DATA PRE-PROCESSING IN NLP

STARTER GUIDE

swipe right 

LOWER CASING

Transfer all characters to lowercase

- It removes unecessary variation
- No differenece between "Hi" and "hi"
- It reduces the vocabulary size and thus time and compute required

DON'T DO IF

- you want your model to recognize abbrevations
- Your downstream task is NER
- If upper-case denotes a special format

swipe right 

NOISE REMOVAL

Removing data that is unecessary and adds noise

- Remove special characters and punctuations as they don't add any information
- Remove HTML tags from web data
- Also check for leading, trailing and consecutive spaces
- Remove numbers if they are not required for your task.
- Focuses your model on data that has content
- Reduces vocabulary size

"Noise" depends on your data and problem

- There may be cases where special characters hold meaning, for emails, etc
- Recommended to manually review a small subset of your data

swipe right 

STOPWORD REMOVAL

Removing common words which carry low value

- Words like "the","a","and" are used in every sentence but don't add any context
- 60% of your raw data would be these stop-words which contribute to noise in your data
- Helps your model focus on content-rich words
- Reduces your vocabulary

BUT

- It can cause issues in some cases. In "New york", 'new' shoudln't be removed as it adds context.
- For real-time scenarios, create your own stop-word list and remove those from the data

swipe right 

Tokenization

Converting data into structured tokens

- Splitting text into units.
- A unit can be a word, character, group of words etc.
- Makes it easy for the model to analyze
- I made a seperate post on tokenization. Link in comments.

swipe right 

Stemming

Reducing words to their root form

- "walking", "walked","walkes" ,"walker"-> "walk"
- Removes the ending part of root words
- Different variations of a root are the same to the model
- Reduces vocabulary size
- Computationally inexpensive

BUT

- Linguistically inaccurate. Reduce unrelated words to same.
- "Universal", "University" are stemmed to "Univers" .

swipe right 

Lemmatization

Sophisticated version of stemming

"studies" -> "study"; "is","are","was" -> "be"

- Context aware compared to stemming
- reduced words to their lemma (base form)
- Utilizes dictionary to understand word structure
- Preferred for most NLP tasks

BUT

- Slower than Stemming

Don't apply both stemming and lemmatization as it is unecessary and can add noise

swipe right 

VECTORIZATION

Converting text to numbers

- Machines can only understand numbers
- Each input text needs to be converted to a vector of numbers

Count-based (simple)

- One-hot encoding
- Bag-of-words
- TF-IDF

Embeddings (Complex)

- Word2Vec
- GloVe
- FastText

PS: Will make a seperate post on these

swipe right 

- **NLTK** and **spaCY** are popular python libraries for NL pre-processing and analysis
- Follow these steps need to be followed in that order and the data is now ready to be passed on to a model
- **And follow me fore more content like this**