**Sentiment classification using AUM-ST**

**Motivation:**

Sentiment classification is a well-known problem in natural language processing (NLP), where the objective is to identify and classify the sentiment or emotion conveyed by a customer based on textual input or tone. While various approaches have been developed to address this problem, supervised learning is commonly used. However, a significant limitation of supervised learning is the need for large amounts of labeled data to train the model effectively. Acquiring labeled data in real-time is both challenging and costly, as it requires human experts to manually annotate the data, which is time-consuming and resource intensive.

To overcome the limitations of supervised learning, researchers have introduced semi-supervised learning methods. These methods aim to leverage a small set of labeled data to guide the model in making use of a larger pool of unlabeled data. By utilizing semi-supervised learning, models can effectively label and classify data with minimal human intervention. However, a key challenge in semi-supervised learning is handling noisy pseudo-labels, as noisy or incorrect labels can lead to poor model generalization and reduce overall performance. Reducing the impact of noisy labels remains a critical area of focus in advancing semi-supervised learning techniques for sentiment classification.

To address the challenge of noisy labeled data in semi-supervised learning, Tiberiu Sosea et al. proposed a method that leverages the **area under margin (AUM)** for text classification. This approach helps in filtering out noisy pseudo-labeled data, thereby improving model performance. We aim to apply this method to sentiment classification, with the goal of enhancing classification accuracy while using a smaller amount of labeled data. By incorporating the AUM-based approach, we seek to improve the filtering of noisy pseudo-labels, ensuring better generalization and more reliable sentiment predictions in semi-supervised settings.

**Objectives:**

The main motivation of these projects is to successfully classify the sentiment into positive tone, negative tone, or neutral tone using a smaller number of labeled data using the AUM-ST algorithm. The main objectives of these projects are:

1. Identify how well the model can classify the sentiment analysis using the different numbers of less label samples.
2. To successfully create the unlabeled data using back translation or other approaches because getting real time unlabeled data is not available for us.
3. To successfully set the algorithm for sentiment classification.
4. To successfully overcome the noisy pseudo labeled data using the area under margin curve.

**Significance:**

1. It minimizes the reliance on large, labeled datasets and makes sentiment analysis more accessible and cost-effective in real-world scenarios.

2.Filters out noisy pseudo-labels and improves the accuracy and reliability of sentiment classification, essential for applications like customer feedback analysis.

3. Use of AUM-ST addresses a key challenge in semi-supervised learning, which handles noisy labels contributing to progress in this area of NLP.

4.Enhances sentiment analysis, which benefits businesses and research by providing more precise insights into market trends and social dynamics.

5.Demonstrates that sentiment analysis models can be scaled effectively with minimal labeled data, making the technology more adaptable across various sectors.

**Technical Characteristics:**

To classify the sentiment classification using AUM-ST, we use the following technical characteristics:

1. Semi-Supervised Learning Framework: To reduce dependency on large amounts of labeled data, we use the semi supervised learning framework.
2. AUM Filter: Filtering noisy pseudo-labels often leads to better results in model generalization.
3. Accuracy: To compare the results with previous methods, we want to use accuracy as an evaluation method because we want to use the balanced data among positive, negative, and neutral tones.

**Deliverables:**

1. Sentiment classification model:The best sentiment classification model that can classify the sentiment into positive, negative, and neutral tones and is trained on a smaller number of labels.
2. Performance reports:proper documentation of code implementation as well as effective results comparisons of the model with existing methods.

**Milestones:**

1. Preprocessed dataset
2. Preparing unlabeled data sets
3. Modifying existing code for sentiment classification
4. Comparing model performance with different label samples.

**Dataset:**

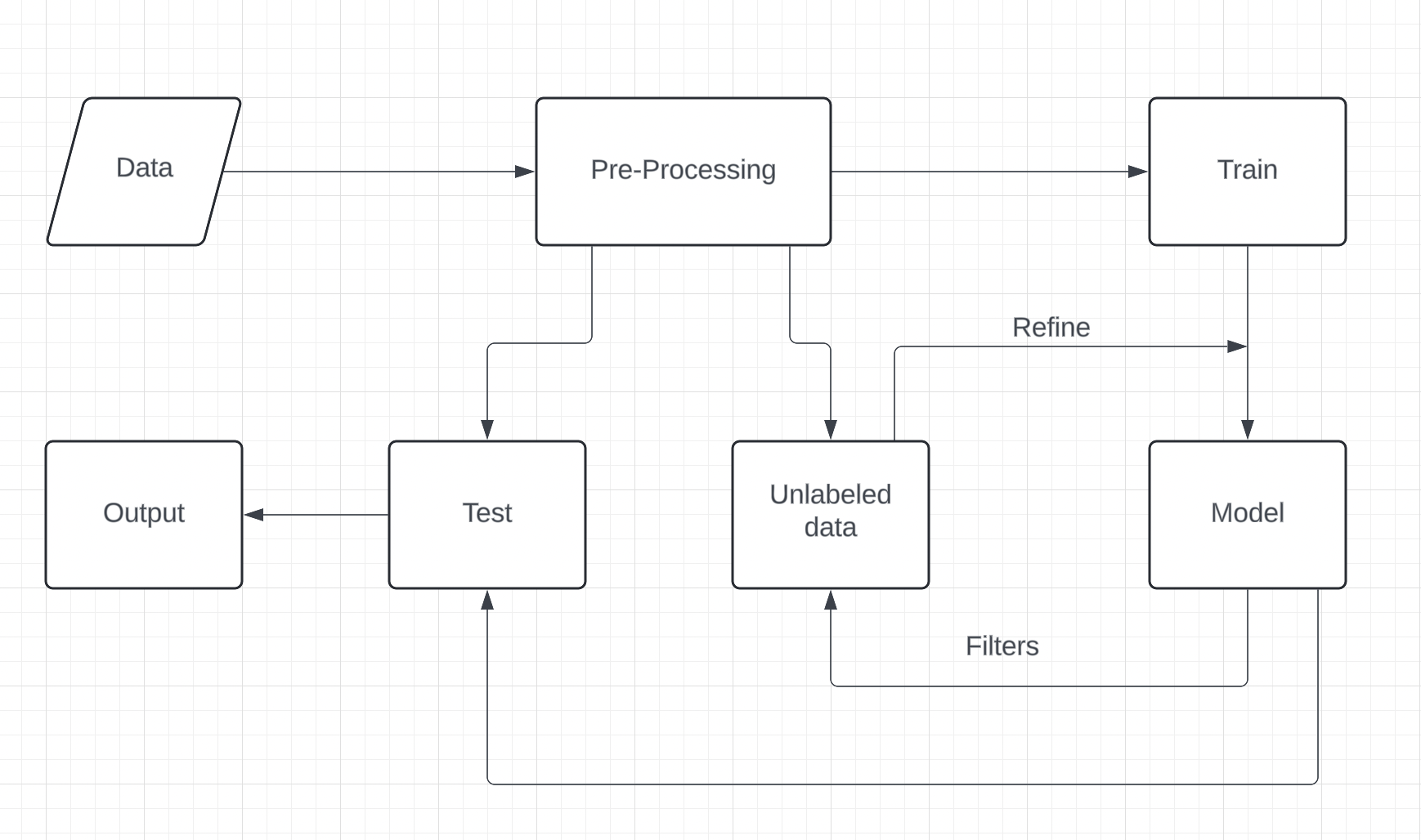
Sentiment analysis dataset is taken from the Kaggle which has the training dataset and testing dataset. The dataset contains the tweets with emotions removed.

The dataset contains the data having 6 fields

* The sentiment of the tweet
* The ID of the tweet
* The tweets date and time
* The search term used
* The username of the person who tweeted
* The content of the tweet

**Source:**<https://www.kaggle.com/datasets/abhi8923shriv/sentiment-analysis-dataset?resource=download>

**Visualization:**



1. Data: The project begins with data collection; it includes both labeled data and unlabeled data.

2. Pre-processing: Next the data undergoes pre-processing, which involves cleaning and transforming the data so that it gets ready for training and the input data is now available in the desired format for the model.

3. Train: Now Labeled data is used to train the model, setting the foundation for future refinement.

4. Model: Next, the trained model is used to produce pseudo-labels for the unlabeled data. This model's output is refined by filtering out noisy pseudo-labels using techniques such as AUM.

5. Unlabeled Data: The refined model utilizes the unlabeled data, integrating it into the training loop to improve its learning with minimal labeled data.

6. Test: The model is tested to evaluate its performance using a test dataset. This step measures the model's ability to accurately classify sentiments.

7. Output: Finally, the tested model produces the output, which includes classified sentiments, indicating whether the input data conveys a positive, negative, or neutral tone.