Day 17 Training Report

15 July 2025

TF-IDF Vectorization + Feature Extraction

On **Day 17**, we learned to **convert text into numerical form**, a necessary step because machine learning models can only process numbers. The main focus was **TF-IDF (Term Frequency-Inverse Document Frequency)**.

---

1. Objectives

- Convert raw text into **numerical feature vectors**.
- Understand how **word importance** is quantified in documents.

---

2. TF-IDF Concept

- **Term Frequency (TF):** How often a word appears in a document.
- **Inverse Document Frequency (IDF):** Measures how unique a word is across documents.
- **TF-IDF Score:** High for words that appear **frequently in a document but rarely in others**.

TF-IDF=TF×log⁡(Total DocumentsNumber of Documents containing the word)\text{TF-IDF} = \text{TF} \times \log(\frac{\text{Total Documents}}{\text{Number of Documents containing the word}})TF-IDF=TF×log(Number of Documents containing the wordTotal Documents)

Example:

```
from sklearn.feature_extraction.text import TfidfVectorizer

documents = ["I love machine learning", "Machine learning is amazing"]
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(documents)

print(vectorizer.get_feature_names_out())
print(X.toarray())
```

---

3. Hands-on Practice

- Applied TF-IDF to **sample product reviews and tweets**.
- Visualized top words with **highest TF-IDF scores**.
- Compared TF-IDF with **simple bag-of-words** for efficiency and informativeness.