Introduction to Natural language Generation

① What is Natural Language Generation?

Definition and overview:-

Natural language generation (NLG) is a branch of AI that generates human-like text from non-linguistic data, such as numbers, databases, or structured inputs. Unlike other ai tasks, NLG focuses on producing text that is coherent, meaningful, and contextually appropriate.

Analogy:- Imagine NLG as a storyteller who takes raw facts (like a list of events or numbers) and weaves them into a compelling narrative. Just as storyteller turns a list of historical events into an engaging tale, NLG transform data into sentences that humans can easily understand. Think of it like a chef turning raw ingredients (data) into a delicious dish (text).

Why NLG matters for scientists:-

As a budding scientist, NLG is powerful because it:-

- Communicate complex data:- Turns raw research data into readable summaries.
- Automates Tasks:- Saves time by generating reports or documents.
- Explore AI creativity:- Helps you study how machines mimic human language. NLG is used in chatbots, automated journalism, and scientific reporting, making it a key area for advancing AI research and communication.

② NLP vs NLG

Key difference :-

Natural Language processing (NLP) and NLG are related but distinct :-

- NLP (Natural language processing) :-
  - Focus :- Understanding and interpreting human language
  - Tasks :- Sentiment analysis, text classification, machine translation, named entity recognition
  - Example :- Analyzing a movie review to determine if it's positive or negative.
  - Analogy :- NLP is like a detective who reads and decodes clues from text.
- NLG (Natural language generation) :-
  - Focus :- Generating human-like text from data.
  - Tasks :- Creating reports, writing stories, generating dialogue.
  - Example :- Producing a weather forecast from temperature and humidity data.
  - Analogy :- NLG is like a writer who crafts a new story from raw ideas.

How they work together :-

NLP and NLG are complementary. For example :-

- In a chatbot, NLP interprets the user's query ("What's the weather?"), and NLG generates the response ("It's sunny with a temperature of 25°C")
- In summarization, NLP extracts key points from a document, and NLG rephrases them into a concise summary.

① Key Components of NLG

NLG system follows a pipeline with distinct stages. These components are the foundation of how NLG works.

② Data Input and Processing :-

NLG starts with raw data, such as :-

- Numerical Data :- E.g. sales figures ($5000, 20% increase)
- Structured data :- Eg. a database with customer names and purchase.
- Unstructured data :- Eg. sensor logs or user inputs.

Preprocessing involves cleaning and organizing data such as removing outliers, handling missing values, or standardizing formats.

Example :- For weather data {temp : 25.6, condition : sunny}, preprocessing might round the temperature to 26°C for simplicity.

③ Content planning :-

This stage decide what to say :-

- selects relevant information from the data.
- organizes it logically (e.g. in a weather report, start with temperature, then condition).

example :- From {temp : 25, condition : sunny, humidity : 60}, the system might select temperature and condition as key facts.

④ Sentence planning and Realization :-

This stage determine how to say it :-

- Sentence Planning :- Choose words and sentence structures (e.g., "It's running" vs. "The weather is sunny today").

- Realization:- Ensures grammatical correctiveness and proper styles.

Example:- Mapping {temp: 25} to "temperature is 25°C" with correct grammers.

9 Evaluation:- The generated text is evaluated for:-
- Fluency:- Does it sound natural and readable?
- Accuracy:- Does it correctly represent the input data.
- Relevance:- Is it appropriate for the context.

Example:- The output "It is a sunny day with a temperature of 25°C" is fluent, accurate, and relevant.

Q How NLG works: The process:-
1. Data input:- Recieve raw data (e.g. {temp: 25, condition: sunny})
2. Content selection:- Identify key facts (e.g., temperature first, then condition).
3. Content structuring:- Organize facts logically (e.g. temperature first, then condition).
4. Sentence planning:- Map facts to natural language (e.g., "The temperature is 25°C").
5. Surface Realization:- Add grammer and style (e.g., "It's a sunny day with a temperature of 25°C").
6. Output:- Deliver the final text.

③ Mathematical foundations of NLA

NLA relies on probabilistic models and neural networks. Let's :-

1. Probability and language model :-

Language models predict the likelihood of a word or phrase given the context. For example, in "The cat is ___", a model might predicting "sleeping" is more or less are more likely than "flying".

N-gram models :- These calculates the probability of a word based on the previous n-1 words. In a bigram model (n=2), the probability depends on the previous word.

Formula :-

$$P(w_n | w_{n-1}) = \frac{\text{Count}(w_{n-1}, w_n)}{\text{count}(w_{n-1})}$$

Example Calculation : Bigram probability :-

Consider a small corpus :-

- "The cat is sleeping"
- "The cat is eating"

Calculate the probability of "is" given "cat" :-

- Count of "cat" is 2.
- Probability :-

$$P(\text{sleeping} | \text{is}) = \frac{\text{Count}(\text{is sleeping})}{\text{Count}(\text{is})}$$

$$= \frac{1}{2} = 0.5$$

These prob. help NLA systems choose words that form coherent text.

Date
Page

⑤ Neural networks in NLG:-

Modern NLG use transformers, powerful neural networks for modeling language. A Transformers consists of:

- Encoder:- Understands the input data (e.g. {temp: 25, condition: sunny}).
- Decoder:- Generates the output text (e.g. "It's a sunny day.").
- Attention mechanism:- focuses on relevent parts of input when generating each word.

Example:- A transformer encodes {temp: 25, condition: sunny} as a vector, and the decoder generates "It's sunny day with a temperature of 25°C" by predicting words sequentially.

⑥ Type of NLG system:-

NLG systems vary in complexity and approach. Here are the three main types:-

1. Rule-Based Systems:- These use predefined templates and rules to map data to text.
   - How it works:- Uses templates like "The temperature is [TEMP]°C" to generate text.
   - Pros:- Simple, accurate for structured data, easy to control.
   - Cons:- Limited flexibility, struggles with creative or complex text.
   - Example:- Generating financial reports with fixed formats.

- Statistical Systems :- These use statistical models like n-grams to generate text based on word probabilities.
- How it works :- predicts the next word based on the frequency of word sequences in a corpus.
- Pros :- More flexible than rule-based systems.
- Cons :- Less coherent than neural systems, require large datasets.
- Example :- Early chatbots or machine translation systems.

- Neural Systems :-

These use neural networks, particularly Transformers (e.g., GPT, BERT), to generate text.
- How it works :- Trained on massive text datasets to learn language patterns, then fine tuned for specific tasks.
- Pros :- Highly fluent, creative, and adaptable to various contexts.
- Cons :- Computationally expensive, potential for bias.
- Example :- Modern chatbots like Cyrok or story generators.