Smart chatbot for problem Redressal

MLPR EndSem presentation

TABLE OF CONTENTS

Problem Statement

MOTIVATION

- Everyday 1000’s of messages are flooded university chat groups.
- We find messages ranging from serious campus issues to memes.
- Important student issues are lost and never addressed leading to frustrations and bad student life

THE GOAL

- Develop a smart chatbot that automatically identifies student problems when posted on university chat groups
- Post relevant information about how to fix said issue.
- This would possibly reduce the student’s reliance on student life officials for basic issues and make the whole process more efficient

Literature Review

- There are 3 sections to our project:
  - Data preprocessing
  - Problem Classification
  - Data tagging and chatbot
- In essence, our project's foundation is fortified by a three-fold exploration of literature
- Problem classification:
  - Multiple papers explored problem identification classification with RNN, CNN and transformer encoders.
  - Since our dataset was quite large we weren't able to label the data manually so we researched semi-supervised models to help label and classify.

Why Transformers?

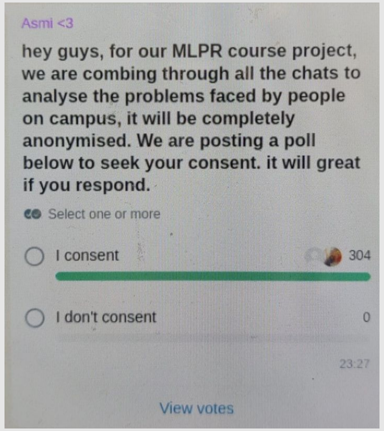
“One of the computational bottlenecks suffered by **RNNs is the sequential processing of text**. Although CNNs are less sequential than RNNs, the computational **cost to capture meaningful relationships between words in a sentence** also grows with increasing length of the sentence, similar to RNNs. Transformers overcome this limitation by **computing in parallel** for every word in a sentence or document an “attention score” to model the influence each word has on another. Due to this feature, Transformers allow for much more parallelization than CNNs and RNNs, and make it possible to efficiently train very big models on large amounts of data on GPU clusters.”

Deep Learning Based Text Classification: A Comprehensive Review

Dataset and data preprocessing

Data

We used the the whatsapp chat history to train our models. Before extracting this data we floated a poll on these whatsapp groups where people consented to this data being used in our project. Everyone who didn't reply or answered no, their data was deleted before any further processing



Asmi <3

hey guys, for our MLPR course project, we are combing through all the chats to analyse the problems faced by people on campus, it will be completely anonymised. We are posting a poll below to seek your consent. it will great if you respond.

Select one or more

I consent 304

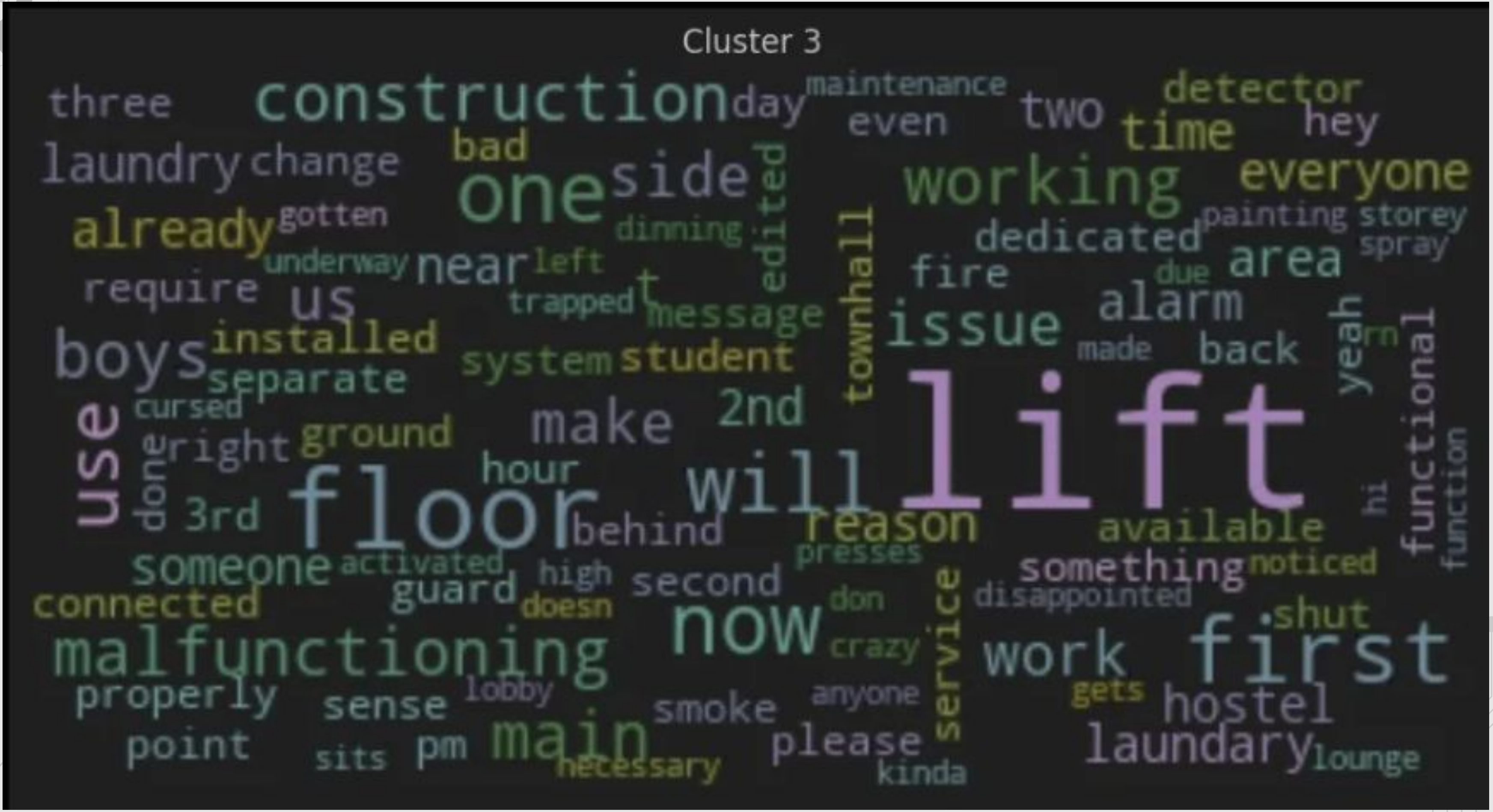
I don't consent 0

23:27

View votes

Data Preprocessing And EDA

- Cleaning the data:
  - Anonymising the data, removing punctuation,, names, numbers, @person messages
- Clustering to see if we could identify any student problems
  - Plotting elbow plots
  - Making wordclouds
- Plotting to see other trends in the data to be able to extract more features
  - What are the common problems faced by students


Cluster 3
three
construction
day
maintenance
two
detector
even
time
hey
laundry
change
bad
one
side
edited
working
everyone
already
gotten
dinning
townhall
dedicated
painting
storey
spray
underway
near
left
fire
area
due
require
us
trapped
t
message
issue
alarm
yeah
rn
functional
boys
installed
system
student
made
back
separate
cursed
2nd
lift
use
right
ground
make
done
hour
will
hi
function
3rd
floor
behind
reason
available
someone
activated
presses
something
noticed
high
guard
second
disappointed
connected
doesn
don
service
shut
malfunctioning
now
crazy
work
first
properly
lobby
sense
smoke
anyone
gets
hostel
point
sits
pm
main
please
laundary
lounge
necessary
kinda

Methodology

- Our solution is in 2 parts
  - First the identification of whether a statement is a problem on campus or not using a Transformer Encoder
  - And then passing it to our chatbot which extracts the relevant details to be given to the students
- Data Labelling
  - Labelled 600 rows on our own. (300 campus problems texts, 300 irrelevant)
  - Used GPT-4 with few shot and zero shot prompting techniques and asked it to label some of the data to see if it is labelling it right, then tested the accuracy of those labels on our manually labelled dataset.
  - GPT-4 actually achieved a 96% accuracy for labelling all the texts with the few shot prompting technique
  - We then used GPT-4 to label 10000 texts in batches over days using the previous prompting technique.

Transformer Encoder

- Data Preparation and Label Mapping
  - Split dataset into training and testing sets.
  - Extract labels and create label-to-ID mappings.
- Tokenizer and Vocabulary Building
  - Use tokenizer to convert text to tokens.
  - Build vocabulary from the training dataset's tokens.
- Word Embedding
  - Implement an embedding layer to convert tokens to vector representations.
- Positional Encoding
  - Add position information to embeddings using a positional encoding module.

Transformer Encoder

- Self-Attention and Multi-Head Attention
  - Compute attention scores with a single-head attention module.
  - Apply multi-head attention for diverse feature representation.
- Residual Connections and Layer Normalization
  - Incorporate residual connections after attention layers.
  - Stabilize learning with layer normalization.
- Feed-Forward Network
  - Include a feed-forward network in each encoder layer.
- Encoder Stack
  - Stack multiple encoder layers for deep representation.
- Transformer Encoder Model
  - Integrate all components into the Transformer Encoder model.
  - Add a classification layer for output prediction.

Transformer Encoder

- DataLoader and Training
  - Define custom dataset and batching functions.
  - Train model with cross-entropy loss and Adam optimizer.
  - Employ gradient clipping for training stability.
- Prediction and Evaluation
  - Implement function for new data prediction.
  - Evaluate model using accuracy, precision, recall, F1 score.

Transformer Encoder

- Positional Encoding: With dropout of 10%
- Encoder Layers: 6 stacked layers
- Attention Heads: 4 per layer
- Feed-Forward Network: Hidden layer size of 50
- Residual Connections: With Layer Normalization
- Output Classes: 2 (binary classification)
- Regularization: Dropout with a rate of 0.1

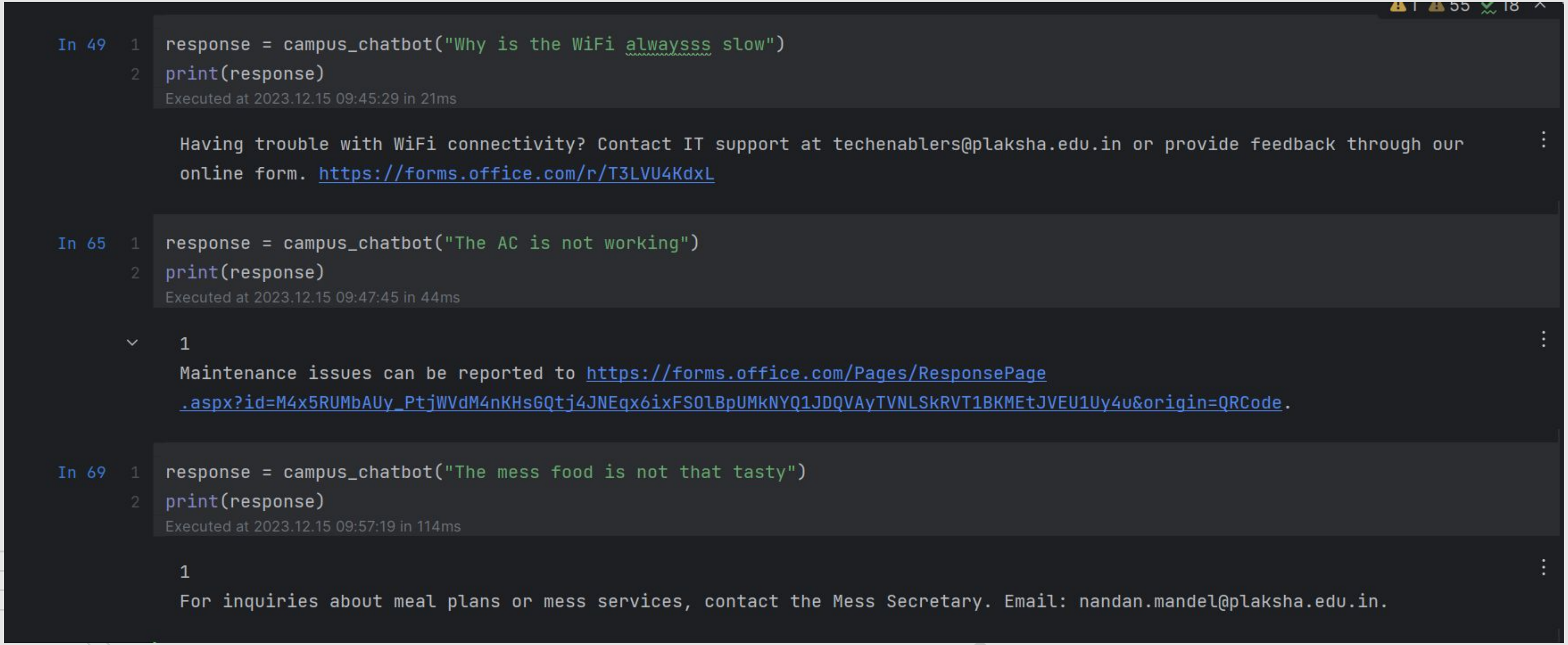
Transformer Encoder

- Positional Encoding: With dropout of 10%
- Encoder Layers: 6 stacked layers
- Attention Heads: 4 per layer
- Feed-Forward Network: Hidden layer size of 50
- Residual Connections: With Layer Normalization
- Output Classes: 2 (binary classification)
- Regularization: Dropout with a rate of 0.1

Chatbot

The chatbot does simple keyword matching(a diverse range of keywords for each category) for different categories of university campus issues and responds appropriately by fetching responses for that category from the solutions database.

Demo



```
In 49
response = campus_chatbot("Why is the WiFi alwaysss slow")
print(response)
Executed at 2023.12.15 09:45:29 in 21ms
```

```
Having trouble with WiFi connectivity? Contact IT support at techenablers@plaksha.edu.in or provide feedback through our
online form. https://forms.office.com/r/T3LVU4KdxL
```

```
In 65
response = campus_chatbot("The AC is not working")
print(response)
Executed at 2023.12.15 09:47:45 in 44ms
```

```
1
Maintenance issues can be reported to https://forms.office.com/Pages/ResponsePage
.aspx?id=M4x5RUMbAUy_PtjWVdM4nKHsGQtj4JNEqx6ixFSOlBpUMkNYQ1JDQVAyTVNLSkRVT1BKMEtJVEU1Uy4u&origin=QRCode.
```

```
In 69
response = campus_chatbot("The mess food is not that tasty")
print(response)
Executed at 2023.12.15 09:57:19 in 114ms
```

```
1
For inquiries about meal plans or mess services, contact the Mess Secretary. Email: nandan.mandel@plaksha.edu.in.
```

Evaluation

Accuracy: 85.4%
Precision: 79%
Recall: 72%
F1 Score: 75.6%

THANKS!