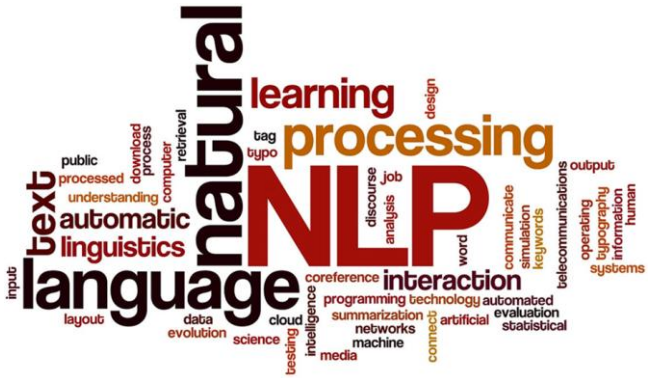
6CS012 – Artificial Intelligence and Machine Learning.
Tutorial – 08
Text Data Pre – processing and Representations.

Siman Giri {Module Leader – 6CS012}



Terminology Alert!!

- **Text data:**
  - Documents.
  - Data that is in the form of text file.
- **Corpus:**
  - Collection of Documents.

- **Vocabulary**
  - Collection of all the **unique terms(words)** in the corpus.
  - Please Note: This Lecture focus on text representations and the process of creating vocabulary will be discuss on your tutorial session.

The New Challenge!!!

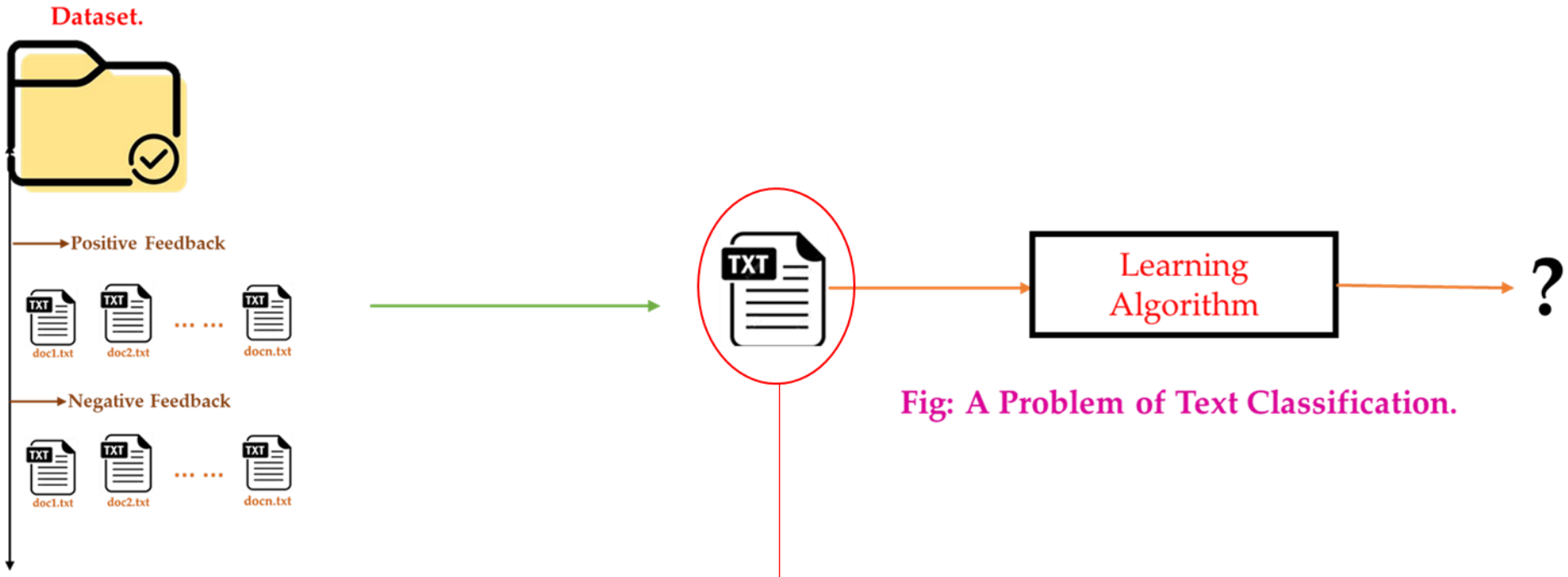


Fig: A Problem of Text Classification.

- First Let's Answer:
  - How Does Computer Understand Text?

1.1 Natural Language Processing: Introduction.

- **Natural language** is one of the **most complex tools** used by humans for a wide rage of reasons, for instance,
  - to communicate with others, to express thoughts or feelings, and ideas to ask questions, or to give instructions.
- Therefore, it is **essential for computers (intelligence system)** to possess the ability
  - to use the same tool in order to effectively interact with humans.
- The **field** of **Natural Language Processing** is a field of research in computer science including AI and Deep Learning 4
  - concerned with giving computers the ability to understand text and spoken words.
- Any NLP application consists of two tasks:
  - **Natural Language Understanding:**
    - **NLU** deals with understanding the meaning of human language, usually expressed as a piece of text.
  - **Natural Language Generations:**
    - The goal is for a computer to generate text, or in other words to talk to humans through natural language
      - Either to verbalize and idea or meaning, or to provide a response.
        - **NLG gives answer**.

1.2 NLP: Example.

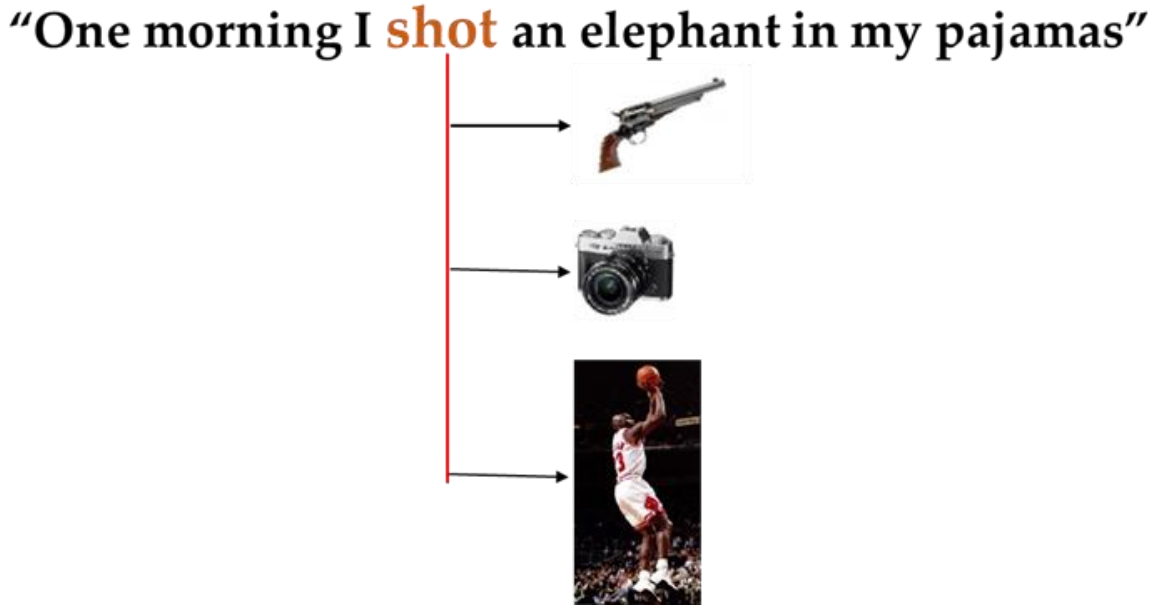
- For instance: **You ask a question to any chatbots:**
  - **"do penguins fly?"**
    - Before chatbot can answer the question:
      - the very first step for chatbot is **to understand the question**:
        - which in turn depends on **the meaning of penguin and fly**, and their **composition**.
      - **Task of NLU.**
    - To answer the question, it must be able to **generate a response** i.e.
      - **Yes** or **No** – such that we can understand.
      - **Task of NLG.**

1.3 Some Common Application of NLP.

- Majority of the NLP tasks break down human text/audio in ways that help computer understand and make sense out of it.
  - **Language Translation or Machine Translation:**
    - Translating from one language to another.
    - Google translate.
  - **Text Classification or Sentiment analysis:**
    - attempts to extract subjective qualities—attitudes, emotions, sarcasm, confusion, suspicion—from text.
  - **Speech recognition:**
    - Converts voice data into text.
    - Required for any application that follows voice commands or answers spoken question.
  - **Text summarization:**
    - Summarizing the text from any literature.

1.4 Challenges of Natural Languages.

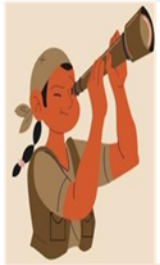
- {Cautions: Our Discussions will be based on English language, but challenges are transferrable to Other language as well.}
- **What makes language hard?**
  - **Ambiguity:**
    - One of the most important difficulties with human language lies in its **ambiguous nature**. Ambiguity can arise at different levels.
      - Ambiguity in **Word** Level:



1.4.1 Challenges of Natural Languages.

- {Cautions: Our Discussions will be based on English language, but challenges are transferrable to Other language as well.}
- **What makes language hard?**
  - **Ambiguity:**
    - One of the most important difficulties with human language lies in its **ambiguous nature**. Ambiguity can arise at different levels.
      - Ambiguity in **Sentence** Level:

She saw the man with the telescope.



This sentence is ambiguous because it could mean either that the woman used a telescope to see the man, or that she saw a man who was using a telescope.

1.4.2 Challenges of Natural Languages.

- **What makes language hard?**
  - **Use of common sense.**
    - Humans do not learn language by observing an endless stream of text
      - For example: What does the following phrase means?
        - " **I ordered a mouse from Amazon**."
        - **"No body orders a mouse(animal) from amazon(rain forest)."**

1.4.3 Challenges of Natural Languages.

- **What makes language hard?**
  - **Figurative language:**
    - Given that the interpretation of these expressions is not a direct function of the meanings of their constituent words, they pose a serious challenge for language understanding algorithms.
      - **For example: What is the meaning of:**
        - **"fingers crossed"**
        - **"all ears"**

Plan for the Week

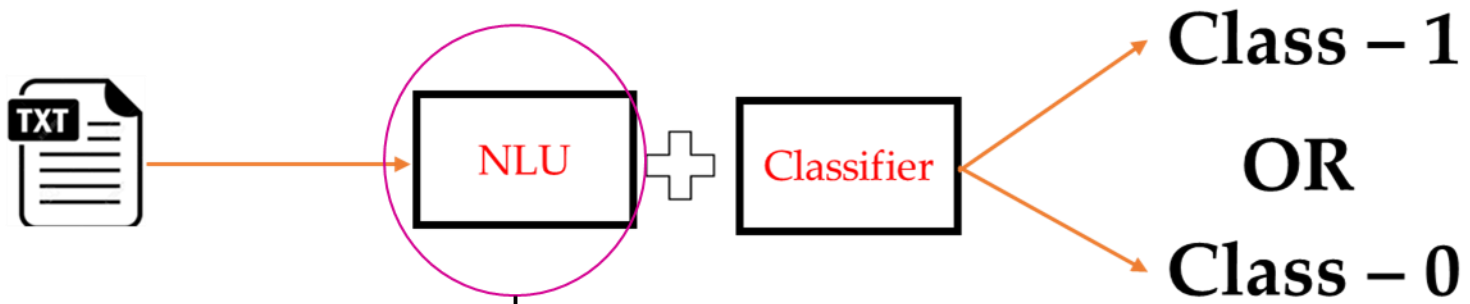


Fig: A Problem of Text Classification.

- Text Pre – Processing or Cleaning. {Focus of Our Tutorial.}
- Text Pre – Representations. {Focus of Our Lecture.} ✓

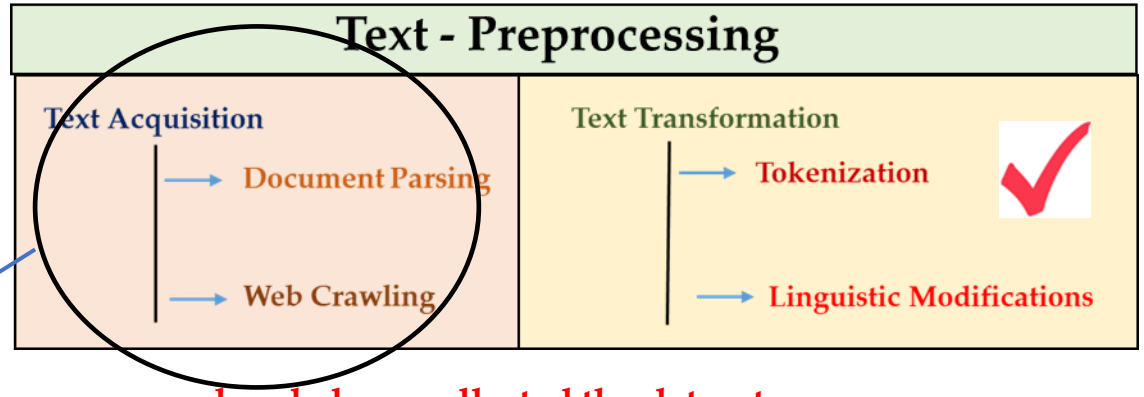
Combine both to build a Classifier. {Plan for Workshop.}

2. Before Text Representation.

{ Text Pre – processing for Natural Language Understanding Task.}

2.1 Text Pre – processing: Introduction.

- Text processing is the process of **converting** **documents** to **{index} terms**.
- **Why?**
  - **Vocabulary Creation:**
    - **Modification required to make the text suitable for further processing.**
    - **To understand which pre-processing steps must be followed.**
    - **Pre-processing steps might be text dependent.**
    - **Improper pre-processing steps may lead to loss of lexical content.**



Not under the scope of this course. For this course we already have collected the dataset.

2.2 Text Transformation: Introduction.

- **Preprocessing (Text Transformation) is the first and a crucial step of NLP task**
  - The **objective of preprocessing** is to **clean/harmonize** the **text**,
    - reduce language fluctuations, if necessary,
    - and **prepare the tokens** for being processed in the next steps
- **Agenda for this class:**
  - **Text cleaning/harmonization/reduction of fluctuations – How?**
    - **Text normalization**
    - **Segmentation**
    - **Stop words**
    - **Stemming & Lemmatization**

2.3 Tokenization.

- Tokenization in NLP is the process of breaking down a piece of text into smaller chunks, called tokens, such as words, phrases, symbols, or other meaningful elements.
- It's a fundamental step in most NLP tasks, as it helps to standardize text and make it more manageable for further analysis.

2.3.1 Challenges of Tokenization Process.

- **Token is an atomic indexing unit i.e.**
  - [“end”] is not same as [“ends”].
- **Handling symbols and abbreviations.**
  - Split I.B.M à [“I”,”B”,”M”] {accidental matches **pronoun “I”** will match **I.B.M**}
- Fusing of the words:
  - “pre-diabetes” → will not match “prediabetes”
- **One word or several:**
  - data base
  - Los Angeles-based company
  - State-of-the-art
- **Dates:**
  - 3/20/91
  - 20/3/91
- **Numbers:**
  - B-52
  - 100.2.86.144

2.3.2 Challenges of Tokenization Process.

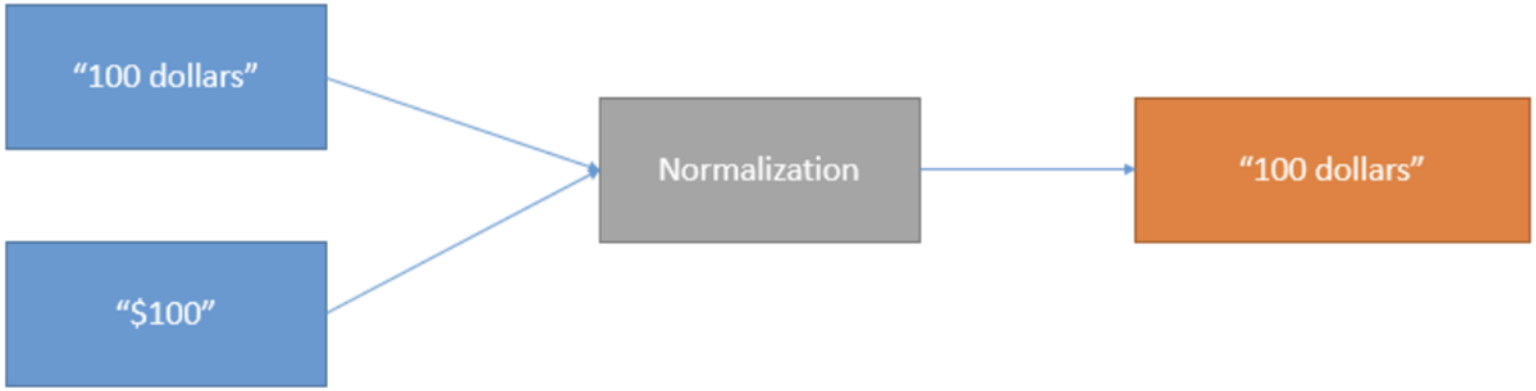
- Languages:
  - **No white-space – How to split?**
    - ノーベル平和賞を受賞したワンガリ・マータイさんが名誉会長を務め るＭＯＴＴＡＩＮＡＩキャンペーンの一環として、毎日新聞社とマガ ジンハウスは「私の、もったいない」を募集します。皆様が日ごろ 「もったいない」と感じて実践していることや、それにまつわるエピ ソードを８００字以内の文章にまとめ、簡単な写真、イラスト、図 などを添えて１０月２０日までにお送りください。大賞受賞者には、 ５０万円相当の旅行券とエコ製品２点の副賞が贈られます。
  - **Arabic Script:**
    - كِ تَ ا بٌ ⇐ اَبٌ
      - **un b ā t i k**
      - **/kitābun/ ‘a book’**
- We will be using **English language** – so not a challenge for us.

2.3.3 To Summarize: A Tokenization Process.

- First step is to use parse to identify appropriate parts of the document to tokenize.
  - Defer complex decisions to other components
    - word is any sequence of alphanumeric characters terminated by a space or special character, with everything converted to lower-case
    - Everything is indexed i.e. 92.3 → 92 3 but search finds documents with 92 and 3 adjacent
  - Incorporate some rules to reduce dependence on query transformation components.
- **Create a pipeline:**
  - **Normalizations → Stopping word removal → Stemming/Lemmatization →**

2.4 Text Normalizations.

- Textual data might have multiple words that share similar meanings.
  - Normalization is the process that standardizes text by reducing the different variations of words with similar meanings and transforming them into a single canonical form.
  - This accounts for variations in spelling, inflection, or other linguistic features, where different forms of the same word can be treated as a single term.
  - This process aims to minimize randomness to improve the quality of the text, reduce the vocabulary size, and improve the efficiency of text processing and the performance of your NLP model.



2.4.1 Text Normalizations: Example.

- Idea → Normalization harmonizes the written forms of the words with same meanings
  - **Some examples:**
    - **deleting periods**
      - ***U.S.A. → USA***
    - **deleting hyphens**
      - ***anti-discriminatory → anti discriminatory***
    - **Accents**
      - French ***résumé → resume***
    - **Case folding: reduce all letters to lower case**
      - It may cause ambiguity but typically helpful
        - **General Motors vs. general motors**
        - **Fed vs. fed**
        - **CAT (City Airport Train) vs. cat**

- **Handling Numbers and Dates:**
  - **Do the numbers, dates, etc. bring information?**
    - If included, the dictionary size may explode!
    - Numbers and dates are commonly replaced by special tokens, e.g.
    - Numbers with <num>
    - Dates with <dates>

2.5 Text Segmentations.

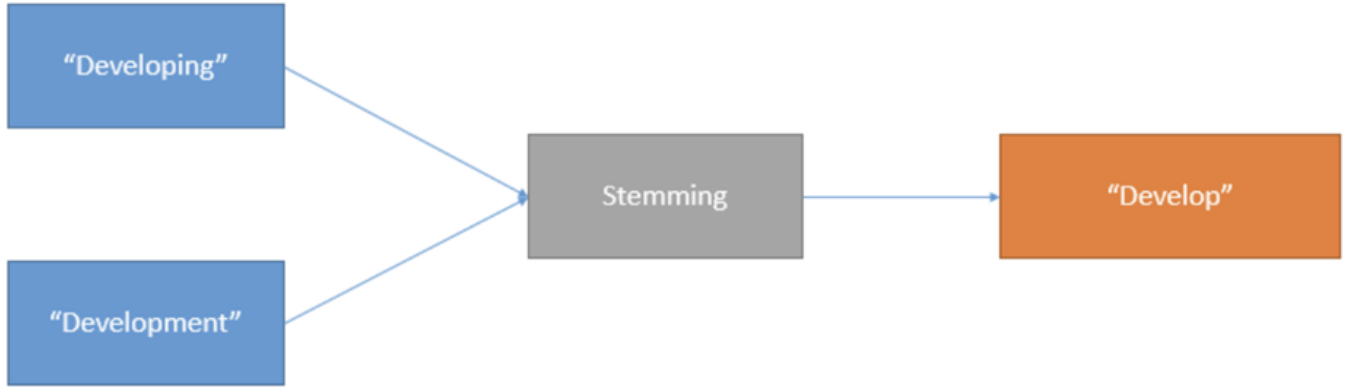
- Segmentation → Splitting a compound words into tokens.
  - **For Example:**
    - **Brainstorm** → [“**Brain**”, “**storm**”]
    - **Bookworm** → [“**Book**”, “**worm**”]

2.6 Stop – words Removal.

- Words like **"is"** and **"are"** are abundant in textual data,
  - appearing so frequently and less significant in semantic understanding of the data
    - Thus, they don't need processing as thoroughly as nouns, other verbs, and adjectives.
- NLP refers to these as stop words, which usually don't add meaning to the data.
  - Stop word removal means removing these commonly used words from the text you want to process.
- **Stop words Example:**
  - The commonest words, like ***the, a, and, to, be***
    - **The pre-positions like "on the to ...."**
  - They carry **little or no semantic information**
  - In Practice we remove all those kind of stop words,
    - Stop-words are also task dependent.

2.7 Stemming.

- In linguistics and information retrieval fields, stemming means reducing inflected (or sometimes derived) words to their stem, base, or root form.
  - **politician, politicians, policy → politics.**
  - **policeman, policemen,→ police**
- The **stem** can be different from the **word's root form**.
- Related words are usually sufficient to map to the same stem, even if the stem isn't a valid root.
  - The stemming process(may) remove redundancy in the data.



2.8 Lemmatizer.

- **Stemming is Hard:**
  - The goal in the stemming process is to iteratively reach to word form, multiple algorithms are proposed most popular being "porter stemming algorithm"
  - Stemming in general is hard.
- **Lemmatization:**
  - A lemmatizer uses a knowledge resource (like WordNet) to find and replace base forms
  - Lemmatizer reduces inflectional/variant forms to base forms.
    - Examples:

      - am, are, is → be

      - car, cars, car's, cars' → car

      - the boy's cars are different colors → the boy car be different color

2.8.1 Lemmatizer Vs. Stemming.

- Both reduce variation
- Stemming is typically faster
- Stemming may harm precision and increase ambiguity
- If a given word does not exist in the knowledge resource, lemmatization may not be able to process it

Fig: Stemming Vs. Lemmatization

3. Text Pre – processing to Text Representations.

{ A Hands – On Exercise on TF – IDF Vectorization.}

2.3.2 TF – IDF weights.

- **Combining tf and idf:**
  - **Common in doc → high tf → high weight.**
    - If a term appears frequently in a document (high TF), it suggests that the term is **important** in that specific document.
  - **Rare in Corpus → high idf → high weight.**
    - If a term appears in only a few documents (high IDF), it suggests that the term is **rare** and thus more **informative**.
    - Common terms (like "the", "and", "is") have low IDF, so they don't contribute much to distinguishing documents.
  - **TF-IDF** combines these to reflect both the **local importance** and **global rarity** of a term in a document, making it a useful measure for distinguishing terms that truly capture the **essence** of a document in the corpus.
    - **tf – idf score is given as:**
      - $W_{tf-idf} = tf_{t,d} \times idf_t$
      - Proposed by G. Salton et. al. 1983 – must probably the most well-known document representation schema.
- **Example Computations:**
  - **For the following corpus, compute the TF – IDF weights for all the unique terms (vocabulary):**
  - **Data – Documents:**

Document ID	Text
Doc 1	"desk and table"
Doc 2	"table and chair"
Doc 3	"chair and lamp"

A Corpus.

Example Computations: TF and IDF weights.

Compute Term Frequency (TF)

- TF for a term t in document d is:
  - $\mathbf{TF_{t,d} = \frac{Number\ of\ times\ t\ appers\ in\ d}{Total\ numbers\ of\ terms\ in\ d}}$

Term	TF – DOC 1	TF – DOC 2	TF – DOC 3
and	0.33	0.33	0.33
chair	0	0.33	0.33
desk	0.33	0	0
lamp	0	0	0.33
table	0.33	0.33	0

- A sample computations:
  - $TF_{and} = \frac{\text{Number of times "and" appears in Doc 1}}{\text{Total number of terms in Doc 1}} = \frac{1}{3} \approx 0.33$

Inverse Document Frequency (IDF)

- IDF for term t is:
  - $\mathbf{IDF_t = \log_{10}\left(\frac{N}{df_t}\right)}$
    - **Here N = 3**

Term	$\mathbf{df_t}$	$\mathbf{idf_t}$
and	3	$\log\left(\frac{3}{3}\right) = \log(1) = 0.00$
chair	2	$\log\left(\frac{3}{2}\right) \approx 0.176$
desk	1	$\log\left(\frac{3}{1}\right) \approx 0.477$
lamp	1	$\log\left(\frac{3}{1}\right) \approx 0.477$
table	2	$\log\left(\frac{3}{2}\right) \approx 0.176$

TF – IDF: Example Computations.

- tf – idf score is given as:
  - $W_{tf-idf} = tf_{t,d} \times idf_t$

Term	DOC 1 – TF × IDF	DOC 2 – TF × IDF	DOC 3 – TF × IDF
and	0.33 × 0.00 = 0.00	0.33 × 0.00 = 0.00	0.33 × 0.00 = 0.00
chair	0 × 0.176 = 0.00	0.33 × 0.176 = 0.058	0.33 × 0.176 = 0.058
desk	0.33 × 0.477 = 0.158	0 × 0.477 = 0.00	0 × 0.477 = 0.00
lamp	0 × 0.477 = 0.00	0 × 0.477 = 0.00	0.33 × 0.477 = 0.158
table	0.33 × 0.176 = 0.058	0.33 × 0.176 = 0.058	0 × 0.176 = 0.00

- Representation for "desk" is:
  - [0.158, 0.00, 0.00]

Towards Exercise Sheet.