Module 1 : Introduction

1.1 Introduction to Text Mining

- Text mining is an evolving field driven by advances in hardware and software, enabling the rapid generation & storage of vast amount of text data, particularly from web platforms and social networks. Unlike structured data managed by databases, text data often lacks structure and is processed via search engines. While information retrival focuses on connecting users with relevant data, text mining aims to analyze and discover patterns within text for decision-making and insight generation.
- Text mining is "Process of extracting meaningful information from unstructured text data.
- Key characteristic of text data include:

- High Dimensionality and Sparsity: Text data is often represented in large, sparse team-document matrices.
- Varied Representation Levels: Text can be analyzed as a "bag-of-words" or as semantically rich entities and relationships. Current methods often rely on simpler representations due to challenges in achieving robust semantic analysis
- Heterogeneous Domains: Text often appears alongside other data types (e.g. images, multimedia) or across lang., making cross-domain mining relevant.

1.1.1 Algorithms for text mining.

Text mining involves diverse techniques to extract insights from unstructured text. Key areas include:

1) Information Extraction: Identifies entities and relationships within text, moving beyond simple word representations to semantic analysis.

2) Text Summerization: Generates consise summeries of text, either by extraction (selecting key text) or abstraction (synthesizing new text).

3) Unsupervised Learning: • Clustering - Groups documents into topics.
• Topic Modeling - Probablistically assigns documents to topics for more flexible clustering and dimension reduction

4) Dimensionality Reduction: Techniques like Latent Semantic Indexing (LSI) reduce data dimensions and improve semantic insights, mitigating issues like synonymy and polysemy. Probabilistic topic modeling (eg. LDA) enhance representation.

5) Supervised Learning: Uses labeled data (training) for classification and prediction tasks. Include transfer learning for domains with labeled data.

6) Mining Text Streams: Addresses challenges of continuously processing large text streams from sources like social media or news feeds under real-time constrains.

7) Cross-Lingual and Multimedia mining: Cross lingual & techniques enable clustering and knowledge between languages. Multimedia mining integrates text with other media like images and videos.

8) Application-Specific Techniques: Text mining is tailored for domains like social media, opinion mining.

1.1.2 Future Directions

The field of text mining continues to expand with several promising areas of research and development:

1) Scalable and Robust Natural Language Understanding

- More beyond "bag-of-words" to semantic representation
- Develop robust information extraction techniques that require minimal training data.

2) Domain Adaptation and Transfer Learning:

- Address the scarcity of labeled data by leverging knowledge from related domain.
- Focus on cross-domain learning for heterogeneous data like multimedia and multilingual corpora.

3) Dynamic and Real-Time Mining:

- Develop methods to handle large-scale, dynamic streams of text data from platforms like social media.

4) Integration with Emerging Technologies:

- Combine text mining with advancements in AI, machine learning and multimodel processing for richer insights.

Text mining is multidisciplinary field intersecting with NLP, data mining & ML, offering broad application across industries like business intelligence, healthcare & social media analytics.

Subject: Text, Web and Social Media Analytics

1.2 INFORMATION EXTRACTION FROM TEXT

Information extraction refers to the process of automatically extracting structured information from unstructured text. The goal is to transform raw text data into a form that is easier to analyze and process.

- Key tasks in IE :

1) Named Entity Recognition (NER)
2) Relation Extraction
3) Unsupervised Information Extraction

1.2.1 Named Entity Recognition (NER)

NER is a specific subtask of IE that focuses on identifying named entities in text. Named entities are typically proper names that refer to specific objects, individuals, locations, organization etc.

Example of Named Entities.

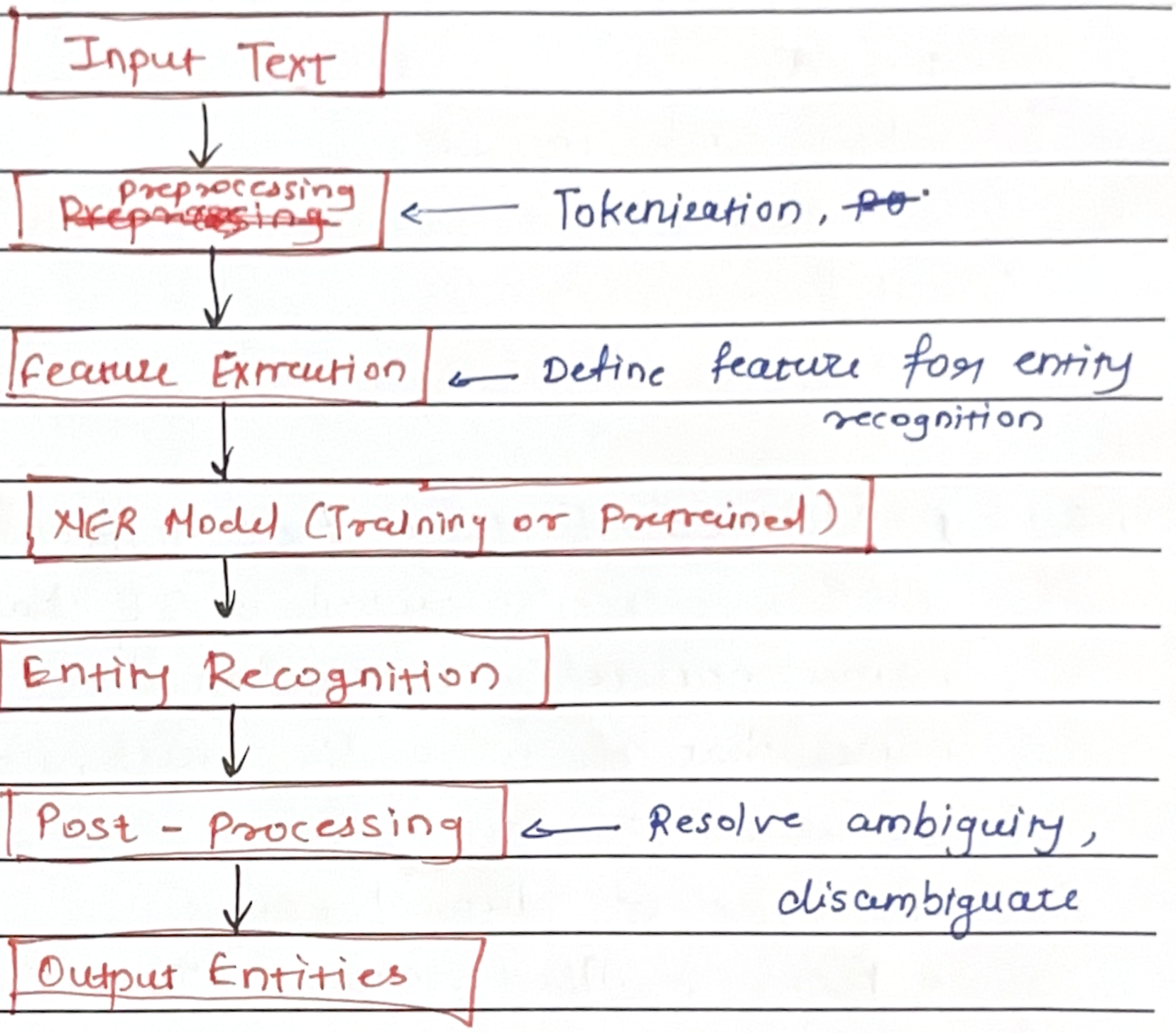
- People :- "Albert Einstein", "Taylor Swift"
- Organizations :- "NASA", "Tesla"
- Locations :- "Paris", "Mount Everest"
- Time Expressions :- "January 1, 2025", "next Monday"
- Miscellaneous Entities :- "COVID 19", "Bitcoin".

- The main objective of NER is to classify and categorize entities in a text, identifying their type (eg. person, organization, location)

eg: In the sentence "Apple Inc. in based in Cupertino, California" an NER system should recognize

"Apple Inc." as an organization and "Cupertino" and "California" as locations.

- Block diagram of NER Application.



1. Input Text: Raw text from various sources (e.g. news, social media etc) to extract named entities.

2. Preprocessing: Tokenization, Lowercasing, Removing stop words.

3. Feature Extraction: Word-based, contextual, lexical

4. NER Model : A trained model (HMM, CRF, LSTM) identifies named entities

5. Entity Recognition : The model classifies entities in the text (eg. "California" as a location)

6. Post-Processing : Ambiguity resolution, Linking entities.

7. Output Entities : Final o/p as structured entities
eg. ["Barak obama" (Pearson), "US" (Location)].

1.2.2 Relation Extraction

RE refers to identifying semantic relationships between entities in text. For eg. identifying that "Mark Zuckerberg" is the "Founder of" "facebook" from the sentence "facebook co-founder Mark Zukerberg".

- Types of Relations: Binary relations, Physical, Personal/social, Employment/Affiliation.
- Key Tasks : 1) Relation Extraction - Identifying the relationship between two entities in a text.
2) Relation Mention Extraction - Identifying individual mentions of relations.
- Approaches:

a) Feature-based Classification : Treats RE as a classification problem, where feature are crafted to represent relations between entities.

- Features Include

Entity Features : Names & types of entities (e.g. Father for family relation)

Lexical Contextual Features : Words or phrases between entities (eg. "founded" may indicate "foundedOf")

Syntactic Contexual Features : Relationships derived from sentence structure (eg. verb-object relations)

Background knowledge : Using external sources like wikipedia to identify relations between entities.

b) Kernel-based Classification : Uses kernel (eg. tree kernels) to measure similarities between entities based on their syntactic or structural relationship. Tree kernels are used with parse trees, and Sequence kernel are used with dependency paths between words.

- Training Methods : Supervised learning requires a labeled training corpus to train classifiers. Weakly supervised learning in which less labeled data is used.

- Challenges : Ambiguity - The same pair entities might have multiple possible relations, depending on context. Noisy Data - Automated extraction may include false patterns or incorrect relationships, especially in weakly supervised methods.

Contextual complexity - Relation mentions might span multiple sentences or be implied through complex sentence structures

• Relation Extraction framework :
- ACE Relation Feature Representation - Relations are often represented as graphs, where nodes represent entities & edges represent relations.
- Graph-based Features - A Relation instance can be represented as a directed graph with attributes like entity types and syntactic structure.

• Evaluation : Precision and Coverage - In methods like bootstraping, its important to evaluate the quality of extraction patterns based on how many true relation instances are covered and how accurate those instances are.

• Applications : - Building knowledge bases (eg. extracting facts from text and integrating them into databases) o
- Questioning answering systems (where recognizing relationships between entities is essential for providing accurate answers.
- Social network analysis (detecting relationships between entities or people).

1.2.3 Unsupervised Information Extraction

Unsupervised IE aims to extract info from large corpora without predefined labels. This method alleviates the challenges of manually annotating data and defining structures. Here

A) Relation Discovery and Template Induction

Relation discovery is about discovering relations without predefined types. For instance, given hurricane news articles, an algorithm might discover a relation like "hit" between a hurricane and the affected place.

- Shinyama & Sekine's Approach: They used lexical similarity to group similar articles and then performed syntactic parsing. Entity pairs were clustered based on contextual patterns, achieving around 75% accuracy.
- Rosenfeld & Feldman: Their method clusters entity pairs using surface patterns features like "arg1 based on arg2." They used algorithm clustering like k-means and hierarchical agglomerative clustering.
- Template Induction: Goes beyond binary relation to identify templates that contain multiple semantic roles. The approach include clustering role fillers and assigning labels to discored slots.

B) Open Information Extraction

This method extracts relations from a large, diverse corpus (eg. the web) without predefined relation types. It works by generating tuples of the form (arg1, relation, arg2)

- Banko and Etzioni's Approach: They introduced a CRF-based method to extract relations without lexical features. The focus was on syntactic patterns rather than specific relation types.
- Heuristic improvements: Later added heuristics like ensuring multi-word relation phrases start with verbs and end with prepositions. These heuristics led to better extraction results.

Unsupervised IE methods like relation discovery and open IE are powerful tools for extracting useful information without predefined relation types. They use clustering, syntactic patterns, and heuristics to find relations across diverse corpora.