**Fine-Tuned Text Classification and Summarization for Focused Information Extraction**

**Introduction:**

Social media platforms provide a wealth of information that reflects public opinion. Understanding the emotions behind this perspective and the reasons for it can provide valuable insights for businesses, organizations and individuals. This function is an NLP pipeline for classifying tweets according to sentiment and provides a summary that reveals the cause of that sentiment.

**Problem statement:**

Sentiment analysis and Social Media Text:

* Social media platforms provide a wealth of data that reflects public opinion, but manually sifting through this data can be overwhelming.
* Understanding the emotions (good, bad) behind public opinion and why those emotions occur is valuable for businesses, organizations and individuals.

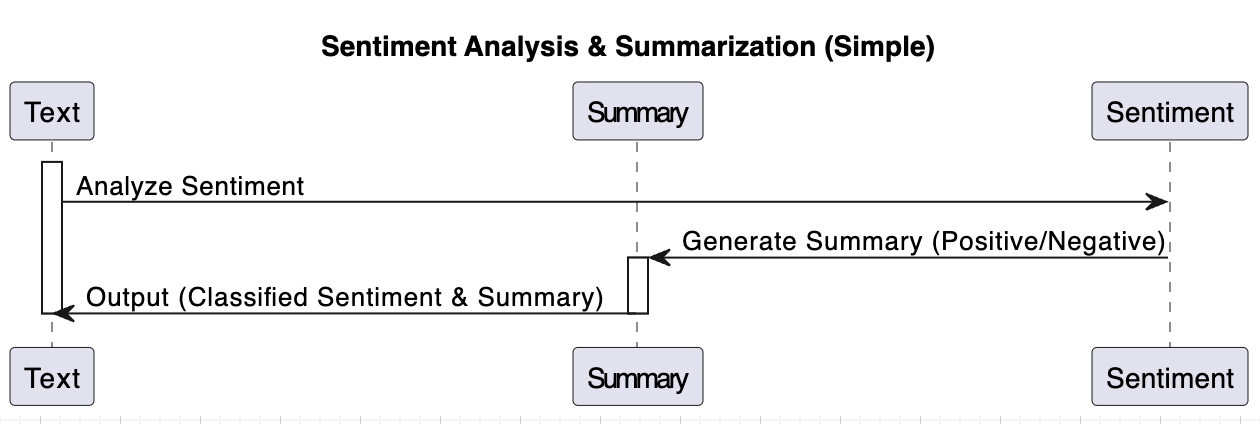
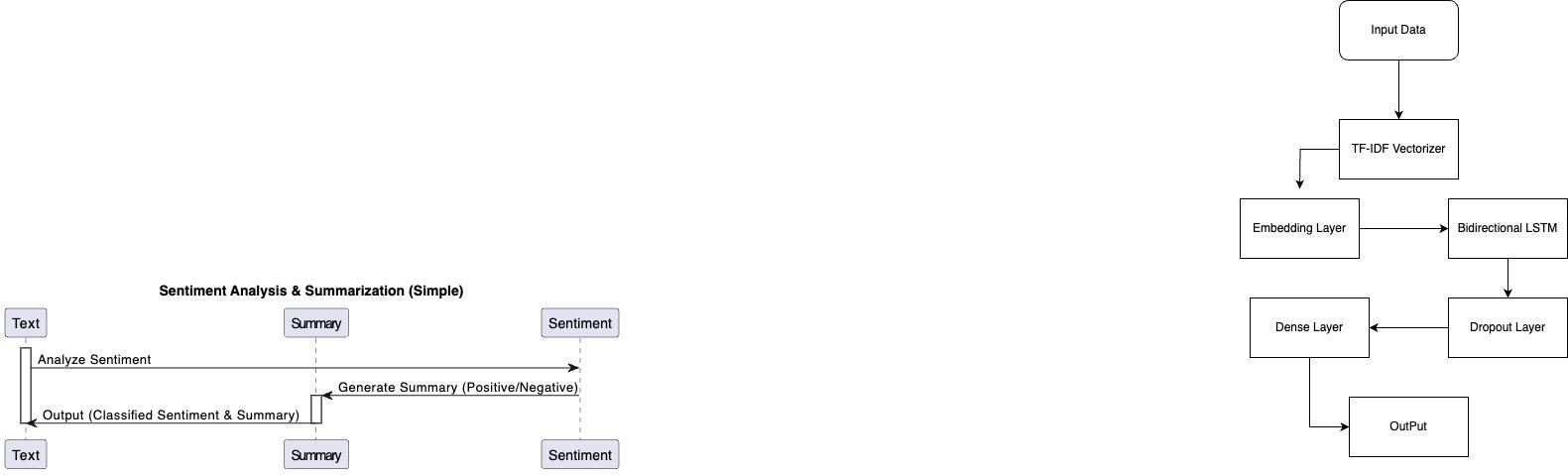
The aim of this project is to develop a system that will:

* Classifies social media posts (e.g., tweets) as positive or negative sentiments.

Provide a social media summary, highlighting the aspects that contribute to the predicted sentiment.

To achieve these goals, the project will provide a means to analyze the emotions behind social media conversations and understand the key drivers of those emotions.

**Your Model/Methodology:**

* Overall workflow diagram
  + 
* Architecture diagram of model with explanation
  + 

**Data sets:**

**Description:**

A sensory analysis data set is a collection of text data tagged with sensory categories. These categories generally have positive, negative, and neutral emotions. The data set may also include additional information such as content, metaphor, or thematic classification.

**Representatives: .**

* Text: Text data can include various formats such as sentences, paragraphs, tweets, product reviews, and news reports.
* Labeling: Emotional labels typically consist of categorical variables such as "positive," "negative," or "neutral." Some data structures may use numeric notation (e.g. 1 for positive, 0 for negative).

**Design of Features**:

* This represents the sensitivity group assigned to the texture data. It can be a category (positive, negative, neutral) or a more granular classification

**Exploratory data analysis:**

****

**Positive Sentiment:**

* family: This appears to be the most prominent word, suggesting a strong focus on family in the text data.
* love
* happy
* good
* great
* next (might be related to phrases like "next chapter" or "next adventure" implying positive anticipation)
* better

All of these words express feelings and experiences, indicating that the text is based on positive emotions.

**Other Observations:**

* morning: This can indicate good relations in the morning, or be part of a greeting such as "good morning."
* think and thought might reflect introspection or contemplation within the text.
* Food can be associated with positive food experiences.
* The party means a social gathering that can be good
* twitter can indicate that the source of the text data is social media..

**Limitations:**

* It is difficult to draw definite conclusions about the source of the textual information without obtaining additional context.
* Word clouds prioritize frequently used words, not necessarily. A word can often have a greater meaning depending on the context.

****

**Negative Sentiment:**

* hate: This is the most prominent word, indicating strong negativity in the text data.
* never (often used to express negation or dissatisfaction)
* bad
* fail
* worst
* late (can be negative depending on the context, like being late for something important)
* cold (could describe physical coldness or a negative emotional state)

These words econvey frustration, disappointment, anger, and other negative emotions.

**Other Observations:**

* Time: This can be related to negative experiences or frustrations related to lack of time.
* money: Financial problems or frustration can be an issue with text data.
* school (may be negative depending on context, such as schoolwork complaints);
* work (which, like school, can be negative if it involves job stress or dissatisfaction);
* People (if they are associated with conflict or uncomfortable relationships with others);
* Know (it may mean doubt or confusion) .

**Implementation:**

**Libraries:**

* **NLTK (Natural Language Toolkit):** This library provides tools for manipulating textual data in Python. Your code uses features from NLTK:
  + **Stemming**: Reducing words to bases (e.g., "running" -> "run").
  + **Stop words**: Remove common words that don't contribute much to comprehension (e.g., "the," "a").
  + **Text parsing**: Breaking down information into sentences and tokens (individual words).
* **Sumy**: This library provides various data collection algorithms, including Latent Semantic Analysis (LSA). Your code uses the LsaSummarizer class from Sumy for LSA-based summarizing.

**Sentiment Analysis (LSTM):**

* Imagine you have a box filled with reviews (text data) and you want to determine whether they are positive, negative, or neutral. This piece of legislation does just that!
  + LSTM: Think of it as an intelligent machine that learns from patterns. It already reads many reviews labeled positive, negative or neutral. Based on this study, it can analyze new reviews and predict their emotions.
  + approach:
  + The code passes the probe information (words) to the LSTM device.
  + LSTM analyzes words and their structure to understand the overall feeling.
  + Finally, it predicts the sentiment (positive, negative, or neutral) to review.

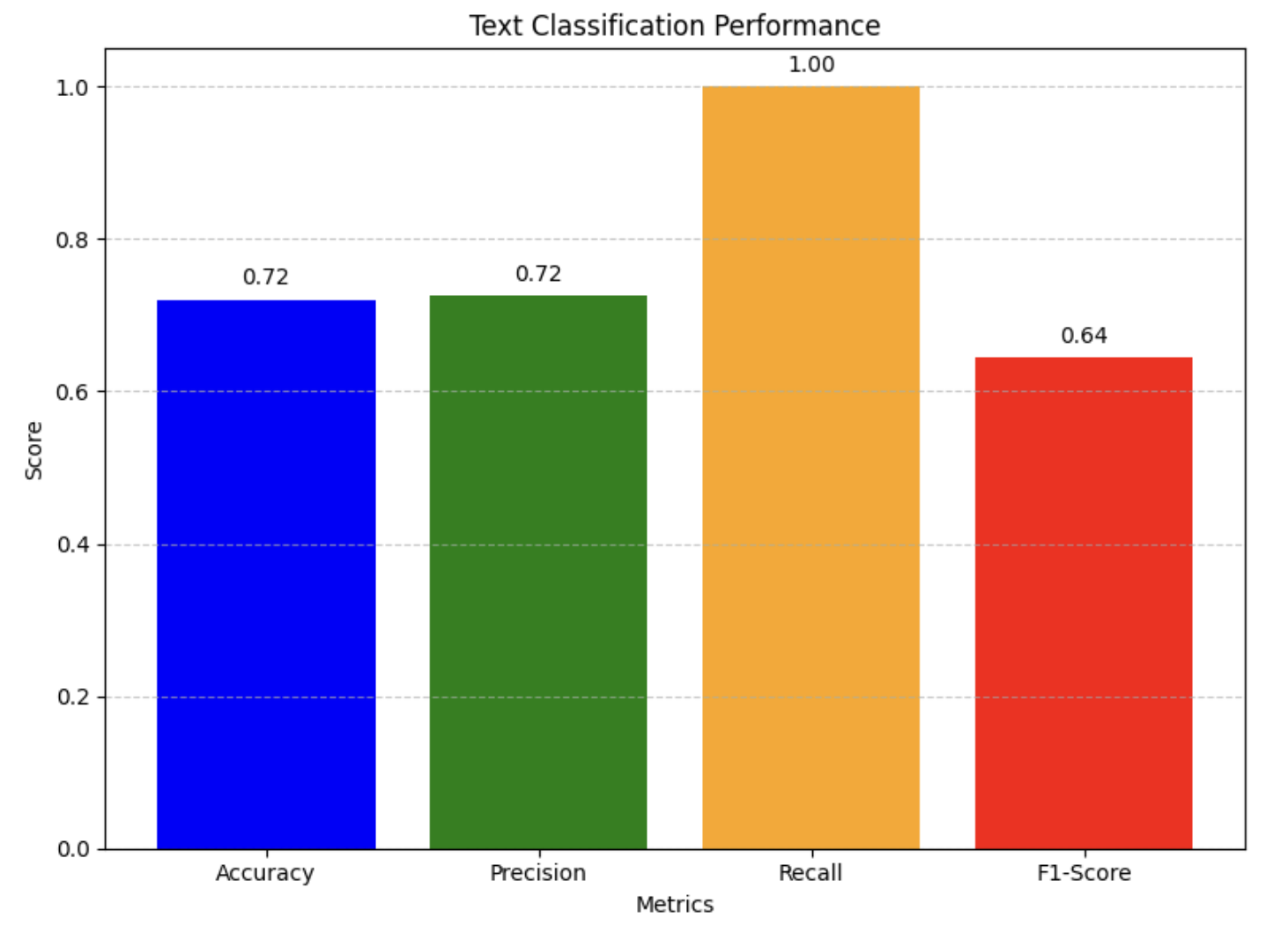
**Text Summarization (lsa\_summary function):**

* 1. **LSA (Latent Semantic Analysis)**: is a fancy way of saying that the code is looking for points in the text.
  2. approach:
  3. The program cleans the text (removes unnecessary words, and merges similar ones).
  4. It then analyzes relationships between words to understand key concepts.
     + Based on this analysis, it identifies important sentences that summarize the main points of the text.
     + Finally, it returns a summary that captures the essence of the story.

**Result:**

**Text Classification:**

* **Accurate: 72%** - This means that the model correctly classified 72% of the textual data into positive, negative, or neutral categories (assuming this is your classification task).
* **Accuracy: 0.725** - This metric shows you how often the model predicted positive/negative/neutral emotions, and was in fact correct for those emotions (positive equity, negative equity, neutral equity based distribution on the project). A value of 0.725 indicates that the model does well in most cases when predicting emotions.
* **Remember: 1.0** - This metric indicates how well the model identified all actual positive/negative/neutral patterns in the data. A perfect score of 1.0 indicates that the model captured all relevant patterns. However, the trade-off between accuracy and recall is important. In some cases, very high recall may exist at the expense of lower precision (i.e. the model may classify some irrelevant samples as positive/negative/neutral).
* **F1-score: 0.6444** - This is a compromise between precision and recall, giving a balanced view of the model's performance. A score of 0.6444 indicates that there is improvement, which can be achieved by trying to increase accuracy without sacrificing too much memory.

****

**Text Summarization (ROUGE Scores):**

* ROUGE-1: This metric measures the degree of overlap (considering unigrams or single words) between your summaries created and reference summaries created by humans The scores awarded are:
  + Precision: 0.23 (less than expected) - This means that only 23% of the total number of terms in your summary that were generated by the samples used in the ROUGE-1 analysis were present actually exist in the human-written collections.
  + Recall: 0.42 (moderate) - This means that 42% of the variance in the human condition variables was accounted for by the model in its output
  + F1-measurement: 0.30 (low) - Harmonic mean accuracy and recall, indicating overall performance of ROUGE-1.

| Description | Responsibility (Task,) | Person | Contributions (%) | Issues/Concerns |
| --- | --- | --- | --- | --- |
| Text Classification Model Training | name1 | Model Development | 35 | - Potential data imbalance requiring investigation. |
| Text Summarization Model Implementation | name2 | Model Selection & Integration | 30 | - Delays due to challenges in integrating the summarization model with the overall pipeline. |
| Data Cleaning & Visualization Preparation | name3 | Data Preprocessing & Exploration | 35 | - Encountered missing values in a specific data source - Identifying the best visualization techniques for the project's specific needs. |

**References:**

* <https://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html>
* <https://pranalipardeshi30.medium.com/machine-learning-for-sentiment-analysis-e17b089384a3>
* <https://machinelearningmastery.com/machine-learning-with-python/>
* <https://radimrehurek.com/gensim_3.8.3/auto_examples/tutorials/run_summarization>
* <https://huggingface.co/docs/transformers/en/tasks/summarization>
* <https://en.wikipedia.org/wiki/Automatic_summarization>