Group 26

TASK: H2

Hate and Offensive Language Identification

Sarthak Maini (2020576)

Saharsh Dev (2020572)

Ashwin Tomer (2020289)

Abhit Rana (2020421)

Problem Definition

Binary Classification Problem

Dataset

Hate Offensive tweets (HOF)

Non-Hate Offensive tweets (NOT)

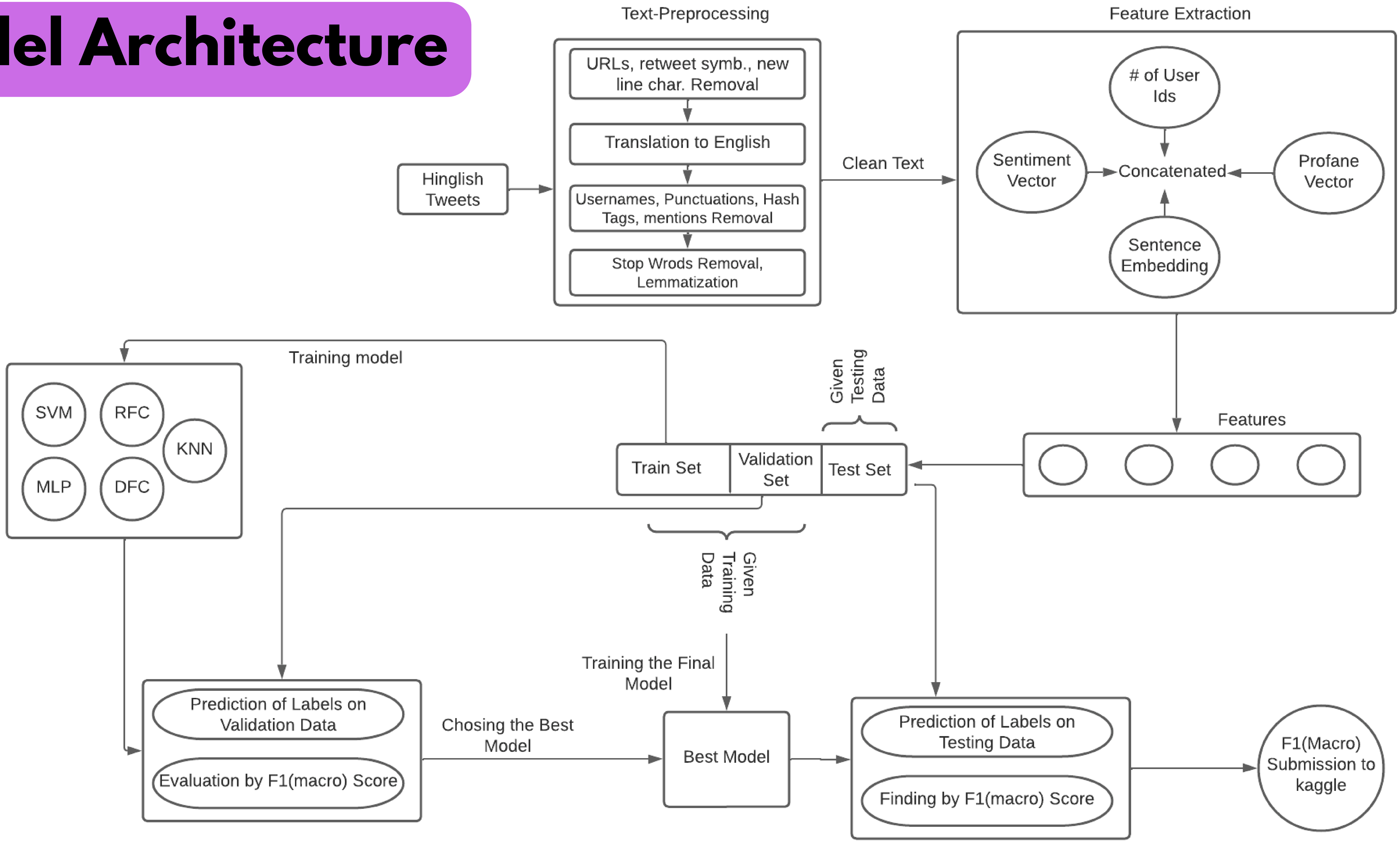
HASOC 2021

Twitter posts in Hindi and Hinglish

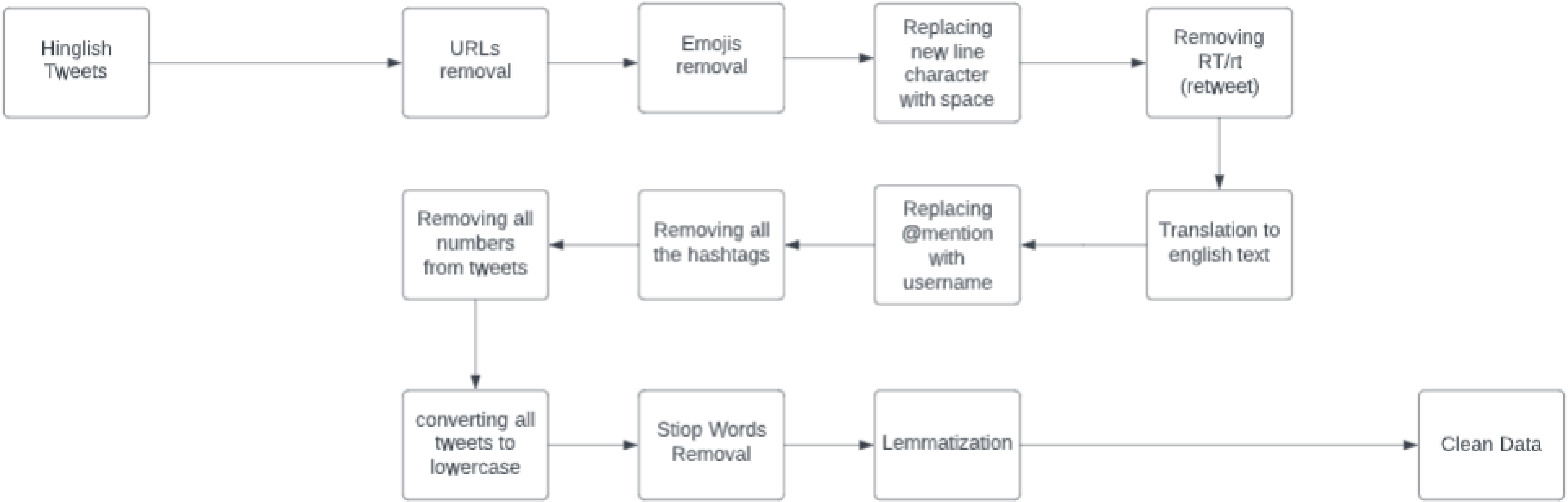
Literature Review

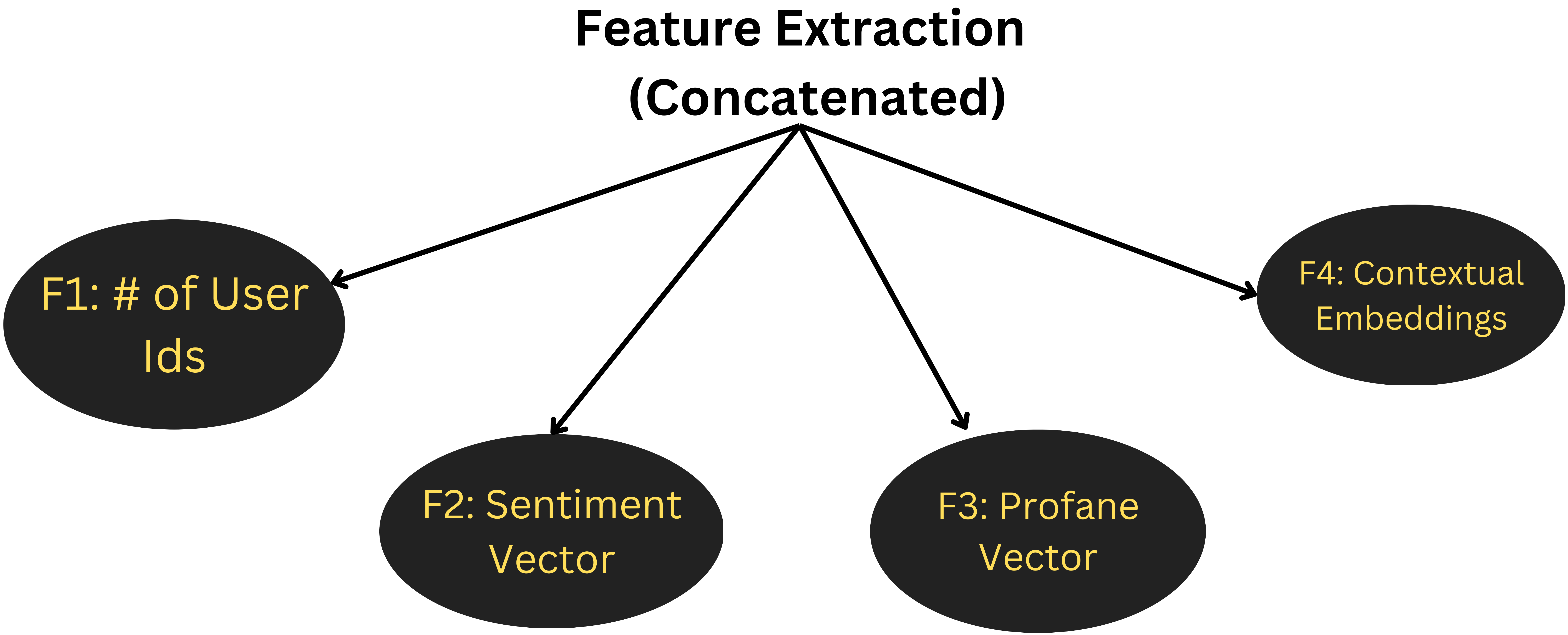
- **Peter Burnap et al. (2015)** used an N-gram technique to create profane vectors from a list of profane phrases
- **Sreelakshmi K et al. (2020)** observed that fastText features gave better feature representation than wor2vec and doc2vec with SVM -RBF
- **Kamble et al. (2018**) trained typical DL Models (CNN, LSTM, BiLSTM etc.) for generating domain-specific word embeddings.
- **Chopra et al. (2020)** used social network-based features and targeted hate embeddings to train the model.
- **Jahan et al. (2021)** translated Hindi to English and Hinglish was phonetically converted to Hindi then synonym replaced into English and was then trained on LR, CNN and BERT.

Model Architecture



Text Preprocessing




Feature Extraction
(Concatenated)
F1: # of User
Ids
F2: Sentiment
Vector
F3: Profane
Vector
F4: Contextual
Embeddings

Experiments

Embeddings

Tf-Idf+Vectorizer,
ULMFIT, BERT, XLNET,
distilroberta-hatespeech,
uncased-hatexplain

Classification Models

SVM, XGBOOST,
MLP, RFC, KNN,
DTC

BEST-1 -> BERT(FINE TUNED) + SVM
BEST2 -> TF-IDF+VECTORIZER+ RFC or SVM, BERT(FINE TUNED)+XGBOOST
OTHERS-> OTHER POSSIBLE RESULTING COMBINATION OF THESE MODELS

Results

Hate specific neural architectures

Transformer	RFC	MLP	SVM
distilroberta-hatespeech	0.642	0.671	0.668
uncased-hatexpain	0.647	0.676	0.666
facebook/roberta	**0.657**	**0.697**	**0.697**

Highest Score -
MLP

Fine-Tuned BERT

Architechture	RFC	MLP	SVM
1 layer 1epoch	**0.729**	**0.75**	**0.773**
4 layer 2 epoch	0.728	0.724	0.767

Highest Score -
SVM

TF-IDF Vectorizer

Embedding	DT	RFC	MLP	KNN
TF-IDF	0.693	**0.762**	0.693	0.565

Highest Score -
RFC

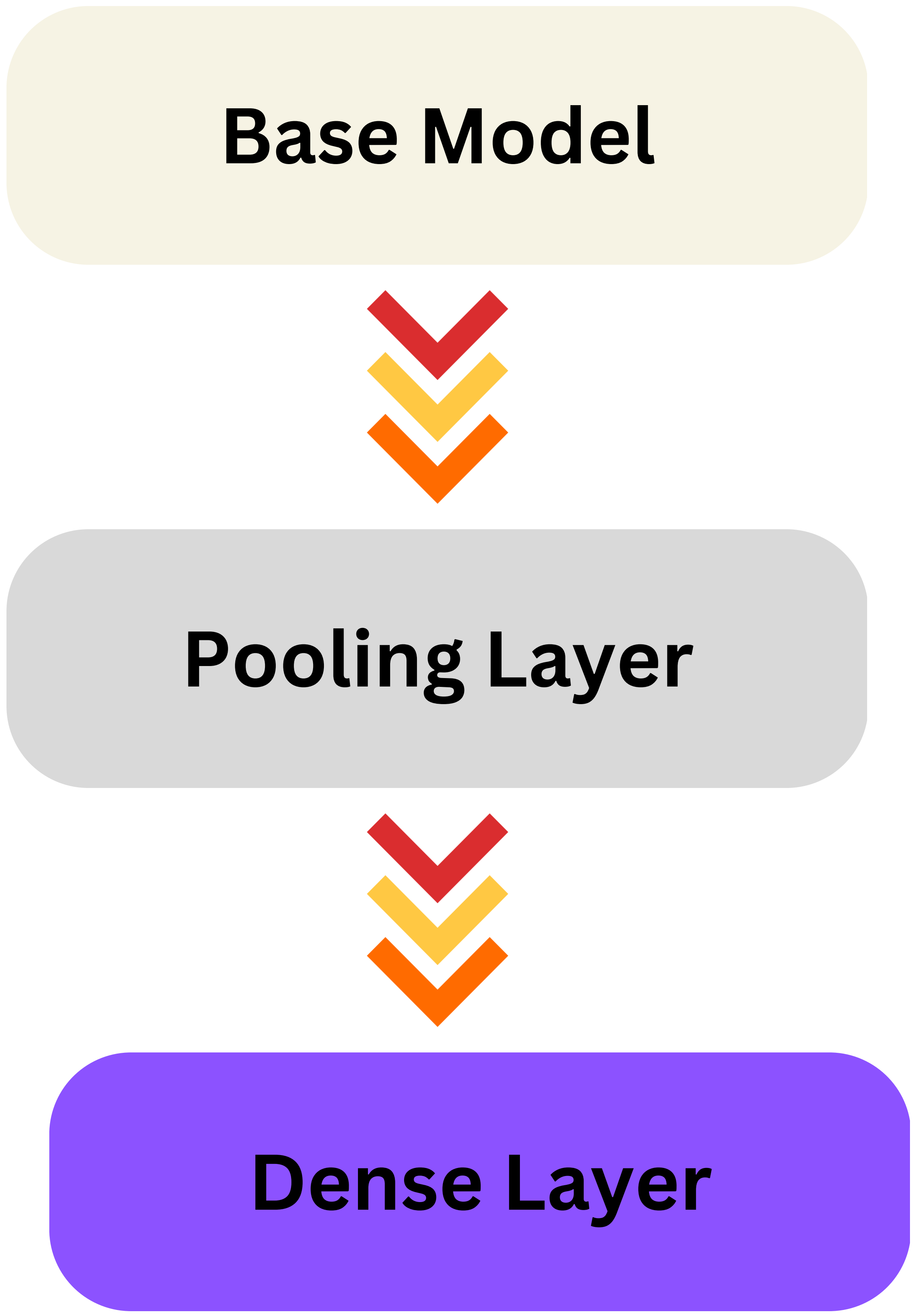
Analysis

- Using contextual embedding resulted in a higher F1 score
- Concatenation of contextual embeddings with more information gave the model better context
- Fine-tuning the pre-trained language model improved the results even further

BERT Best Model Parameters

Hyperparameters

Batch Size - 25
Epoch -1
Output Dim - 414
Loss Used -
BatchAllTripletLoss



Individual Contributions

Text Pre-processing -	Saharsh Dev, Ashwin Tomer, and Abhit Rana
Feature Extraction -	Sarthak Maini, Abhit Rana, and Saharsh Dev
Model Formation -	Sarthak Maini, Ashwin Tomer and Abhit Rana

Thank you!