Libraries: NLTK; SPACY

Text – Unstructured Data

<https://towardsdatascience.com/introduction-to-natural-language-processing-for-text-df845750fb63>

NLP > NLU > NLG: <https://blog.marketmuse.com/whats-the-difference-between-nlp-nlu-and-nlg/>

NLU has Lexical, Syntactic, Referential Ambiguity

Lexical – A word may have different meanings

Syntactic – A sentence/phrase can have different meanings

Referential – Refer to something use pronouns. The mother scolded her child. She is upset. “She” is referring to who?

NLP? Natural Language Processing – AI method for communicating with an intelligent agent using Natural Language.

Word embedding. Word embeddings are **representations of words as vectors.**

**1 word = 1 vector.** Each vector has a value (weight). The value is used to represent meaning of word. **Semantically similar words have similar vectors.**

Tokenization – Breaking down complex sentence into words

|  |  |
| --- | --- |
| Information Extraction | Rule-Based – Extract information from text using the pre-defined set of rules (syntax/grammar/vocab)  <https://www.analyticsvidhya.com/blog/2019/09/introduction-information-extraction-python-spacy/> |
| Named Entity Recognition (NER) | Locate and **Classify Named Entities** in the text into pre-defined categories e.g. Name of person/Organization/Location.  Steps:  <https://towardsdatascience.com/named-entity-recognition-with-nltk-and-spacy-8c4a7d88e7da>  <https://medium.com/@b.terryjack/nlp-pretrained-named-entity-recognition-7caa5cd28d7b> |
| Sentiment Analysis (Classification) | People express emotions in their text.  So, we want to **Interpret** and **Classify** emotions within the text. Convert the unstructured data into a structured data.  Words or Sentences have Positive/Negative/Neutral sentiments,  e.g. “I like it, great job!” > Positive Sentiment  “It sucks and it is the worst one” > Negative Sentiment  “It is okay I guess” > Neutral Sentiment  Rule-Based – Perform analysis based on manual set of rules. The set of rules can be e.g. “negative words list”, “positive words list”. If text has more positive words than negative, text = positive.  Automatic – Relies on ML to learn from data. Convert text input to **feature vector** using **feature extractor (bag of words/bag of ngrams/word vector)**. Train the model to associate **feature vectors** with **TAGS** (Positive/Negative/Neutral). Then predict.    <https://monkeylearn.com/sentiment-analysis/> |
| Machine Translation | e.g. Google Translate  Automatically converting one natural language into another, while preserving the meaning/grammar of the text.  Rule-Based Machine Translation (RBMT) – Uses built-in linguistic rules and bilingual dictionaries for each language pair. e.g. (Enligsh>Chinese)  Statistical Machine Translation (SMT) – Uses **statistical models** that learns to translate text from SOURCE -> TARGET language (bilingual text corpora <pls google corpora). Maximizes the **probability** of the output sequence given the input sequence. P(Output|Input).  Neural Machine Translation (NMT) – Uses **neural network models to learn statistical models** for machine translation. It builds a **SINGLE** large neural network that reads in a sentence and outputs the translated sentence.  Sequence 2 Sequence, Encoding and Decoding  NMT used in modern day.  <https://machinelearningmastery.com/introduction-neural-machine-translation/>  <https://towardsdatascience.com/machine-translation-a-short-overview-91343ff39c9f>  Steps in python:  <https://www.analyticsvidhya.com/blog/2019/01/neural-machine-translation-keras/> |
| Text Summarization | Creating short, accurate, summary of a long text.    Extractive – **Selects key phrases/sentences from the source** to make new summary  Abstractive – **Generates NEW phrases/sentences** to interpret the meaning of the source  Steps in python:  <https://blog.floydhub.com/gentle-introduction-to-text-summarization-in-machine-learning/> |
| Chatbot | e.g. Think website chatbots  Rule-Based/Scripted – Only can respond to certain questions  Pattern Matching (ML)– e.g. “QN: Can I know the price of x” will be matched to the template of “ANS: **The price of x is …**”  Uses pre-generated templates, still limited questions.  NLP – Natural Language Understanding (NLU) > Natural Language Generation (NLG)  Steps:  Tokenize > Normalize > Recognize Entities > Dependency Parsing > Generate  <https://towardsdatascience.com/how-to-build-a-chatbot-a-lesson-in-nlp-d0df588afa4b>  <https://medium.com/analytics-vidhya/building-a-simple-chatbot-in-python-using-nltk-7c8c8215ac6e> |
| Speech Recognition | e.g. (Google/Siri/Cortana)  Processing the speech, converting it to text (Sound > Text)  NLP is the ability to understand the meaning of the text    <https://www.analyticsvidhya.com/blog/2019/07/learn-build-first-speech-to-text-model-python/> |
| Spell Checking | Requires text corpus |
| Keyword Extraction |  |
| Topic Segmentation |  |
| Advertisement Matching | Application  <https://blog.marketmuse.com/natural-language-processing-for-content-marketers/> |

Text input > Tokenization > Stop word filter > Negation Handling > Stemming > Classification > Sentiment Class