BASIC NLP CONCEPTS

Nandana Ranjish

The NLP Pipeline for Legal Text Processing

- In machine learning we begin doing complicated things by building a pipeline
- break up your problem into very small pieces -> use machine learning to solve each smaller piece separately -> chaining together several machine learning models that feed into each other
- An NLP pipeline is a series of steps that take raw text data and process it to make it understandable for a machine, transforming it into actionable insights or for use in an application
- By carefully adapting each component—sentence segmentation, tokenization, stop-word removal, lemmatization, POS tagging, and NER—to the unique characteristics of legal language, we can build systems that significantly enhance legal research, document analysis, and compliance monitoring.

1. SENTENCE SEGMENTATION

- Sentence segmentation (or sentence boundary detection) is the process of dividing a continuous stream of text into individual sentences.
- We first try to code so that the machine understand these individual segments rather than trying to translate the whole data.
- Modern NLP pipelines often use more complex techniques that work even when a document isn’t formatted cleanly.
- challenges: U.S., Smith v. Jones, See § 123.45(a). Should remain intact

2.TOKENIZATION

- Tokenization splits text into individual units (tokens), which can be words, punctuation, or symbols.
- Challenges: Preserve legal citations, Handle hyphenated terms, Maintain case names, Preserve parentheses in references
- Applications: Preparing text for legal search engines, Feature extraction for machine learning models

3. STOP-WORDS REMOVAL

- Stop-words removal eliminates common, low-information words that appear frequently but carry little semantic meaning. Contract interpretation: "shall" and "may" have legal significance
- When to be careful about stop-words removal: Negation contexts, Obligation words, Conditional terms, Legal-Specific Strategy
- Legal applications: Improving legal document search relevance, Reducing dimensionality for text classification, Enhancing topic modeling in case law

4. LEMMATIZATION

- Lemmatization reduces words to their base or dictionary form (lemma), considering the word's part of speech.
- Normalizing case law for comparison, Improving legal search recall
- Latin terms: "habeas corpus", "prima facie" should remain unchanged
- Possessive forms: "defendant's" → "defendant" but preserve meaning
- Legal phrases: "force majeure" should not be lemmatized separately
- Legally distinct forms: "breach" vs "breached" may need preservation in some contexts

5. PART-OF-SPEECH TAGGING

- assigns grammatical categories to each token (noun, verb, adjective, etc.).
- Legal-Specific POS Tags: Modal verbs, Proper nouns, Legal nouns, Conditional markers

6. NAMED ENTITY RECOGNITION

- NER identifies and classifies named entities in text into predefined categories.
- they are using the context of how a word appears in the sentence and a statistical model to guess which type of noun a word represents.
- The goal of *Named Entity Recognition*, or *NER*, is to detect and label these nouns with the real-world concepts that they represent.
- **Legal-Specific Entity Types: case name, legal document, law firm**
- Legal applications: Automated legal research, Case law analysis and summarization, Contract entity extraction, Legal document organization

Legal Domain Challenges

- Legal terms often have multiple interpretations
- Heavy use of citations and references
- Legal language evolves over time
- Training data scarcity
- Legal documents can be extremely long