ruch.ai

YOUR EMAIL MOM.

the team.

your inbox janitors -

Bhavi

Ex-Computer Science and Artificial Intelligence

Rishi

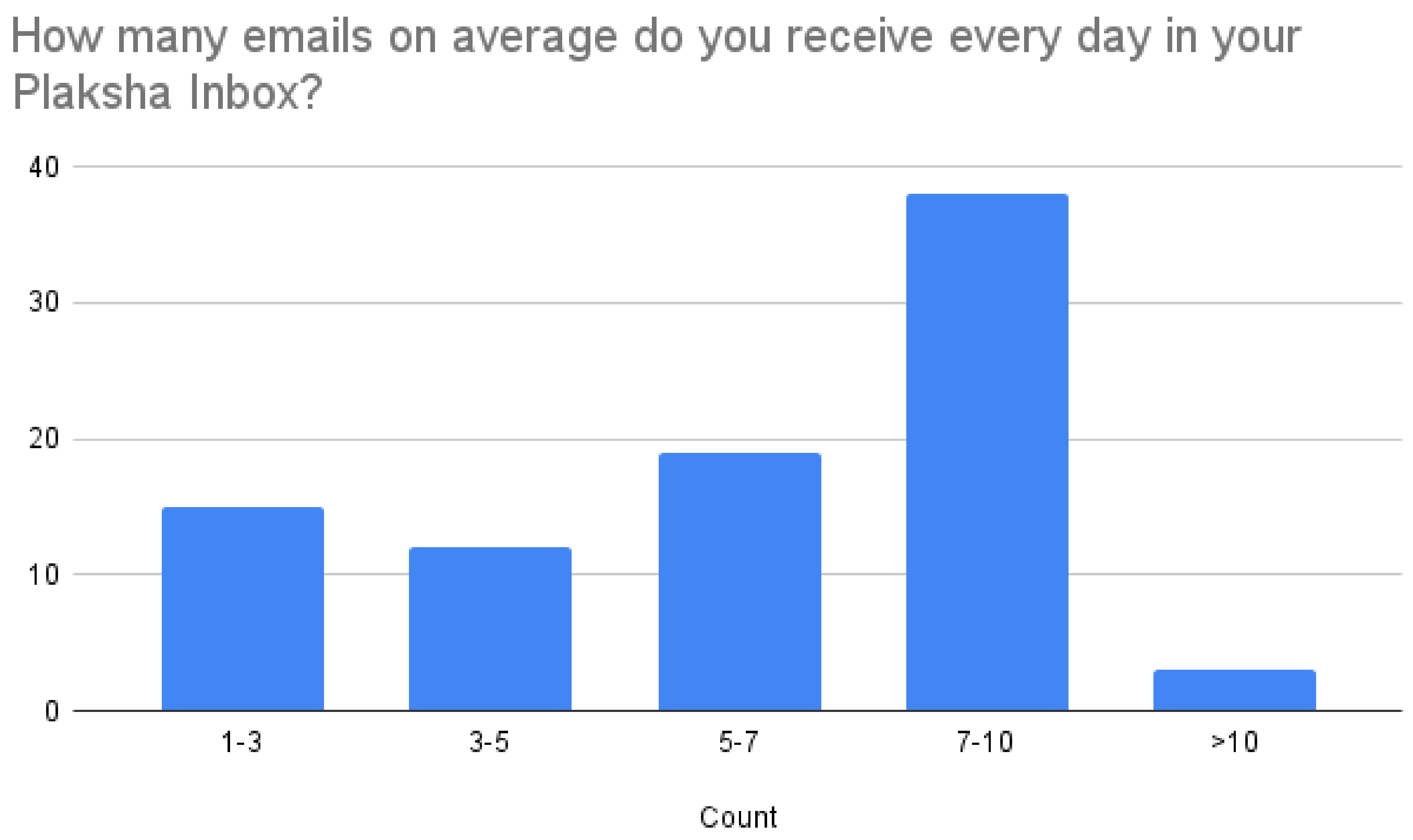
Computer Science and Artificial Intelligence

Shaurya

Computer Science and Artificial Intelligence

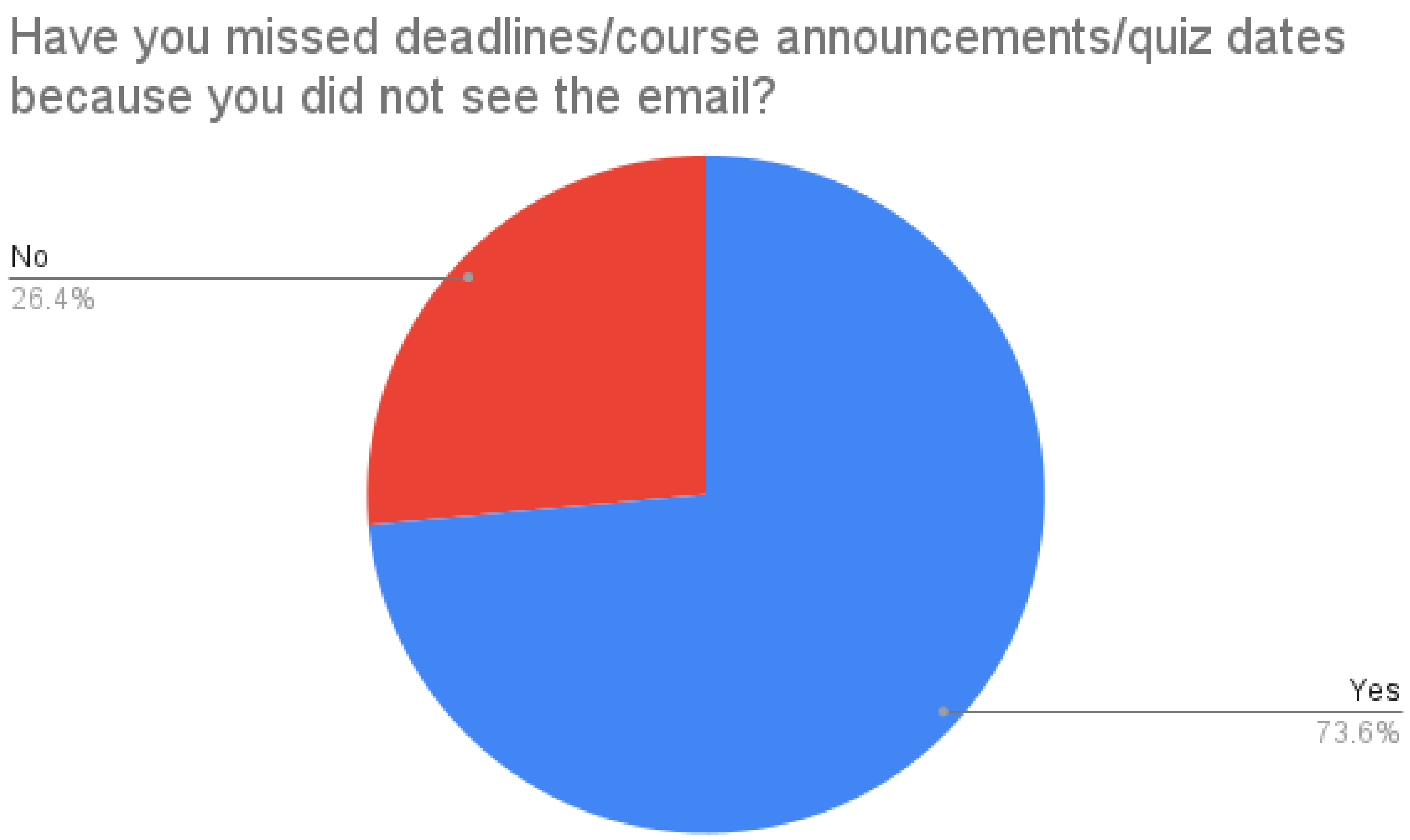
the problem.

outlook - i hate you, i love you.



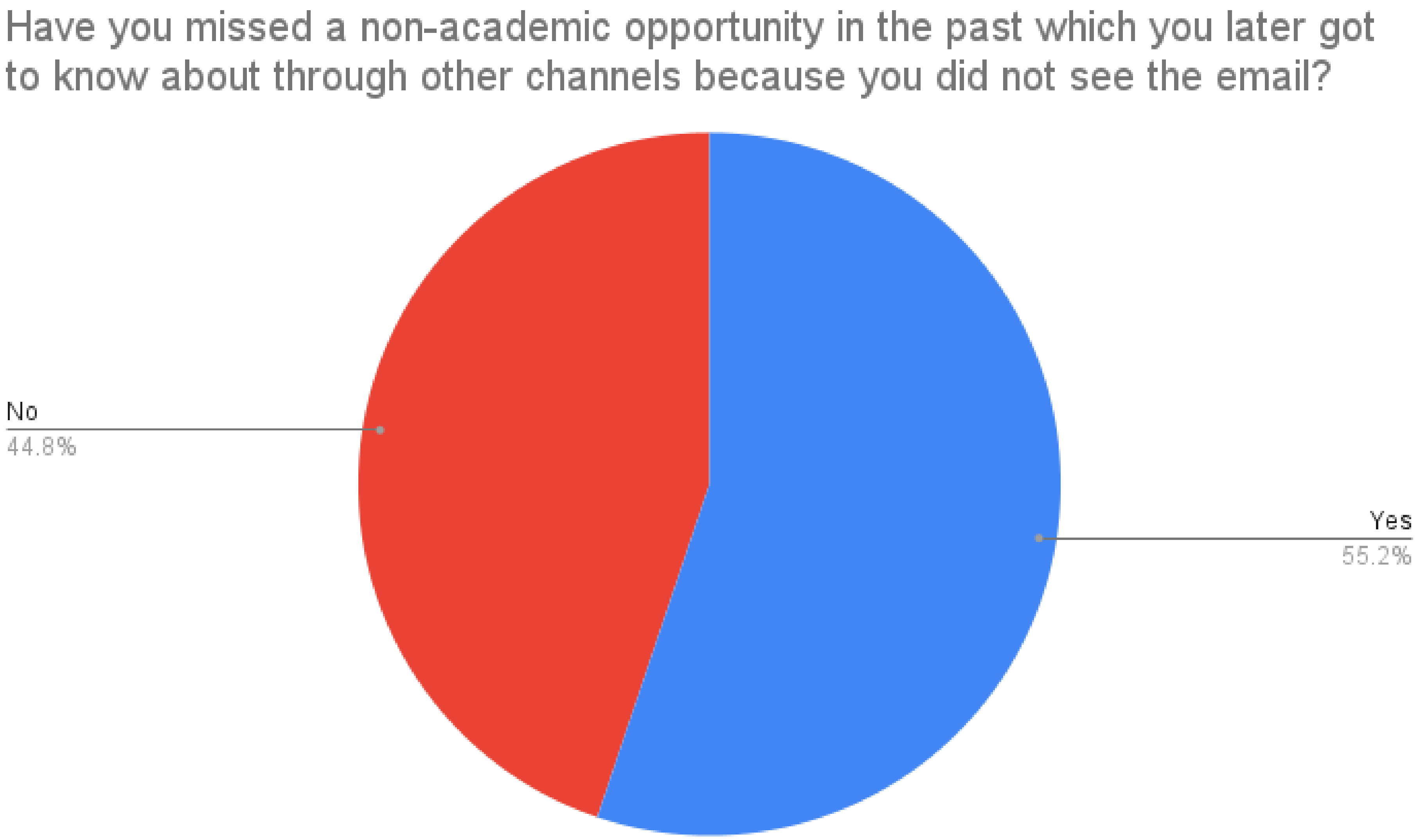
the problem.

some cheesy lyric.



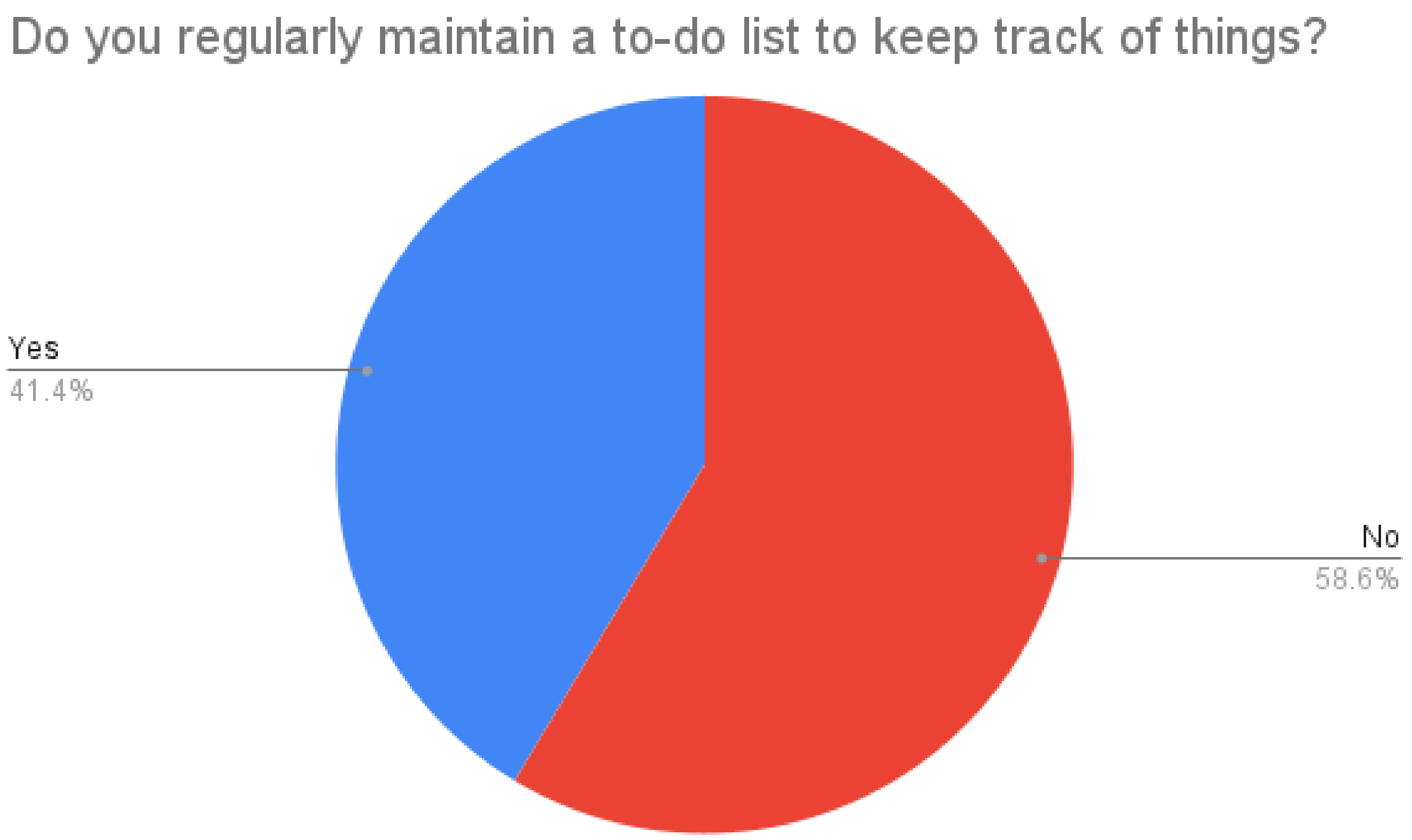
the problem.

trust me, i'm trying.



the problem.

sorry, i give up.



the problem.

no, but seriously -

168 million

emails sent every minute

39%

emails are information only

29%

emails have unnecessary CCs

17%

emails are irrelevant/spam

15%

emails are important

(and they end up getting lost)

Louis Eugene, Isaac Caswell, Making a Manageable Email Experience using Deep Learning,
https://cs224d.stanford.edu/reports/EugeneLouis.pdf

the problem.

what's going on?

a plaksha student -

enrols in
6-8 courses/sem

receives emails
from 16-18
channels

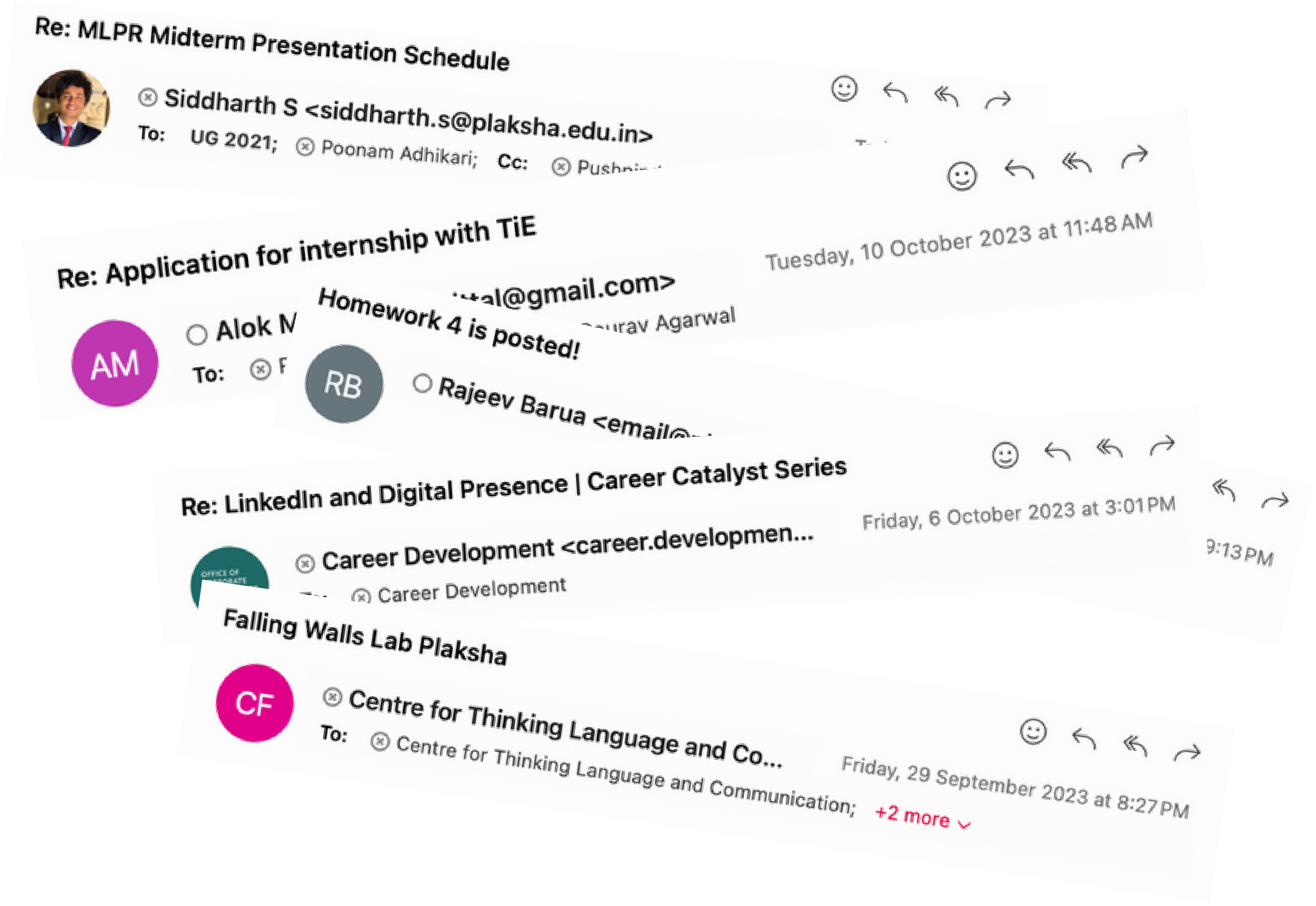
gets overwhelmed

misses deadlines
and opportunities

proposed solution.

your email mom.

remembering what needs to be done, so you don't have to.



Hi Rishi,

Here's what landed in your inbox today -
Important - MLPR Final Presentation on Friday
Important - PA Assignment due on Wednesday
Moderate - TiE internship follow-up with Mr. Alok Mittal
Relaxed - Become audience at Falling Labs Plaksha
Relaxed - Session on LinkedIn presence on Saturday

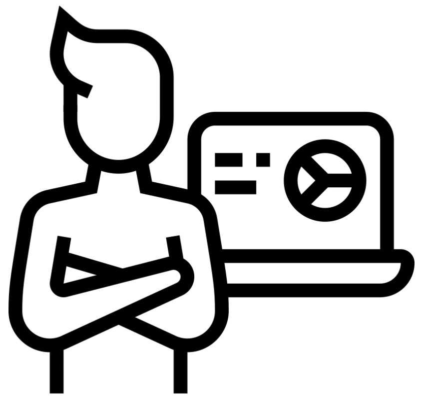
Cope well, don't jump off a cliff.

Best,
ruch.ai

proposed solution.

paisa, kya cheez hai paisa -

with rich and diverse data, this approach has potential beyond the walls of MLPR.

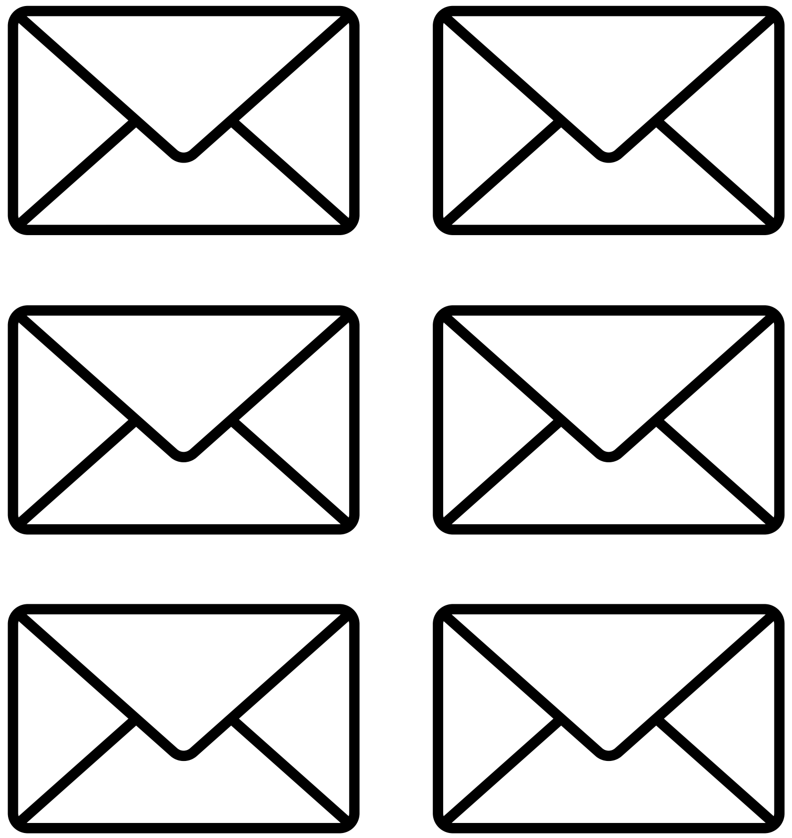


working professionals

government officials

customer support

collecting the data



Considerations during data collection:

- the information we need: Senders, Date, Subject, Body

How we collected:

- IMAP- failed
- Microsoft Graph API- credits to Devesh Shah, our amazing Tech. Sec.

Ethical concerns:

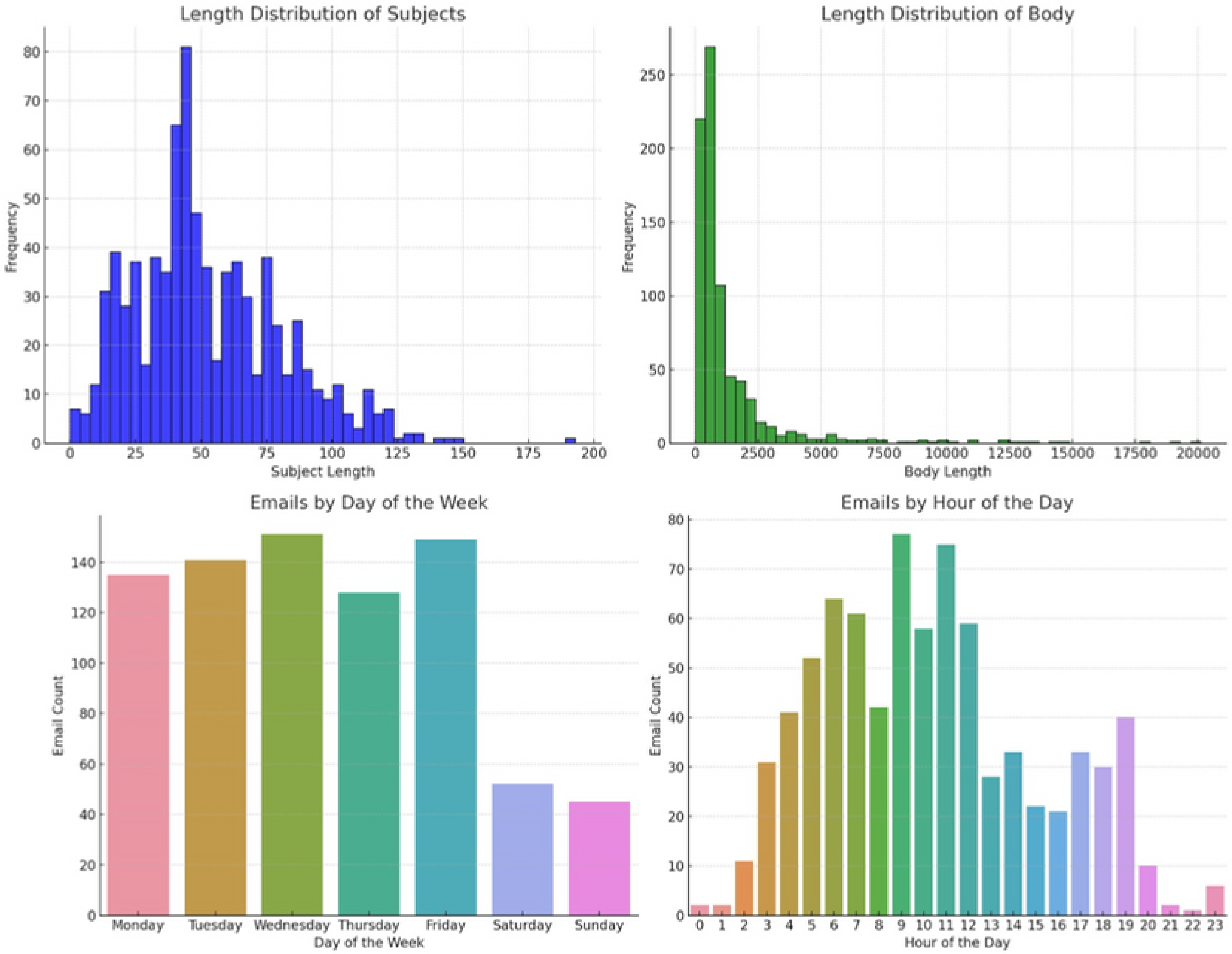
- Private information about certain organisations- created a condition to skip those emails

pre-processing the data

1. removing HTML or Markup
2. lowercasing
3. tokenization
4. stopword removal
5. special character and number removal
6. expanded contractions
7. whitespace trimming
8. NaN values were dropped

the data.

eda



Length Distribution of Subjects
Frequency
Subject Length
Length Distribution of Body
Frequency
Body Length
Emails by Day of the Week
Email Count
Day of the Week
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Emails by Hour of the Day
Email Count
Hour of the Day

literature review.

existing work -

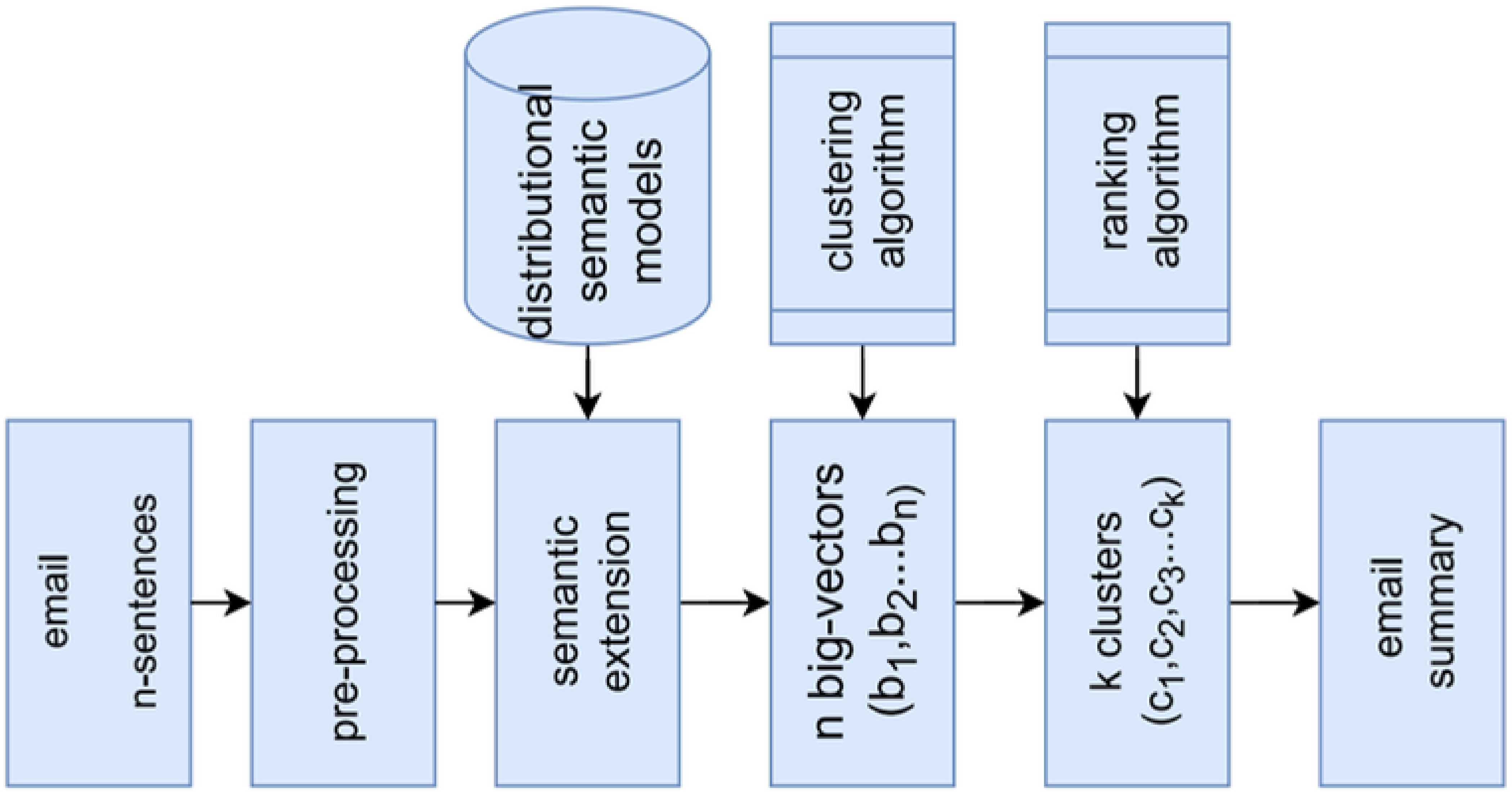
extensive work has been done to mitigate email overload, specifically in the domain of **email classification**.

here, we illustrate a few examples of how this task has been tackled by other groups.

since there is no uniform way to rank email prioritisation systems, the **performance** of these methods depends largely on the **datasets** used.

literature review.

ranking sentences in semantically similar big vectors



Mahira Kirmani, Gagandeep Kaur, Mudasir Mohd,ShortMail: An email summarizer system, Software Impacts, Volume 17, 2023, 100543, ISSN 2665-9638, https://doi.org/10.1016/j.simpa.2023.100543.

literature review.

ranking sentences in semantically similar big vectors

pre-processing -
remove irrelevant content, threads, punctuations, etc. tokenize, convert to lowercase and lemmatize.

→

semantic extension -
use Google's BERT (Bidirectional Encoder Representations from Transformers) to obtain semantics of the text.

→

big vector generation -
concatenate similar words as obtained during semantic extension to rich big vectors

↓

clustering -
use the K-means algorithm to create clusters of semantically similar sentences from big vectors.

←

features -
1. *sentence position*
2. *frequency (TF-IDF)*
3. *proper nouns*
4. *cosine similarity*

←

ranking -
rank sentences in the clusters based on features, the total score is the sum of the normalized score of each sentence in the cluster. Present top sentences to user

Mahira Kirmani, Gagandeep Kaur, Mudasir Mohd,ShortMail: An email summarizer system, Software Impacts, Volume 17, 2023, 100543, ISSN 2665-9638, https://doi.org/10.1016/j.simpa.2023.100543.

literature review.

ranking sentences in semantically similar big vectors

algorithm for big-vector generation

Let D be input email
s_i is the sentences of email D

for all $s_i \in$ D **do**
  $W \leftarrow Tokenization(s_i)$
  where $W = \{w_1, w_2, w_3, ..., w_n\}$

  for all $w_i \in W$ **do**
    $V_i \leftarrow BERT(W_i)$
  end for
  $BV = V_1 \oplus V_2 \oplus \cdots \oplus V_{|W|}$
  $\oplus$ is concatenation
end for

Let $\delta(w)$ is a function for retrieving a top list of 'm' words from a semantic model. The function is given as $w' = \delta(w) = w'_1 \oplus w'_2 \oplus, \ldots, \oplus w'_m$. For a sentence with $W = \{w_1, w_2, w_3, \ldots., w_k\}$ as the sequence of k tokenized words, a big-vector BGV is populated by concatenating respective top m similar words for each word i.e $BGV = \{\delta(w_1) \oplus \delta(w_2) \oplus, \ldots, \oplus \delta(w_k)\}$.

Mahira Kirmani, Gagandeep Kaur, Mudasir Mohd,ShortMail: An email summarizer system, Software Impacts, Volume 17, 2023, 100543, ISSN 2665-9638, https://doi.org/10.1016/j.simpa.2023.100543.

summarization using statistical methods

each sentence is represented as a vector of **7 features**, each feature being a value between 0 and 1.

130 emails were summarised using this approach and by a human for reference. these summaries were then compared.

Mithak I. Hashem, Improvement of Email Summarization using Statistical Based Methods,
https://ijcsmc.com/docs/papers/February2014/V3I2201488.pdf

summarization using statistical methods

title feature

$$S.S.(1) = \frac{No.of\ Title\ Words\ in\ Sentence\ S}{No.of\ Words\ in\ title}$$

term weight feature

$$Wi = tfi\ x\ logN/n \qquad (2)$$

where *tfi* is the term frequency of word *i* in the document, *N* is the total number of sentences, and *n* is number of sentences in which word *i* occurs. This feature can be calculated as follows [19]:

$$S.S.(2) = \frac{\sum_{i=1}^{m} Wi(S)}{Max[\sum_{i=1}^{m} Wi(S)]_{i=1}^{n}} \qquad (3)$$

Where *m* is number of words in sentence.

sentence position feature

$$S.S.(3) = \frac{1}{5}\ for\ the\ fist\ sentence, \frac{5}{5}\ for\ the\ second, \frac{4}{5}\ for\ the\ third, \frac{3}{5}\ for\ the\ fourth, \frac{2}{5}\ for\ the\ second, \frac{0}{5}\ for\ the\ others. \qquad (4)$$

event feature

$$S.S.(4) = \frac{2}{2}\ if\ place\ and\ time\ have\ been\ mentioned\ in\ the\ sentence\ S, or \quad \frac{1}{2}\ if\ one\ factor\ (place\ or\ time)\ have\ been\ mentioned\ in\ the\ sentence\ S, or \quad \frac{0}{2}\ for\ others. \qquad (5)$$

Mithak I. Hashem, Improvement of Email Summarization using Statistical Based Methods, https://ijcsmc.com/docs/papers/February2014/V3I2201488.pdf

literature review.

summarization using statistical methods

proper nouns feature

$$S.S.(5) = \frac{No.\,of\ Proper\ Nouns\ in\ the\ single\ sentence\ S}{Maximum\ number\ of\ Prper\ Nouns\ within\ Email}$$

numerical data feature

$$S.S.(6) = \frac{No.\,of\ Numerical\ Data\ in\ the\ single\ sentence\ S}{Length\ of\ the\ Sentence\ S}$$

topical words feature

$$S.S.(7) = \frac{No.\,of\ topical\ words\ in\ the\ single\ sentence\ S}{maximum\ No.\,of\ topical\ words\ in\ the\ Sentence\ S}$$

sentence score calculation

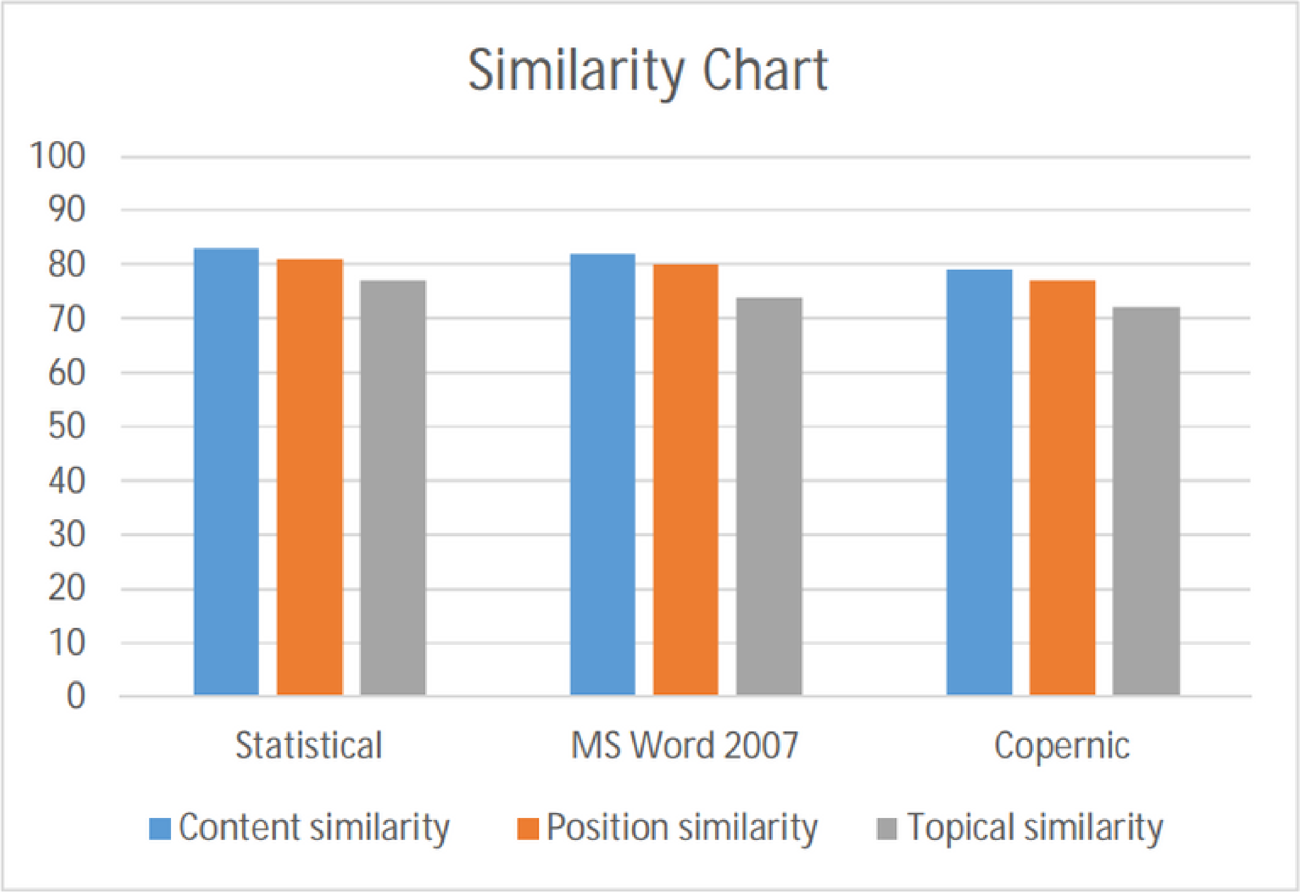
$$Score(S) = \sum_{k=1}^{7} S.S.(k)$$

Mithak I. Hashem, Improvement of Email Summarization using Statistical Based Methods,
https://ijcsmc.com/docs/papers/February2014/V3I2201488.pdf

literature review.

summarization using statistical methods

Summarizer	Average		
	Content Similarity	Position Similarity	Topical Similarity
Statistical	0.83	0.81	0.77
MS Word 2007	0.82	0.80	0.74
Copernic	0.79	0.77	0.72

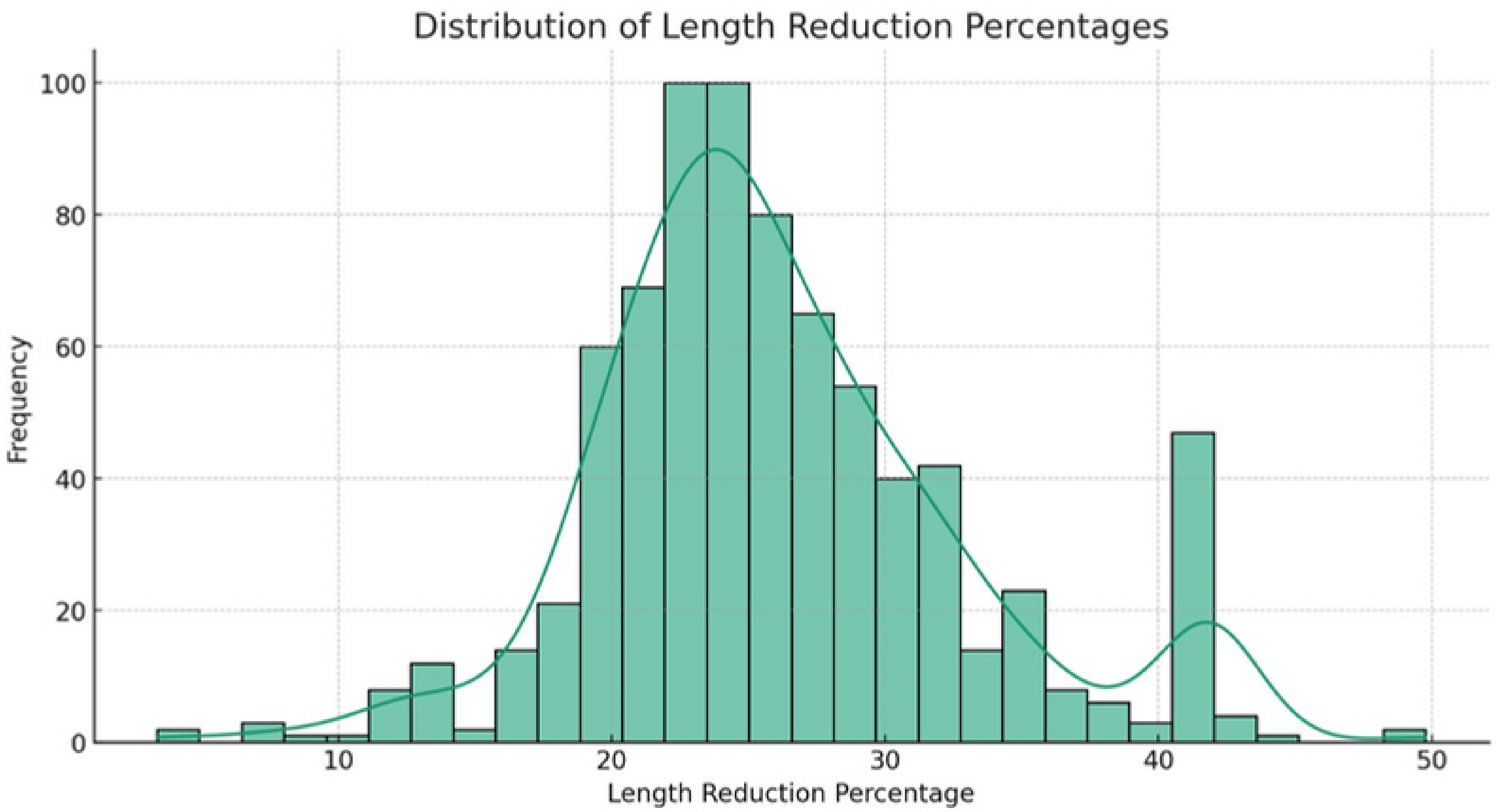


Mithak I. Hashem, Improvement of Email Summarization using Statistical Based Methods, https://ijcsmc.com/docs/papers/February2014/V3I2201488.pdf

our approach.

summarization - tf-idf

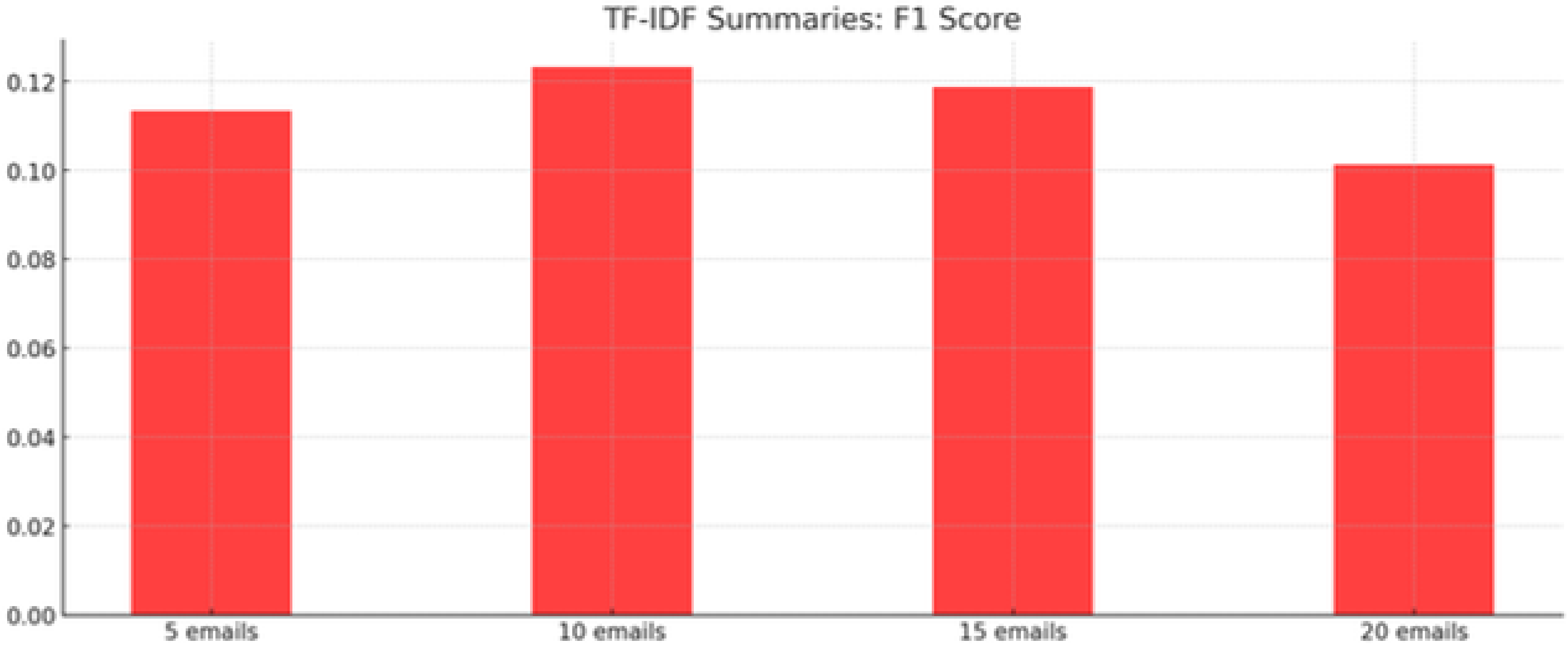
length reduction:



our approach.

summarization - tf-idf

f1-score:

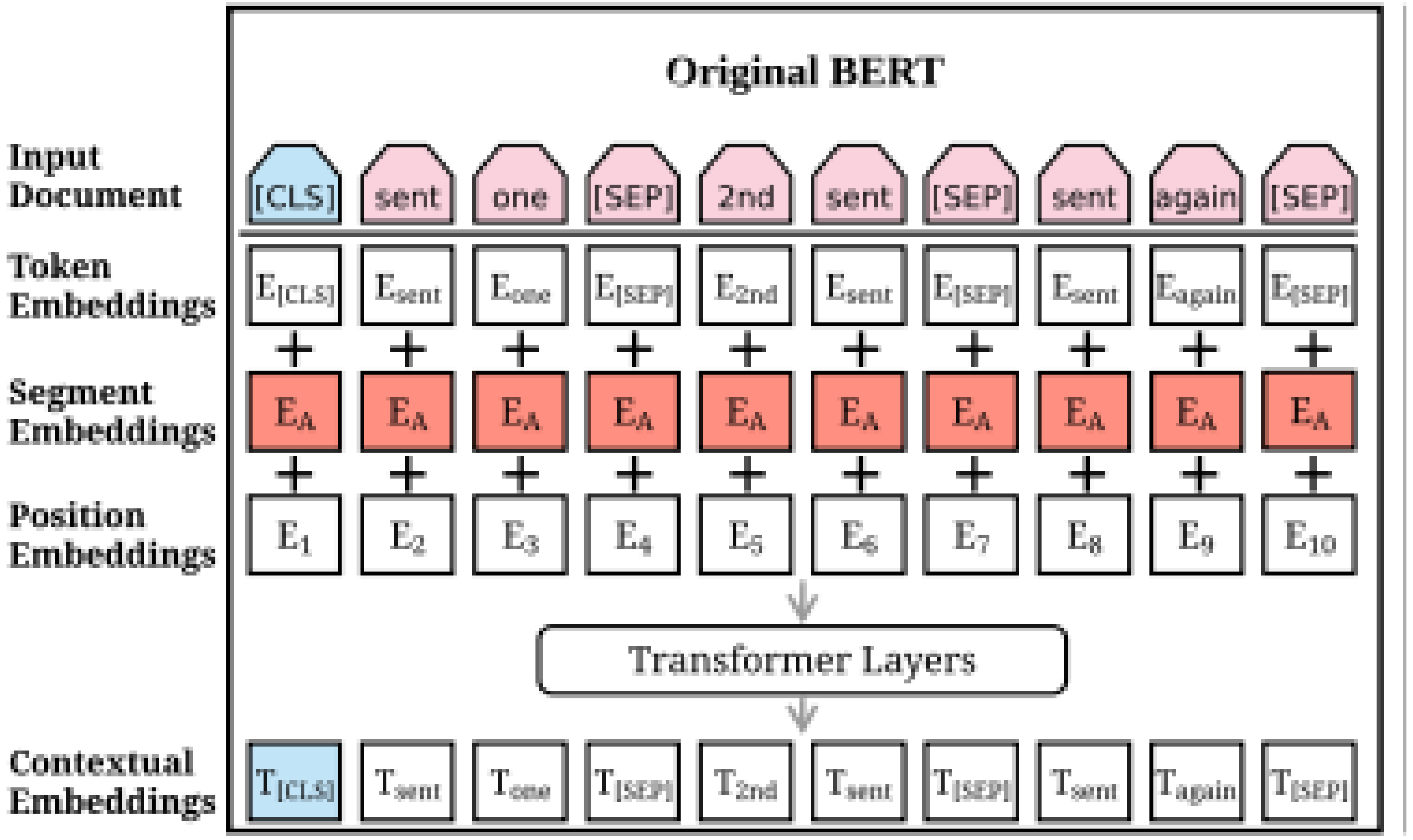


our approach.

summarization - bert

Model	R1	R2	RL
ORACLE	29.79	8.81	22.66
LEAD	16.30	1.60	11.95
Abstractive			
PTGEN (See et al., 2017)	29.70	9.21	23.24
PTGEN+COV (See et al., 2017)	28.10	8.02	21.72
TCONVS2S (Narayan et al., 2018a)	31.89	11.54	25.75
TransformerABS	29.41	9.77	23.01
BERT-based			
BERTSUMABS	38.76	16.33	31.15
BERTSUMEXTABS	38.81	16.50	31.27

Table 4: ROUGE F1 results on the **XSum** test set. Results for comparison systems are taken from the authors' respective papers or obtained on our data by running publicly released software.

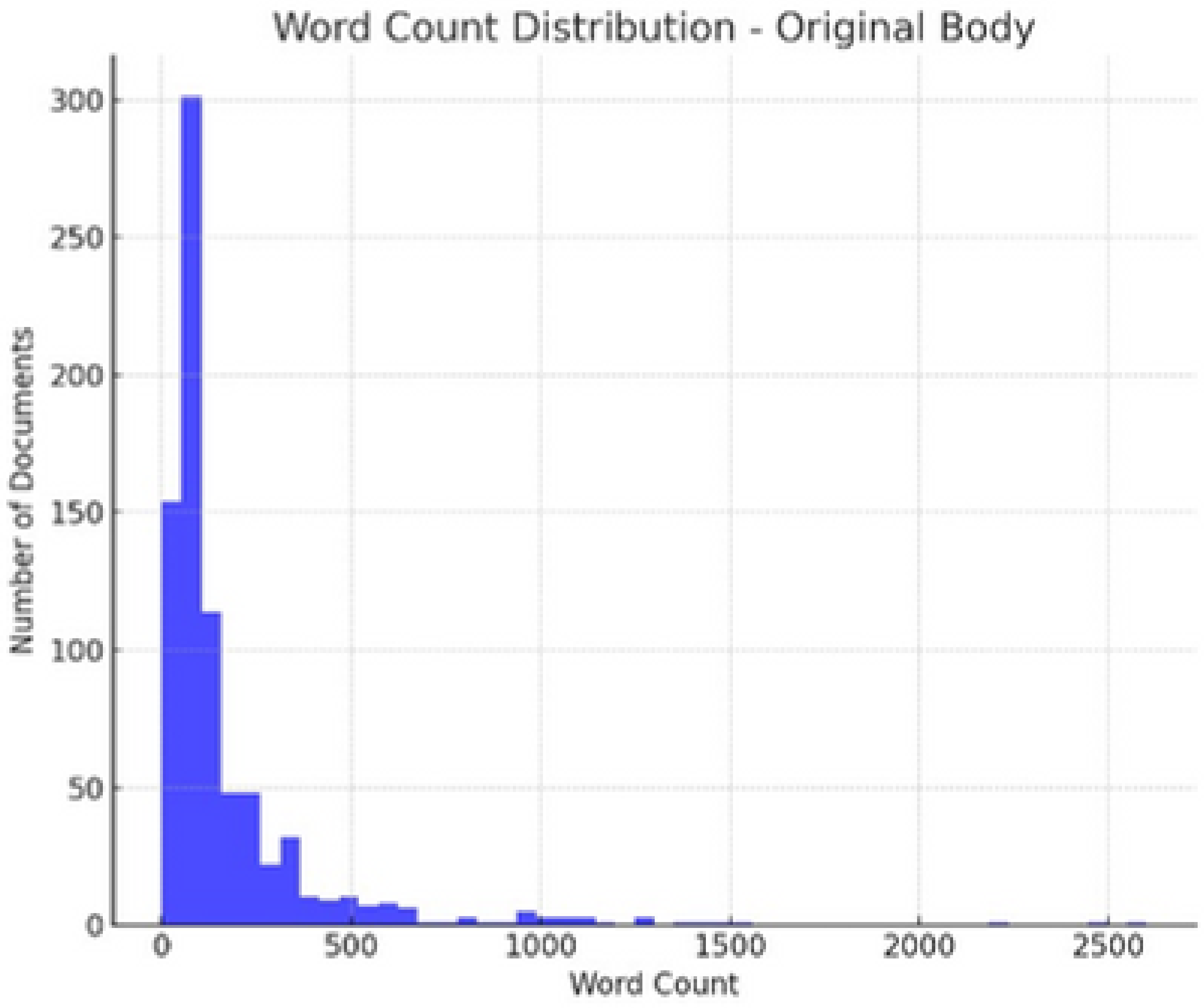


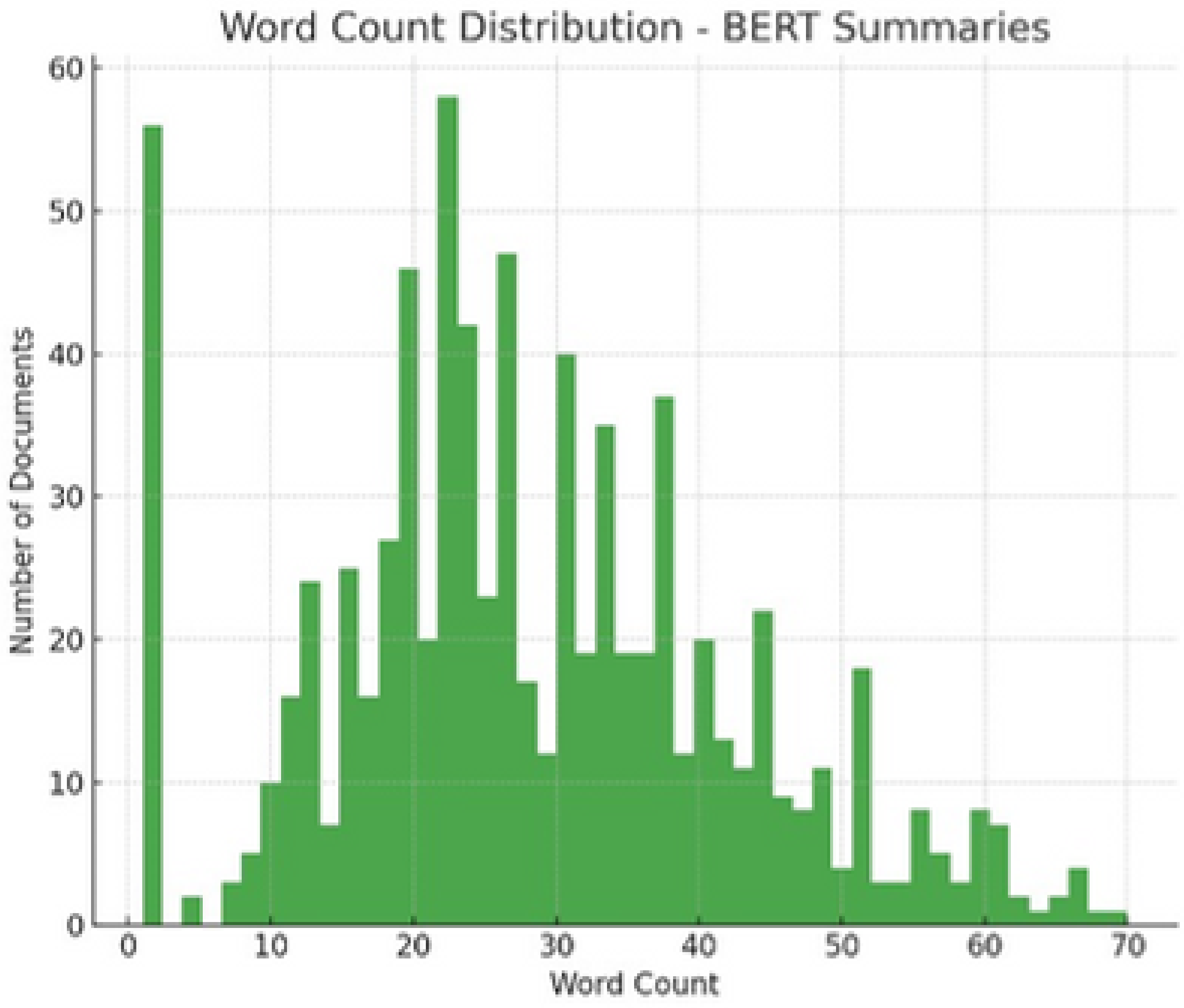
Liu, Y., & Lapata, M. (2019, August 22). Text Summarization with Pretrained Encoders. arXiv.Org. https://arxiv.org/abs/1908.08345v2

our approach.

summarization - bert

length reduction:

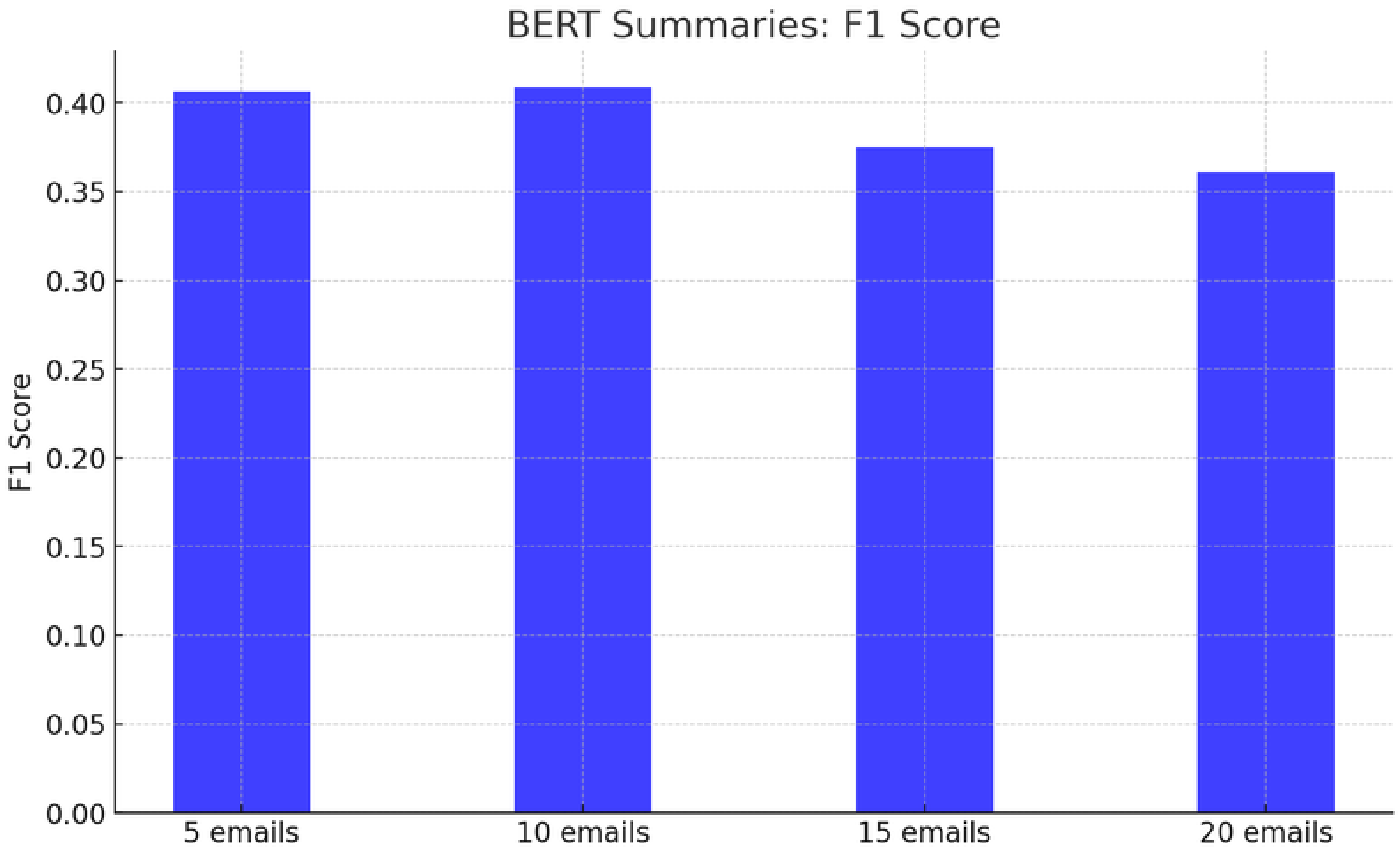




our approach.

summarization - bert

f1-score:

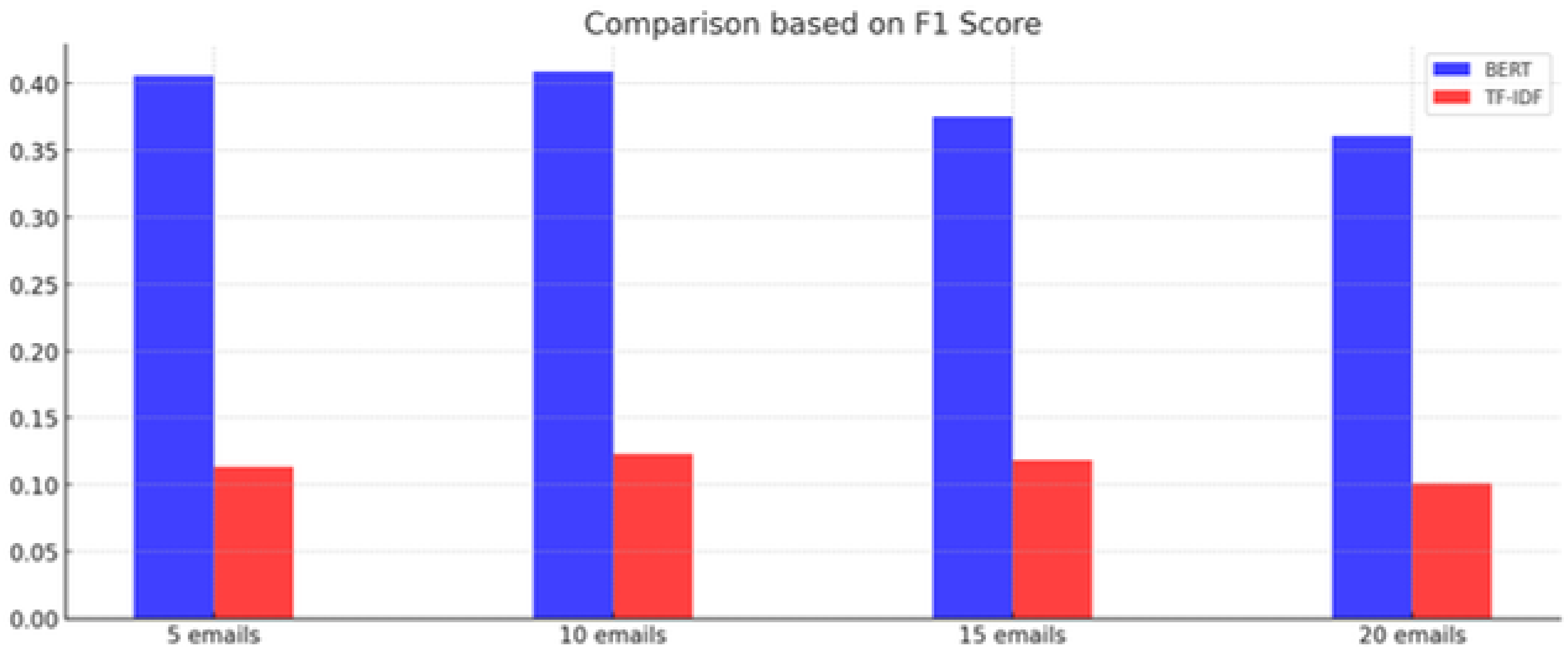


our approach.

tf-idf vs bert

comparison

```
PS C:\Users\bhavi> python -u "c:\Users\bhavi\Downloads\performaces.py"
   Number of Emails  BERT Precision  BERT Recall  BERT F1 Score  TF-IDF Precision  TF-IDF Recall  TF-IDF F1 Score
0                 5           0.320        0.600          0.406             0.069          0.333            0.113
1                10           0.322        0.624          0.409             0.077          0.329            0.123
2                15           0.286        0.617          0.375             0.075          0.327            0.119
3                20           0.265        0.663          0.361             0.063          0.305            0.101
PS C:\Users\bhavi>
```



our approach.

bert summaries

results

```
PS C:\Users\bhavi> python -u "c:\Users\bhavi\Downloads\bert_summaries.py"
•  Program analysis  assignments: due date for the assignment "homework 3" is 2 days aways .

•  A student wants to meet with professor tanmoy to discuss their project topic . The preferred date for this discussion is t Tuesday, October 3, 2023 .

•  Last call to register and be a part of the audience at the falling walls lab plakshal on 30th september 2023 .

•  Please join tomorrow's class on the microsoft teams link given below at 11:00 am . The students are required to provide their project progress update .

•  Search methods in artificial intelligence  "9. smai-astar-space-saving-versions" has been created .

PS C:\Users\bhavi>
```

literature review.

prioritisation using deep learning

svm

(gaussian kernel)

random forests

(100 estimators, split on Gini Criterion)

vs

lstm

&

cnn

features -
bag of words
unigrams and bigrams

features -
dense word vectors
(Word2Vec)

Eugene, Louis, and Isaac Caswell. "Making a Manageable Email Experience with Deep Learning." Department of Management Science and Engineering, Stanford University; Department of Computer Science, Stanford University. https://cs224d.stanford.edu/reports/EugeneLouis.pdf

literature review.

prioritisation using deep learning

data

Parakweet Lab's Email Intent Dataset

comes from the **Enron Data Corpus** (600k emails)
each labelled case is one sentence from an email

training data	testing data
4213	**991**
cases	**cases**
1631	**277**
positives	**positives**

Isaac Caswell's Stanford Inbox

two kinds of labels -
if the email was replied to and gmail's importance flag

~26,000

emails

labels -
former as a proxy for the email's importance
latter as a measure for how well the data agrees with gmail's ranking

Eugene, Louis, and Isaac Caswell. "Making a Manageable Email Experience with Deep Learning." Department of Management Science and Engineering, Stanford University; Department of Computer Science, Stanford University. https://cs224d.stanford.edu/reports/EugeneLouis.pdf

literature review.

prioritisation using deep learning

results

evaluation metric - **F1 score**

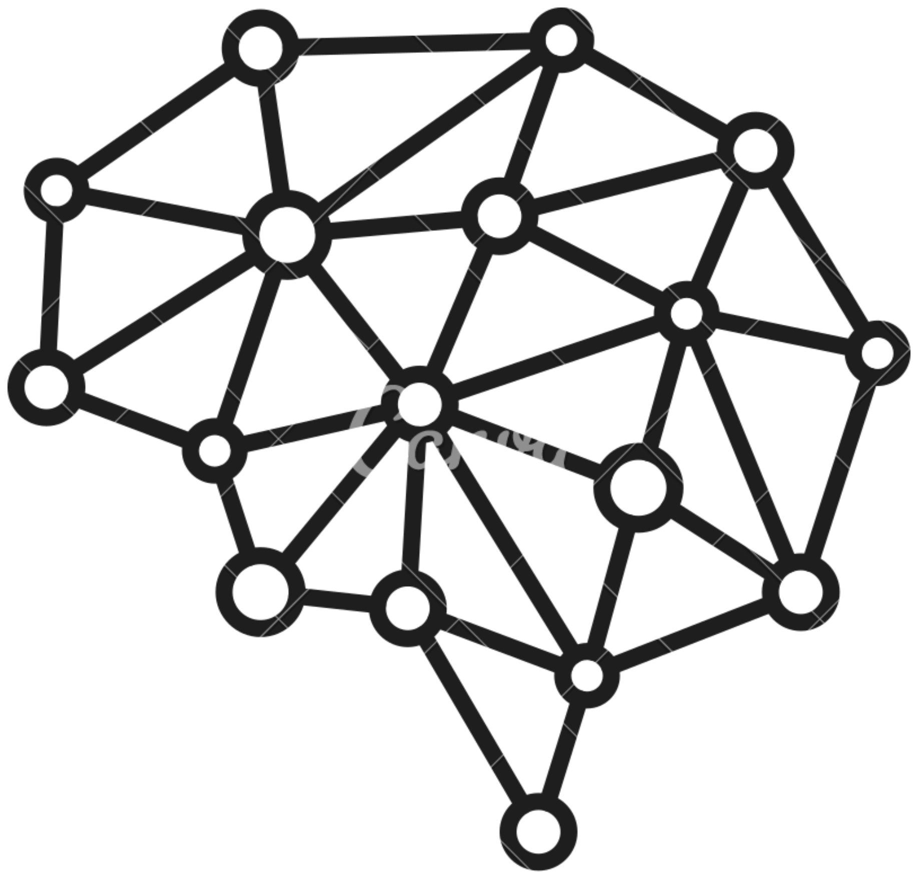
	Parakweet	Isaac-rep-1	Isaac-imp-1	Isaac-rep-all	Isaac-imp-all
RF	0.772	0.887	**0.826**	0.891	**0.904**
SVM	0.778	0.876	0.799	0.882	0.871
LSTM	0.809	0.854	0.801	0.900	0.879
CNN	**0.811**	**0.913**	0.817	**0.914**	0.900

1. cnn and lstm outperformed the baselines **81-91% F1 Scores**
2. cnn - best-performing algorithm (**6.2% better performance** than the previous state-of-the-art F1 on Parakweet - LibSVM)
3. training time for cnn was longer than baseline algorithms (multiple days)

Eugene, Louis, and Isaac Caswell. "Making a Manageable Email Experience with Deep Learning." Department of Management Science and Engineering, Stanford University; Department of Computer Science, Stanford University. https://cs224d.stanford.edu/reports/EugeneLouis.pdf

ml methodology.

where and how ML is used



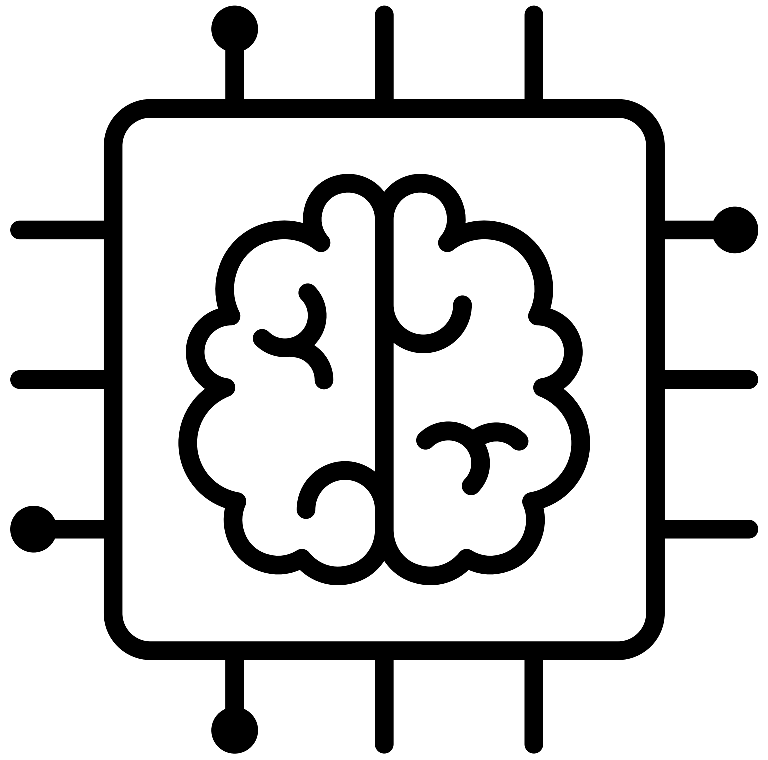
we will be evaluating multiple machine learning algorithms on our pre-processed data to prioritize emails based on their importance and whether they require a response

the ml model would be to prioritise basis these 3 features:

1. does the message require action?
2. has the email service labeled this email as important?

ml methodology.

support vector machines



- Support Vector Machine (SVM) is a machine learning algorithm that finds the best hyperplane to separate different classes in a dataset, maximizing the margin between them.
- Our baseline model employs a Support Vector Machine with a RBF kernel, processing unigram and bigram features extracted through TF-IDF vectorization of the BERT Summaries.
- it served as an excellent base classical machine learning model used for multi -class classification.

ml methodology.

performance- support vector machine

```
Accuracy: 0.7181208053691275

Classification Report:
              precision    recall  f1-score

           1       0.63      0.85      0.72
           2       0.78      0.76      0.77
           3       0.92      0.38      0.54

    accuracy                           0.72
   macro avg       0.78      0.66      0.68
weighted avg       0.75      0.72      0.71
```

ml methodology.

CNN

What is CNN?

A Convolutional Neural Network (CNN) is a deep learning architecture designed for image processing and pattern recognition. It uses convolutional layers to automatically learn and extract hierarchical features from input data.

Why CNN?

CNN was the best-performing deep approach in the Stanford study. Moreover, CNNs capture local patterns and hierarchical features in text, enabling effective feature extraction for text prioritisation.

ml methodology.

performance- CNN

Performance Metrics:

	Metric	Value
0	Accuracy	0.677852
1	Precision	0.677718
2	Recall	0.677852
3	F1 Score	0.676188

ml methodology.

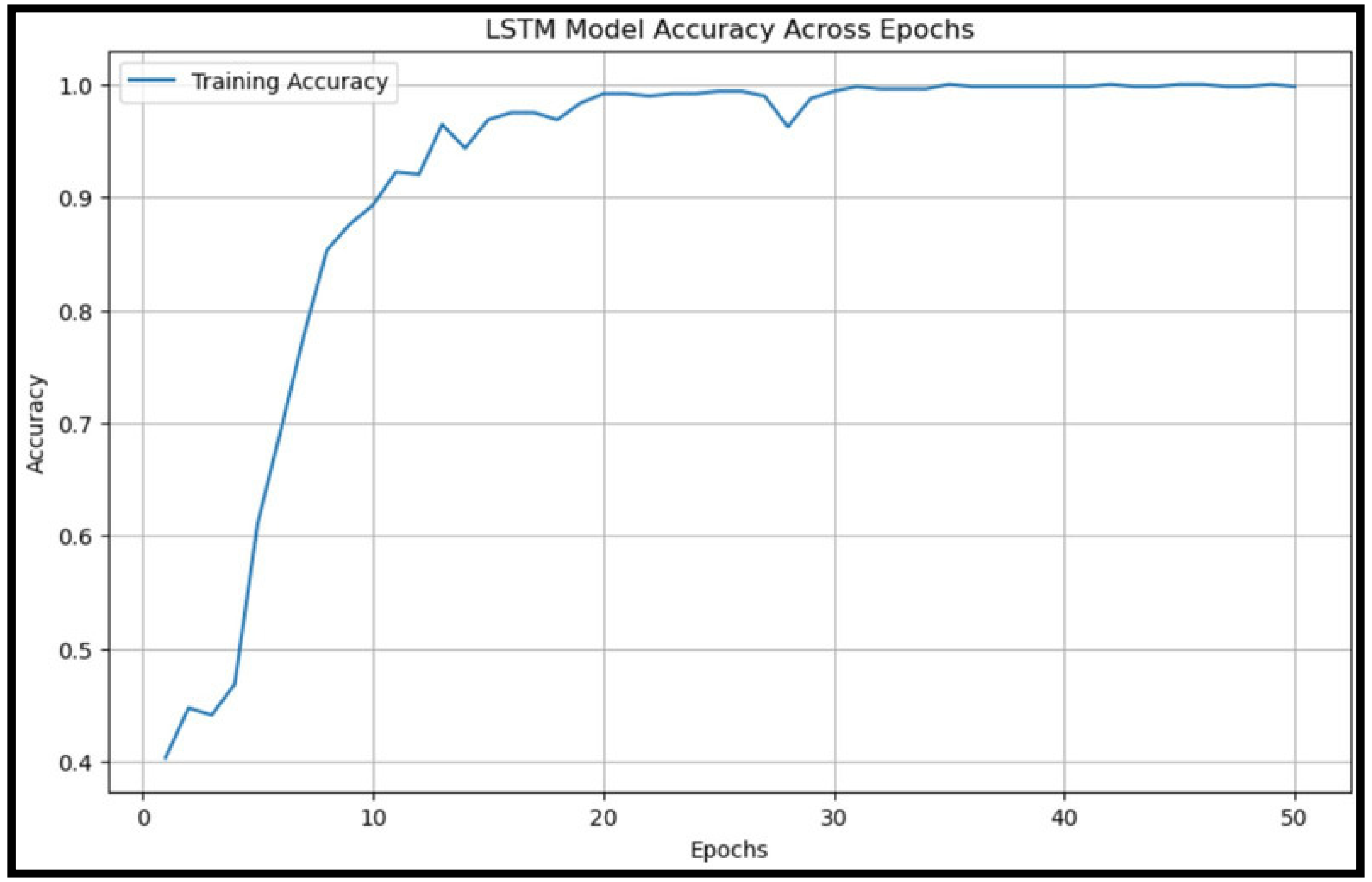
LSTM

- LSTM captures short and long-term dependencies.
- It identifies keywords, sentiment, and important phrases (sequential and contextual understanding) for relevance and urgency assessment.
- The model is trained on labeled email data with a focus on optimizing accuracy.
- Once trained, it can efficiently evaluate new emails based on content, enabling automated email prioritization.

ml methodology.

performance- LSTM

```
The optimal dropout rate is 0.2.
The optimal recurrent dropout rate is 0.30000000000000004.
The optimal learning rate for the optimizer is 0.00031418620419296233.
```




```
Accuracy: 0.7785
Precision: 0.7787
Recall: 0.7785
F1-Score: 0.7764
```

ml methodology.

comparitive analysis

Metric	SVM	CNN	LSTM
Accuracy	0.7181	0.677	0.7785
Precision	0.75	0.677	0.7787
Recall	0.72	0.677	0.7785
F1-Score	0.71	0.676	0.7765

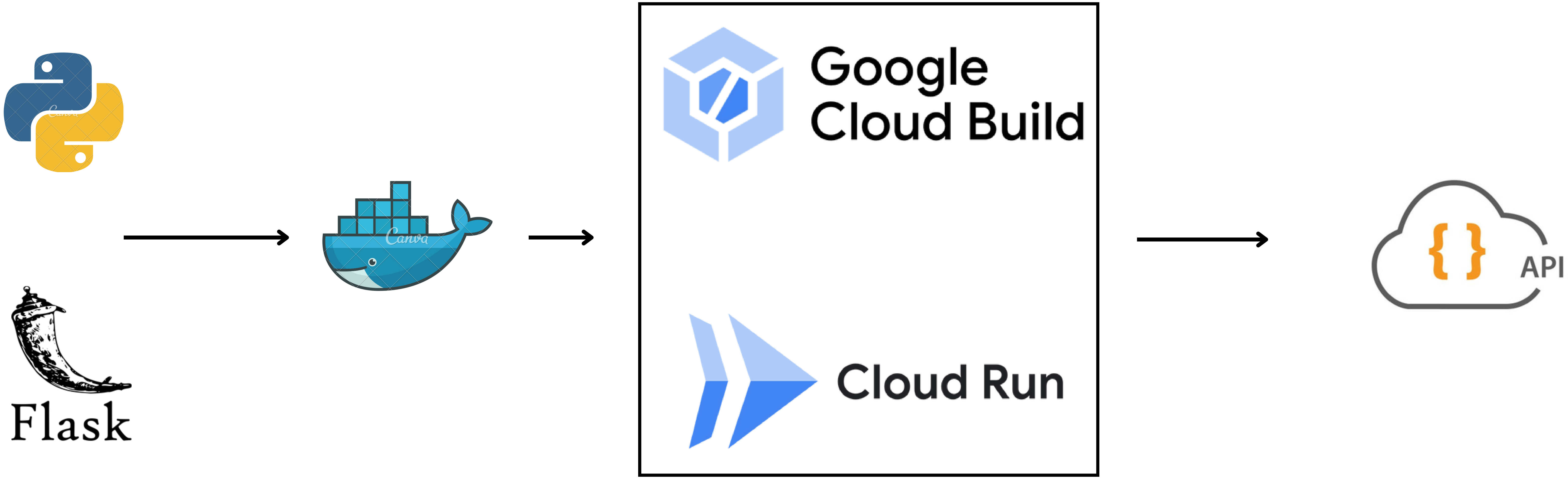
Table 1: Comparison of SVM, CNN, and LSTM

winner- LSTM!!

deployment

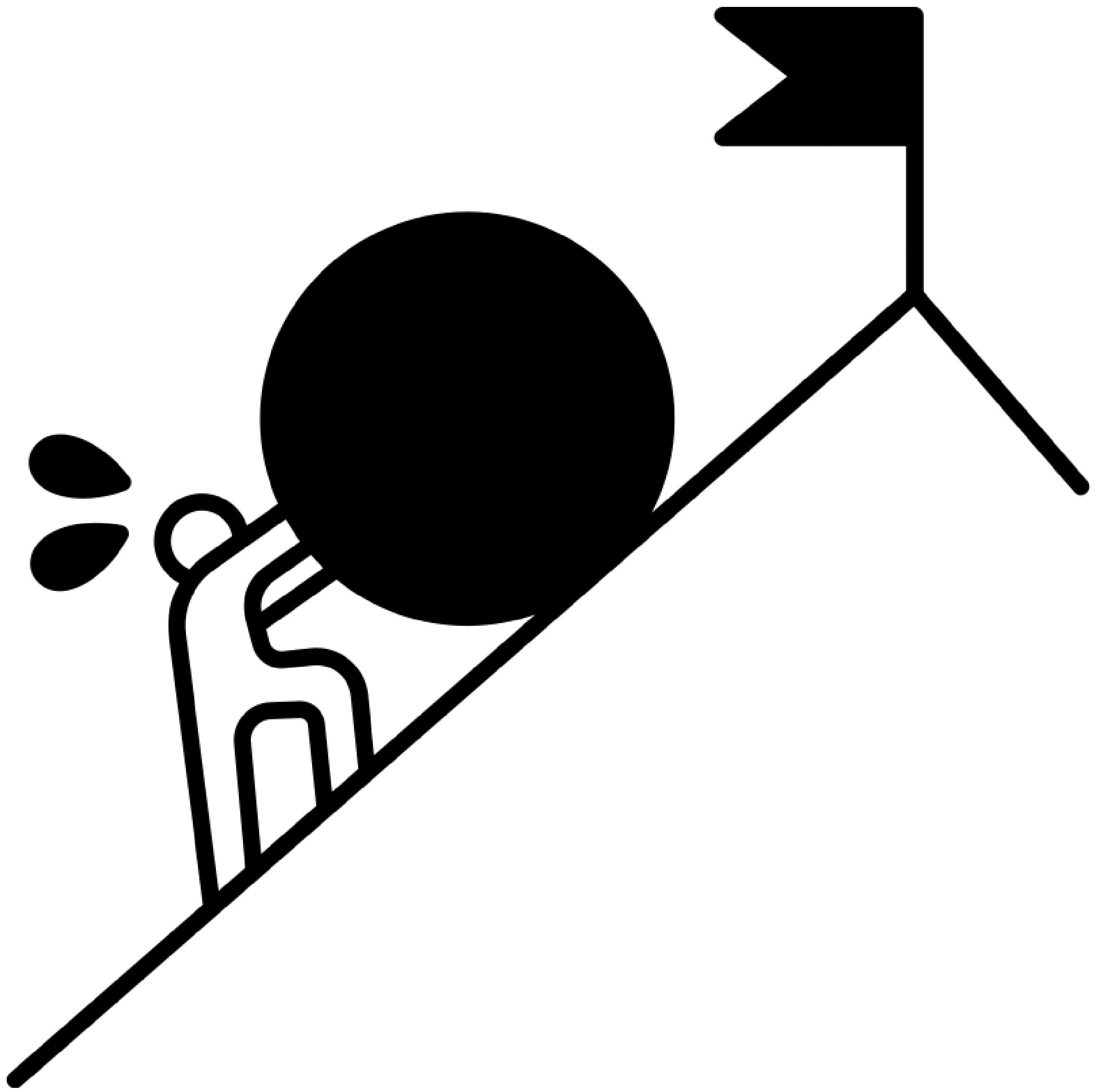
to the moooooooooon 🚀

deployment pipeline -



what else

challenges and future possibilities



1. **labeling and upscaling training data** - only option was and to manually label all the emails which was time-consuming and had 3 different perspectives. For better accuracy, we can upscale the data.
2. **ethical concerns over data** - users may be not comfortable with the idea of an ai tool having access to potentially sensitive information.
3. **task ambiguity -** prioritization and summarization tasks are highly subjective and context-dependent. Defining what is important and summarizing information accurately is challenging and highly personal. For the future, we can emply a personalised priority rank system using outlook plugins and personalised model training.
4. **plugin**- we plan to integrate our model into an Outlook add-in, designed for the academic community at Plaksha University.

thank you.

ruch.ai

(Scout consents to this image being taken.)