Machine Learning for Data Science

M. H. Rahman

Associate Professor
Department of Statistics and Data Science
Jahangirnagar University
Bangladesh
E-mail: habib.drj@juniv.edu

Twenty-Fifty

Text Mining

Syllabus

Syllabus Text Mining

- Introduction and overview of quantitative text analysis and its applications.
- Information Extraction
- Basics of Text Mining
- Common Text Mining Visualizations
- Sentiment Scoring
- Hidden Structures
- Text Vectors and
- Topic Modeling.

Text Mining

What is text mining?

Text mining, also known as text data mining, is the process of transforming unstructured text into a structured format to identify meaningful patterns and new insights. It can use text mining to analyze vast collections of textual materials to capture key concepts, trends, and hidden relationships.

Text is one of the most common data types within databases. Depending on the database, this data can be organized as:

- Structured data: This data is standardized into a tabular format with numerous rows and columns, making it easier to store and process for analysis and machine learning algorithms. Structured data can include inputs such as names, addresses, and phone numbers.

Text Mining

- Unstructured data: This data does not have a predefined data format. It can include text from sources, like social media or product reviews, or rich media formats like video and audio files.
- Semi-structured data: As the name suggests, this data is a blend between structured and unstructured data formats. While it has some organization, it doesn't have enough structure to meet the requirements of a relational database. Examples of semi-structured data include XML, JSON, and HTML files.

JSON stands for JavaScript Object Notation. JSON is a text format for storing and transporting data. JSON is "self-describing" and easy to understand. Extensible Markup Language (XML) is a markup language and file format for storing, transmitting, and reconstructing arbitrary data. XML is a software- and hardware-independent tool for storing and transporting data.

Text Mining

Text mining vs. Text analytics

- The terms, text mining and text analytics are largely synonymous in meaning in conversation, but they can have a more nuanced meaning.
- Text mining and text analysis identify textual patterns and trends within unstructured data through the use of machine learning, statistics, and linguistics.
- By transforming the data into a more structured format through text mining and text analysis, more quantitative insights can be found through text analytics.
- Data visualization techniques can then be harnessed to communicate findings to wider audiences.

Text Mining

Text mining techniques

- The process of text mining comprises several activities that enable to deduce information from unstructured text data.
- Before applying different text mining techniques, it must start with text preprocessing, which is the practice of cleaning and transforming text data into a usable format.
- This practice is a core aspect of natural language processing (NLP) and it usually involves the use of techniques such as language identification, tokenization, part-of-speech tagging, chunking, and syntax parsing to format data appropriately for analysis.
- When text preprocessing is complete, it can apply text mining algorithms to derive insights from the data.

Text Mining

Text mining techniques

Some of these common text-mining techniques include:

- Information retrieval
- Natural language processing (NLP)
- Information Extraction
- Data mining

Text Mining

Information retrieval

Information retrieval (IR) returns relevant information or documents based on a pre-defined set of queries or phrases. IR systems utilize algorithms to track user behaviors and identify relevant data. Information retrieval is commonly used in library catalog systems and popular search engines, like Google. Some common IR sub-tasks include:

- **Tokenization**: This is the process of breaking out long-form text into sentences and words called "tokens". These are, then, used in the models, like bag-of-words, for text clustering and document matching tasks.
- **Stemming**: This refers to the process of separating the prefixes and suffixes from words to derive the root word form and meaning. This technique improves information retrieval by reducing the size of indexing files.

Text Mining

- **Natural language processing (NLP)**
  Natural language processing, which evolved from computational linguistics, uses methods from various disciplines, such as computer science, artificial intelligence, linguistics, and data science, to enable computers to understand human language in both written and verbal forms. By analyzing sentence structure and grammar, NLP sub-tasks allow computers to "read". Common sub-tasks include:
  - **Summarization**: This technique provides a synopsis of long pieces of text to create a concise, coherent summary of a document's main points.
  - **Part-of-Speech (PoS) tagging**: This technique assigns a tag to every token in a document based on its part of speech—that is, denoting nouns, verbs, adjectives, and so on. This step enables semantic analysis of unstructured text.

Text Mining

- Natural language processing (NLP)...
  - **Text categorization**: This task, which is also known as text classification, is responsible for analyzing text documents and classifying them based on predefined topics or categories. This sub-task is particularly helpful when categorizing synonyms and abbreviations.
  - **Sentiment analysis**: This task detects positive or negative sentiment from internal or external data sources, allowing to track changes in customer attitudes over time. It is commonly used to provide information about perceptions of brands, products, and services. These insights can propel businesses to connect with customers and improve processes and user experiences.

Information Extraction

Information extraction (IE) surfaces the relevant pieces of data when searching various documents. It also focuses on extracting structured information from free text and storing these entities, attributes, and relationship information in a database. Common information extraction sub-tasks include:

- **Feature selection, or attribute selection**, is the process of selecting the important features (dimensions) to contribute the most to the output of a predictive analytics model.
- **Feature extraction** is the process of selecting a subset of features to improve the accuracy of a classification task. This is particularly important for dimensionality reduction.
- **Named-entity recognition (NER)** also known as entity identification or entity extraction, aims to find and categorize specific entities in text, such as names or locations. For example, NER identifies "California" as a location and "Mary" as a woman's name.

Text Mining

- **Data mining**

  Data mining is the process of identifying patterns and extracting useful insights from big data sets. This practice evaluates both structured and unstructured data to identify new information, and it is commonly utilized to analyze consumer behaviors within marketing and sales. Text mining is essentially a sub-field of data mining as it focuses on bringing structure to unstructured data and analyzing it to generate novel insights. The techniques mentioned above are forms of data mining but fall under the scope of textual data analysis.

Overview of Text Mining

- Text Preprocessing phase
  - Tokenization
  - Compound word identification
  - Normalization and noise reduction
  - Linguistic analysis
- Content Analysis
  - Dictionary-based
  - Supervised learning
  - Unsupervised learning

Text Mining

Text mining applications

- Customer service
- Risk management
- Maintenance
- Healthcare
- Spam filtering
- Social Media Analysis
- Business Intelligence

Text Mining

Key Info

- Text mining is extracting insights from text. Example: analyzing customer reviews to identify sentiments and preferences.
- NLP is Natural Language Processing, and text mining uses NLP techniques to analyze unstructured text data for insights.
- Text mining in Python involves using libraries like NLTK or spaCy for natural language processing tasks.
- Text mining is used to extract insights from unstructured text data, aiding decision-making and providing valuable knowledge across various domains.

Tokenization

Tokenization is the process of breaking up a given text into units called tokens. Tokens can be individual words, phrases or even whole sentences. In the process of tokenization, some characters like punctuation marks may be discarded. The tokens usually become the input for the processes like parsing and text mining.

In other words, Tokenization is the process of breaking down a piece of text, like a sentence or a paragraph, into individual words or "tokens." These tokens are the basic building blocks of language, and tokenization helps computers understand and process human language by splitting it into manageable units.

Lemmatization

Lemmatization is a text normalization technique that reduces different forms of a word to a single root form, or lemma. For example, "systems" becomes "system" and "changes" becomes "change".

In other words, Lemmatization is the process of reducing the different forms of a word to one single form, for example, reducing "builds" "building" or "built" to the lemma "build". Lemmatization is also the process of grouping inflected forms together as a single base form.

Sentiment score

In text mining, a sentiment score quantifies the sentiment expressed in a piece of text, such as a sentence, paragraph, or document. Sentiment analysis, also known as opinion mining, involves identifying and categorizing opinions expressed in the text to determine whether the writer's attitude towards a particular topic, product, or event is positive, negative, or neutral.
Components of Sentiment Score

- Polarity:
  Positive: Indicates favorable sentiment.
  Negative: Indicates unfavorable sentiment.
  Neutral: Indicates neither favorable nor unfavorable sentiment.
- Intensity (Magnitude): Reflects the strength of the sentiment. For instance, "I love this product!" has a higher intensity positive sentiment compared to "I like this product."

Text Mining

Methods to Compute Sentiment Scores

- Lexicon-Based Approaches: Use predefined dictionaries (sentiment lexicons) where words are associated with specific sentiment scores.

  Example lexicons: AFINN, SentiWordNet, VADER (Valence Aware Dictionary and sEntiment Reasoner).

- Machine Learning-Based Approaches:
  Train models on labeled datasets where text samples are annotated with sentiment labels.

  Techniques: Naive Bayes, Support Vector Machines, Deep Learning models like LSTMs and Transformers (BERT, GPT).

Text Mining

AFINN Lexicon: AFINN Lexicon is the most simplest and popular lexicon for sentiment analysis. The current version is AFINN-en-165.txt (** Check) and it contains more than 3000 words along with its polarity score. Head over to the official repository to know more.

SentiWordNet: SentiWordNet is a lexical resource for opinion mining. SentiWordNet assigns to each synset of WordNet three sentiment scores: positivity, negativity, and objectivity.

VADER: VADER (Valence Aware Dictionary and Sentiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. It is fully open-sourced and is available in the NLTK package which can be applied directly to unlabelled text data. VADER is capable of detecting of polarity and intensity of emotion.

Example of Lexicon-Based Sentiment Scoring

Here's a Python example using the VADER sentiment analysis tool, which is particularly good for social media text:

```
!pip install vaderSentiment
from vaderSentiment.vaderSentiment import
                SentimentIntensityAnalyzer

# Initialize VADER sentiment analyzer
analyzer = SentimentIntensityAnalyzer()
# Sample text
text = "I absolutely love this! It's fantastic."

# Get sentiment scores
scores = analyzer.polarity_scores(text)
print(scores)
{'neg': 0.0, 'neu': 0.323, 'pos': 0.677, 'compound': 0.855}
```

Example of Machine Learning-Based Sentiment Scoring

Using a pre-trained transformer model from Hugging Face's transformers library:

```
#!pip install transformers
from transformers import pipeline

# Initialize sentiment-analysis pipeline
nlp = pipeline("sentiment-analysis")

# Sample text
text = "I absolutely love this! It's fantastic."

# Get sentiment scores
result = nlp(text)
print(result)
[{'label': 'POSITIVE', 'score': 0.9998749494552612}]
```

Applications of Sentiment Scores

- Customer Feedback Analysis: Understanding customer satisfaction and dissatisfaction.
- Social Media Monitoring: Gauging public opinion on brands, products, or events.
- Market Research: Analyzing sentiment trends over time.
- Product Reviews: Summarizing consumer opinions on e-commerce platforms.

Sentiment scores provide actionable insights by converting qualitative text data into quantitative metrics, enabling better decision-making and strategic planning.

What is lexicon-based sentiment analysis

Lexicon-based sentiment analysis is a technique used in natural language processing to detect the sentiment of a piece of text. It uses lists of words and phrases (lexicons or dictionaries) that are linked to different emotions to label the words (e.g. positive, negative, or neutral) and detect sentiment.

In our example, the words are labeled with the help of a so-called valence dictionary. Each word in the text can have some type of emotional valence, such as "great" (positive valence) or "terrible" (negative valence), which gives us a positive or negative impression.

What is lexicon-based sentiment analysis...

Take the phrase "Good airlines sometimes have bad days.". A valence dictionary would label the word "Good" as positive; the word "bad" as negative; and possibly the other words as neutral.

Once each word in the text is labeled, we calculate an overall sentiment score by counting the numbers of positive and negative words and then combining the values. A popular formula to calculate the sentiment score (StSc) is:

$$\text{StSc} = \frac{\text{Number of positive words} - \text{Number of negative words}}{\text{Total number of words}}$$

If the sentiment score is negative, the text is classified as negative. A positive score means a positive text and a score of zero means the text is classified as neutral.

Text Mining

Breiman (2017), Han et al. (2022)

References I

Breiman, L. (2017). *Classification and regression trees*, Routledge, Newyork, USA.

Han, J., Pei, J. and Tong, H. (2022). *Data Mining: Concepts and Techniques*, 4th edn, Morgan Kaufmann, Burlington, MA.

***** - /// - There are no End .. - /// - *****

****

****