Sentiment Analysis

CE807-25-SP

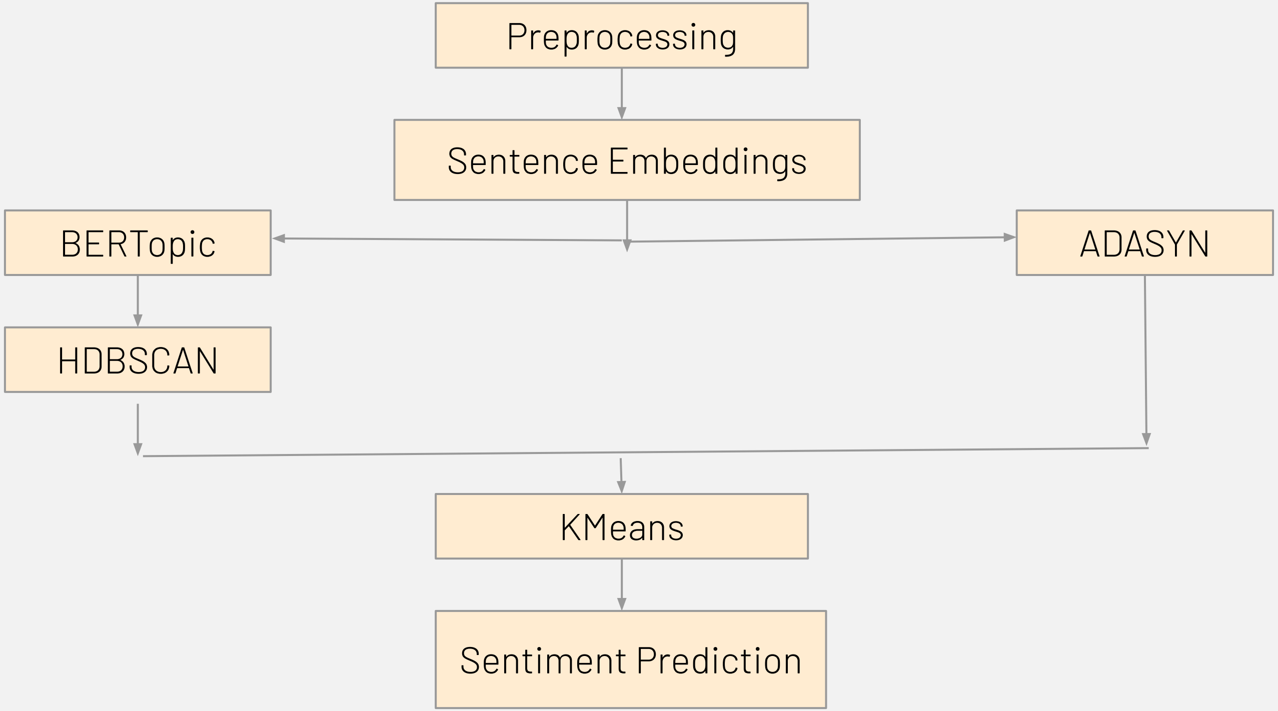
Name: Samia Rahman Misty
Student ID: 2400570

Approach

Unsupervised

Discriminative

Unsupervised Approach



Unsupervised Approach

1. **Why BERTopic for topic modeling?**

   Uses **transformer embeddings + HDBSCAN**

   Extracts **interpretable**, high-quality topics from unstructured text

2. **Why HDBSCAN after clustering training dataset based on their semantic content?**

   Detects **dense clusters** in semantic space

   No need to predefine number of clusters

   Great for **variable-length, noisy data**

3. **Why ADASYN for oversampling?**

   Creates **synthetic samples** for minority classes

   Improves **class balance** and model performance

4. **Why finally using KMeans for prediction?**

   Efficiently clusters encoded data

   Maps clusters to **sentiment labels**

   Scalable and fast for inference

Discriminative Approach

Bidirectional Long Short-Term Memory (BiLSTM)

BiLSTM Model Configuration

Component	Details
Input	10,000-word vocab, max length: 100 tokens
Embedding Layer	128-dimensional vectors
BiLSTM Layer	64 hidden units
Dense Layer	64 units, ReLU activation
Dropout	0.5 to prevent overfitting
Output Layer	Sigmoid (binary classification)

Training Setup

Parameter	Value
Optimizer	Adam (learning rate: 0.01)
Loss Function	Binary Cross-Entropy
Batch Size	64
Epochs	5 (early stopping at epoch 5)
Observation	Training PRF peaked, val PRF plateaued → early signs of overfitting

Discriminative Approach

1. **Why not RNN?**

   - Struggles with long-range dependencies

   - Poor performance on noisy, short text

2. **Why not GAN?**

   - Designed for generation, not classification

3. **Why not LSTM?**

   - Unidirectional; misses backward context

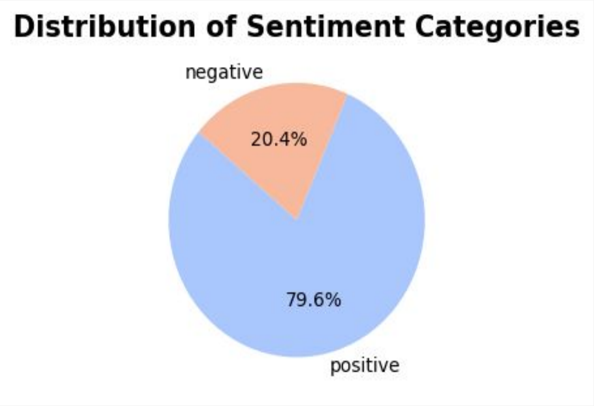
4. **Why BiLSTM?**

   - Captures context in both directions

   - Handles noisy, imbalanced short texts well

   - Efficient and robust for sentiment tasks

Data Set & Feature Analysis



Feature	Description
Syllable Count Distribution	Right-skewed; most under 100 syllables. Outliers suggest complex/noisy data.
Sentence Length Distribution	Peaks around 15 words; rare short/long sentences affect tokenization.
Root Word and Lemmatization	140k words in root form; 10k inflected forms may benefit from lemmatization.
Suffixes and Pluralization	150k root words; fewer than 25k have suffixes/plurals. Stemming improves uniformity.

Data Preprocessing(Contd.)

Common Preprocessing

- Lemmatization (e.g., *"running"* → *"run"*) to preserve meaning
- Lowercasing, punctuation/URL/whitespace removal
- Stopwords removed; only meaningful tokens retained

Discriminative Approach

Tokenization

- Text → Integer sequences
- Vocabulary capped at **10,000** (frequent words only)

Padding

- Padded to **90th percentile length** to avoid over/underfitting

Label Encoding

- **Positive = 1**, **Negative = 0**
- Simple binary classification setup

Data Preprocessing(Contd.)

Unsupervised Approach

Sentence Embedding

- Uses **all-MiniLM-L6-v2** for speed + accuracy
- Alternatives (BERT, RoBERTa) more accurate but slower

Feature Extraction

- **CountVectorizer** with unigrams & bigrams
- **No TF-IDF** – lacks contextual depth

Clustering

- **HDBSCAN + KMeans** used for semantic clustering
- DBSCAN & Agglomerative are options but less scalable

Class Balancing

- **ADASYN** oversampling for hard-to-classify minorities
- SMOTE is an alternative, but ADASYN is more adaptive

Hyperparameter Tuning

Threshold Calibration

- **Tested Range**: 0.3 to 0.8
- **Optimal Threshold**: **0.7**
  - Best **precision-recall** trade-off
  - Minimizes **false positives/negatives**
  - Tuned for **79.1% positive** class imbalance

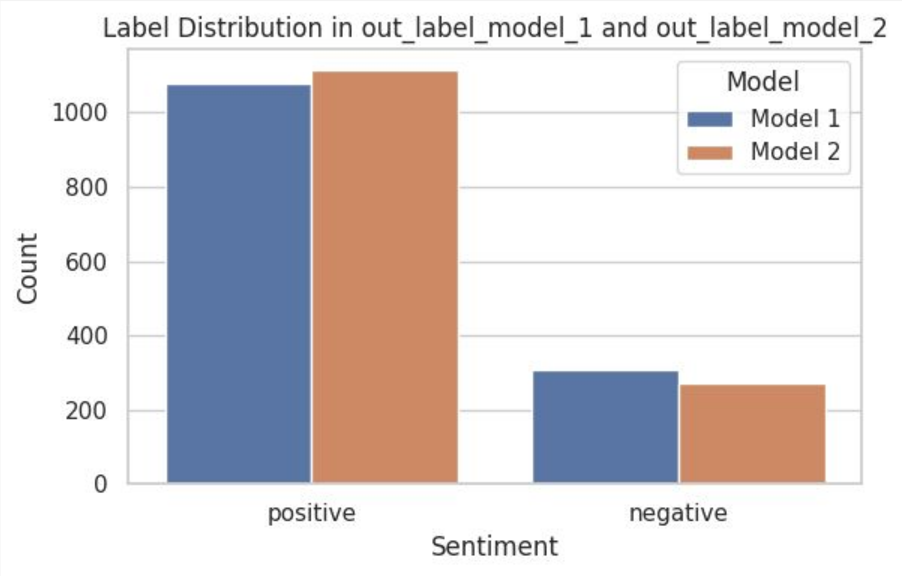
KMeans Clustering

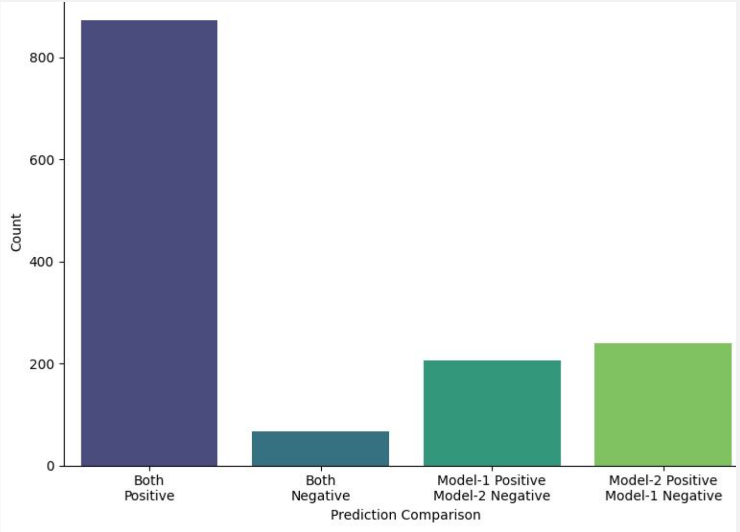
- **Cluster Count Tested**: 2 to 7
- **Best Performance**: **n_clusters = 2**
  - Clear separation of **positive vs. negative** sentiment
  - Higher cluster counts → confusion & drop in **true negatives**
  - **n=2** ensures best **F1-score**, **accuracy**, and **generalization**

Model Performance

Metric	Model 1	Model 2	SOTA	Explanation
Precision vs Recall	Recall ≈ Precision	Recall > Precision	Balanced	Model 1 balances both; Model 2 favors recall; SOTA typically maintains balance.
Bias Toward Positive	Moderate	Strong	Low	KMeans helps reduce bias in Model 1; Model 2 suffers due to high threshold.
Overfitting Signs	None	None	None	All show stable train/validation performance; no signs of overfitting.
Generalization Ability	Moderate	Strong	Strong	Model 2 generalizes well; SOTA trained on diverse data, boosting adaptability.
Class Imbalance Handling	Moderate	Weak	Strong	KMeans + ADASYN help Model 1; SOTA more robust due to architecture and tuning.
F1 Score (Validation)	69.90%	87.16%	~90%	Model 1 is moderate; Model 2 excels with threshold tuning; SOTA slightly better.

Model Comparison





Unsupervised Approach (Model 1) & Discriminative Approach (Model 2)

Example & Justification

Text	GT	SoTA	Model 1	Model 2	Explanation
great received quickly	Positive	Positive	Positive	Positive	All models correctly identify positive sentiment. "Received quickly" is well-handled by semantic clustering, BiLSTM, and SoTA transformers.
work messed ever thing	Negative	Negative	Positive	Positive	Both custom models misclassify due to ambiguity in “work.” SoTA's contextual depth correctly captures the negative tone.
better paper filter mess	Negative	Neutral	Negative	Negative	Model 1 & 2 detect negative hints. SoTA labels neutral, possibly due to short length and missed subtle cues.
easy install water taste good	Positive	Positive	Negative	Positive	Model 1 misclassifies, likely confused by neutral phrasing. Model 2 and SoTA identify “taste good” as a strong positive indicator.
needed needed thank dm	Positive	Neutral	Positive	Negative	Model 1 captures sentiment through gratitude cues. Model 2 misclassifies due to strict thresholding; SoTA stays neutral due to vagueness.

Discussion & Summary

Limitations Highlighted

- **Standard Metrics Fall Short**
  Accuracy, F1-score, and precision/recall can **mislead** in imbalanced datasets.
- **Model Bias**
  - Unsupervised clustering → prone to **biased groupings**.
  - Deep learning models tend to **favor majority class**.
- **Threshold Calibration**
  - Effective for **tuning decision boundaries**.
  - Improves **precision, recall**, and **F1-score** on imbalanced data.

Future Improvements

- **Contrastive Learning**
  → Better representation of **minority classes**.
- **Variational Autoencoders (VAEs)**
  → Unsupervised **feature disentanglement** for diverse patterns.
- **Adaptive Reweighting (e.g., Focal Loss)**
  → Handles class imbalance by **emphasizing hard examples**.

Thank You