Com o início da construção, verificou-se a necessidade de apresentar pessoas capazes de comandar e manobrar as referidas naus. E daí surge, na cidade do LOC Porto, a Aula de Náutica MISC .



Project Examples

NLP in PRI Projects

- In summary, for PRI, NLP can be an important tool to:
  - understand
  - characterize
  - clean
  - enrich
  - structure
  - entity linking

References

Real-World Natural Language Processing
Masato Hagiwara
Manning, 2021

Natural Language Processing in Action
Maria Dyshel and Hobson Lane
Manning, 2023

Practical Natural Language Processing
Vajjala S. et al.
O’Reilly, 2020