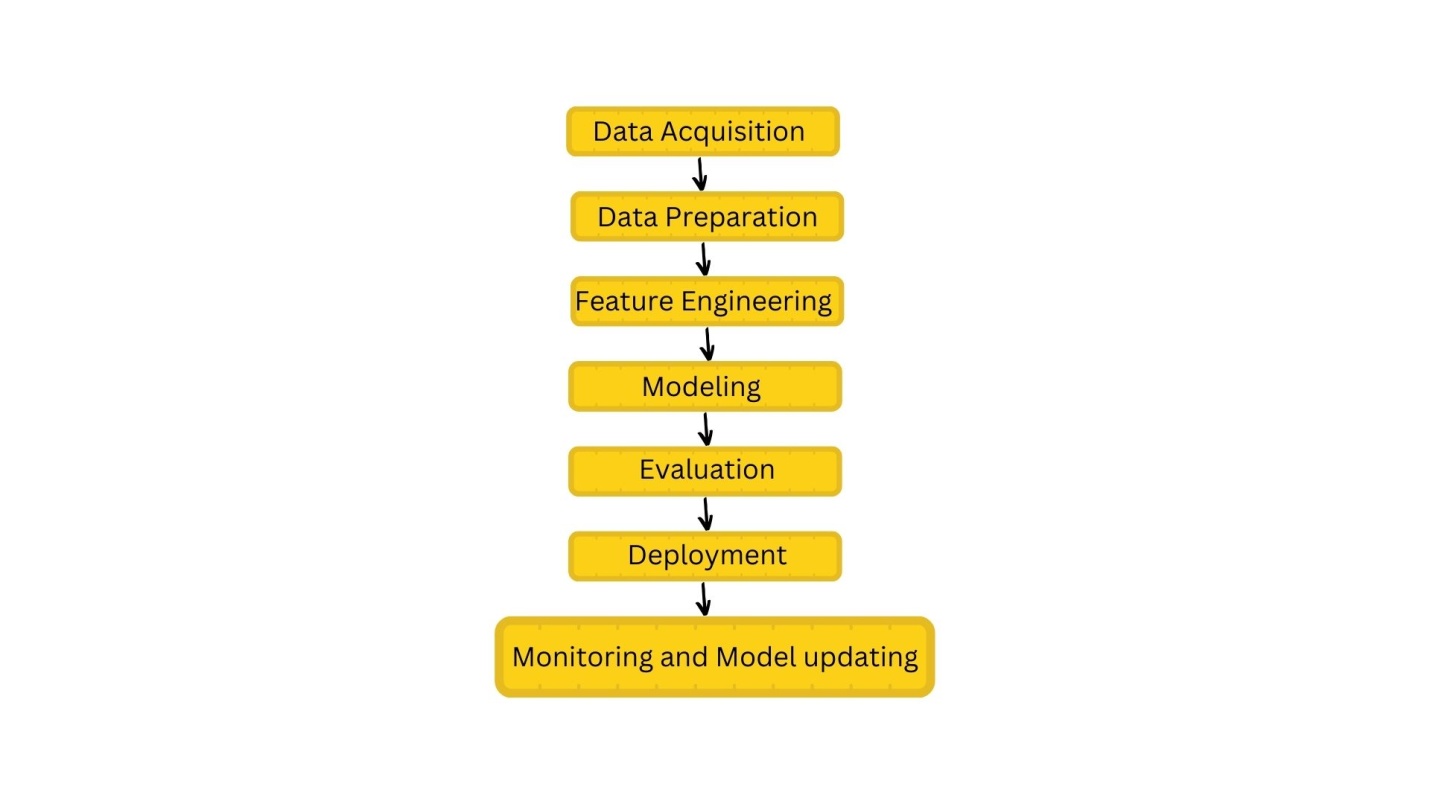
**End to End Pipeline of Generative AI**

1. **Data Acquisition**
2. **Data Preparation**
3. **Feature Engineering**
4. **Modeling**
5. **Evaluation**
6. **Deployment**
7. **Monitoring and Model updating**

****

**1. Data Acquisition**

The first step is to gather relevant data for training. Data can come from various sources:

* **Available Data**: Structured files like CSV, text, PDF, DOCX, and XLSX formats.
* **Other Data**: Collected from APIs, databases, web scraping, or the internet.
* **No Data**: If no data is available, it can be generated using LLMs (Large Language Models) like OpenAI’s GPT. For instance, by prompting models to generate sentences.

**Data Augmentation:**

When the data volume is insufficient, augmentation techniques are applied to increase the dataset size artificially:

* **Synonym Replacement**: Substituting words with their synonyms.
  + E.g., "I am an AI Engineer" → "I am a Data Scientist."
* **Bigram Flip**: Reordering phrases.
  + E.g., "I am Malavika" → "Malavika is my name."
* **Back Translation**: Translating a sentence to another language and back to the original.
  + E.g., "Data augmentation is the process of artificially generating new data" translated to another language and back to English might become, "Data mining is the process of routinely generating new data."

**Adding Noise:**

* **Additional Data/Noise**: Adding extra information or slight variations to existing sentences to create variations.
  + E.g., "I am an AI engineer. I love this job." → "I love my job as an AI engineer."

**2. Data Preparation/Preprocessing**

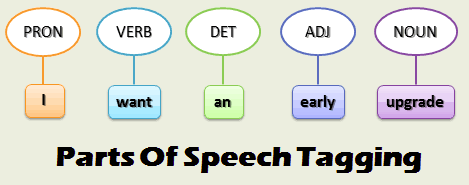
This stage involves cleaning and preparing data for model input:

**Basic Preprocessing:**

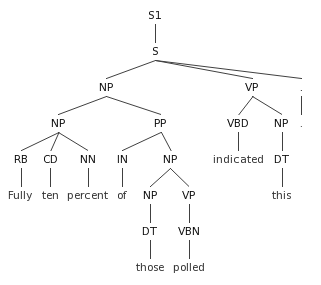
1. **Tokenization**: Splitting text into smaller units like words or sentences.
   * Word Tokenization: "My name is Malavika" → ['My', 'name', 'is', 'Malavika']
   * Sentence Tokenization: "My name is Malavika. I am an AI engineer." → ['My name is Malavika', 'I am an AI engineer']
2. **Stemming**: Reducing words to their base form (root).
   * E.g., "play, played, playing" → "play"
3. **Lemmatization**: A more advanced version of stemming that returns readable root forms.
4. **Punctuation Removal**: Removing unnecessary punctuation marks (e.g., /, ?).
5. **Lowercase Conversion**: Converting text to lowercase to maintain uniformity.
6. **Language Detection**: Identifying the language of the text.

**Advanced Preprocessing:**

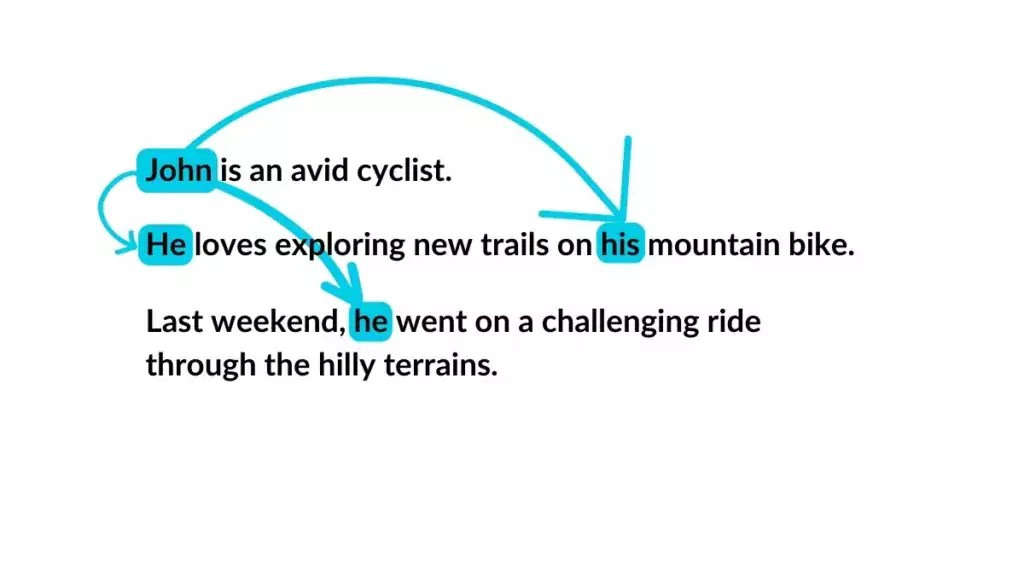
1. **Part-of-Speech (POS) Tagging**: Labeling words based on their grammatical roles.

****

1. **Parsing Trees**: Analyzing the grammatical structure of sentences.

****

1. **Coreference Resolution**: Identifying which words refer to the same entities (e.g., resolving pronouns like "he" or "it").



#### 3. ****Feature Engineering****

Transforming raw text into numerical representations suitable for machine learning models.

* **TF-IDF Vector**: Measures word importance relative to a document and across a corpus.
* **Count Vector**: Counts the occurrences of words.
* **Hashing Vector**: Uses hash functions to convert text to numerical representations.
* **Bag of Words**: Represents text as a collection of words without considering order.
* **One-Hot Encoding**: Represents individual words as binary vectors.
* **Transformer Models**: Advanced deep learning models like BERT or GPT to convert text into dense vectors.
* **Word2Vec**: Converts words into dense numerical vectors that capture semantic meanings.

#### 4. ****Modeling****

You can either use open-source or paid models depending on the resources available.

* **Open-source LLM**: These require infrastructure (CPU/GPU, memory) but can be run locally.
* **Paid LLM (e.g., OpenAI)**: No infrastructure needed, just API integration.

#### 5. ****Evaluation****

Evaluation involves two aspects:

* **Intrinsic Evaluation**: Metrics evaluated by the generative AI engineer (e.g., perplexity, BLEU score, etc.).
* **Extrinsic Evaluation**: Feedback collected from users in real-world deployment.

#### 6. ****Deployment****

This phase involves deploying the model to production. It requires setting up systems for continuous monitoring and periodic updates:

* **Monitoring**: Track the model's performance, ensuring it meets desired metrics and outcomes.
* **Retraining**: Periodically retrain the model with new data or when performance degrades.

### Common Terms:

1. **Corpus**: The entire collection of sentences or texts used.
2. **Vocabulary**: Unique words present in the corpus.
3. **Documents**: Individual files (e.g., DOCX, PDF) containing text.
4. **Words**: Single units of meaning arranged in order to create text.

This pipeline provides an end-to-end process for building generative AI systems, from gathering data to deploying the model and maintaining its performance.