Lecture 6 - NLP 2

ELIZA – The Early Chatbot

ELIZA, developed by Joseph Weizenbaum in 1966, is one of the earliest examples of a chatbot that simulates conversation by using pattern matching and substitution techniques.

- **Core Mechanism:** ELIZA analyzes input sentences using decomposition rules triggered by keywords and then generates responses based on reassembly rules.
  - Identification of key words
  - Extraction of minimal context
  - Transforming input into appropriate responses
    ELIZA laid the groundwork for subsequent developments in conversational agents and remains a seminal work in NLP history.

> **ELIZA Technical Description - Weizenbaum (1966)**
>
> "ELIZA is a program [...] which makes certain kinds of natural language conversation between man and computer possible. Input sentences are analyzed on the basis of decomposition rules which are triggered by key words..."

Evolution of NLP Approaches

NLP has undergone significant transformations since its inception:

- **Rule-Based Approaches (1950s–1980s):**
  Early systems relied on handcrafted rules to process language.
- **Statistical Approaches (1980s–2010s):**
  The advent of machine learning introduced probabilistic models that improved language understanding.
- **Deep Learning Approaches (2010s–Present):**
  Neural networks, such as recurrent and transformer-based models, have revolutionized the field with their ability to learn complex patterns from vast datasets.

Approach Type	Characteristics	Example Applications
Rule-Based	Handcrafted rules, deterministic	Early chatbots, simple parsers
Statistical	Probabilistic models, relies on large corpora	Machine translation, tagging
Deep Learning	Neural network architectures, data-intensive	Modern chatbots, sentiment analysis

> **Advantages vs. Disadvantages**
>
> Rule-based systems are interpretable but inflexible, while deep learning models require large datasets and computing power but can generalize better.

Text Analytics: Concepts and Applications

Definition: The use of machine learning and statistical techniques to uncover patterns and trends in textual data.

Text analytics is the process of deriving high-quality information from text.

- **Applications:**
  - Analyzing customer reviews
  - Monitoring social media sentiment
  - Extracting key themes from documents
  - Facilitating automated translation and summarization
- **Core Tasks:**
  - **Language Detection:** Identifying the language of a given text using APIs and libraries.
  - **Named Entity Extraction:** Identifying names of people, places, and organizations.
  - **Sentiment Analysis:** Classifying text as positive, negative, or neutral.
  - **Text Summarization:** Reducing texts to their essential points.

> **Text Analytics Definition**
>
> "Text data mining applies machine learning and statistical methods to texts to discover useful patterns."

Language Identification in Text Analytics

Language identification is fundamental for processing multilingual text:

- **Purpose:** Determines the language of a text to facilitate subsequent processing like translation or sentiment analysis.
- **Tools and APIs:**
  - IBM Language Translator
  - Microsoft Azure Translator
  - Google Translate Language Detection
  - Python libraries (e.g., langdetect, NLTK examples)
- **Process:** The text is analyzed to match linguistic features against known language models.

> **Proper language detection is essential, as misidentification can lead to cascading errors in further NLP tasks.**

Sentiment Analysis

Sentiment analysis aims to determine the emotional tone behind a text:

- **Process:**
  - The text is divided into sentences or phrases.
  - Each segment is analyzed for sentiment using a combination of sentiment libraries and rule-based scoring.
- **Examples:**
  - Classifying a product review as positive or negative.

    - Analyzing social media posts to gauge public opinion.
- **Challenges:**
    - Handling negations (e.g., "not good").
    - Differentiating subtle emotional nuances.

> **Sentiment Analysis with NLTK**
>
> "Sentiment analysis involves breaking text into parts, scoring each, and aggregating these scores for an overall sentiment."

> **VADER - Hutto & Gilbert (2014)**
>
> "VADER is a parsimonious rule-based model for sentiment analysis of social media text."

Text Summarization Techniques

Text summarization reduces a large text into a concise version that retains the key information:

- **Two main approaches:**
    - **Extraction:** Directly selecting important sentences from the text.
    - **Abstraction:** Generating a new summary that conveys the critical content in a rephrased manner.

> **Extraction vs. Abstraction**
>
> Extraction is simpler as it selects verbatim sentences, whereas abstraction requires deeper semantic understanding to generate new text.

- **Workflow Example:**
    1. **Sentence Segmentation:** Convert paragraphs into sentences.
    2. **Preprocessing:** Clean and normalize the text.
    3. **Tokenization:** Break sentences into words.
    4. **Frequency Analysis:** Identify weighted word frequencies.
    5. **Sentence Scoring:** Replace words with frequency scores and rank sentences.
    6. **Summary Generation:** Combine top-ranked sentences into a final summary.

> **Summarization Example**
>
> "Text summarization extracts the most relevant parts of a text to create a concise version that still conveys the original message."

Preprocessing Techniques in NLP

Effective text analysis relies on several core preprocessing steps:

- **Tokenization:** Breaking text into words or sentences.
- **Stop Words Removal:** Eliminating common words that add little semantic value.
- **Normalization:** Converting text to a uniform format (e.g., lowercasing, removing punctuation).
- **Stemming and Lemmatization:** Reducing words to their base or root form.
- **Part-of-Speech Tagging:** Identifying the grammatical roles of words.
- **Named Entity Extraction:** Detecting and classifying key entities such as names, locations, and dates.

How do preprocessing steps improve the effectiveness of NLP applications?

Preprocessing removes noise and standardizes text, thereby enhancing the accuracy and efficiency of subsequent analyses.
