- Text and Web Mining
  - Text Mining and Its Applications
    - Overview
    - Types of Data
    - Importance
    - Applications of Text Mining
    - Text Analysis Techniques
    - Natural Language Processing (NLP)
  - Text Analytics
    - Overview
    - Need for Text Analytics
    - Traditional Text Mining Process
  - Text Preprocessing
    - Overview
    - Noise Removal
    - Preprocessing Stages
    - NLP Pipeline Steps
  - Text Transformation using BoW and TF-IDF
    - Bag-of-Words (BoW)
    - Vector Space Modeling
    - Term Frequency-Inverse Document Frequency (TF-IDF)
  - Dimensionality Reduction
    - Introduction
    - Curse of Dimensionality
    - Benefits of Dimensionality Reduction
  - Techniques for Dimensionality Reduction
    - Feature Selection
      - 1. Variance Thresholds
      - 2. Correlation Thresholds
      - 3. Genetic Algorithms
      - 4. Stepwise Regression
    - Feature Extraction
      - 1. Linear Discriminant Analysis (LDA)
      - 2. Principal Component Analysis (PCA)
      - 3. t-distributed Stochastic Neighbor Embedding (t-SNE)
      - 4. Autoencoders
  - Web Mining
    - Overview
    - Features of Web Mining
    - Web Mining Tasks
    - Applications of Web Mining
  - Types of Web Mining
    - Web Content Mining
    - Web Structure Mining
    - Web Usage Mining
  - Mining Multimedia Data on the Web

  - Automatic Classification of Web Documents
  - Check Your Progress-1
  - Check Your Progress-2

Text and Web Mining

Text Mining and Its Applications

Overview

- **Text mining**: Process of transforming unstructured text into a structured format to identify patterns and insights.
- **Techniques**: Naïve Bayes, Support Vector Machines (SVM), deep learning algorithms.
- **Purpose**: Discover hidden relationships within unstructured data.

Types of Data

- **Structured data**:
  - Standardized tabular format with rows and columns.
  - Examples: names, addresses, phone numbers.
- **Unstructured data**:
  - No predefined format.
  - Examples: social media text, product reviews, videos, audio files.
- **Semi-structured data**:
  - Blend of structured and unstructured formats.
  - Examples: XML, JSON, HTML files.

Importance

- **80% of data** in the world is unstructured.
- **Text mining tools** and **NLP techniques** transform unstructured data into structured formats for analysis.
- **Benefits**: Improved decision-making and business outcomes.

Applications of Text Mining

- **Customer service**:
  - Use feedback systems (chatbots, surveys, online reviews, support tickets, social media).
  - Combine with text analytics tools for better customer experience.
  - Prioritize key pain points, respond to urgent issues, increase customer satisfaction.
- **Risk management**:
  - Monitor industry trends and financial markets.
  - Extract information from analyst reports and whitepapers.
  - Valuable for banking institutions.
- **Maintenance**:
  - Provide insights into product and machinery operations.

    - Automate decision-making by revealing patterns for maintenance procedures.
    - Help professionals find root causes of challenges and failures faster.
- **Healthcare**:
    - Valuable for biomedical research.
    - Automate extraction of valuable information from medical literature.
- **Spam filtering**:
    - Filter and exclude spam emails to improve user experience and reduce cyber-attack risks.

Text Analysis Techniques

- **Information Extraction**:
    - Extract domain-specific information from texts.
    - Map text fragments to field or template slots with definite semantics.
- **Text Summarization**:
    - Identify, summarize, and organize related text.
    - Help users deal with large documents efficiently.
- **Text Categorization**:
    - Organize documents into a taxonomy.
    - Assign subject descriptors or classification codes to complete texts.
- **Text Clustering**:
    - Automatically group documents with common features.

Natural Language Processing (NLP)

- **Definition**: AI method of communicating with intelligent systems using natural language.
- **Applications**:
    - Organize massive chunks of textual data.
    - Perform automated tasks like summarization, translation, speech recognition, and topic segmentation.

Text Analytics

Overview

- **Text mining vs. Text analytics**:
    - Text mining: Focuses on the process.
    - Text analytics: Focuses on the result.
- **Purpose**: Transform text data into high-quality information or actionable knowledge.
- **Sub-set of NLP**:
    - Automates extraction and classification of actionable insights.
    - Works with unstructured text from emails, tweets, chats, tickets, reviews, and survey responses.

Need for Text Analytics

- **Maintain Consistency**:
    - Manual tasks are repetitive, tiring, and error-prone.
    - Cognitive bias can hinder consistency in data analysis.

    - Advanced algorithms ensure quick, rational, reliable, and consistent data analysis.
- **Scalability**:
    - Process enormous data from social media, emails, chats, websites, and documents.
    - Improve business efficiency with more structured information.
- **Real-time Analysis**:
    - Real-time data evaluation is crucial.
    - Detect and address urgent matters promptly.
    - Monitor and automate flagging of tweets, shares, likes, and sentiments indicating urgency or negativity.

Traditional Text Mining Process

1. **Text preprocessing**:
    - Initial preparation of text data.
2. **Text Transformation (attribute generation)**:
    - Generate attributes from text.
3. **Feature Selection (attribute selection)**:
    - Select relevant attributes.
4. **Data Mining**:
    - Apply data mining techniques to the processed text.
5. **Evaluation**:
    - Assess the results and insights generated.

Text Preprocessing

Overview

- **Purpose**: Clean and prepare text data for specific contexts.
- **Uses**: Essential in NLP pipelines (voice recognition, search engines, machine learning models).
- **Goal**: Reduce text to only the necessary words for NLP goals.

Noise Removal

- **Text cleaning**: Remove unwanted information based on the project's goal and data source.
- **Types of noise**:
    - Punctuationts and accents
    - Special characters
    - Numeric digits
    - Leading, ending, and vertical whitespace
    - HTML formatting

Preprocessing Stages

- **Stemming, Lemmatization, and Normalization**: Standardize vocabulary size and form.

NLP Pipeline Steps

- **Common steps**:
  - Sentence segmentation
  - Word tokenization
  - Lowercasing
  - Stemming or lemmatization
  - Stop word removal
  - Spelling correction
  - Normalization
  - **Segmentation**
    - **Definition**: Breaking text into sentences.
    - **Challenges**: Periods in abbreviations and fractional numbers can create uncertainty.
  - **Tokenization**
    - **Definition**: Breaking text into smaller components (tokens).
    - **Uses**:
      - Counting words or sentences
      - Finding specific words or phrases
      - Identifying co-occurring terms
    - **Tokens**: Usually words, but can be sentences or other text pieces.
  - **Normalization**
    - **Common tasks**:
      - Uppercasing or lowercasing
      - Stop word removal
      - Stemming
      - Lemmatization
    - **Change Case**
      - **Lowercasing**: Common in NLP software for consistency.
    - **Spell Correction**
      - **Purpose**: Correct spelling errors in text.
    - **Stop-Words Removal**
      - **Stop words**: Frequently occurring words (e.g., is, the, are).
      - **Purpose**: Remove redundant words for specific NLP applications.
    - **Stemming**
      - **Definition**: Converting words to their base form (stem).
      - **Uses**: Search engines, emotion identification, text classification.
    - **Lemmatization**
      - **Definition**: Advanced stemming converting words to their root form (lemma).
      - **Advantages**: Considers parts of speech and context.
      - **Example**:
        - Stemmer: right (both "turn right" and "always right")
        - Lemmatizer: right (direction) and correct (rightness)
  - **Parts of Speech Tagging**
    - **Definition**: Augment text with grammatical structure information.
    - **Categories**: Noun, verb, adjective, etc.
    - **Alternate name**: Grammatical tagging

Text Transformation using BoW and TF-IDF

Bag-of-Words (BoW)

- Converts text into numerical format.
- Helps machines read and analyze text data.

-Example Reviews:

- Review 1: "This movie is very scary and long"
- Review 2: "This movie is not scary and is slow"
- Review 3: "This movie is spooky and good"

-Vocabulary:

- Unique words: 'This', 'movie', 'is', 'very', 'scary', 'and', 'long', 'not', 'slow', 'spooky', 'good'

-Vector Representation:

- Review 1: [1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0]
- Review 2: [1, 1, 2, 0, 1, 1, 0, 1, 1, 0, 0]
- Review 3: [1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1]

-Drawbacks of BoW:

- Vocabulary size increases with new words.
- Results in sparse matrices.
- No information on grammar or word order.

Vector Space Modeling

- Treats each distinct term as a dimension.
- Document D: "He is neither a friend nor is he a foe"
  - M = 10, w3 = "neither"
  - Term space: V = {"He", "is", "neither", "a", "friend", "nor", "foe"}
- **Vector Representation**:
  - D||B = (2, 2, 1, 2, 1, 1, 1)

Term Frequency-Inverse Document Frequency (TF-IDF)

- Reflects word importance in a document within a corpus.
- **Term Frequency (TF)**:
  - Measures how frequently a term, t, appears in a document, d.
    - TF = (Number of times term appears in document) / (Total number of terms in document)
- **Example TF for Review 2:**
  - Vocabulary: 'This', 'movie', 'is', 'very', 'scary', 'and', 'long', 'not', 'slow', 'spooky', 'good'
  - Number of words in Review 2: 8
    - TF('this') = 1/8
    - TF('movie') = 1/8
    - TF('is') = 2/8 = 1/4
    - TF('very') = 0/8 = 0

        - TF('scary') = 1/8
        - TF('and') = 1/8
        - TF('long') = 0/8 = 0
        - TF('not') = 1/8
        - TF('slow') = 1/8
        - TF('spooky') = 0/8 = 0
        - TF('good') = 0/8 = 0
- **Inverse Document Frequency (IDF):**
    - Measures word importance across the corpus.
        - IDF = log(Total number of documents / Number of documents containing the word)
- **Example IDF for Review 2:**
    - IDF('this') = log(3/3) = 0
    - IDF('movie') = log(3/3) = 0
    - IDF('is') = log(3/3) = 0
    - IDF('not') = log(3/1) = 0.48
    - IDF('scary') = log(3/2) = 0.18
    - IDF('and') = log(3/3) = 0
    - IDF('slow') = log(3/1) = 0.48
- **TF-IDF Calculation for Review 2:**
    - TF-IDF('this') = TF('this') * IDF('this') = 1/8 * 0 = 0
    - TF-IDF('movie') = 1/8 * 0 = 0
    - TF-IDF('is') = 1/4 * 0 = 0
    - TF-IDF('not') = 1/8 * 0.48 = 0.06
    - TF-IDF('scary') = 1/8 * 0.18 = 0.023
    - TF-IDF('and') = 1/8 * 0 = 0
    - TF-IDF('slow') = 1/8 * 0.48 = 0.06
- **TF-IDF Benefits:**
    - Higher scores for important, less frequent words.
    - Highlights significant words in a document.

Dimensionality Reduction

Introduction

- **Dimensionality**: Number of input features, variables, or columns in a dataset.
- **Dimensionality Reduction**: Process of reducing the number of random variables or attributes in a dataset.
- **Importance**: Essential in data pre-processing for real-world applications, especially to address the "Curse of Dimensionality."

Curse of Dimensionality

- **Definition**: Handling high-dimensional data is challenging.
- **Problems**:
    - Increased complexity of machine learning algorithms.
    - Higher chance of overfitting, leading to poor performance.
- **Solution**: Reduce the number of features to improve model performance.

Benefits of Dimensionality Reduction

- Reduces storage space required for datasets.
- Decreases computation time for training.
- Helps in visualizing data quickly.
- Removes redundant features and addresses multicollinearity.

Techniques for Dimensionality Reduction

Feature Selection

- **Objective**: Omit features that do not contribute to class separability.

1. Variance Thresholds

- **Method**: Remove features with low variance across observations.
- **Pros**: Easy and safe way to start reducing dimensions.
- **Cons**: Subjective; requires manual tuning.

2. Correlation Thresholds

- **Method**: Remove one of the features if they are highly correlated.
- **Pros**: Intuitive.
- **Cons**: Subjective; requires manual tuning and is less preferred compared to algorithms like PCA.

3. Genetic Algorithms

- **Method**: Inspired by evolutionary biology; find an optimal binary vector representing feature inclusion.
- **Pros**: Efficient in traversing large solution spaces.
- **Cons**: Complex and computationally intensive.

4. Stepwise Regression

- **Method**: Add or remove variables based on statistical tests.
- **Types**:
  - **Forward Selection**: Start with no features and add one at a time.
  - **Backward Elimination**: Start with all features and remove one at a time.
- **Pros**: Automated variable selection.
- **Cons**: Lower performance compared to supervised methods.

Feature Extraction

- **Objective**: Create a new, smaller set of features that capture most of the useful information.

1. Linear Discriminant Analysis (LDA)

- **Method**: Uses multiple features to create a new axis that maximizes class separability.
- **Pros**: Effective for labeled data.
- **Cons**: Requires normalization; supervised method.

2. Principal Component Analysis (PCA)

- **Method**: Identifies relationships among features, transforms data, and retains principal components.
- **Steps**:
  i. Identify relationships through a Covariance Matrix.
  ii. Perform eigen-decomposition to get eigenvectors and eigenvalues.
  iii. Transform data using eigenvectors into principal components.
  iv. Quantify importance using eigenvalues and keep significant components.
- **Pros**: Creates orthogonal, uncorrelated features.
- **Cons**: Only useful for linearly correlated variables; requires normalization.

3. t-distributed Stochastic Neighbor Embedding (t-SNE)

- **Method**: Non-linear technique for visualizing high-dimensional datasets.
- **Applications**: NLP, speech processing.
- **Pros**: Effective for visualizing complex data.
- **Cons**: Computationally intensive; not suitable for all types of data.

4. Autoencoders

- **Method**: Neural networks trained to reconstruct original inputs.
- **Components**:
  - **Encoder**: Compresses input data, removing noise.
  - **Decoder**: Reconstructs original input from compressed form.
- **Pros**: Effective for non-linear transformations.
- **Cons**: Requires a lot of data for training; complex to implement.

Web Mining

Overview

- Web mining involves mining web data using data mining techniques.
- Extracts information from websites, including:
  - Hyperlinks
  - Text or content
  - User activity across web pages

Features of Web Mining

- Web search engines (Google, Yahoo, MSN, etc.)
- Different from relational data: includes text content and linkage structure
- Rapidly increasing user-generated data (e.g., Google's usage logs)
- Real-time reactions with dynamic patterns
- Web server logs identify loyal or potential customers
- Web pages as graphs:
  - Nodes: pages
  - Edges: hyperlinks

- Directed graph with high linkage (8-10 links/page on average)

Web Mining Tasks

1. Generate patterns on websites (e.g., customer buying behavior).
2. Retrieve faster results for search queries.
3. Classify web documents to enhance business transactions.

Applications of Web Mining

- Personalized customer experience in B2C
- Web search
- Web-wide tracking
- Understanding web communities and auction behavior
- Personalized portals and recommendations (e.g., Netflix, Amazon)
- Improving conversion rates and advertising (e.g., Google AdSense)
- Fraud detection
- Enhancing website design and performance

Types of Web Mining

Web Content Mining

- Extract useful information from web document contents.
- Includes text, images, audio, video, and structured records.
- Text mining includes:
  - Topic discovery
  - Extracting association patterns
  - Clustering web documents
  - Classifying web pages

Web Structure Mining

- Discovering structure information from the web.
- Two kinds based on structure information:
  - Hyperlinks (intra-document and inter-document)
  - Document structure (HTML/XML tags forming tree-structured format)

Web Usage Mining

- Applying data mining to web usage data to understand user behavior.
- Types of usage data:
  - Web server data (IP address, page reference, access time)
  - Application server data (business events in server logs)
  - Application level data (custom events and histories)

Mining Multimedia Data on the Web

- Web multimedia data: video, audio, images, graphs.
- Multimedia data has different characteristics and retrieval methods.
- Techniques:
  - **PageRank**: Measures page importance based on connectivity.
  - **HITS**: Rates pages using hubs and authorities.
  - **Page Layout Analysis**: Maintains relationships from link structure.
  - **VIPS Algorithm**: Segments pages into blocks for better content aggregation.
  - **Block-level Link Analysis**: Useful for web image retrieval and categorization.

Automatic Classification of Web Documents

- Categorizes web pages into subjects or domains.
- Issues:
  - Constructing models for classification is a mammoth task.
  - Large number of unorganized pages may have redundant documents.
- Automated classification based on textual content.
- Process:
  - Collect documents from various sources.
  - Data cleansing using extraction, transformation, and loading.
  - Group documents by similarity and TF-IDF.
  - Create and execute machine learning models to generate clusters.
- Benefits:
  - Efficient and accurate classification.
  - Reduces operational costs.
  - Easy data storage and retrieval.
  - Organizes files and documents effectively.

Check Your Progress-1

1. Define structured, un-structured and semi-structured data with some examples for each.
2. Differentiate between Text Mining and Text Analytics.

Check Your Progress-2

1. What are the techniques to analyze the web usage pattern?
2. What are the other applications of Web Mining which were not mentioned?
3. What are the differences between Block HITS and HITS?
4. List some challenges in Web Mining.