NLP Pipeline

06 December 2025 11:51

Focus on the steps (not the sequence)

step # 1: Sentence Segmentation

paragraph

London is the capital and most populous city of England and the United Kingdom. Standing on the River Thames in the southeast of the island of Great Britain, London has been a major settlement for two millennia. It was founded by the Romans, who named it Londinium.

source: wikipedia

sentence segmentation

1. London is the capital and most populous city of England and the United Kingdom.
2. Standing on the River Thames in the southeast of the island of Great Britain, London has been a major settlement for two millennia.
3. It was founded by the Romans, who named it Londinium.

- it is the very first step in the NLP pipeline
= it divides the entire paragraph into different sentences for clarity and better understanding.

step # 2 Word tokenization:

What is a token?

In the context of large language models (LLMs) a token is a small unit of text that the model processes at once.

Sentence # 1 "I am learning" → 3 tokens: ["I", "am", "learning"]

every token is a word.

Sentence # 2 "Chatgpt is [illegible]

every token is a word.

Sentence # 2 "ChatGPT is powerful" → 4 tokens ["chat", "G", "PT", "is powerful"]

tokens ⇒ words

Unbelievable → 2 tokens ["un", "believable"]

word tokenization is the process of splitting a sentence or paragraph into individual words tokens (word tokens)

these are building blocks used for further analysis such as POS (parts of speech) tagging, sentiment analysis or even input to machine learning models.

Sentence: I love teaching NLP.

word tokens: ["I", "love", "teaching", "NLP"]

why is word tokenization important?

① Feature Extraction

- Many NLP algorithms work on the individual word and each distinct word should be tagged as ONE token even when it is repeating multiple times

① learning	② teaching	③	④

② Reducing Complexity

Breaking down a sentence or text into tokens such as words, group of words → it simplifies the processing tasks

[Text → millions of words → thousands of sentences]

How does LLM internally divide them? Will it follow any algorithm to make it happen?

intent
↓
(action)

Sub-word tokens

I am unhappy with the taxation rate.

(omg! it's such a positive sentiment) X → interpretation is WRONG!!

it's actually -ve sentiment not positive

- unhappiness → ["un", happiness"]
- unwell → ["un", "well"]

} sub-word tokens to understand the context better.

Rule-based tokenization

- it handles punctuations & contractions

Don't go ! → ["Don", "'", "t", "go", "!"]

I've cold.

We'll meet.

U.S.A. is a big country → ["U", ".", "S", ".", "A", "."]

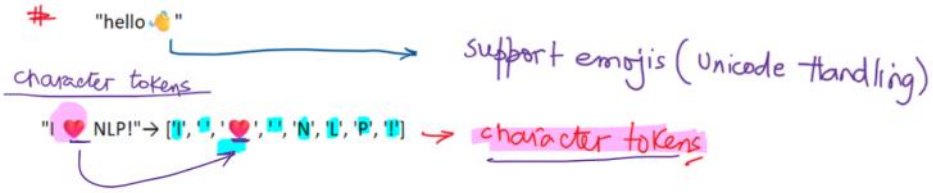
↳ Avoid splitting U.S.A.

↓

needs to be considered as one token.

ask2abc@gmail.com

one token



RegEx-based tokenizations

↳ Regular Expressions

- splits the text into tokens (words, phrases, characters, symbols) based on custom-pattern matching rules.

Regex	Meaning	Example Input	output
\W+	word characters	let's go!	["let", "s", "go"]
r '#\W+'	Hashtags	#AI is awesome	["#AI"]

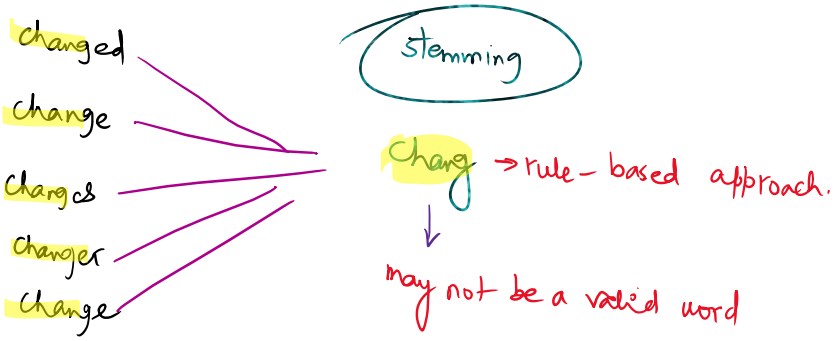
Note: • Highly customized for specific text formats
• No dependency on pretrained models → rule based tokenization
• Lightweight and fast

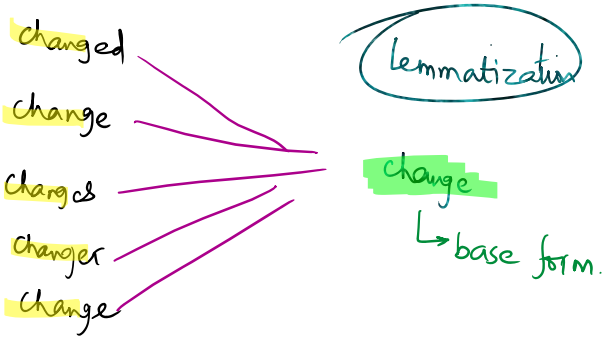
Cons: • Regex doesn't have linguistic awareness
• It's hard to handle context while doing Regex.

Pro-tip

Stemming vs Lemmatization

Both stemming & lemmatization are techniques used to normalize words by reducing them to their root or base form.





Stemming → is a rule-based approach or process of trimming the words to reduce them to their stem (root)

Note: Stemming may not produce words that are actual or valid dictionary words but share the same meaning root.

How does stemming work?

- uses simple heuristics (rules)

- uses simple heuristics (rules)
- trims the prefixes / suffixes
- ignores grammar and context

Porter Algorithm : is a stemming alogorithm used in NLP to reduce words to their root (stem) form
↳ Martin Porter in 1980

original word	stemmed word	lemmatized word
studying	studi	study
studies		
happier	happi	happy
organization	organ	organization
easily	easili	easy

 # lemmatization - it reduces words to their dictionary form called 'lemma' using morphological analysis and contextual pos tagging.

Note: It always returns valid words and handles word inflection and derivation properly.

good vs goods
advice vs advise } can be easily handled using lemmatization
noun verb

Criteria	Stemming	Lemmatization
Speed	Fast (rule-based truncation)	Slower (dictionary & POS lookup required)
Accuracy	Lower – may produce non-existent or incorrect base forms	Higher – produces correct base words (lemmas)
Grammar Awareness	Ignores grammatical context	Considers POS (Part-of-Speech) for accurate results
Output Example	studies → studi, better → better	studies → study, better → good (with POS)
Handling Errors	High chance of over-stemming or under-stemming	Very low – uses validated lemmas
Tool Examples	Porter, Snowball, Lancaster stemmers (from NLTK)	WordNet Lemmatizer (NLTK), spaCy, TextBlob, Stanza
Language Support	Primarily English	Supports multiple languages (esp. with spaCy, Stanza)
POS Requirement	Not required	Required for optimal performance
Data Use-case	Search engines, indexing, real-time filtering	Chatbots, text summarization, grammar checking
Output Validity	May produce roots not present in the dictionary (e.g., argu, easili)	Always yields valid dictionary words
Customization	Custom rules difficult to apply	Easier to customize via lexical resources
Ideal For	High-speed needs, low precision tolerance (e.g., search filters)	NLP pipelines needing high precision (e.g., translation, inference, NLU)

Stop words Removal

Stopwords Analysis

Stopwords are commonly used words in a language that carry little to no meaningful information in text analysis.

Such words are typically ignored during text pre-processing because they don't contribute significantly to the overall context or meaning of the text.

Articles: a, an, the

Prepositions: in, on, at, with

Pronouns: he, she, it, we, they

Why do we remove stopwords?

① Reduce Dimensionality

By removing stopwords, the no of unique words (features) in the dataset is reduced making

(features) in the dataset is reduced making computationally faster and less memory intensive.

② Improves the models' efficiency

Removing the noise (stopwords) helps the models' efficiency by making the learning process efficient.

Do not remove stop words

Sentiment Analysis – I am not happy with the taxation rate.

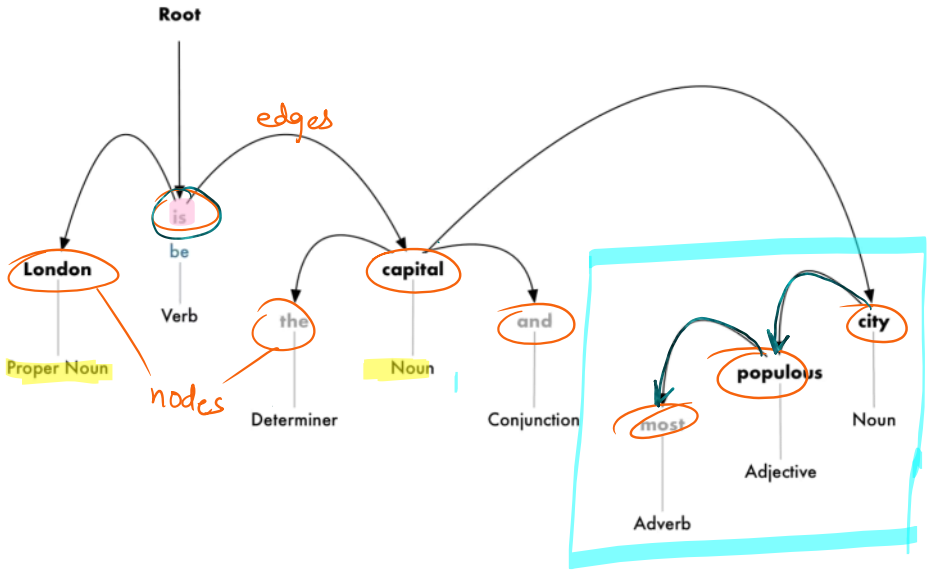
can't remove the word 'not'.

NLG Tasks: for tasks where stopwords provide meaningful context or structure such as natural language generation tasks, machine translation or question-answering — keep the stopwords in general.

keep the stop words or drop selectively

Dependency Parsing

Sentence: London is the capital and most populous city



Dependency parsing is a process in NLP used to analyze the grammatical structure of a sentence. It identifies syntactic relationships b|w words in a sentence, often in the form of a tree or graph structure.

Note: There are nodes (words) and edges (grammatical relationships) in dependencing parsing.

POS: Parts of speech tagging

It is the process of assigning a grammatical label such as noun, verb, adverb, adjective etc to each word in a sentence based on it's role and context

(not an exhaustive list)

POS Tags	Meaning	Example
NOUN	Person, place, thing	dog, car, London
PRON	Pronoun	he, she, it
VERB	Action/state	run, is, has
ADJ	Describes nouns	big, red, fast
ADV	Describes verbs/adjectives	quickly, very
DET	Determiner	the, a, an
ADP	Preposition	in, on, by
CONJ	Conjunction	and, but, or

(not an exhaustive list)

POS Tags	Meaning	Example
NOUN	Person, place, thing	dog, car, London
PRON	Pronoun	he, she, it
VERB	Action/state	run, is, has
ADJ	Describes nouns	big, red, fast
ADV	Describes verbs/adjectives	quickly, very
DET	Determiner	the, a, an
ADP	Preposition	in, on, by
CONJ	Conjunction	and, but, or
NUM	Number	one, twenty
PART	Particle	to (as in "to go"), not
INTJ	Interjection	wow, ouch, hey

Lemmatization ← with pos

Named Entity Recognition (NER) ← pos tagging

NER is a framework to do fundamental task in NLP that involves identifying and classifying named entities in text into pre-defined categories

Entity Type	Example
PERSON	Elon Musk, Mahatma Gandhi
ORGANIZATION	OpenAI, Google, United Nations
LOCATION	New York, India, Himalayas
DATE	18 May 2025, yesterday, next month
TIME	5:00 PM, noon
MONEY	$1 million, ₹500
PERCENT	30%, 10 percent
GPE (Geo-Political Entity)	USA, France, Maharashtra
PRODUCT	iPhone, ChatGPT, Toyota Prius
EVENT	Olympics, World War II
LAW	GDPR, Indian Penal Code
LANGUAGE	Hindi, Python, Spanish
WORK OF ART	Mona Lisa, Hamlet

Microsoft will open a new office in Electronics City, Bangalore

(ORG: organization)

Facility or site

GPE: Geo-Political Entity