UTA

THE UNIVERSITY OF TEXAS AT ARLINGTON

UNIVERSITY OF
TEXAS
ARLINGTON

Deep learning-based Sentiment Analysis in the Natural Language Processing Methodology

INTRODUCTION

- The main objective of our project is to create an advanced and robust **Sentiment Analysis model** using Deep Learning methods to gain insights from textual data.
- Our project leverages the power of **two diverse datasets**, each offering unique insights and presenting distinct challenges. This approach ensures a comprehensive analysis that accommodates various linguistic styles and contexts.
- We employ **three different types of embeddings** to enrich our analysis. These embeddings add depth and nuance to the model, allowing it to understand and interpret textual data with a higher level of sophistication.
- We have developed two hybrid models.
  - **CNN-LSTM model**
  - **CNN-LSTM-Transformer model**

DATASETS

- For our project, we are utilizing two distinct types of datasets. One of the datasets is **Twitter US Airline dataset**, which consists of **1 - 2 lines of tweets**, and the other is **IMDB Movie Review dataset**, which consists of **paragraph reviews**.
- Assessing the model's effectiveness in several scenarios, such as the Twitter US Airline dataset and the IMDB Movie Review dataset to determine its flexibility and capacity to generalize sentiment analysis across multiple textual sources.

TWITTER US AIRLINE DATASET

- It is collected from **Kaggle** and includes sentimental **tweets made by travelers** in February 2015 on Twitter.

  It consists of tweets related to **six major U.S. airlines: Delta, Virgin America, Southwest, United, American, and US Airways**.
- This dataset has 14640 rows and 15 attributes. They are as follows:

1. tweet_id
2. airline_sentiment
3. airline_sentiment_confidence
4. negativereason
5. negativereason_confidence
6. airline
7. airline_sentiment_gold
8. name
9. negativereason_gold
10. retweet_count
11. text
12. tweet_coord
13. tweet_created
14. tweet_location
15. user_timezone

IMDB DATASET

- It is collected from **Stanford University**.
- It is a dataset of **50k movie reviews** for **binary sentiment classification**.
- This dataset comprises tweets that are in a **paragraph format**.

EXPLORATORY DATA ANALYSIS

EDA – TWITTER US AIRLINE DATASET

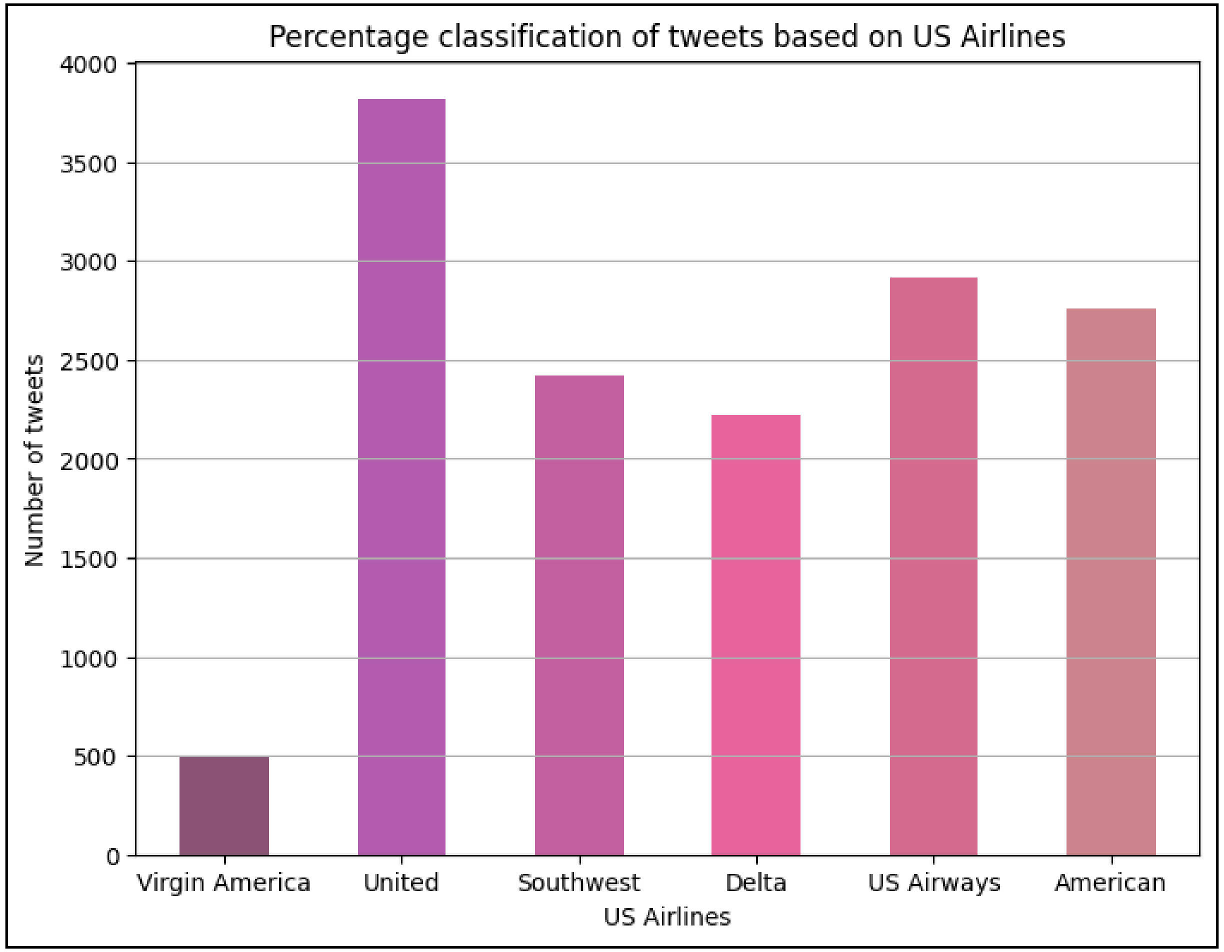


Fig 1.1: Percentage classification of tweets based on US Airlines

EDA – TWITTER US AIRLINE DATASET

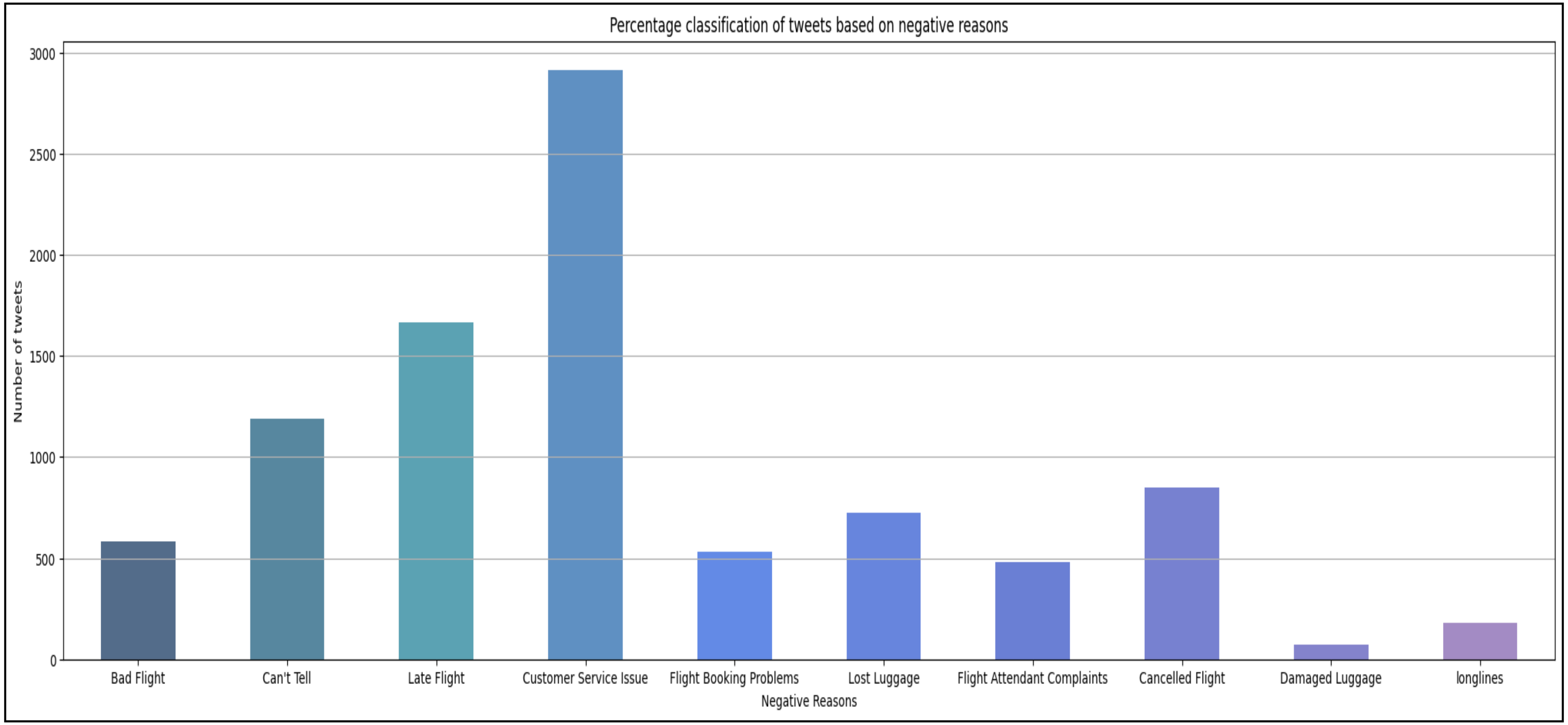


Fig 1.2: Percentage classification of tweets based on negative reasons

EDA – TWITTER US AIRLINE DATASET

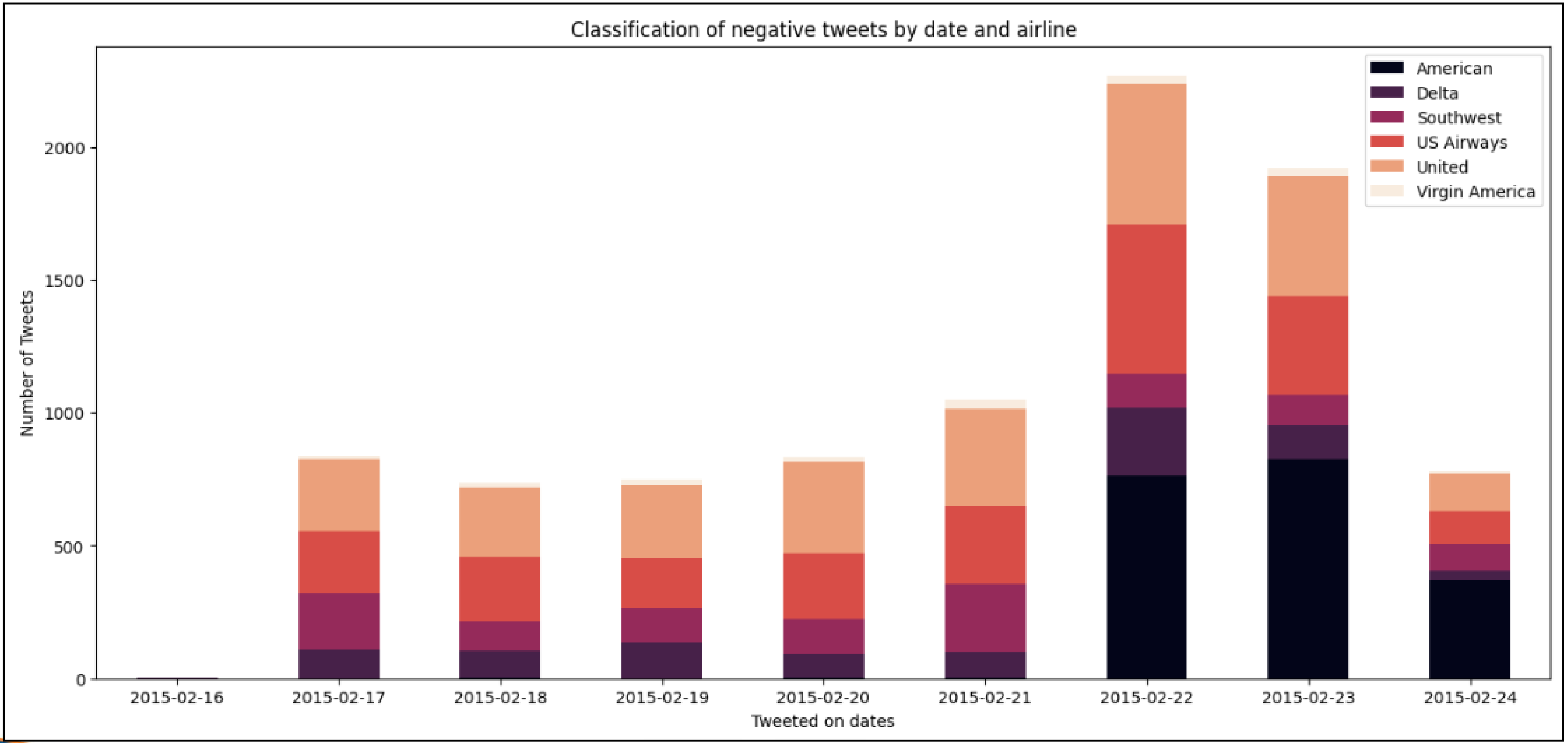


Fig 1.3: Classification of negative tweets by date and airline

EDA – TWITTER US AIRLINE DATASET

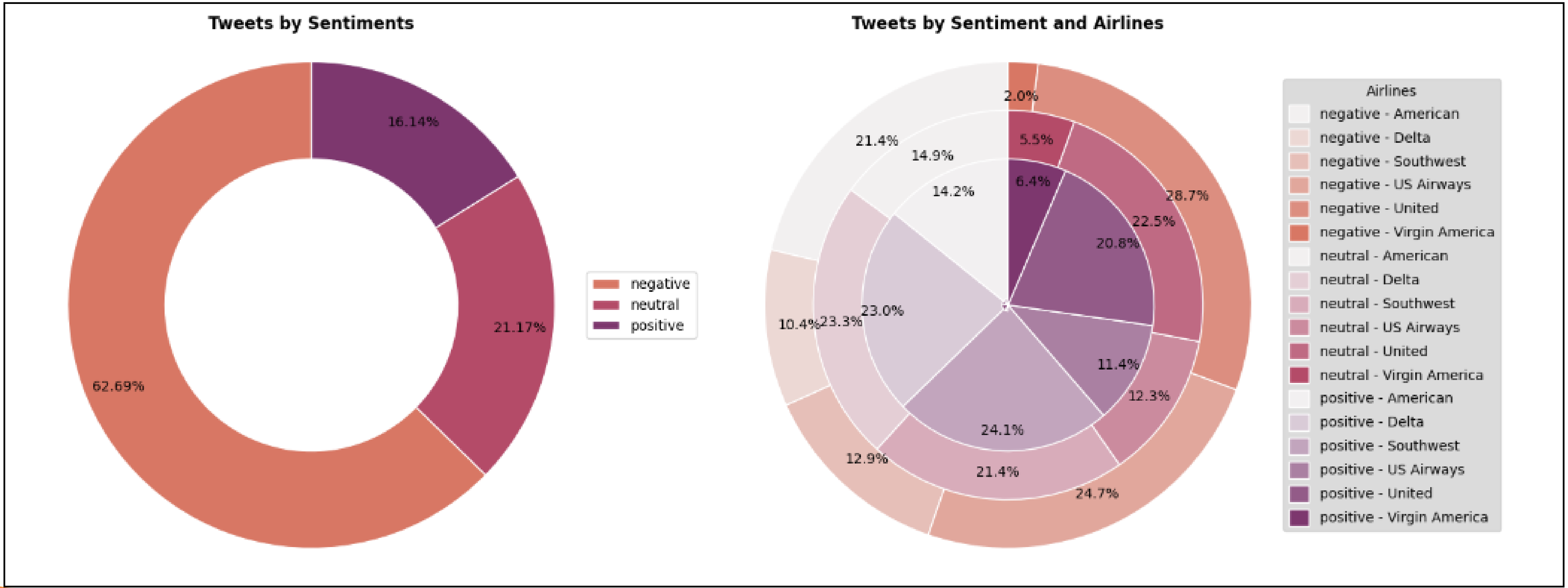


Fig 1.4: Tweets by sentiment

EDA – TWITTER US AIRLINE DATASET

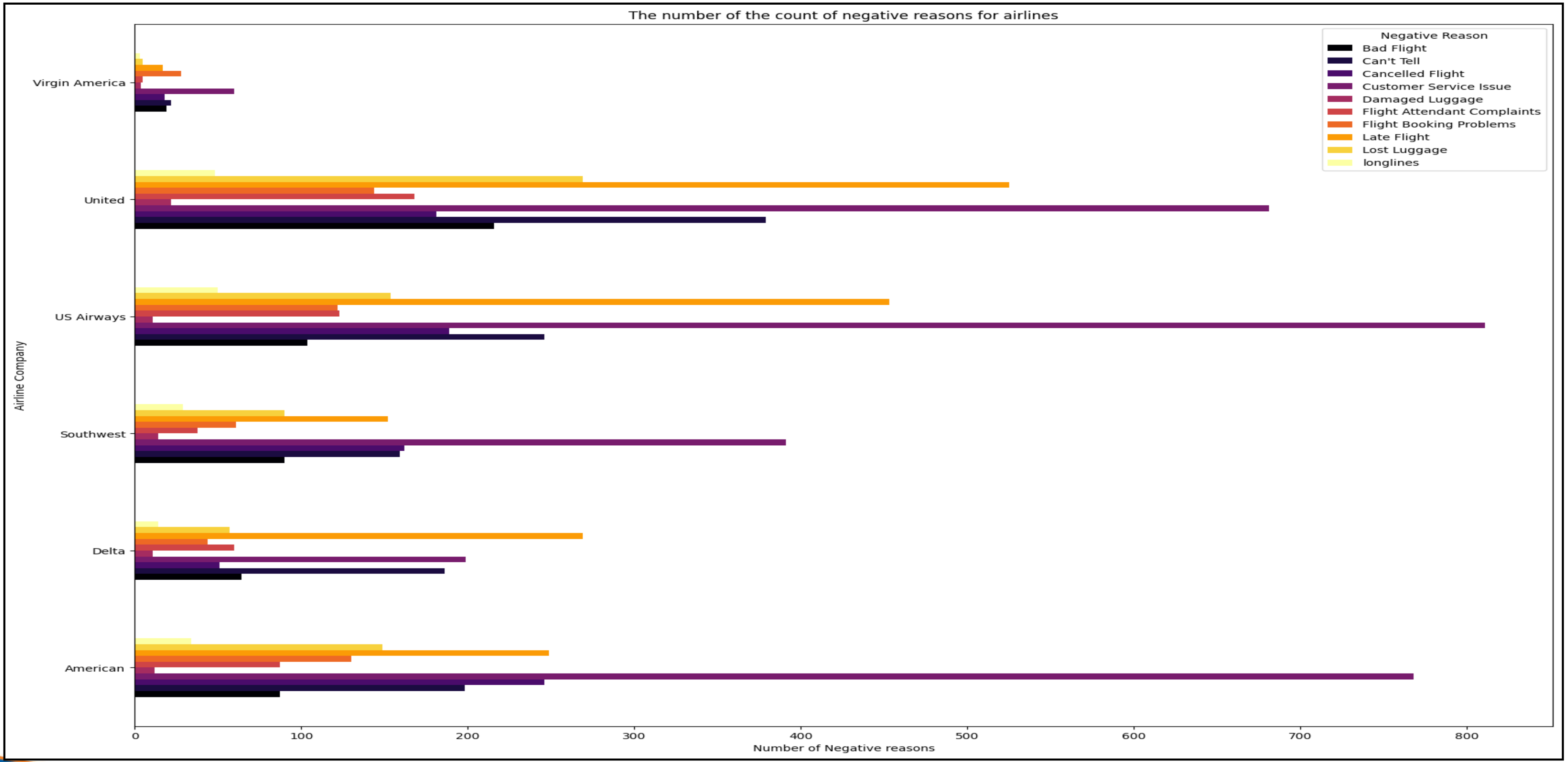


Fig 1.5: Tweets by sentiment

EDA – IMDB MOVIE REVIEWS DATASET

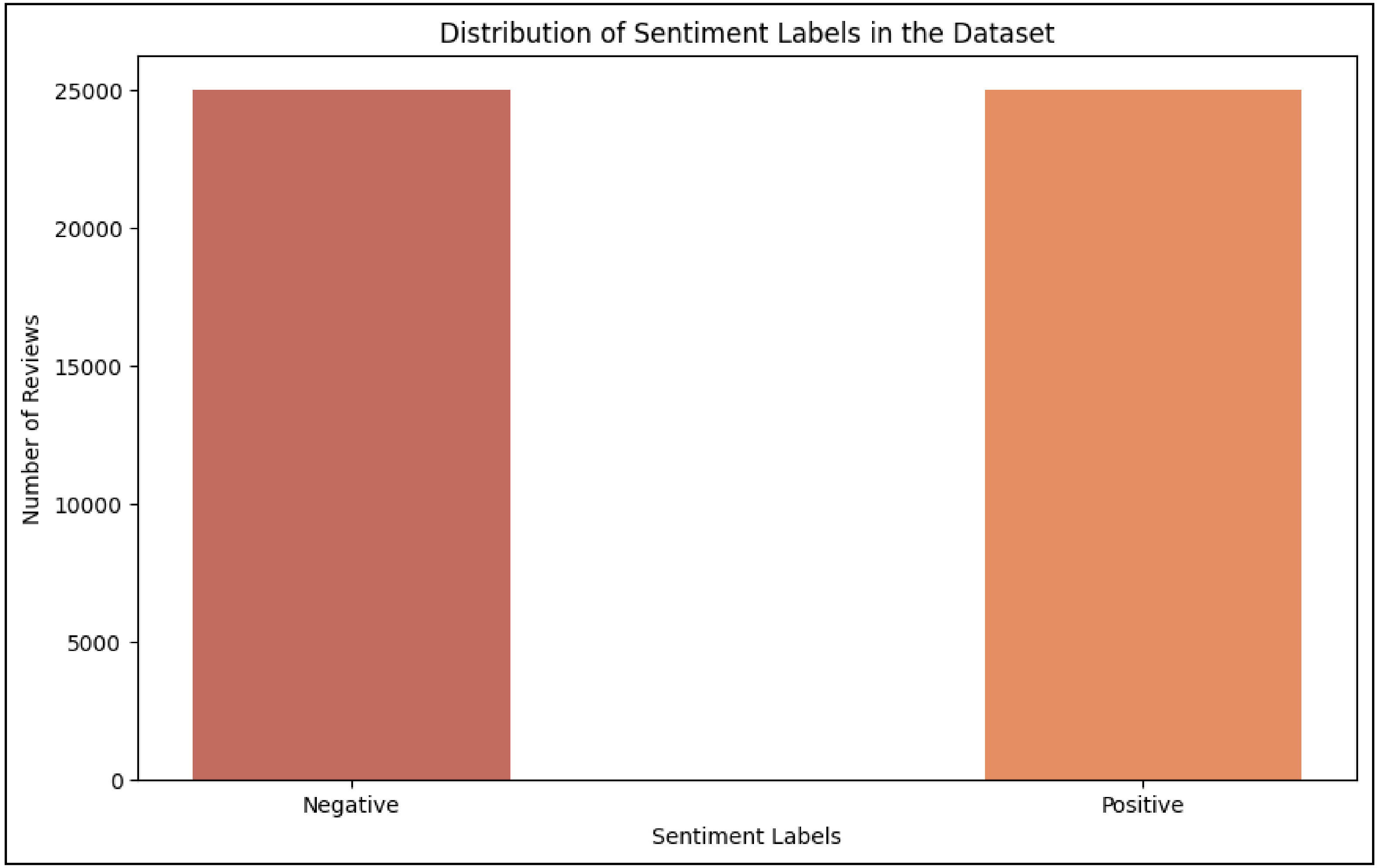
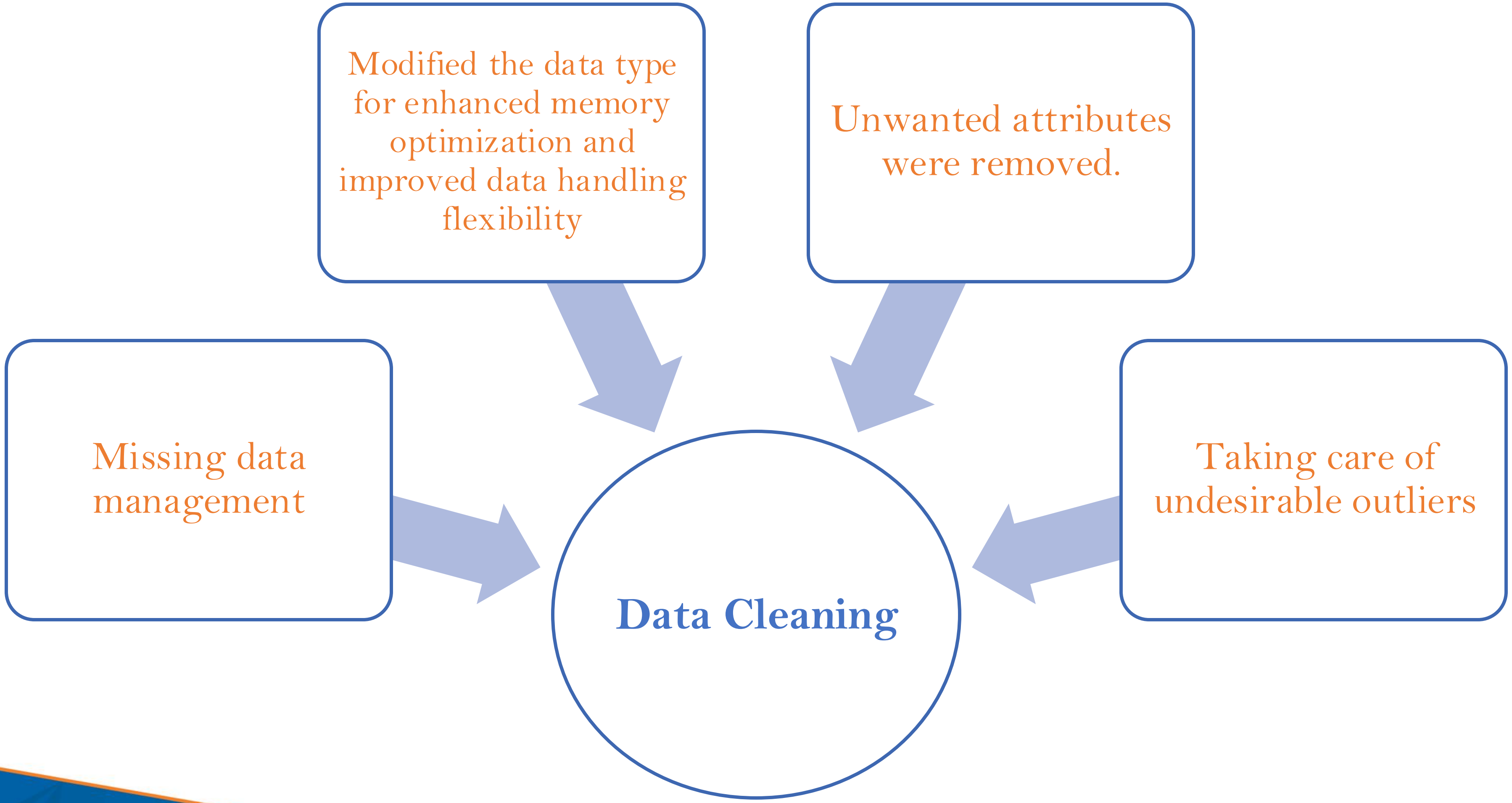


Fig 2.1 Distribution of sentiment analysis in the dataset

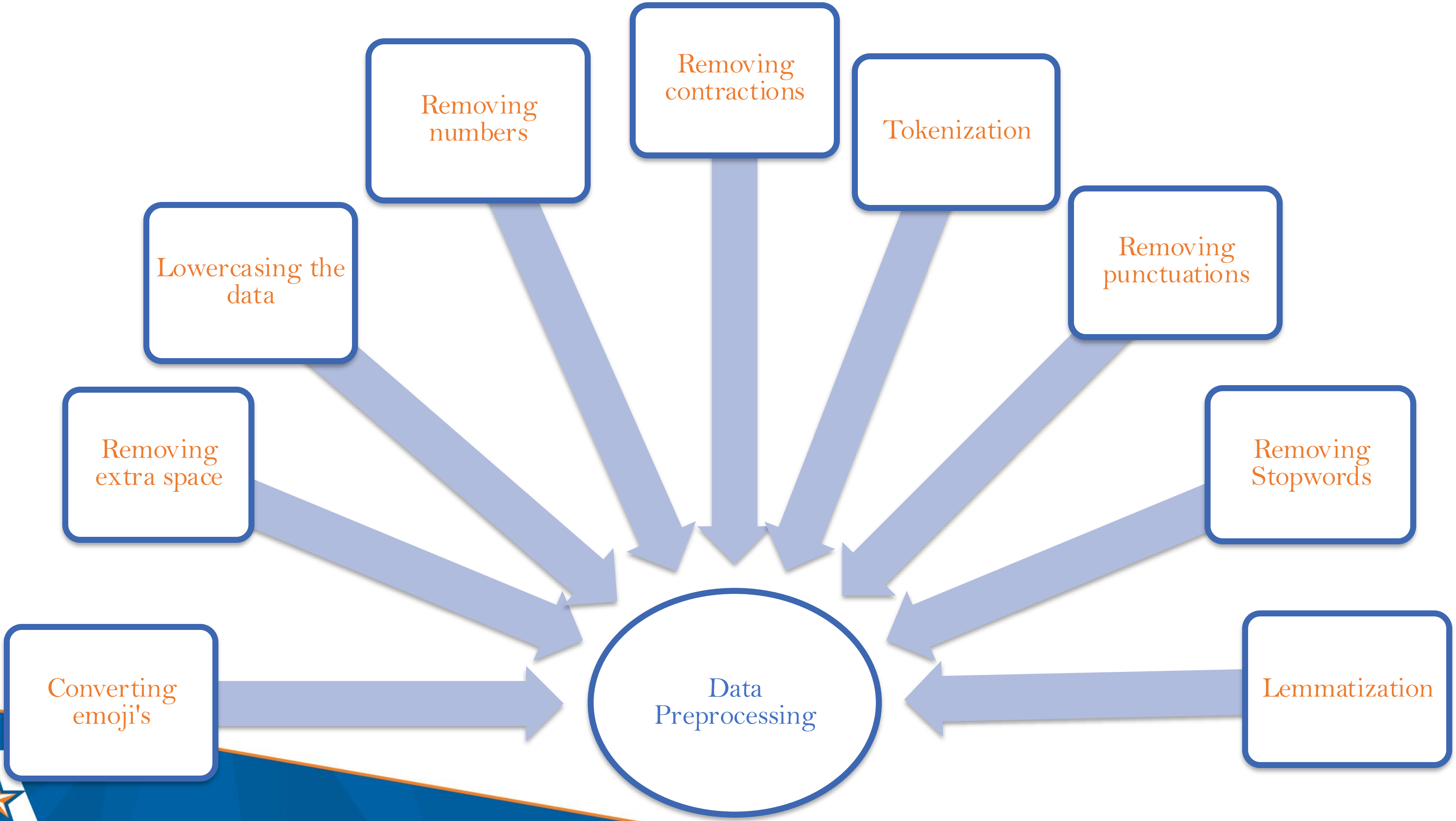
DATA PREPROCESSING

- The initial stage of data preparation, known as Data Preprocessing, involves **cleaning, transforming, and organizing raw data** to make it appropriate for analysis.
- The foundation of sound data analysis is Data Preprocessing. It has a direct impact on the quality and reliability of the outcomes.

DATA CLEANING



DATA PREPROCESSING



TWITTER US AIRLINE DATASET

DATA PREPROCESSING

Before Data Preprocessing

	airline_sentiment	airline	text
0	neutral	Virgin America	@VirginAmerica What @dhepburn said.
1	positive	Virgin America	@VirginAmerica plus you've added commercials to the experience... tacky.
2	neutral	Virgin America	@VirginAmerica I didn't today... Must mean I need to take another trip!
3	negative	Virgin America	@VirginAmerica it's really aggressive to blast obnoxious "entertainment" in your guests' faces & they have little recourse
4	negative	Virgin America	@VirginAmerica and it's a really big bad thing about it
5	negative	Virgin America	@VirginAmerica seriously would pay $30 a flight for seats that didn't have this playing.\nit's really the only bad thing about flying VA
6	positive	Virgin America	@VirginAmerica yes, nearly every time I fly VX this “ear worm” won’t go away :)
7	neutral	Virgin America	@VirginAmerica Really missed a prime opportunity for Men Without Hats parody, there. https://t.co/mWpG7grEZP
8	positive	Virgin America	@virginamerica Well, I didn't…but NOW I DO! :-D
9	positive	Virgin America	@VirginAmerica it was amazing, and arrived an hour early. You're too good to me.
10	neutral	Virgin America	@VirginAmerica did you know that suicide is the second leading cause of death among teens 10-24

DATA PREPROCESSING

After Data Preprocessing

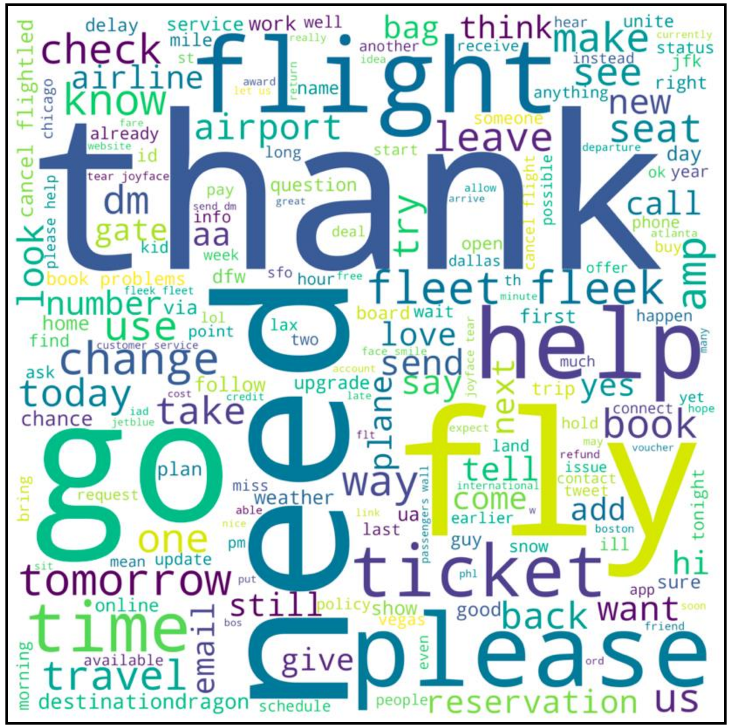
	airline_sentiment	airline	cl_txt
0	neutral	Virgin America	say
1	positive	Virgin America	plus add commercials experience tacky
2	neutral	Virgin America	today must mean need take another trip
3	negative	Virgin America	really aggressive blast obnoxious entertainment guests face amp little recourse
4	negative	Virgin America	really big bad thing
5	negative	Virgin America	seriously would pay seat play really bad thing fly va
6	positive	Virgin America	yes nearly every time fly vx ear worm go away
7	neutral	Virgin America	really miss prime opportunity men without hat parody
8	positive	Virgin America	well didntbut
9	positive	Virgin America	amaze arrive hour early good
10	neutral	Virgin America	know suicide second lead death among teens

WORDCLOUD

NEGATIVE WORDCLOUD

POSITIVE WORDCLOUD

NEUTRAL WORDCLOUD



IMDB DATASET

DATA PREPROCESSING

Before Data Preprocessing

	Reviews	Sentiment
0	Bizarre horror movie filled with famous faces but stolen by Cristina Raines (later of TV's "Flamingo Road") as a pretty but somewhat unstable model with a gummy smile who is slated to pay for her attempted suicides by guarding the Gateway to Hell! The scenes with Raines modeling are very well captured, the mood music is perfect, Deborah Raffin is charming as Cristina's pal, but when Raines moves into a creepy Brooklyn Heights brownstone (inhabited by a blind priest on the top floor), things really start cooking. The neighbors, including a fantastically wicked Burgess Meredith and kinky couple Sylvia Miles & Beverly D'Angelo, are a diabolical lot, and Eli Wallach is great fun as a wily police detective. The movie is nearly a cross-pollination of "Rosemary's Baby" and "The Exorcist"--but what a combination! Based on the best-seller by Jeffrey Konvitz, "The Sentinel" is entertainingly spooky, full of shocks brought off well by director Michael Winner, who mounts a thoughtfully downbeat ending with skill. ***1/2 from ****	Positive
1	A solid, if unremarkable film. Matthau, as Einstein, was wonderful. My favorite part, and the only thing that would make me go out of my way to see this again, was the wonderful scene with the physicists playing badmitton, I loved the sweaters and the conversation while they waited for Robbins to retrieve the birdie.	Positive
2	It's a strange feeling to sit alone in a theater occupied by parents and their rollicking kids. I felt like instead of a movie ticket, I should have been given a NAMBLA membership. Based upon Thomas Rockwell's respected Book, How To Eat Fried Worms starts like any children's story: moving to a new town. The new kid, fifth grader Billy Forrester was once popular, but has to start anew. Making friends is never easy, especially when the only prospect is Poindexter Adam. Or Erica, who at 4 1/2 feet, is a giant. Further complicating things is Joe the bully. His freckled face and sleeveless shirts are daunting. He antagonizes kids with the Death Ring: a Crackerjack ring that is rumored to kill you if you're punched with it. But not immediately. No, the death ring unleashes a poison that kills you in the eight grade. Joe and his axis of evil welcome Billy by smuggling a handful of slimy worms into his thermos. Once discovered, Billy plays it cool, swearing that he eats worms all the time. Then he throws them at Joe's face. Ewww! To win them over, Billy reluctantly bets that he can eat 10 worms. Fried, boiled, marinated in hot sauce, squashed and spread on a peanut butter sandwich. Each meal is dubbed an exotic name like the "Radioactive Slime Delight," in which the kids finally live out their dream of microwaving a living organism. If you've ever met me, you'll know that I have an uncontrollably hearty laugh. I felt like a creep erupting at a toddler whining that his "dilly dick" hurts. But Fried Worms is wonderfully disgusting. Like a G-rated Farrelly brothers film, it is both vomitous and delightful. Writer/director Bob Dolman is also a savvy storyteller. To raise the stakes the worms must be consumed by 7 pm. In addition Billy holds a dark secret: he has an ultra-sensitive stomach. Dolman also has a keen sense of perspective. With such accuracy, he draws on children's insecurities and tendency to exaggerate mundane dilemmas. If you were to hyperbolize this movie the way kids do their quandaries, you will see that it is essentially about war. Freedom-fighter and freedom-hater use pubescent boys as pawns in proxy wars, only to learn a valuable lesson in unity. International leaders can learn a thing or two about global peacekeeping from Fried Worms. At the end of the film, I was comforted when two chaperoning mothers behind me, looked at each other with befuddlement and agreed, "That was a great movie." Great, now I won't have to register myself in any lawful databases.	Positive
3	You probably all already know this by now, but 5 additional episodes never aired can be viewed on ABC.com I've watched a lot of television over the years and this is possibly my favorite show, ever. It's a crime that this beautifully written and acted show was canceled. The actors that played Laura, Whit, Carlos, Mae, Damian, Anya and omg, Steven Caseman - are all incredible and so natural in those roles. Even the kids are great. Wonderful show. So sad that it's gone. Of course I wonder about the reasons it was canceled. There is no way I'll let myself believe that Ms. Moynahan's pregnancy had anything to do with it. It was in the perfect time slot in this market. I've watched all the episodes again on ABC.com - I hope they all come out on DVD some day. Thanks for reading.	Positive
4	I saw the movie with two grown children. Although it was not as clever as Shrek, I thought it was rather good. In a movie theatre surrounded by children who were on spring break, there was not a sound so I know the children all liked it. There parents also seemed engaged. The death and apparent death of characters brought about the appropriate gasps and comments. Hopefully people realize this movie was made for kids. As such, it was successful although I liked it too. Personally I liked the Scrat!!	Positive

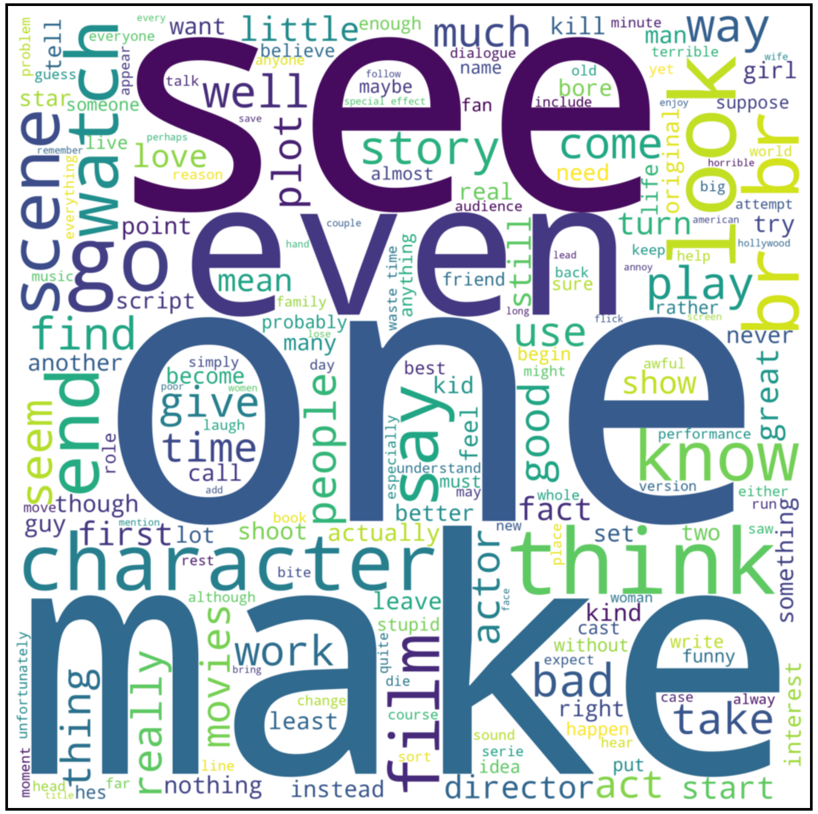
DATA PREPROCESSING

After Data Preprocessing

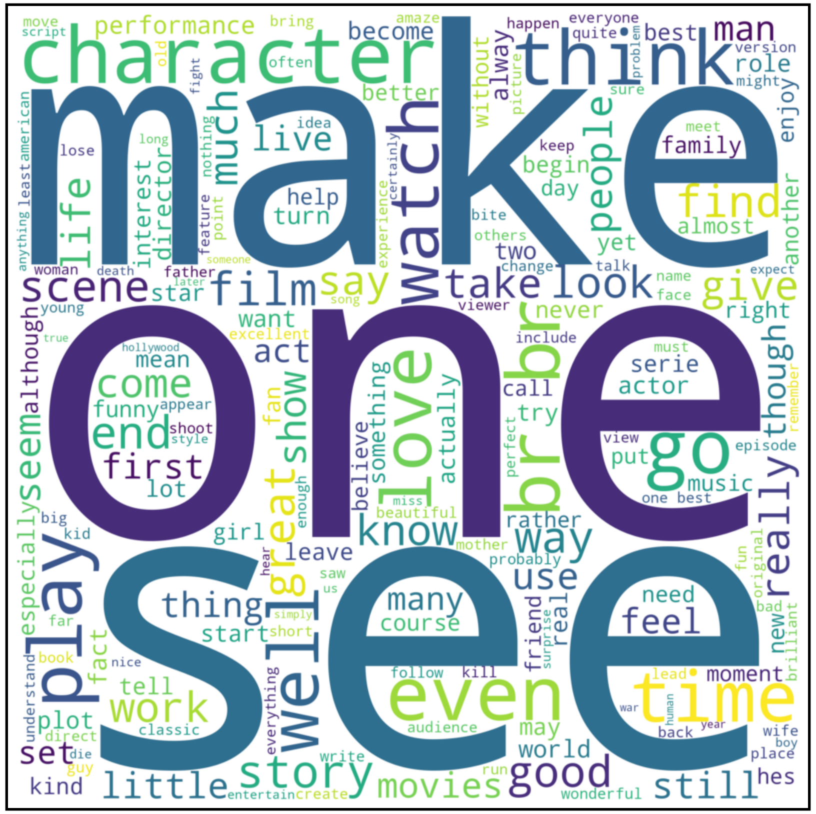
	Sentiment	cl_txt
0	Positive	bizarre fill famous face steal cristina rain later tvs flamingo road pretty somewhat unstable model gummy smile slat pay attempt suicides guard gateway hell scenes rain model well capture mood music perfect deborah raffin charm cristinas pal rain move creepy brooklyn heights brownstone inhabit blind priest top floor things really start cook neighbor include fantastically wicked burgess meredith kinky couple sylvia miles beverly dangelo diabolical lot eli wallach great fun wily police detective nearly crosspollination rosemarys baby exorcistbut combination base bestseller jeffrey konvitz sentinel entertainingly spooky full shock bring well director michael winner mount thoughtfully downbeat end skill
1	Positive	solid unremarkable matthau einstein wonderful favorite thing would make go way see wonderful scene physicists play badmitton love sweaters conversation wait robbins retrieve birdie
2	Positive	strange feel sit alone theater occupy parent rollick kid felt like instead ticket give nambla membershipbr br base upon thomas rockwells respect book eat fry worm start like childrens story move new town new kid fifth grader billy forrester popular start anew make friends never easy especially prospect poindexter adam erica feet giantbr br complicate things joe bully freckle face sleeveless shirt daunt antagonize kid death ring crackerjack ring rumor kill punch immediately death ring unleash poison kill eight gradebr br joe axis evil welcome billy smuggle handful slimy worm thermos discover billy play cool swear eat worm time throw joes face ewww win billy reluctantly bet eat worm fry boil marinate hot sauce squash spread peanut butter sandwich meal dub exotic name like radioactive slime delight kid finally live dream microwave live organismbr br ever meet know uncontrollably hearty laugh felt like creep erupt toddler whine dilly dick hurt fry worm wonderfully disgust like grate farrelly brothers vomitous delightfulbr br writerdirector bob dolman also savvy storyteller raise stake worm must consume pm addition billy hold dark secret ultrasensitive stomachbr br dolman also keen sense perspective accuracy draw childrens insecurities tendency exaggerate mundane dilemmasbr br hyperbolize way kid quandaries see essentially war freedomfighter freedomhater use pubescent boys pawn proxy war learn valuable lesson unity international leaders learn thing two global peacekeeping fry wormsbr br end comfort two chaperon mother behind look befuddlement agree great great register lawful databases
3	Positive	probably already know additional episodes never air view abccom watch lot television years possibly favorite ever crime beautifully write act cancel actors play laura whit carlos mae damian anya omg steven caseman incredible natural roles even kid great wonderful sad go course wonder reason cancel way ill let believe ms moynahans pregnancy anything perfect time slot market watch episodes abccom hope come dvd day thank read
4	Positive	saw two grow children although clever shrek think rather good surround children spring break sound know children like parent also seem engage death apparent death character bring appropriate gasp comment hopefully people realize make kid successful although like personally like scrat

WORDCLOUD

NEGATIVE WORDCLOUD



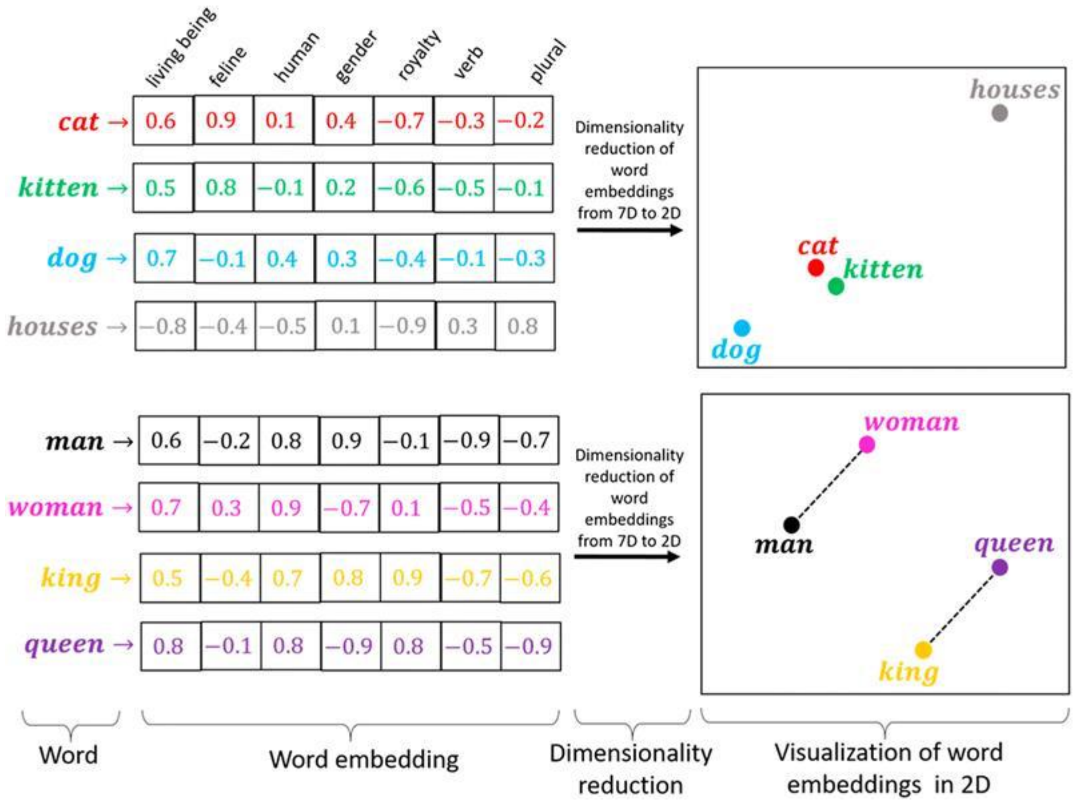
POSITIVE WORDCLOUD



WORD EMBEDDINGS

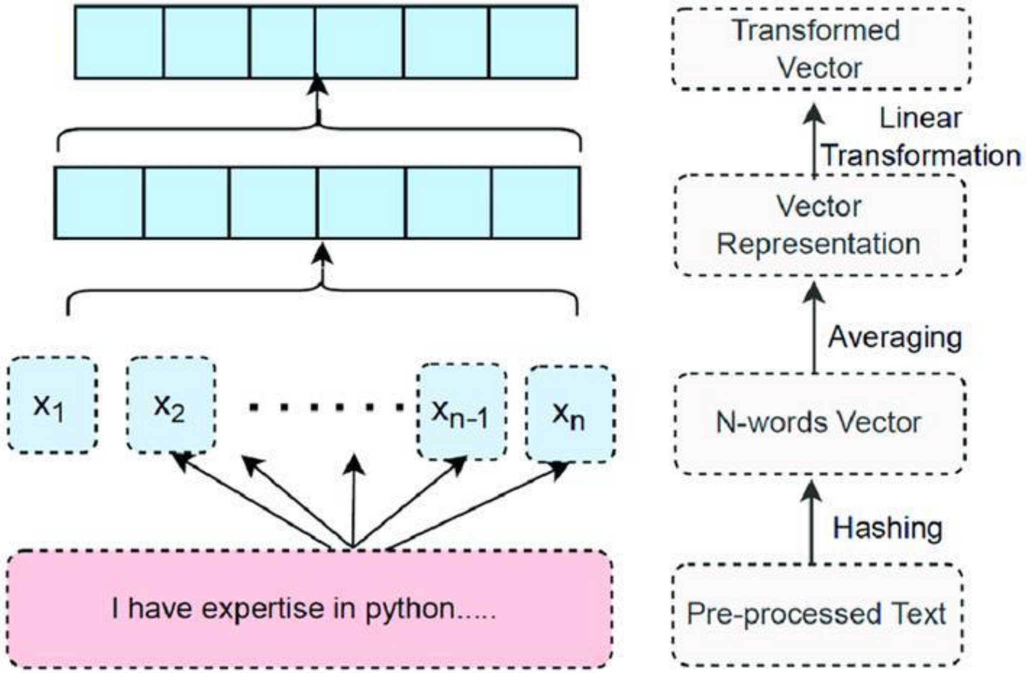
- Word2Vec
- fastText
- Keras Embedding

Word2Vec



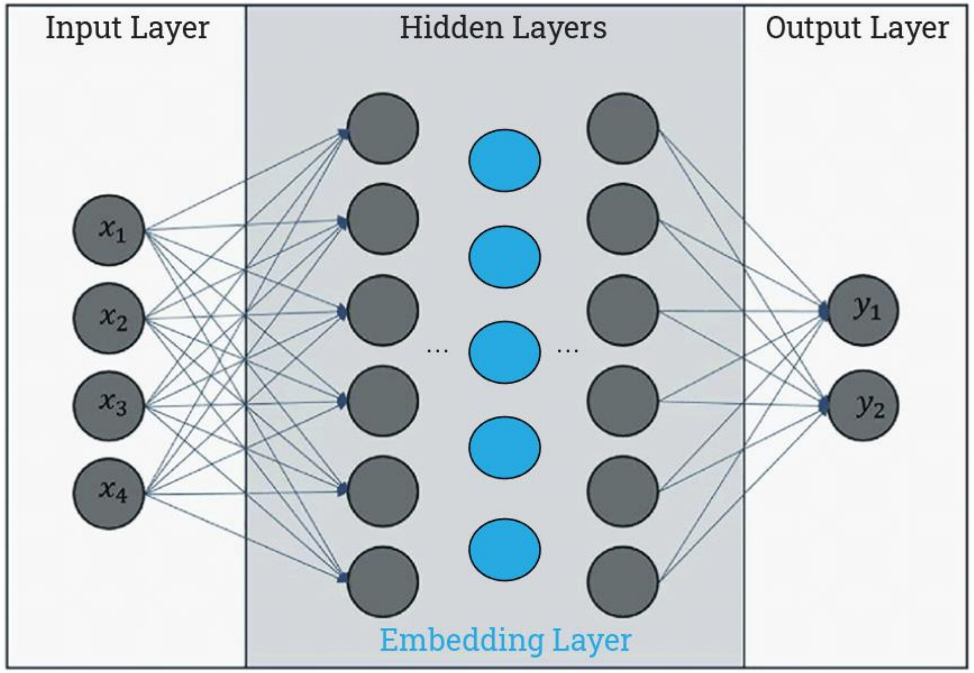
- **Word2Vec (Word to Vector)** is a prominent word embedding technique in natural language processing (NLP) that **expresses words as dense vectors** in a continuous vector space.
- Benefits of Using Word2Vec:
    - Understanding Semantics
    - Reduced Dimensionality
    - Generalization
    - Contextual Information

fastText



- **fastText** is an open-source, free, and lightweight software that enables users to learn text representations and text classifiers.
- **Common Crawl fastText** refers to the usage of the fastText natural language processing toolkit to analyze and process text data taken from the **Common Crawl web dataset**, enabling a wide range of text-based applications such as web content classification and analysis.
- On **Common Crawl**, 2 Million word vectors were learned **using sub-word information** (600B tokens).

Keras Embedding Layer



- Keras provides an Embedding layer for neural networks using text data. It requires integer encoding of the input data, so that each word is represented by a unique integer.
- It is a versatile layer that may be utilized in a variety of ways, including:
  - Learning a word embedding that can be saved and reused in another model later.
  - Can be utilized in a deep learning model, where the embedding is taught alongside the model.
  - Capable of loading a pre-trained word embedding model, which is a sort of transfer learning.

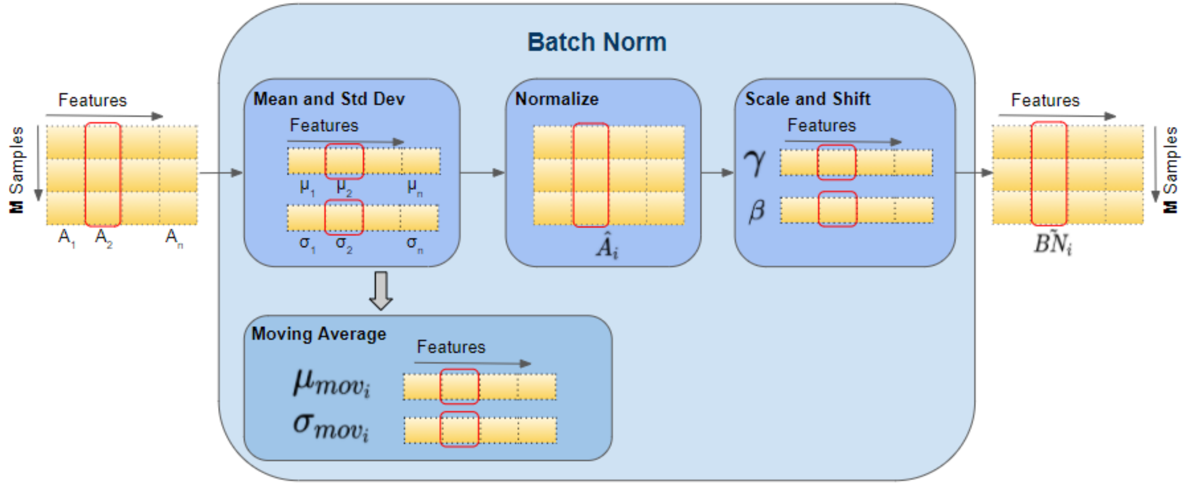
Why three types of Embeddings?

Why three types of Embeddings?

- fastText extends Word2Vec by considering sub-word information, allowing it to represent words as a combination of their sub-word components (character n-grams). This means fastText can handle out-of-vocabulary words and capture the meaning of morphemes and affixes.
- Keras Embeddings are integrated into the Keras deep learning framework. fastText and Word2Vec, on the other hand, are standalone techniques for generating word embeddings.

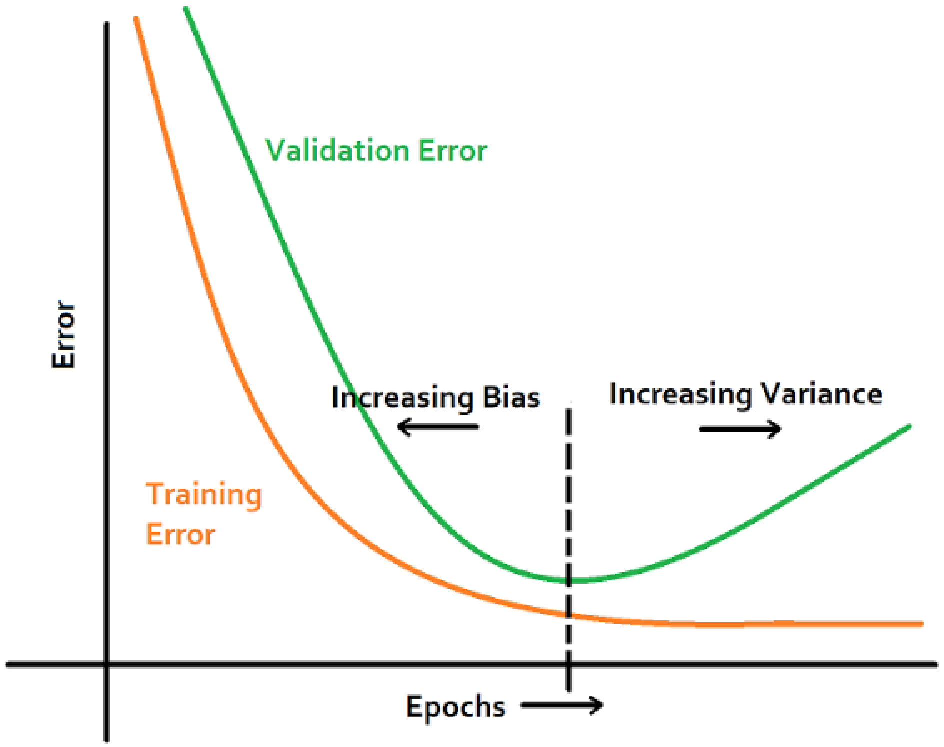
Regularization & Optimization Techniques

BATCH NORMALIZATION



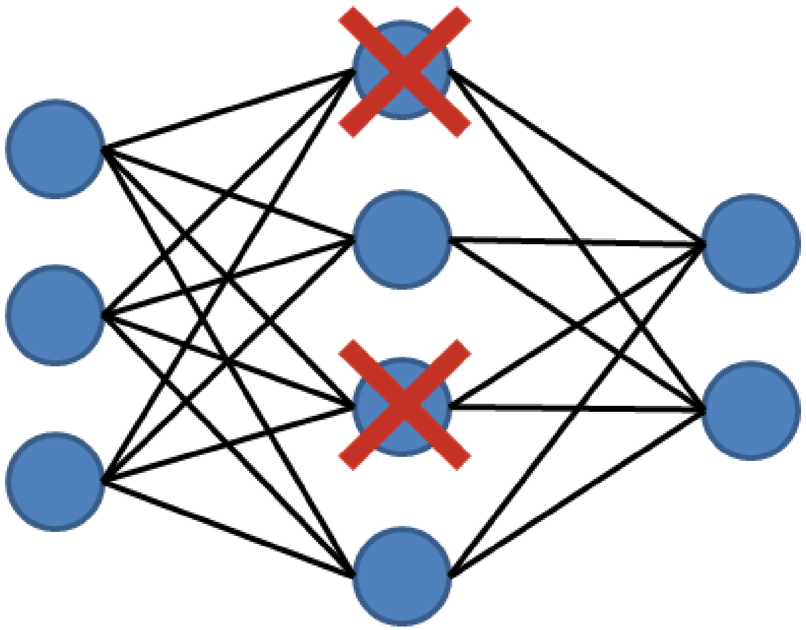
Batch normalization helps in mitigating the internal covariate shift problem, which is the change in the distribution of the inputs to a neural network layer during training. It normalizes the input of each layer in a mini-batch, making the optimization process more stable and reducing the risk of overfitting.

EARLY STOPPING



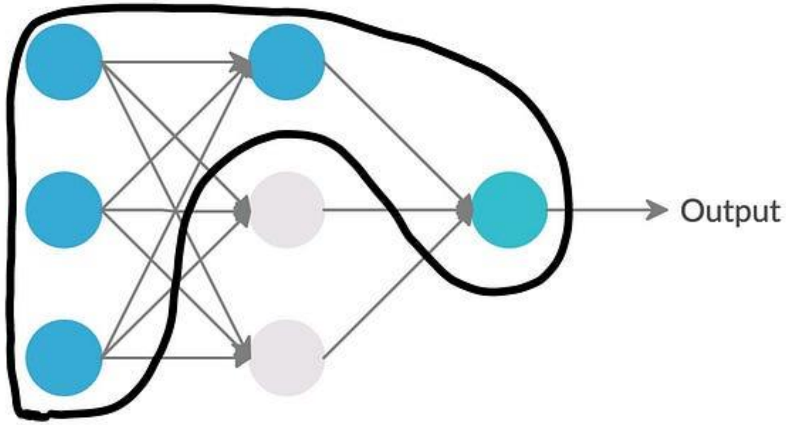
Early stopping is a regularization technique that helps prevent overfitting by stopping the training process once the performance on a validation set starts to degrade. It prevents the model from continuing to learn noise in the training data.

DROPOUT LAYER



Dropout is a regularization technique where randomly selected neurons are ignored during training. This prevents the network from relying too much on specific neurons and helps to improve generalization.

L2 REGULARIZATION

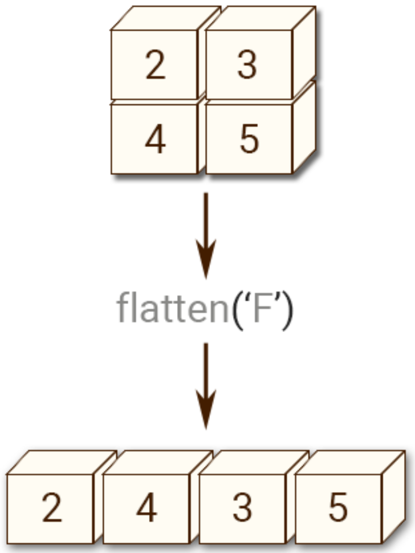


Simplified architecture due to low weights (L2 Norm): Less prone to overfitting

L2 regularization adds a penalty term to the loss function based on the squared magnitudes of the weights. This penalizes large weights and discourages the model from fitting the training data too closely, helping to prevent overfitting.

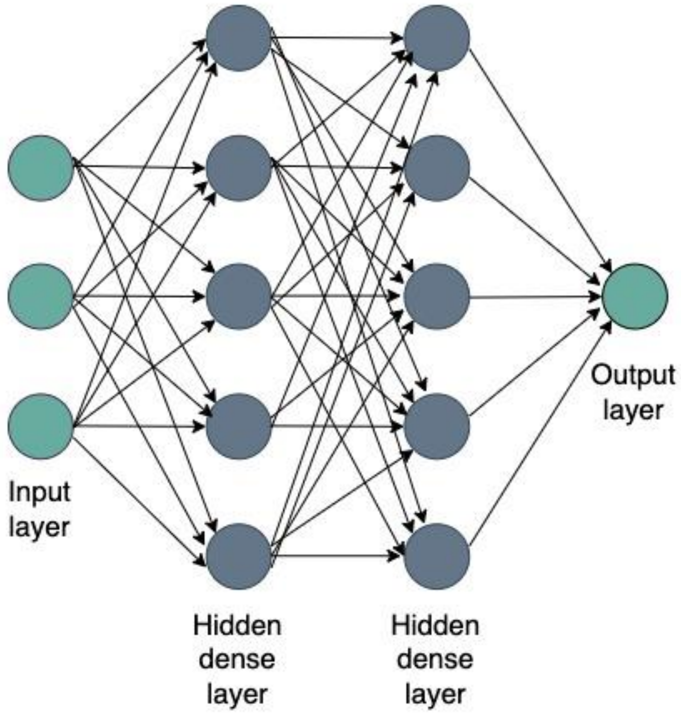
UTA

FLATTENING LAYER



Flattening is a process in neural networks where the multi-dimensional data (e.g., the output of convolutional layers) is converted into a one-dimensional array. It is often used before feeding the data into fully connected layers.

DENSE LAYERS



Employing dense layers alongside dropout or weight regularization techniques and fine-tuning the network's complexity by adjusting the number of layers and neurons can help prevent overfitting in neural networks.

ENHANCEMENTS IN OUR MODEL

- **Emojis as Contextual Cues:** Unlike the reference publications, we acknowledge that the way sentiment is conveyed in textual data is significantly influenced by emojis. These symbols provide our machine a better understanding of the context, which improves the accuracy and sentiment analysis. Emojis have become an important aspect of modern communication, particularly in textual data like social media postings, chats, and online reviews. They act as vital contextual signals, conveying emotions, tone, and mood in ways that mere text cannot do. Understanding the subtleties communicated by emojis is critical for machine and algorithm sentiment analysis.
- **Comparative Analysis:** We evaluated how effectively emojis are incorporated for improved contextual comprehension and how each approach affects sentiment classification accuracy by correlating fastText, Keras Embedding Layer, and Word2Vec in our CNN-LSTM and CNN-LSTM-Transformer sentiment analysis models.

CNN-LSTM
HYBRID MODEL

CONVOLUTION NEURAL NETWORK - LONG SHORT-TERM MEMORY (CNN-LSTM) HYBRID MODEL

In this project we are using CNN-LSTM Hybrid model.

CNN Model in Sentiment Analysis:

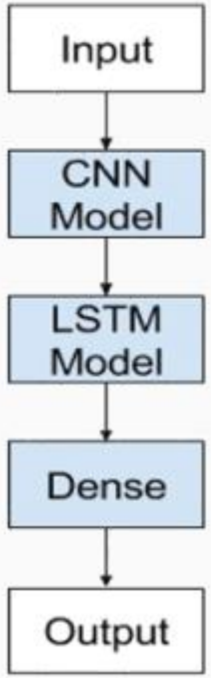
- Local Feature Learning
- Robustness to Noise

LSTM Model in Sentiment Analysis:

- Processing Sequential Information
- Managing Variable-Length Sequences

UTA

ARCHITECTURE OF OUR PROPOSED CNN-LSTM HYBRID MODEL



Convolutional Neural Network Long Short-Term Memory Network Architecture

REASONS FOR IMPLEMENTING CNN-LSTM HYBRID MODEL

- **Strengths When Used Together:** Using the benefits of both CNNs and LSTMs, the hybrid model can simultaneously capture dependence over time and local attributes.
- **Better Feature Extraction:** Local characteristics may be effectively extracted by the CNN component and then fed into the LSTM to capture the context and sequential information, resulting in a more comprehensive understanding of the text.
- **Reduced Overfitting:** By combining the power of the LSTM to generalize patterns across sequences and the CNN's feature extraction capabilities, the combination of the two models lessens overfitting.
- **Complex Context Understanding:** When it comes to understanding both local and global context, our hybrid approach performs better, which may produce better results for sentiment analysis in text data.

TWITTER US AIRLINE DATASET

SUMMARY OF MODELS

Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 300)	3270000
conv1d (Conv1D)	(None, 98, 256)	230656
max_pooling1d (MaxPooling1D)	(None, 49, 256)	0
dropout (Dropout)	(None, 49, 256)	0
conv1d_1 (Conv1D)	(None, 47, 256)	196864
max_pooling1d_1 (MaxPooling1D)	(None, 47, 256)	0
lstm (LSTM)	(None, 47, 256)	525312
lstm_1 (LSTM)	(None, 256)	525312
dropout_1 (Dropout)	(None, 256)	0
dense (Dense)	(None, 128)	32896
dense_1 (Dense)	(None, 1)	129

fastText

Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 100)	1090000
conv1d (Conv1D)	(None, 96, 256)	128256
max_pooling1d (MaxPooling1D)	(None, 48, 256)	0
dropout (Dropout)	(None, 48, 256)	0
conv1d_1 (Conv1D)	(None, 46, 256)	196864
max_pooling1d_1 (MaxPooling1D)	(None, 23, 256)	0
attention_wrapper (AttentionWrapper)	(None, 23, 256)	1
lstm (LSTM)	(None, 23, 128)	197120
lstm_1 (LSTM)	(None, 128)	131584
dropout_1 (Dropout)	(None, 128)	0
batch_normalization (BatchNormalization)	(None, 128)	512
flatten (Flatten)	(None, 128)	0
dense (Dense)	(None, 128)	16512
dense_1 (Dense)	(None, 1)	129

Keras Embedding Layer

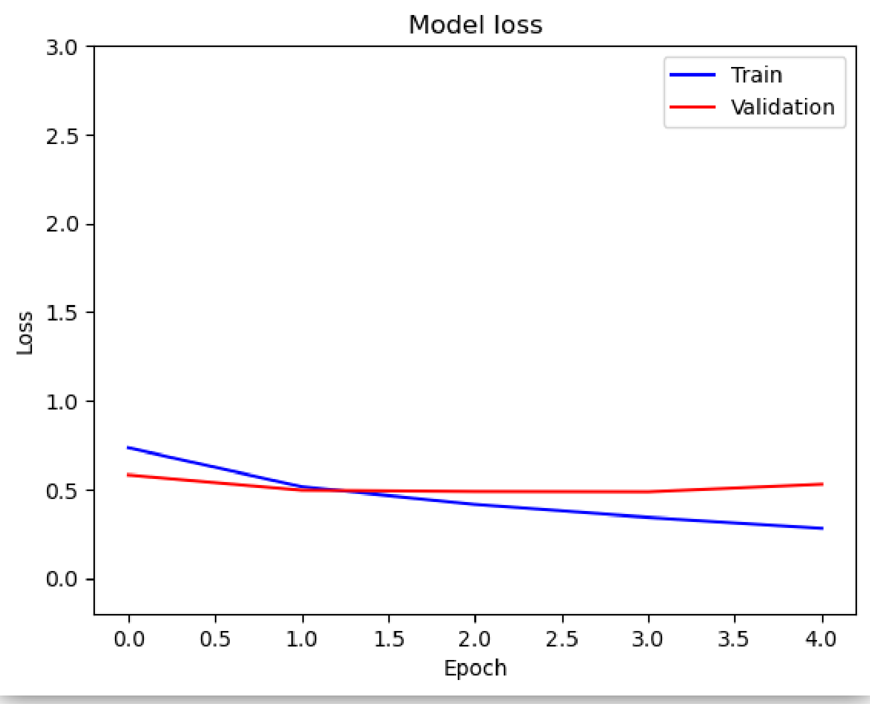
Model: "sequential"

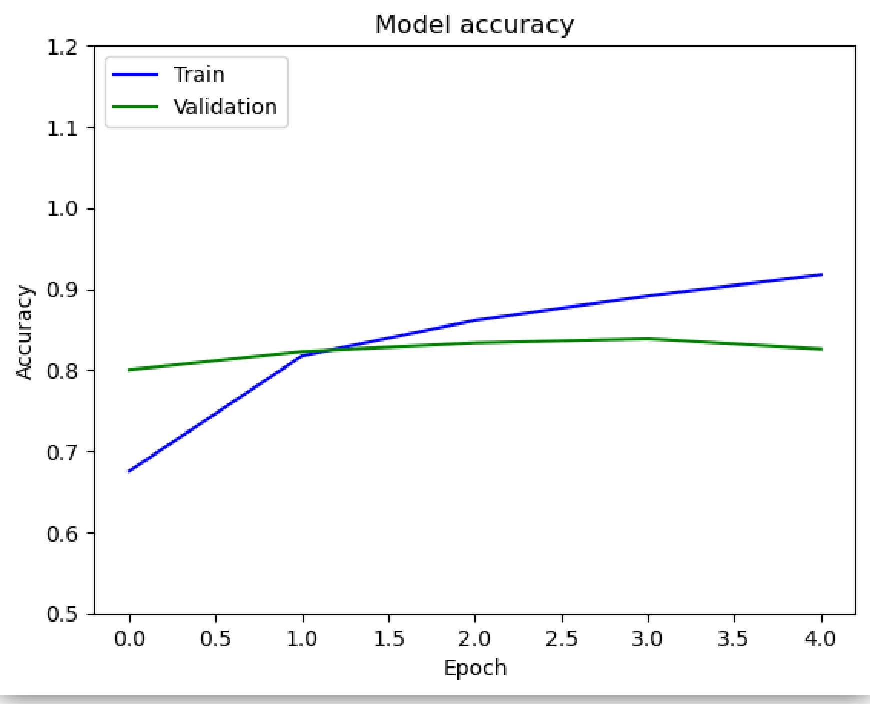
Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 300)	3270000
conv1d (Conv1D)	(None, 98, 256)	230656
max_pooling1d (MaxPooling1D)	(None, 49, 256)	0
dropout (Dropout)	(None, 49, 256)	0
conv1d_1 (Conv1D)	(None, 47, 256)	196864
max_pooling1d_1 (MaxPooling1D)	(None, 47, 256)	0
lstm (LSTM)	(None, 47, 256)	525312
lstm_1 (LSTM)	(None, 256)	525312
dropout_1 (Dropout)	(None, 256)	0
dense (Dense)	(None, 128)	32896
dense_1 (Dense)	(None, 1)	129

Word2Vec

- The summary shows that we have used three different embeddings.
- In each embedding we used different layers to get the highest accuracy possible.

RESULTS – TWITTER DATASET FASTTEXT EMBEDDING

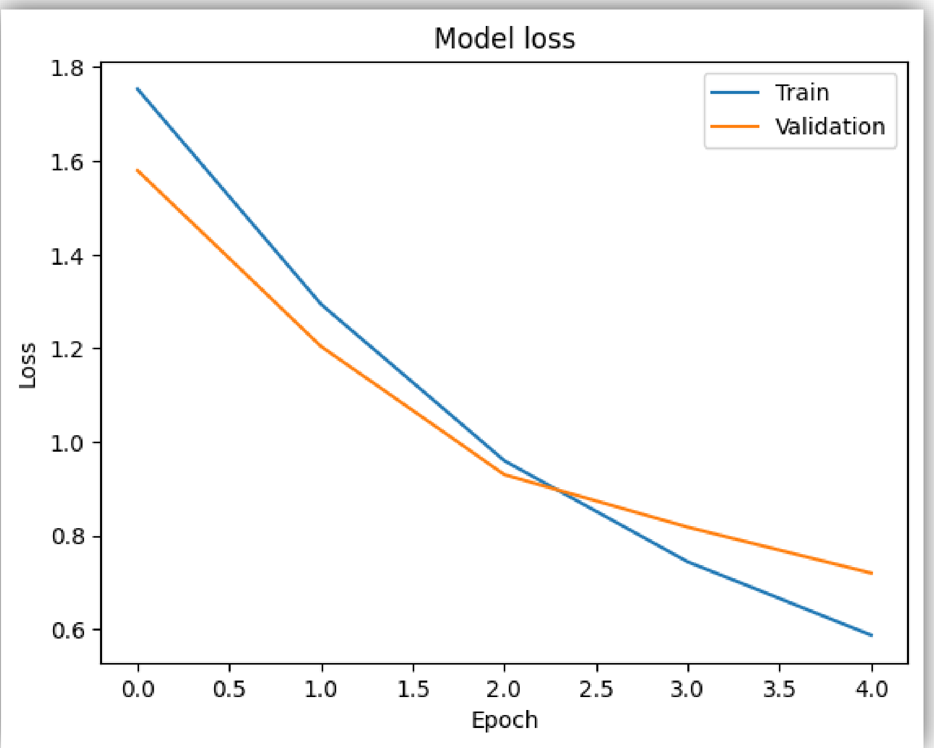


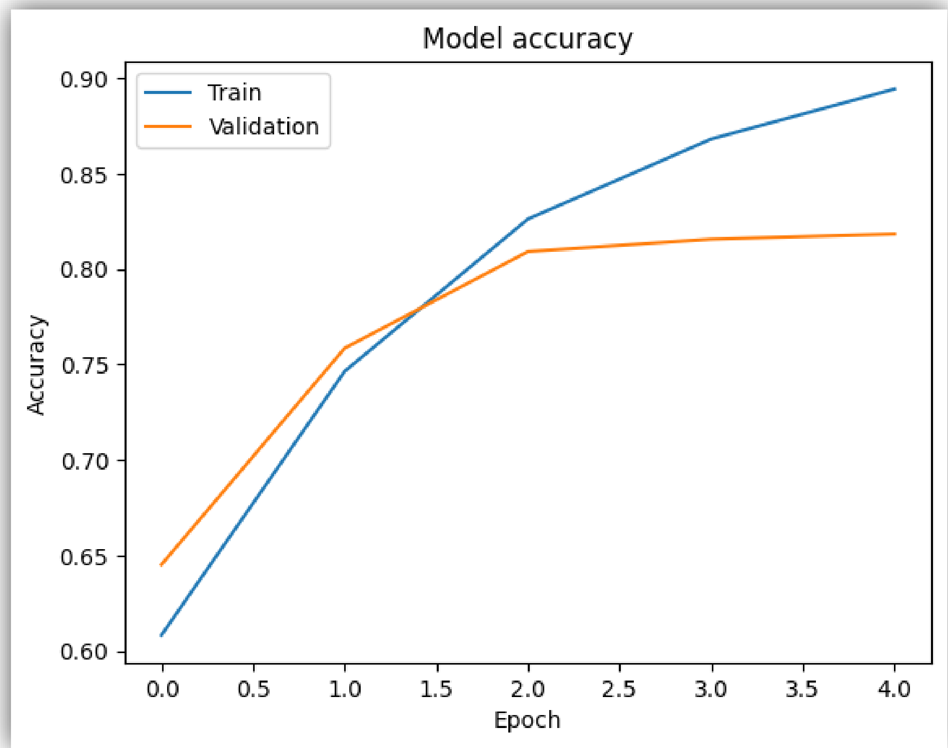


Accuracy: 0.8265027322404371
F1 Score: 0.8652519893899204
Precision: 0.8670919723551302
Recall: 0.8634197988353626

UTA

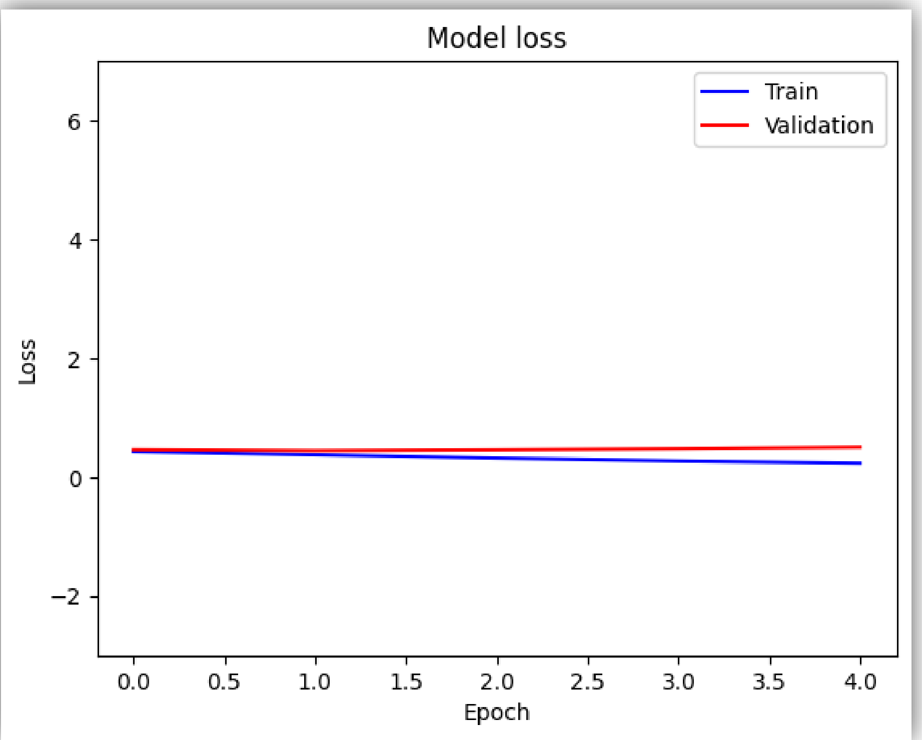
RESULTS – TWITTER DATASET KERAS EMBEDDING

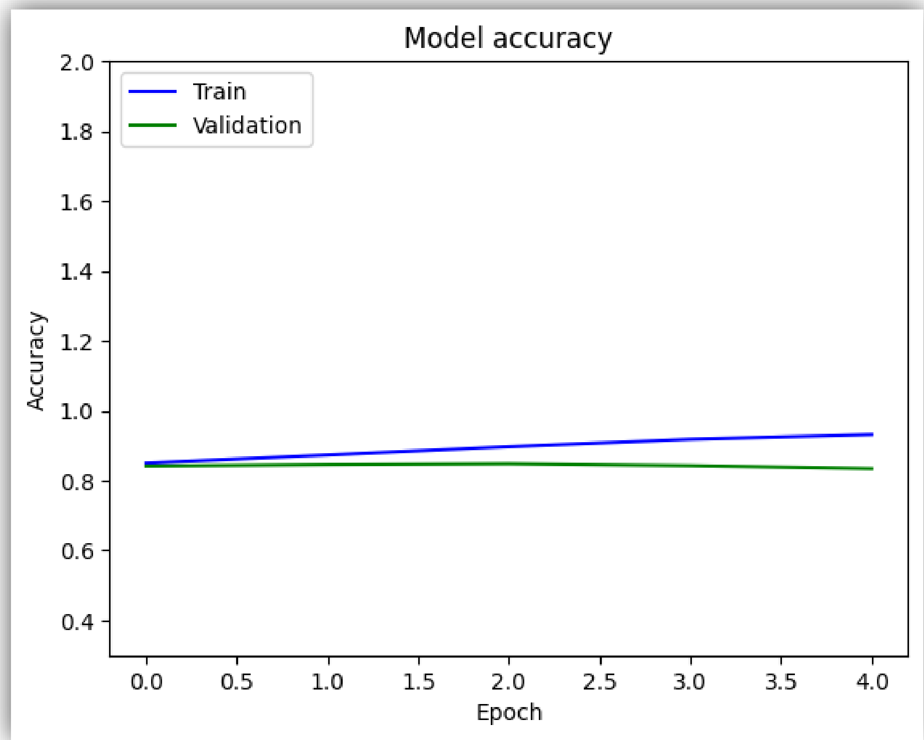





```
Accuracy: 0.8183060109289617
F1 Score: 0.8571428571428571
Precision: 0.8697547683923705
Recall: 0.8448914769719428
```

RESULTS – TWITTER DATASET WORD2VEC EMBEDDING





Accuracy: 0.8456284153005464
F1 Score: 0.8808645229309435
Precision: 0.8771653543307086
Recall: 0.8845950238221281

For **Twitter US Airline** Dataset, implementing **CNN-LSTM** Model with **Word2Vec** embedding gave us highest accuracy of **84.56%.**

IMDB MOVIE REVIEW DATASET

SUMMARY OF MODELS

Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 300)	48751800
conv1d (Conv1D)	(None, 98, 256)	230656
max_pooling1d (MaxPooling1D)	(None, 49, 256)	0
dropout (Dropout)	(None, 49, 256)	0
conv1d_1 (Conv1D)	(None, 47, 256)	196864
max_pooling1d_1 (MaxPooling1D)	(None, 47, 256)	0
lstm (LSTM)	(None, 47, 256)	525312
lstm_1 (LSTM)	(None, 256)	525312
dropout_1 (Dropout)	(None, 256)	0
batch_normalization (BatchNormalization)	(None, 256)	1024
flatten (Flatten)	(None, 256)	0
dense (Dense)	(None, 128)	32896
dense_1 (Dense)	(None, 1)	129

fastText

Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 100)	1915500
conv1d (Conv1D)	(None, 98, 256)	77056
max_pooling1d (MaxPooling1D)	(None, 49, 256)	0
dropout (Dropout)	(None, 49, 256)	0
conv1d_1 (Conv1D)	(None, 47, 256)	196864
max_pooling1d_1 (MaxPooling1D)	(None, 47, 256)	0
lstm (LSTM)	(None, 47, 256)	525312
lstm_1 (LSTM)	(None, 256)	525312
dropout_1 (Dropout)	(None, 256)	0
batch_normalization (BatchNormalization)	(None, 256)	1024
flatten (Flatten)	(None, 256)	0
dense (Dense)	(None, 128)	32896
dense_1 (Dense)	(None, 1)	129

Keras Embedding Layer

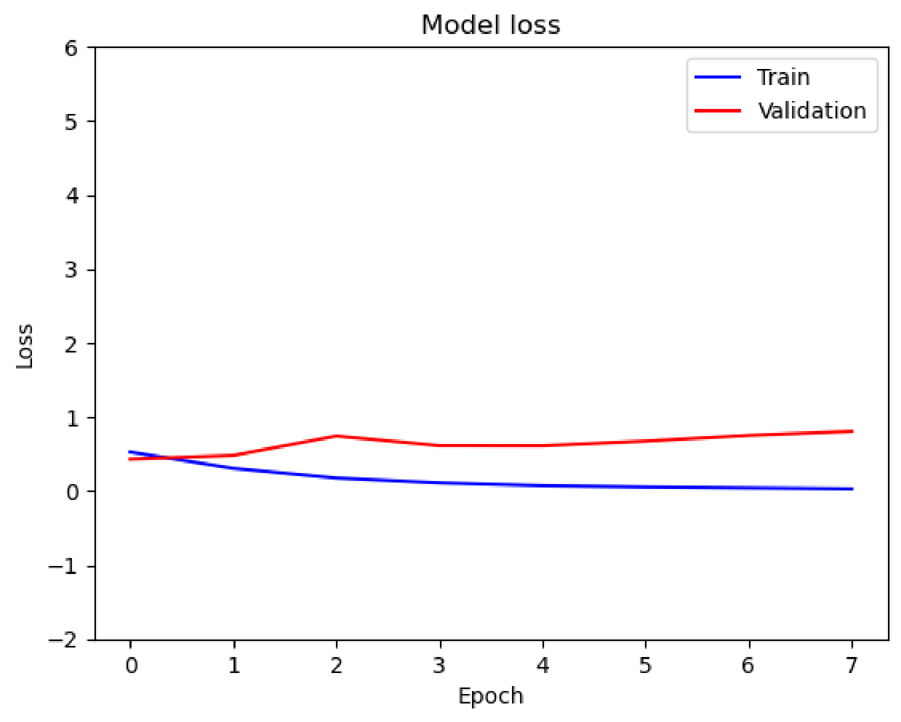
Model: "sequential_1"

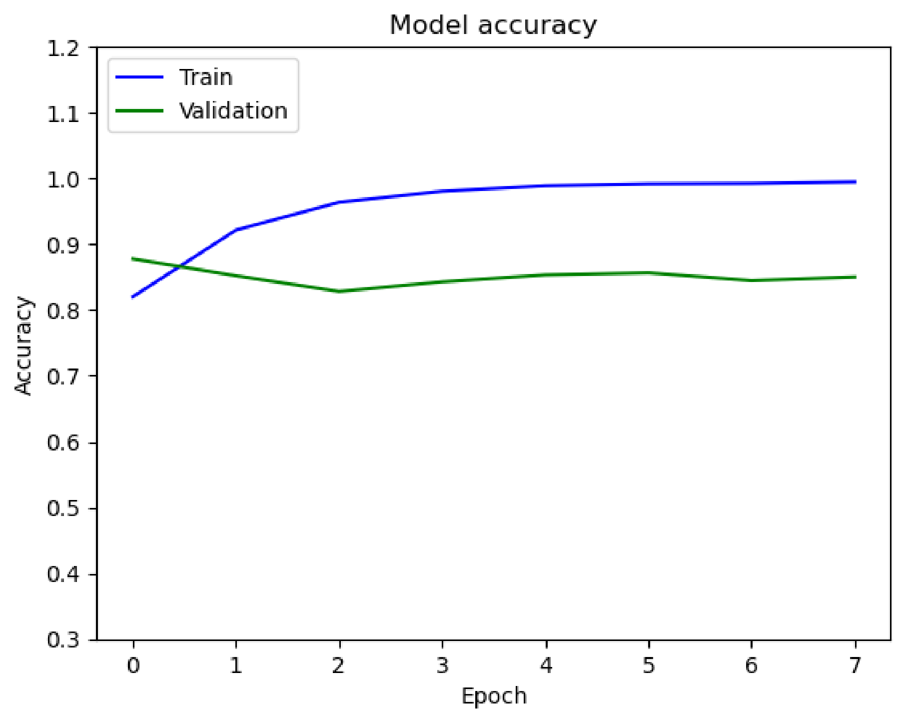
Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 100, 300)	48751800
conv1d_2 (Conv1D)	(None, 98, 256)	230656
max_pooling1d_2 (MaxPooling1D)	(None, 49, 256)	0
dropout_2 (Dropout)	(None, 49, 256)	0
conv1d_3 (Conv1D)	(None, 47, 256)	196864
max_pooling1d_3 (MaxPooling1D)	(None, 47, 256)	0
lstm_2 (LSTM)	(None, 47, 256)	525312
lstm_3 (LSTM)	(None, 256)	525312
dropout_3 (Dropout)	(None, 256)	0
batch_normalization_1 (BatchNormalization)	(None, 256)	1024
flatten_1 (Flatten)	(None, 256)	0
dense_2 (Dense)	(None, 128)	32896
dense_3 (Dense)	(None, 1)	129

Word2Vec

- The summary shows that we have used three different embeddings.
- In each embedding we used different layers to get the highest accuracy possible.

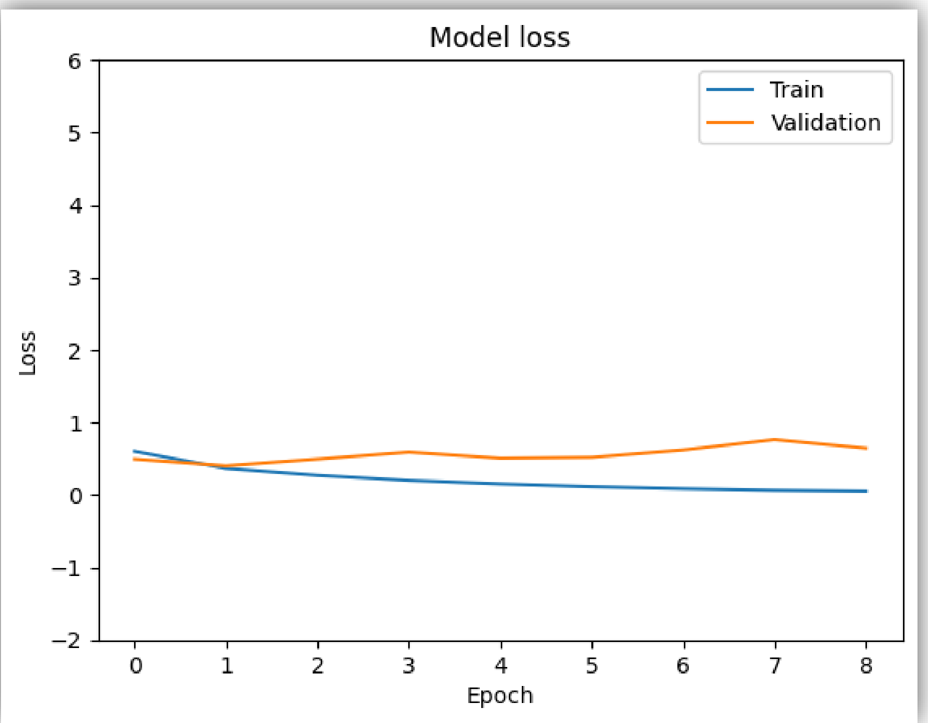
RESULTS – IMDB DATASET FASTTEXT EMBEDDING

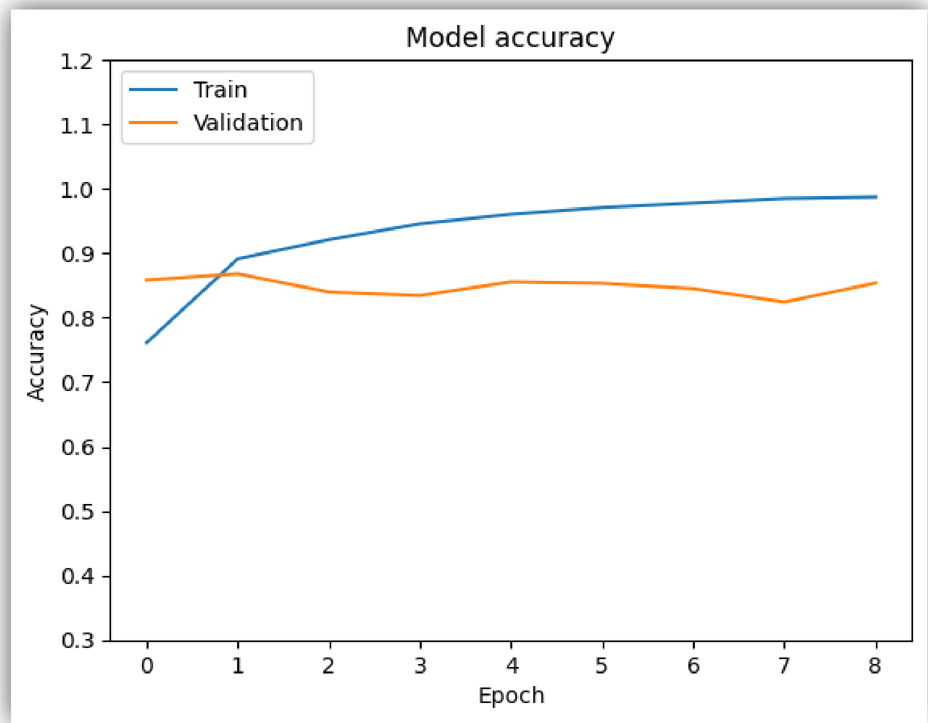





```
Accuracy: 0.8777
F1 Score: 0.8731722493000105
Precision: 0.8961260110685398
Recall: 0.8513650151668352
```

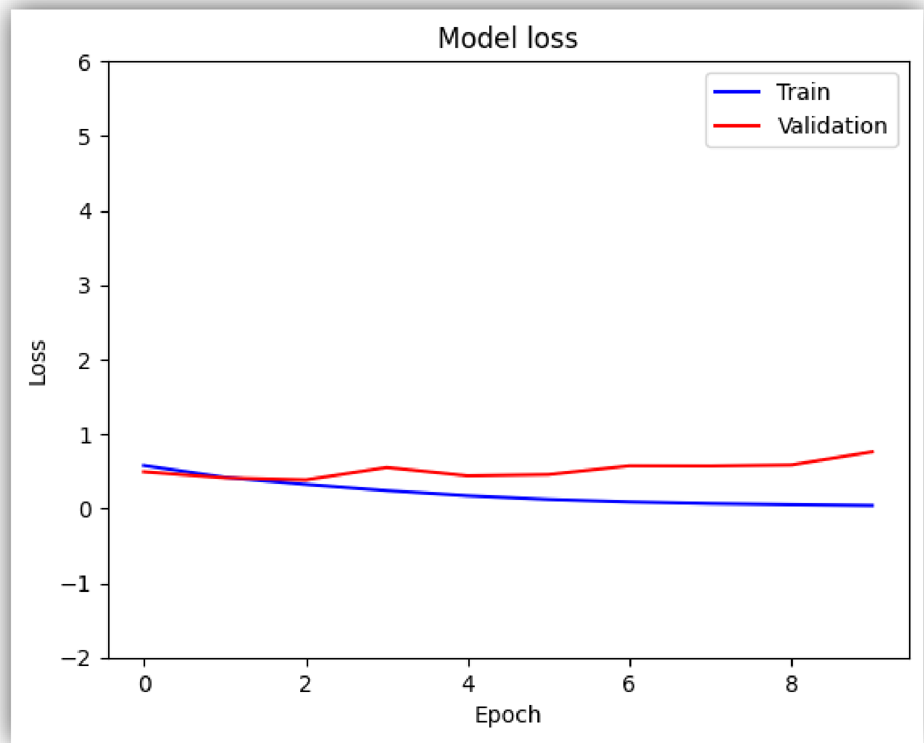
RESULTS – IMDB DATASET KERAS EMBEDDING

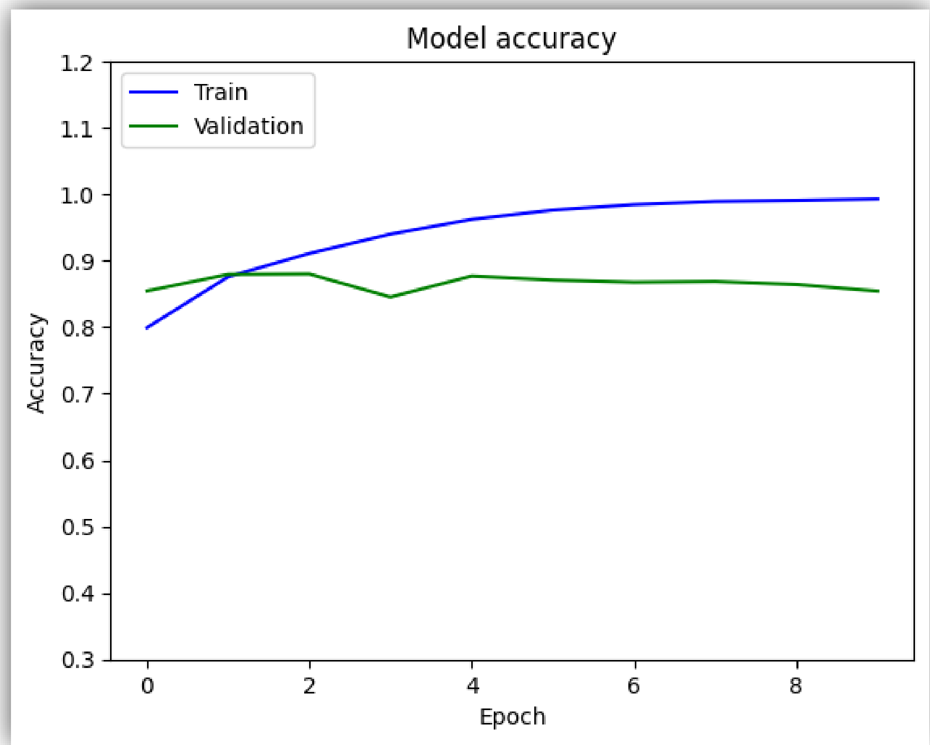





```
Accuracy: 0.8680666666666667
F1 Score: 0.860800450165295
Precision: 0.9033067611455565
Recall: 0.8221147386806396
```

RESULTS – IMDB DATASET WORD2VEC EMBEDDING






```
Accuracy: 0.8801
F1 Score: 0.8740413909024057
Precision: 0.9094884127678181
Recall: 0.8412537917087968
```

UTA

For **IMDB Movie Review** Dataset, implementing **CNN-LSTM** Model with **Word2Vec** embedding gave us highest accuracy of **88.01%.**

CNN-LSTM-TRANSFORMER HYBRID MODEL

CONVOLUTION NEURAL NETWORK - LONG SHORT-TERM MEMORY - TRANSFORMER (CNN-LSTM-TRANSFORMER) HYBRID MODEL

In this project we are using CNN-LSTM-Transformer Hybrid model.

CNN Model in Sentiment Analysis:

- Local Feature Learning
- Robustness to Noise

LSTM Model in Sentiment Analysis:

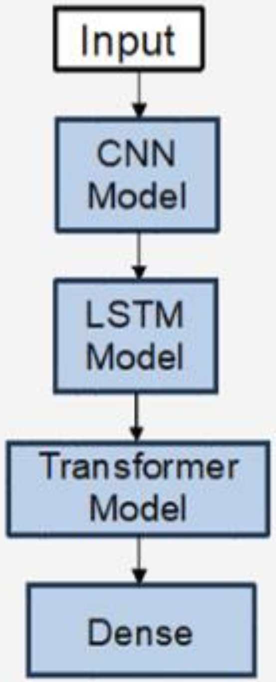
- Processing Sequential Information
- Managing Variable-Length Sequences

Transformer in Sentiment Analysis:

- Global contextual insight
- Improving the model's capacity to capture both short and long-range dependencies in sentiment data.

UTA

ARCHITECTURE OF OUR PROPOSED CNN-LSTM-TRANSFORMER HYBRID MODEL



Convolutional Neural Network Long Short-Term Memory Transformer Network Architecture

REASONS FOR IMPLEMENTING CNN-LSTM-TRANSFORMER HYBRID MODEL

- **Hierarchical Information Processing:** Captures local and global features simultaneously, providing a hierarchical understanding of sentiment-related patterns.
- **Temporal Dependency Management:** Handles variable-length sequences, capturing temporal dependencies and contextual nuances in sentiment expression.
- **Global Contextual Integration:** Incorporates attention mechanisms for comprehensive analysis, considering both short and long-range dependencies for nuanced sentiment comprehension.
- **Synergistic Fusion for Enhanced Performance:** Integrates CNN, LSTM, and Transformer, leveraging their complementary strengths to create a holistic sentiment analysis model with improved adaptability and performance.

TWITTER US AIRLINE DATASET

SUMMARY OF MODELS

Model: "sequential_7"

Layer (type)	Output Shape	Param #
embedding_7 (Embedding)	(None, 100, 100)	1090000
conv1d_14 (Conv1D)	(None, 96, 256)	128256
max_pooling1d_14 (MaxPooling1D)	(None, 48, 256)	0
dropout_14 (Dropout)	(None, 48, 256)	0
conv1d_15 (Conv1D)	(None, 46, 256)	196864
max_pooling1d_15 (MaxPooling1D)	(None, 23, 256)	0
attention_wrapper_7 (AttentionWrapper)	(None, 23, 256)	1
lstm_14 (LSTM)	(None, 23, 128)	197120
lstm_15 (LSTM)	(None, 128)	131584
dropout_15 (Dropout)	(None, 128)	0
dense_14 (Dense)	(None, 128)	16512
dense_15 (Dense)	(None, 1)	129

fastText

Model: "sequential_1"

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 100, 100)	1090000
conv1d_1 (Conv1D)	(None, 96, 256)	128256
max_pooling1d (MaxPooling1D)	(None, 48, 256)	0
dropout_1 (Dropout)	(None, 48, 256)	0
conv1d_2 (Conv1D)	(None, 46, 256)	196864
max_pooling1d_1 (MaxPooling1D)	(None, 23, 256)	0
attention_wrapper (AttentionWrapper)	(None, 23, 256)	1
lstm_2 (LSTM)	(None, 23, 128)	197120
lstm_3 (LSTM)	(None, 128)	131584
dropout_2 (Dropout)	(None, 128)	0
batch_normalization (BatchNormalization)	(None, 128)	512
flatten (Flatten)	(None, 128)	0
dense_1 (Dense)	(None, 128)	16512
dense_2 (Dense)	(None, 1)	129

Keras Embedding Layer

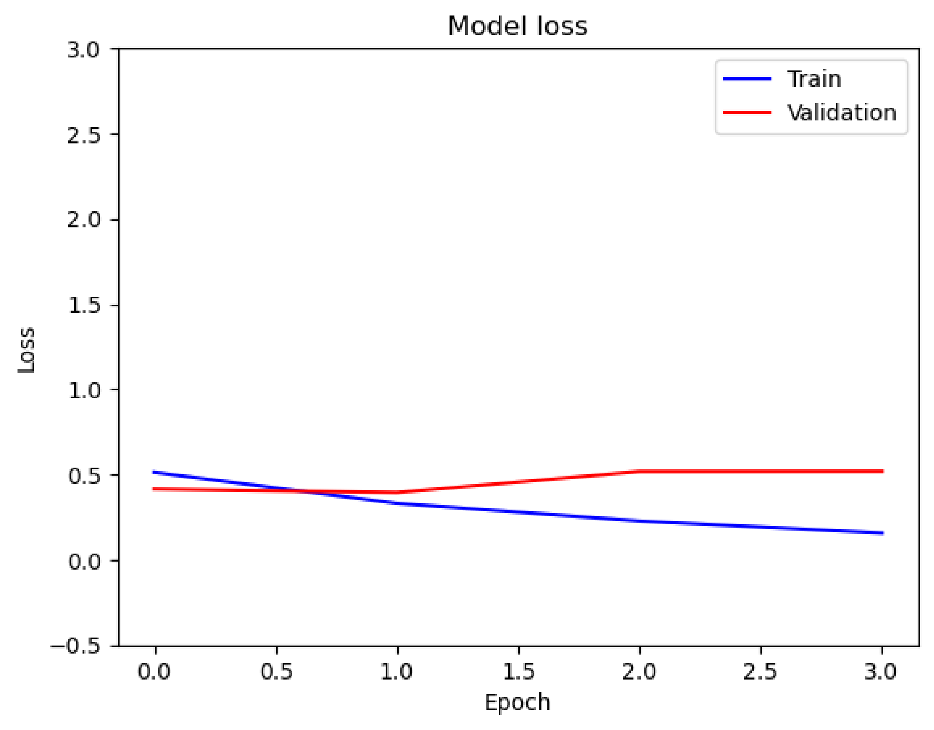
Model: "sequential_11"

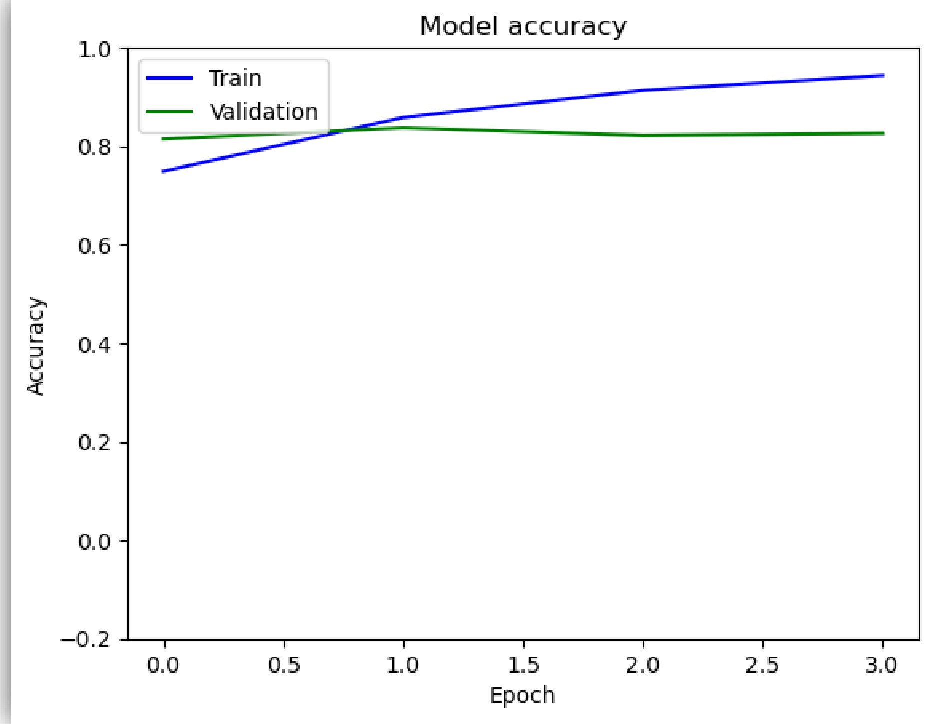
Layer (type)	Output Shape	Param #
embedding_11 (Embedding)	(None, 100, 100)	1090000
conv1d_22 (Conv1D)	(None, 96, 256)	128256
max_pooling1d_22 (MaxPooling1D)	(None, 48, 256)	0
dropout_17 (Dropout)	(None, 48, 256)	0
conv1d_23 (Conv1D)	(None, 46, 256)	196864
max_pooling1d_23 (MaxPooling1D)	(None, 23, 256)	0
attention_wrapper_6 (AttentionWrapper)	(None, 23, 256)	1
lstm_12 (LSTM)	(None, 23, 128)	197120
lstm_13 (LSTM)	(None, 128)	131584
dropout_18 (Dropout)	(None, 128)	0
dense_12 (Dense)	(None, 128)	16512
dense_13 (Dense)	(None, 1)	129

Word2Vec

- The summary shows that we have used three different embeddings.
- In each embedding we used different layers to get the highest accuracy possible.

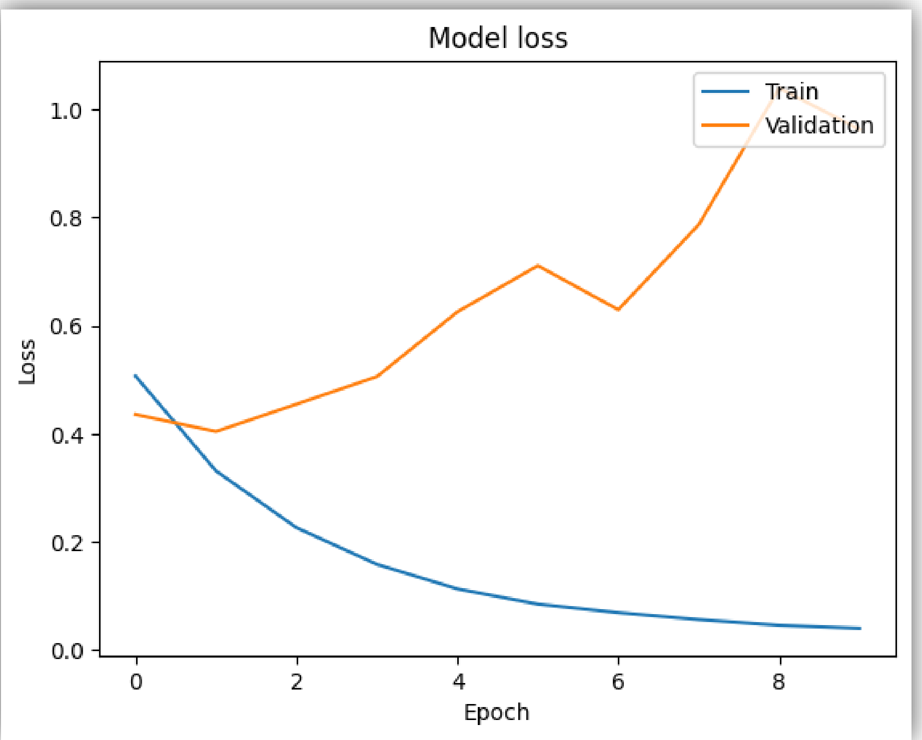
RESULTS – TWITTER DATASET FASTTEXT EMBEDDING

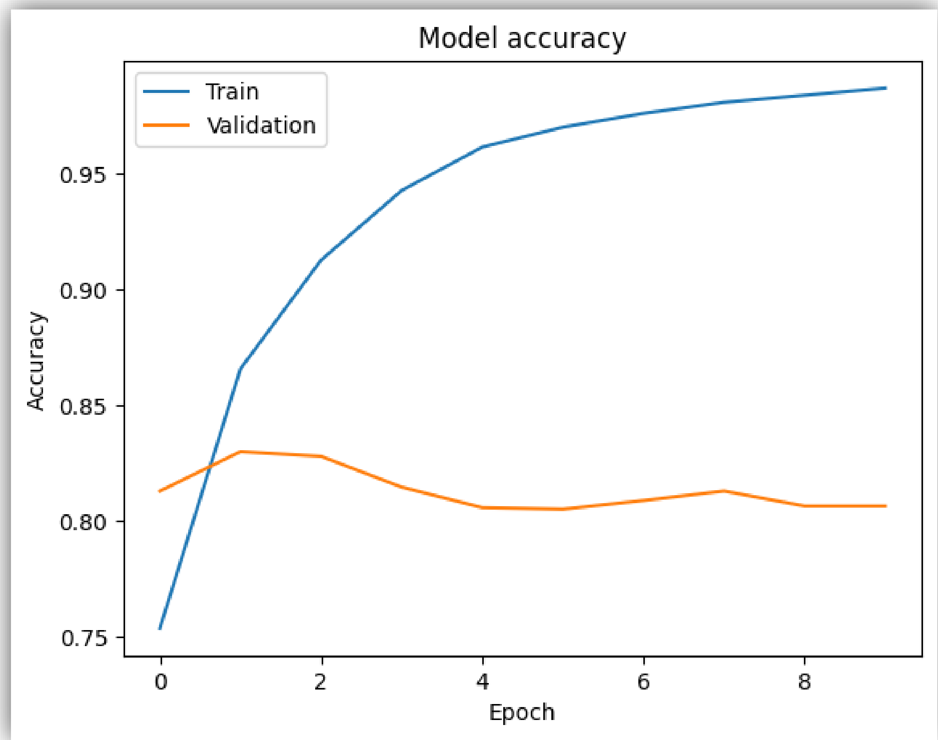




Accuracy: 0.8377732240437158
F1 Score: 0.8782363496539349
Precision: 0.8513916500994035
Recall: 0.9068290100582319

RESULTS – TWITTER DATASET KERAS EMBEDDING

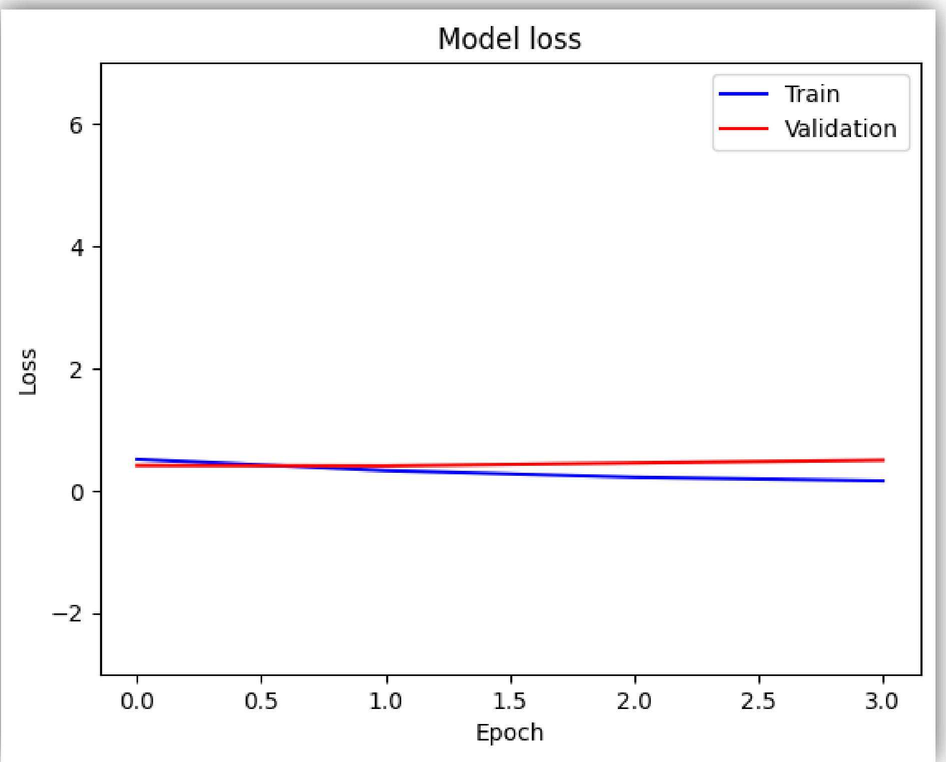


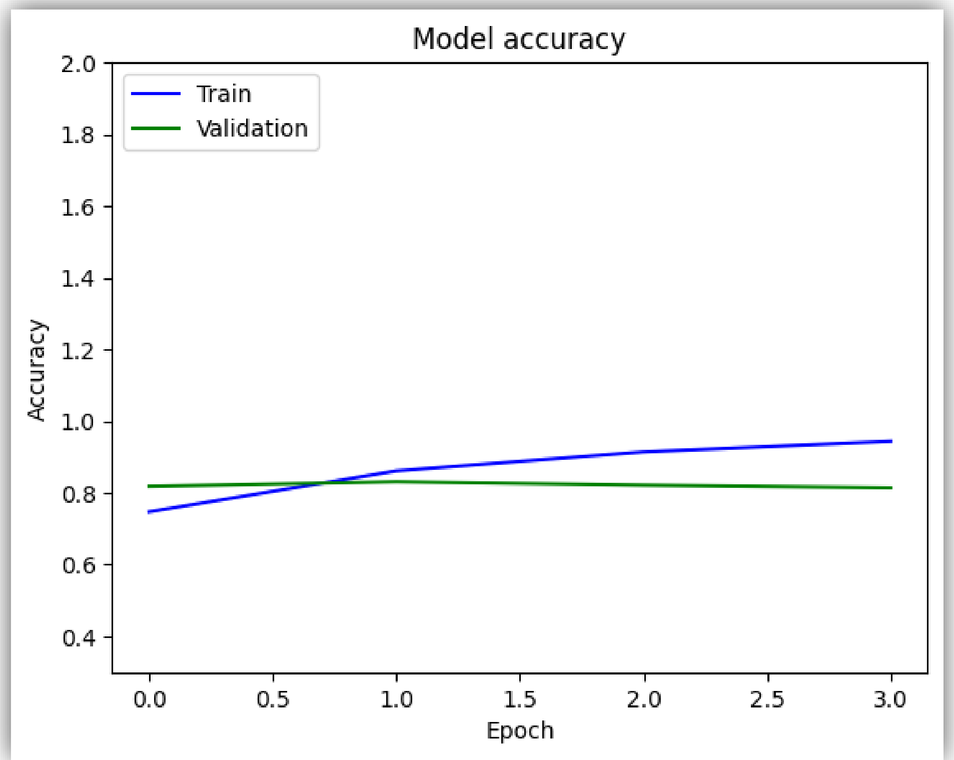



```
Accuracy: 0.8063524590163934
F1 Score: 0.8508287292817
Precision: 0.8457112970711297
Recall: 0.8560084700899947
```

UTA

RESULTS – TWITTER DATASET WORD2VEC EMBEDDING





Accuracy: 0.8340163934426229
F1 Score: 0.8733055265901981
Precision: 0.8602978941961993
Recall: 0.8867125463208047

For **Twitter US Airline** Dataset, implementing **CNN-LSTM-Transformer** Model with **fastText** embedding gave us highest accuracy of **83.7%.**

IMDB MOVIE REVIEW DATASET

SUMMARY OF MODELS

Model: "sequential_6"

Layer (type)	Output Shape	Param #
embedding_3 (Embedding)	(None, 100, 300)	48751800
conv1d_6 (Conv1D)	(None, 96, 256)	384256
max_pooling1d_6 (MaxPooling1D)	(None, 48, 256)	0
conv1d_7 (Conv1D)	(None, 46, 256)	196864
max_pooling1d_7 (MaxPooling1D)	(None, 46, 256)	0
dropout_17 (Dropout)	(None, 46, 256)	0
batch_normalization_17 (BatchNormalization)	(None, 46, 256)	1024
lstm_6 (LSTM)	(None, 46, 256)	525312
dropout_18 (Dropout)	(None, 46, 256)	0
batch_normalization_18 (BatchNormalization)	(None, 46, 256)	1024
lstm_7 (LSTM)	(None, 256)	525312
dropout_19 (Dropout)	(None, 256)	0
batch_normalization_19 (BatchNormalization)	(None, 256)	1024
transformer_block_3 (TransformerBlock)	(None, 256)	35137
batch_normalization_22 (BatchNormalization)	(None, 256)	1024
flatten_2 (Flatten)	(None, 256)	0
dropout_22 (Dropout)	(None, 256)	0
dense_12 (Dense)	(None, 256)	65792
dense_13 (Dense)	(None, 1)	257

fastText

Model: "sequential_2"

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 100, 100)	1915500
conv1d_2 (Conv1D)	(None, 96, 256)	128256
max_pooling1d_2 (MaxPooling1D)	(None, 48, 256)	0
conv1d_3 (Conv1D)	(None, 46, 256)	196864
max_pooling1d_3 (MaxPooling1D)	(None, 46, 256)	0
dropout_6 (Dropout)	(None, 46, 256)	0
batch_normalization_6 (BatchNormalization)	(None, 46, 256)	1024
lstm_2 (LSTM)	(None, 46, 256)	525312
dropout_7 (Dropout)	(None, 46, 256)	0
batch_normalization_7 (BatchNormalization)	(None, 46, 256)	1024
lstm_3 (LSTM)	(None, 256)	525312
dropout_8 (Dropout)	(None, 256)	0
batch_normalization_8 (BatchNormalization)	(None, 256)	1024
transformer_block_1 (TransformerBlock)	(None, 256)	35137
batch_normalization_11 (BatchNormalization)	(None, 256)	1024
flatten_1 (Flatten)	(None, 256)	0
dropout_11 (Dropout)	(None, 256)	0
dense_6 (Dense)	(None, 256)	65792
dense_7 (Dense)	(None, 1)	257

Keras Embedding Layer

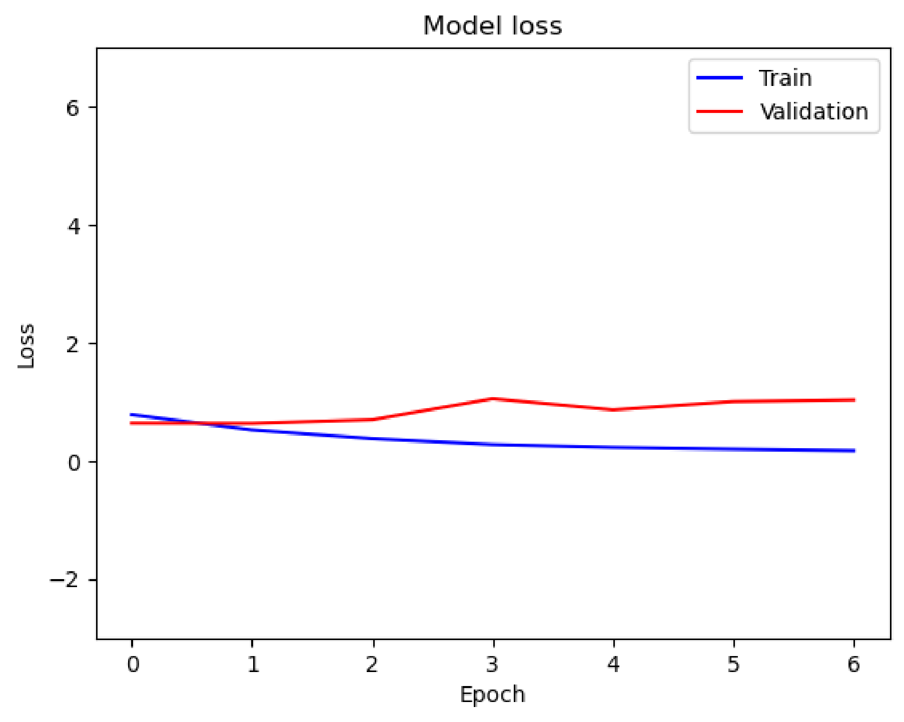
Model: "sequential"

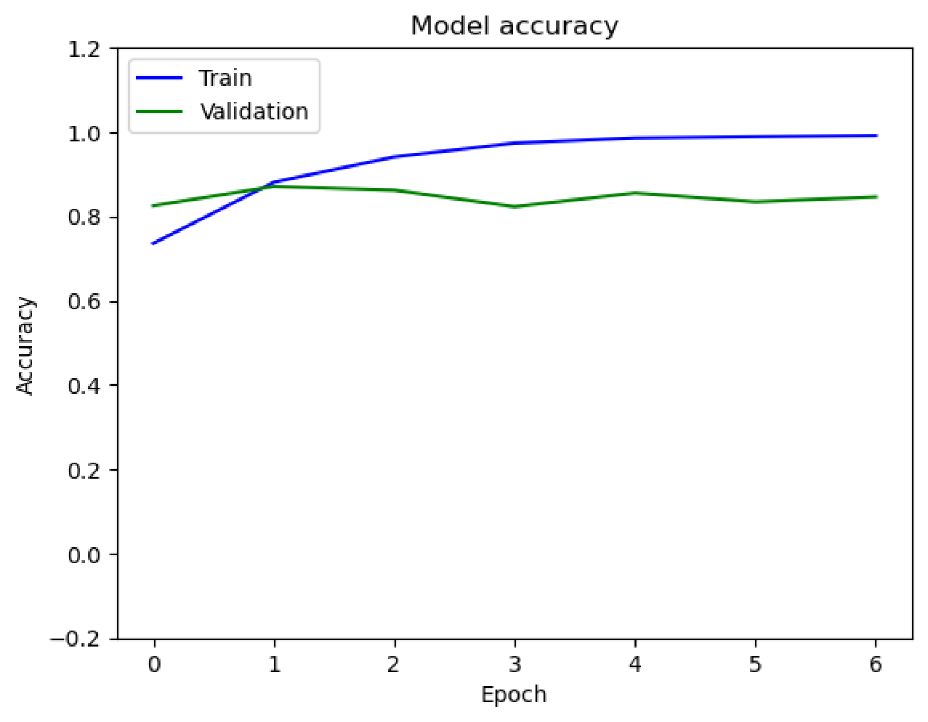
Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 300)	48751800
conv1d (Conv1D)	(None, 96, 256)	384256
max_pooling1d (MaxPooling1D)	(None, 48, 256)	0
conv1d_1 (Conv1D)	(None, 46, 256)	196864
max_pooling1d_1 (MaxPooling1D)	(None, 46, 256)	0
dropout (Dropout)	(None, 46, 256)	0
batch_normalization (BatchNormalization)	(None, 46, 256)	1024
lstm (LSTM)	(None, 46, 256)	525312
dropout_1 (Dropout)	(None, 46, 256)	0
batch_normalization_1 (BatchNormalization)	(None, 46, 256)	1024
lstm_1 (LSTM)	(None, 256)	525312
dropout_2 (Dropout)	(None, 256)	0
batch_normalization_2 (BatchNormalization)	(None, 256)	1024
transformer_block (TransformerBlock)	(None, 256)	35137
batch_normalization_5 (BatchNormalization)	(None, 256)	1024
flatten (Flatten)	(None, 256)	0
dropout_5 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 256)	65792
dense_3 (Dense)	(None, 1)	257

Word2Vec

- The summary shows that we have used three different embeddings.
- In each embedding we used different layers to get the highest accuracy possible.

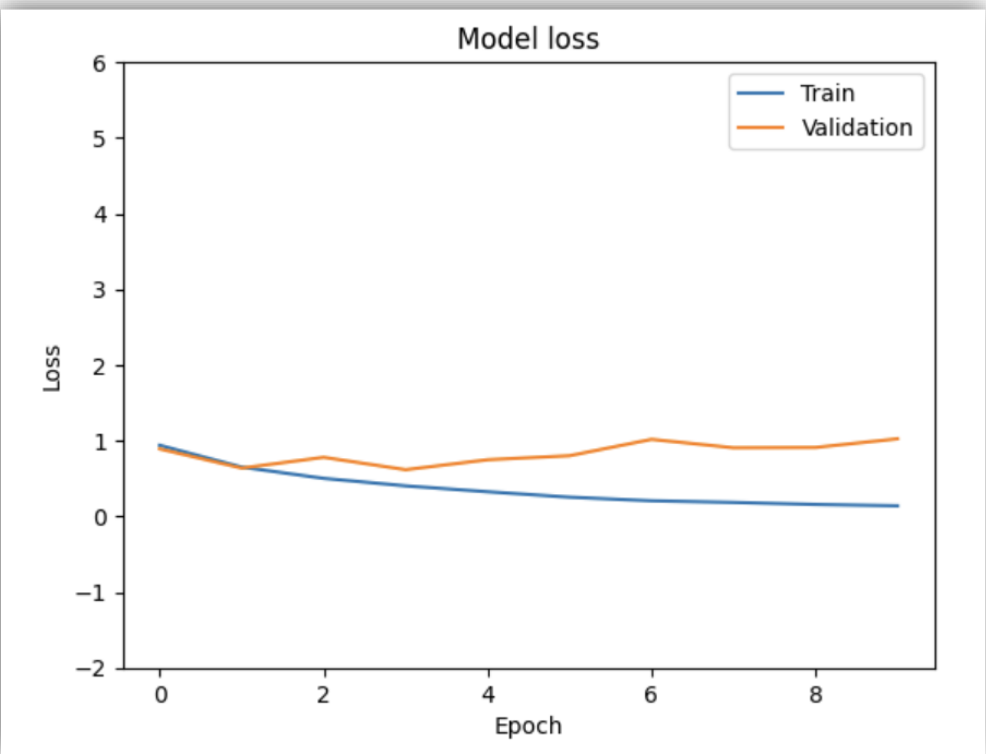
RESULTS – IMDB DATASET FASTTEXT EMBEDDING

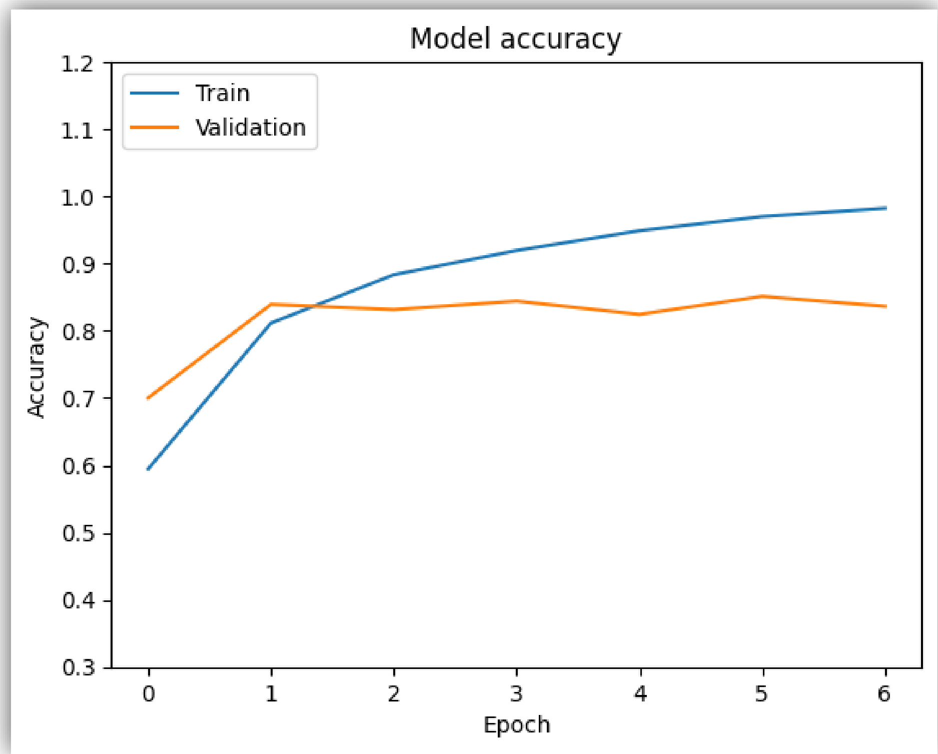




Accuracy: 0.8702666666666666
F1 Score: 0.8675830157866086
Precision: 0.8789466427685095
Recall: 0.8565094719871019

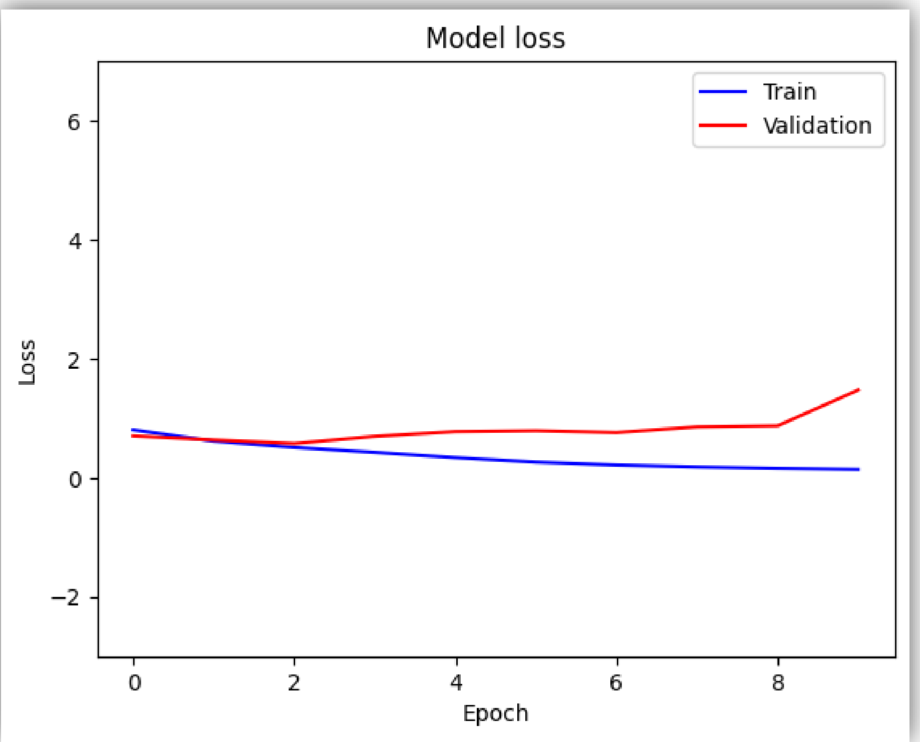
RESULTS – IMDB DATASET KERAS EMBEDDING

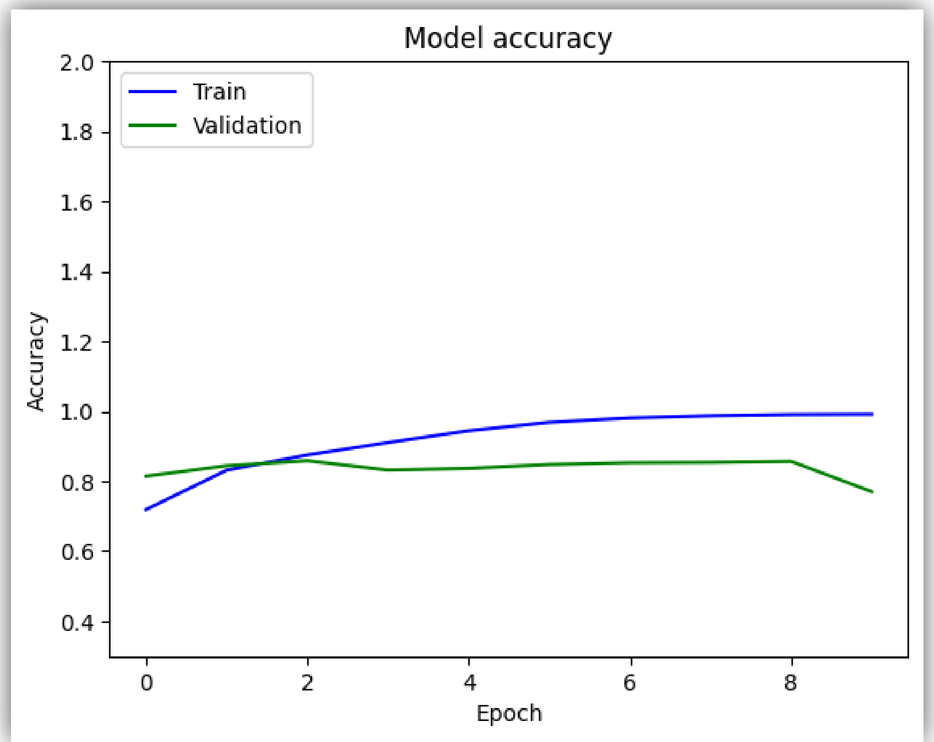





```
Accuracy: 0.8398
F1 Score: 0.8391889178879743
Precision: 0.836
Recall: 0.8424022571543732
```

RESULTS – IMDB DATASET WORD2VEC EMBEDDING






```
Accuracy: 0.8584666666666667
F1 Score: 0.8522513744867425
Precision: 0.8840600635287323
Recall: 0.822652156388553
```

UTA

For **IMDB Movie Review** Dataset, implementing **CNN-LSTM-Transformer** Model with **fastText** embedding gave us highest accuracy of **87.02%.**

SUMMARY

MODELS IMPLEMENTED	ACCURACIES OF THE MODEL
Twitter US Airline Dataset CNN LSTM Keras Embedding	81.83%
Twitter US Airline Dataset CNN LSTM fastText Embedding	82.65%
Twitter US Airline Dataset CNN LSTM Word2Vec Embedding	**84.56%**
IMDB Movie Review Dataset CNN LSTM Keras Embedding	86.81%
IMDB Movie Review Dataset CNN LSTM fastText Embedding	87.77%
IMDB Movie Review Dataset CNN LSTM Word2Vec Embedding	**88.01%**

MODELS IMPLEMENTED	ACCURACIES OF THE MODEL
Twitter US Airline Dataset CNN LSTM Transformer Keras Embedding	80.63%
Twitter US Airline Dataset CNN LSTM Transformer fastText Embedding	**83.78%**
Twitter US Airline Dataset CNN LSTM Transformer Word2Vec Embedding	83.4%
IMDB Movie Review Dataset CNN LSTM Transformer Keras Embedding	83.77%
IMDB Movie Review Dataset CNN LSTM Transformer fastText Embedding	**87.03%**
IMDB Movie Review Dataset CNN LSTM Transformer Word2Vec Embedding	85.8%

CONCLUSION

Word2Vec with CNN-LSTM:

- Demonstrated superior accuracy in both Twitter US Airline dataset and IMDB Movie Reviews dataset.
- Effective in capturing semantic relationships in organized and formal writing.
- Suggests synergy between Word2Vec and CNN-LSTM for diverse textual datasets.

fastText with CNN-LSTM-Transformer:

- Outperformed in both Twitter US Airline dataset and IMDB Movie Reviews scenarios.
- Particularly effective in identifying semantic linkages within sub-words.
- Highlights the importance of leveraging fastText with CNN-LSTM-Transformer for dynamic language tasks.

PROPOSED IMPLEMENTATION VS RESEARCH PAPERS' IMPLEMENTATION

Dataset Diversity:

- Utilization of diverse datasets, including US Airline Twitter dataset (brief tweets) and IMDB movie reviews (longer paragraphs).
- Showcase of adaptability across a broader range of text categorization tasks compared to papers focusing on specific datasets and tasks.

Thorough Preprocessing Techniques:

- Holistic preprocessing approach incorporating standard techniques (lowercasing, punctuation removal, stop word elimination, tokenization).
- Extension of preprocessing with contraction handling and emoji processing.

PROPOSED IMPLEMENTATION VS RESEARCH PAPERS' IMPLEMENTATION

Emphasis on Emojis as Contextual Cues:

- Acknowledgment of the significant influence of emojis on sentiment in textual data.
- Recognition of emojis as vital contextual signals conveying emotions, tone, and mood.
- Improved contextual comprehension and sentiment analysis accuracy through emoji incorporation.

Comparative Analysis of Sentiment Analysis Models:

- Evaluation of fastText, Keras Embedding Layer, and Word2Vec in CNN-LSTM and CNN-LSTM-Transformer models.
- Assessment of semantic relationships between words that enhance contextual comprehension.
- Correlation of different approaches with sentiment classification accuracy.

UTA

PROPOSED IMPLEMENTATION VS RESEARCH PAPERS' IMPLEMENTATION

Innovative Model Architecture:

- Introduction of a hybrid CNN-LSTM model with early stopping and batch normalization.
- Effective capturing of sequential text information, enhancing performance across various text categorization tasks.
- Distinction from research papers that present varied CNN architectures.

Advanced Hyperparameters:

- Incorporation of advancements in hyperparameters, including increased filters, batch normalization, and early stopping.
- Use of distinct dropout rates for different datasets and the integration of three embeddings (Word2Vec, fastText, Keras embedding layers).
- Improved adaptability across datasets, contributing to overall performance enhancement.

UTA

REFERENCES

1. Poornima A, Nataraj , Nithya R, Nirmala D, Divya P. (2022), "Sentiment Analysis of Tweets in Twitter Using CNN," 2022 International Conference on Computer Communication and Informatics (ICCCI)

2. S. Hafeez and N. Kathirisetty. (2022), "Effects and Comparison of different Data pre-processing techniques and ML and deep learning models for sentiment analysis: SVM, KNN, PCA with SVM and CNN" 2022 First International Conference on Artificial Intelligence Trends and Pattern Recognition (ICAITPR)

3. S. Smetanin and M. Komarov. (2019), "Sentiment Analysis of Product Reviews in Russian using Convolutional Neural Networks," 2019 IEEE 21st Conference on Business Informatics (CBI)

UTA

THANK YOU