Final Project Presentation

Matthew Burke, Aaron Finkelstein, Jin-seo Kim and Sindhura Mente

MATT

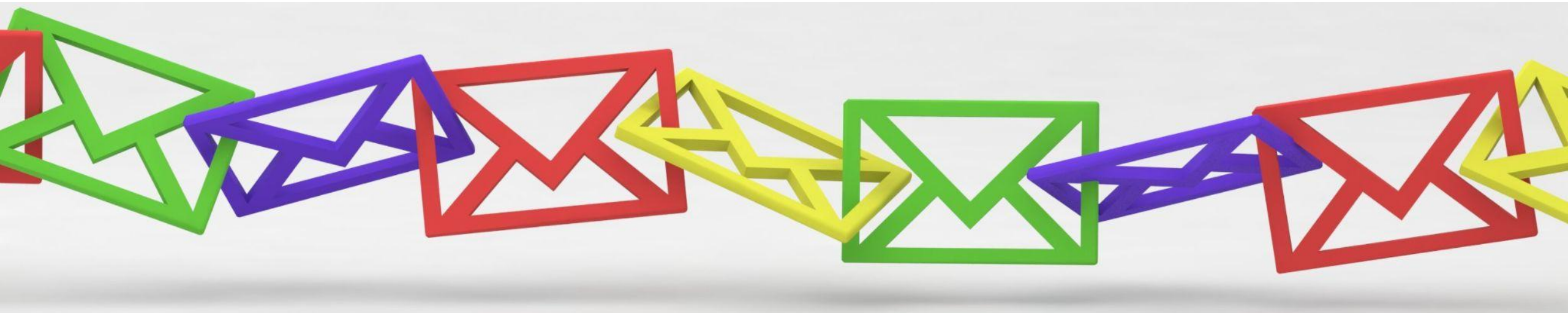
CIS 5300 Prof Mark Yatskar
TA Advisor: Yufei Wang
December 22, 2023

Business Email Abstractive Threads Summarization (BEATS)

NLP models reading a chain of 30 emails beats reading them yourself

Overview & Project Description

MATT



Background

- The 2021 paper by Shiyue Zhang et al.**, "EmailSUM: Abstractive Email Thread Summarization"** is our main reference.
- We want a model that can summarize business email threads: Who participated, what was discussed, and what were the conclusions or pending questions.

Motivation

- We all face a barrage of email chains in everyday life, in fact:
  - Business professionals spend almost **one-third of their day on email**.
  - Most are low information density.
  - Through summarization, the time spent reading emails could be dramatically cut down.

Problem Statement

- A business email thread is not a single document or a dialog; it is an asynchronous branching discussion between multiple writers exchanging information, quoting each other, and succeeding or failing to reach agreements.
- Email threads are hard to summarize, and a good summary can save a lot of time.

Related Work

Includes Hyperlinks for Additional Information

	Lapata & Liu 2019	El-Kassas et al. 2021	Gao and Wan 2022	Zhang et al. 2021
REVIEW	**Practical**	**Comprehensive**	**Cutting Edge**	**Gold Standard**
CONCERNS	**Used ROUGE as a benchmark, which fails to capture semantics.**	**Provides an overview of approaches, rather than a methodology.**	**Proposed several metrics for testing (BARTScore, QA-based).**	**Benchmark only measures n-gram similarity.**
STRENGTHS	**Text Summarization with Pretrained Encoders** examines how well pretrained models do at text summarization with and without fine tuning and presents additional methods to improve them for this task.	**Automatic text summarization: A comprehensive survey** categorizes ATS systems based on input size, summarization approach, output nature, summary language, algorithms used, and summary content, type, or domain.	**DialSummEval: Revisiting Summarization Evaluation for Dialogues** re-evaluate 18 categories of metrics in terms of four dimensions: coherence, consistency, fluency and relevance, as well as a unified human evaluation of various models for the first time.	Developed an abstractive Email Thread Summarization dataset, **EmailSum**, containing human-annotated short and long summaries of over 2500 email threads of various topics.

MATT

What is exciting about your term project?

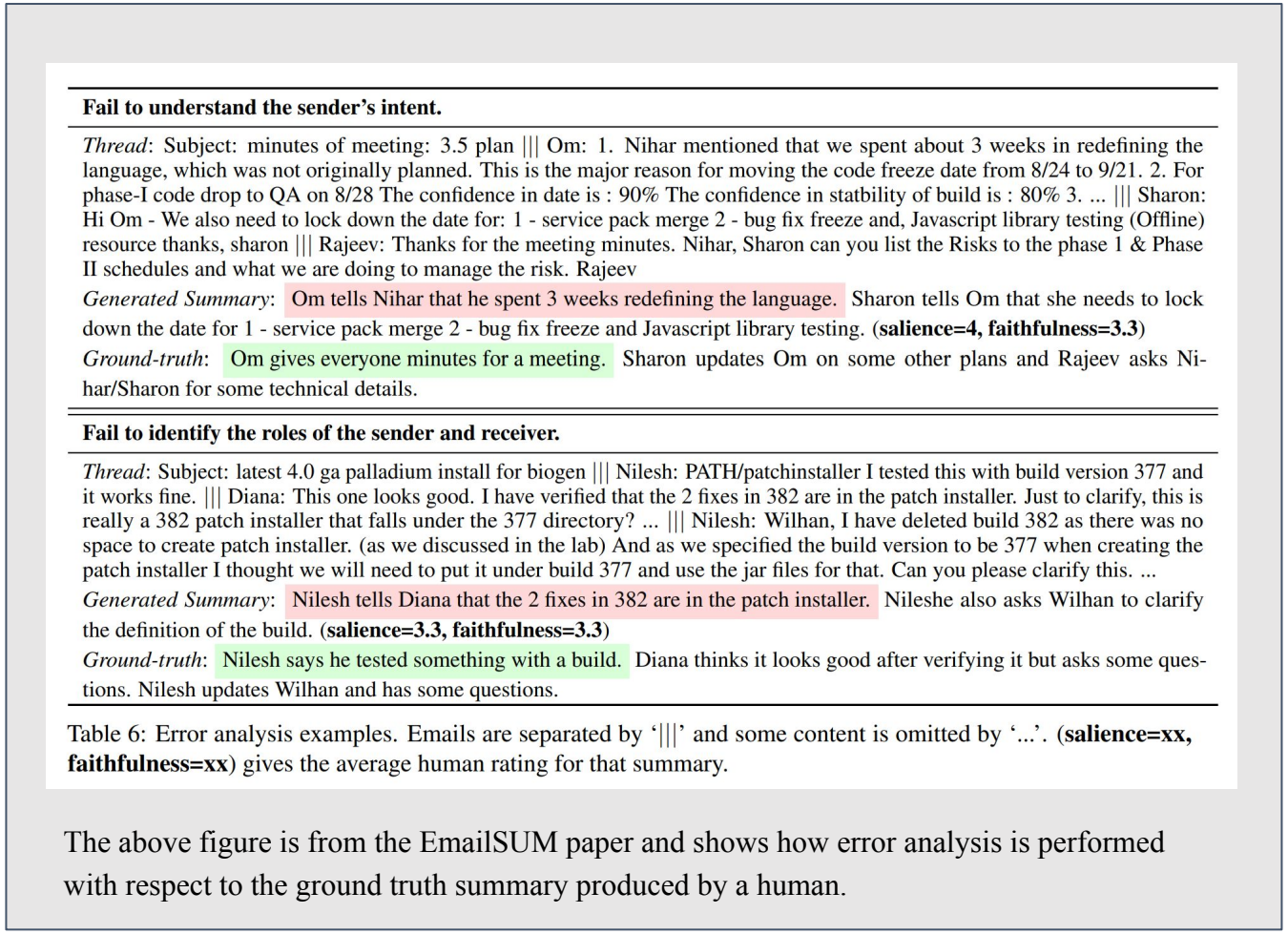
Emails are a huge part of everyday life:

- **86% of professionals** prefer email for business communication.
- On average, professionals receive at least **200 messages and send around 40 emails per day**.
- A small improvement in email efficiency, even in terms of minutes or hours per day, can result in enormous corporate-wide financial savings.

Why did you want to work on this topic?

Our team was interested in applying the tools we have learned in CIS 5300 to a practical everyday problem. We realized:

- A lot of work has been done on summarizing documents, specifically on summarization for the purpose of Text To Speech (TTS)
- There is much less work specific to email summarization.
- Nevertheless, generic methods can be successfully employed to summarize emails
- There is currently a lot of room for improvement, as evidenced by the EmailSUM example, and we thought that the tools we have learned in CIS 5300 could be employed for abstractive summarization



Fail to understand the sender's intent.

Thread: Subject: minutes of meeting: 3.5 plan ||| Om: 1. Nihar mentioned that we spent about 3 weeks in redefining the language, which was not originally planned. This is the major reason for moving the code freeze date from 8/24 to 9/21. 2. For phase-I code drop to QA on 8/28 The confidence in date is : 90% The confidence in statbility of build is : 80% 3. ... ||| Sharon: Hi Om - We also need to lock down the date for: 1 - service pack merge 2 - bug fix freeze and, Javascript library testing (Offline) resource thanks, sharon ||| Rajeev: Thanks for the meeting minutes. Nihar, Sharon can you list the Risks to the phase 1 & Phase II schedules and what we are doing to manage the risk. Rajeev

Generated Summary: Om tells Nihar that he spent 3 weeks redefining the language. Sharon tells Om that she needs to lock down the date for 1 - service pack merge 2 - bug fix freeze and Javascript library testing. (**salience=4, faithfulness=3.3**)

Ground-truth: Om gives everyone minutes for a meeting. Sharon updates Om on some other plans and Rajeev asks Nihar/Sharon for some technical details.

Fail to identify the roles of the sender and receiver.

Thread: Subject: latest 4.0 ga palladium install for biogen ||| Nilesh: PATH/patchinstaller I tested this with build version 377 and it works fine. ||| Diana: This one looks good. I have verified that the 2 fixes in 382 are in the patch installer. Just to clarify, this is really a 382 patch installer that falls under the 377 directory? ... ||| Nilesh: Wilhan, I have deleted build 382 as there was no space to create patch installer. (as we discussed in the lab) And as we specified the build version to be 377 when creating the patch installer I thought we will need to put it under build 377 and use the jar files for that. Can you please clarify this. ...

Generated Summary: Nilesh tells Diana that the 2 fixes in 382 are in the patch installer. Nileshe also asks Wilhan to clarify the definition of the build. (**salience=3.3, faithfulness=3.3**)

Ground-truth: Nilesh says he tested something with a build. Diana thinks it looks good after verifying it but asks some questions. Nilesh updates Wilhan and has some questions.

Table 6: Error analysis examples. Emails are separated by '|||' and some content is omitted by '...'. (**salience=xx, faithfulness=xx**) gives the average human rating for that summary.

The above figure is from the EmailSUM paper and shows how error analysis is performed with respect to the ground truth summary produced by a human.

How Does the Project Relate to the Class?

- We applied our knowledge of **Transformers** to our task.
- The project also uses the **HuggingFace** API and draws inspiration from Homework 4 and applies our knowledge of fine-tuning BERT to the T5 model for the baselines.
- We applied our knowledge of **LLMs and GPT** to the summarization task and better understand how it's performance differs from the strong baseline after fine-tuning.
- **Summarization** is a key task for language models among others that we discussed in class and we explore it further in this project.
- Getting all of our models to work requires a deeper understanding of how the raw data is initially represented and then transformed to be fed into the models, which we learned in class during our discussion of neural vector spaces and embeddings.

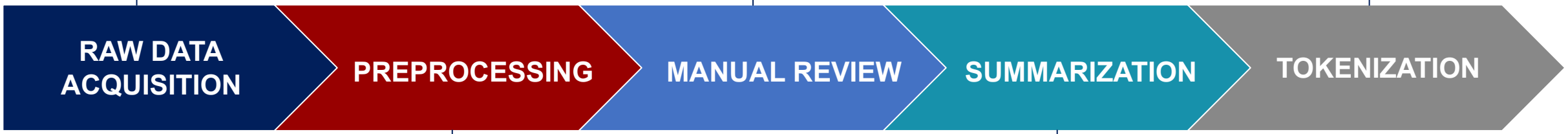
Data Acquisition and Cleaning

Acquisition, Preprocessing, Manual Review, Summarization, Tokenization

Our **raw dataset** was obtained from the **Linguistic Data Consortium** at Penn. It consists of emails taken from 239 accounts of an undisclosed information technology company.

The preprocessed and anonymized threads were **examined to ensure high quality** for the purposes of analysis, yielding 8,116 email threads in total.

The **labeled dataset** was imported into Google Colab in training, development, and test partitions, and properly tokenized for analysis.



To ensure privacy, **personal information** was removed from each email that was irrelevant for analysis. This includes URLs, addresses, phone numbers, and more.

Long (>100 words) and short (<30 words) summaries from **Amazon Mechanical Turk** were concatenated with the threads data, leaving 2,549 entities in total.

Model Design

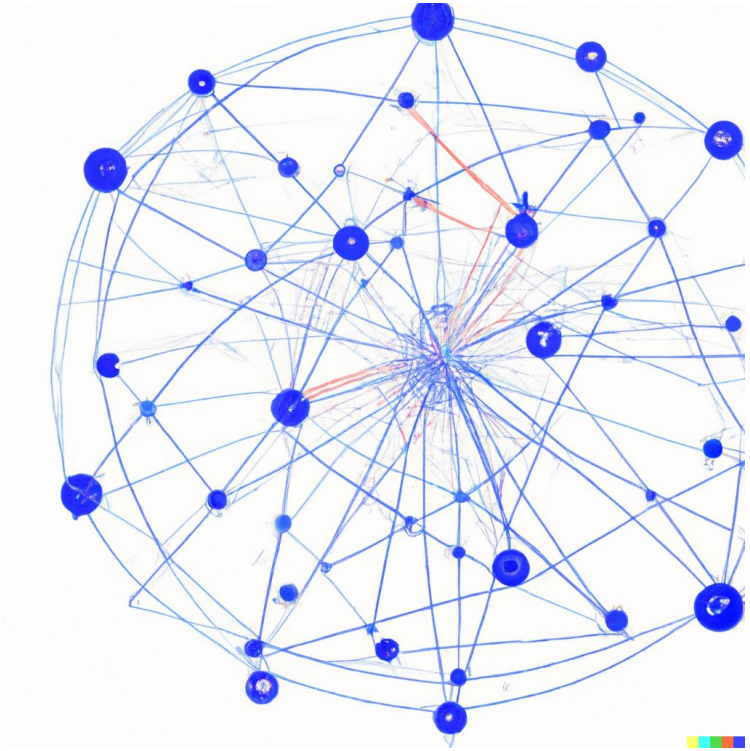
As a weak baseline, we implemented an untuned T5 for conditional generation from Hugging Face.

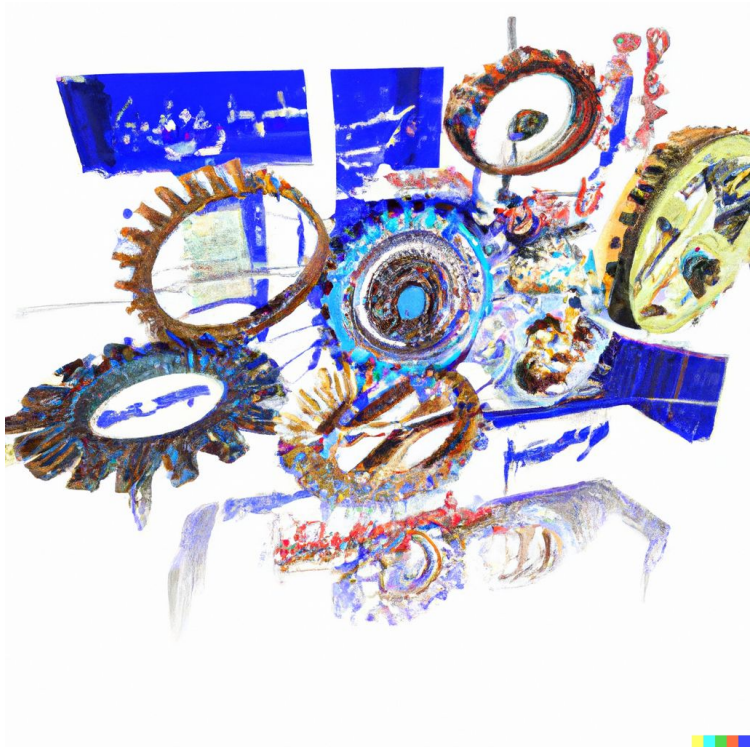
STRONG BASELINE

Our primary baseline is a T5 model for conditional generation from Hugging Face fine-tuned on the development set.

EXTENSION 1

Our first extension model is GPT 3.5 Turbo from OpenAI similarly fine-tuned on the development set.



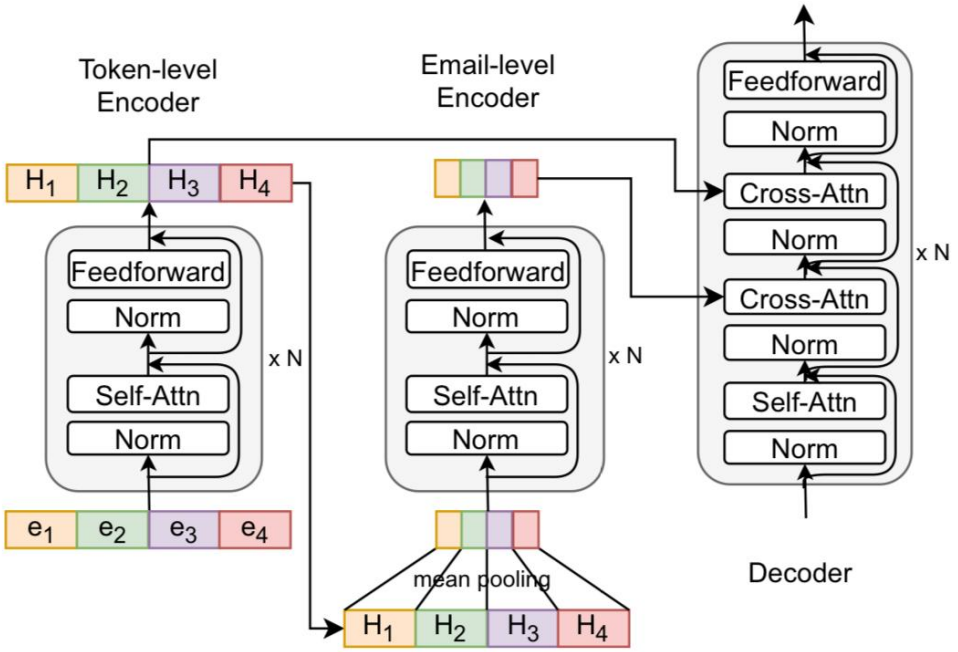


https://arxiv.org/abs/1909.01610

Model Design

Our second extension is T5 fine-tuned using SummaQA metrics

- For extension 2, we applied SummaQA from "Answers Unite! Unsupervised Metrics for Reinforced Summarization Models" **(Scialom et al., 2019)**
- SummaQA applies a BERT-based question-answering model to answer cloze-style questions using generated summaries. Questions are generated by masking named entities in source documents associated with evaluated summaries. The metric reports both the F1 overlap score and QA-model confidence. We can either compare the summary to the original thread (unsupervised) or we can compare the summary against the human-generated summaries (supervised). We applied SummaQA in two ways.
- First, we evaluated the baseline model output and the output from Extension 1, verifying that although the ROUGE scores were similar, the output from GPT contained more of the key pieces of information than the summaries from T5 fine-tuned to improve the ROUGE score.
- Second, we fine-tuned T5 using the SummaQA score as an objective. This increased the SummaQA score further, although it is unclear whether this increase represented a real improvement in quality.



Model Evaluation

Evaluation Metrics:

- **BERTScore**
  - "Bidirectional Encoder Representations from Transformers Score"
  - Word embeddings of the system and reference summaries are compared with cosine similarity
  - Each token in the reference is matched to the most similar token in the summary to compute F1-Score

$$R_{\text{BERT}} = \frac{1}{|x|}\sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \quad , \quad P_{\text{BERT}} = \frac{1}{|\hat{x}|}\sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \quad , \quad F_{\text{BERT}} = 2\frac{P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}$$

$$\hat{R}_{\text{BERT}} = \frac{R_{\text{BERT}} - b}{1 - b}$$

- **ROUGE**
  - "Recall-Oriented Understudy for Gisting Evaluation"
  - Overlap of n-grams between the system and reference summaries, for n = 1 and n = 2 in this study
  - Additionally, the longest common subsequence is found between summaries

$$\text{ROUGE-1}_{\text{recall}} = \frac{\text{unigram cand.} \cap \text{unigram ref.}}{|\text{unigram ref.}|}$$

$$\text{ROUGE-1}_{\text{precision}} = \frac{\text{unigram cand.} \cap \text{unigram ref.}}{|\text{unigram cand.}|}$$

$$\text{ROUGE-1}_{\text{F1}} = 2 * \frac{\text{recall} * \text{precision}}{\text{recall} + \text{precision}}$$

$$\text{ROUGE-L}_{\text{F1}} = 2 * \frac{\text{recall} * \text{precision}}{\text{recall} + \text{precision}}$$

- **Our weak baseline was evaluated with these metrics as follows:**

$R_{BERT} = 0.852$, $P_{BERT} = 0.838$, $F1_{BERT} = 0.844$

$F1_{R1} = 0.177$, $F1_{R2} = 0.0347$, $F1_{RL} = 0.138$

Model Comparison: Evaluation Metrics

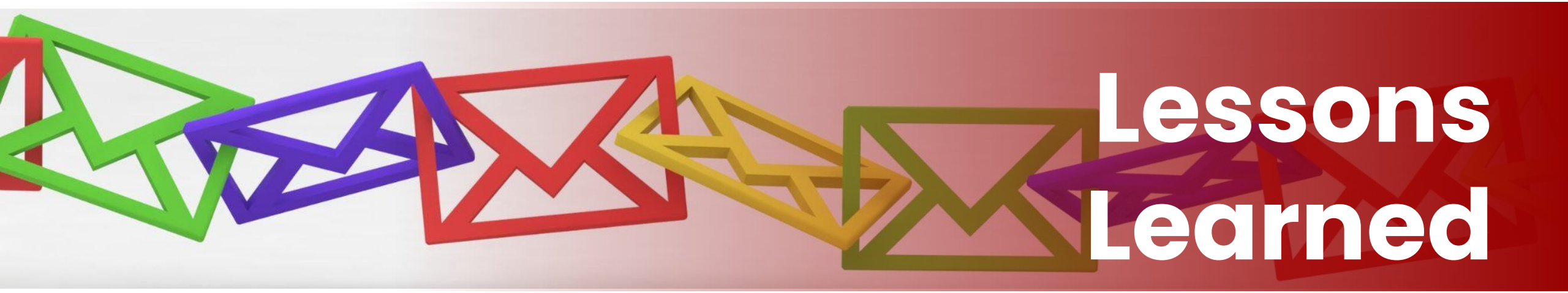
Metric	Weak T5 Baseline	Strong T5 Baseline	GPT3.5	SummaQA-finetuned T5
Rouge1	0.1886	0.3041	**0.3376**	0.3336
Rouge2	0.04235	0.09586	0.07542	**0.1060**
RougeL	0.1416	0.2391	0.2406	**0.2755**
RougeLSum	0.1440	0.2389	0.2418	**0.2752**
BertScore Precision	0.8431	0.8855	0.8804	**0.8856**
BertScore Recall	0.8434	0.8715	0.8813	**0.8871**
BertScore F1	0.8428	0.8783	0.8808	**0.8862**

Model Comparison: Example

The following is an example of the model output (move to scoring.md):

Original email thread: Subject: lunch this week———Susan: All, Regarding our lunch this week to celebrate the one year anniversaries for Michelle David, and Mark's birthday, I have a request to make it Wednesday instead of Tuesday. Does anyone have an objection to this? Susan———David: I have another lunch engagement Wed, but I will skip it if everyone else wants to move our lunch. David———Tamra: Susan, Wednesday works out better for me as well. I have a doctor's appointment tomorrow during lunch. Tamra

- **Human Summary**: Susan emails everyone about an anniversary and offers to change the date. David says he is busy but is willing to go with the majority. Tamra agrees with Susan's date.
- **T5 Weak Baseline:** lunch this week to celebrate the one year anniversaries for Michelle & David, and Mark's birthday . does anyone have an objection to this? *extractive, with verbatim copies of two phrases from the input*
- **T5 Strong Baseline**: Susan has a request to make lunch this week to celebrate the one year anniversaries for Michelle David, and Mark's birthday. David has another lunch engagement wed, but I will skip it if everyone else wants *abstractive,but did not consistently switch to third person, and left out the conclusion*
- **Extension 1: Fine-tuned GPT** : Susan asks coworkers if they can move a lunch to Wednesday. David and Tamra say they can and list their reasons why. *close to the human summary, though more focus on what than why*
- **Extension 2: T5 fine-tuned to SummaQA:** Susan asks for a Wednesday lunch to celebrate the one year anniversary of Michelle & David and Mark's birthday. David says he will skip it. Tamra *provides an efficient abstractive summary of first email, keeping extra information, but provides an incorrect summary of responses*



Lessons Learned

Baselines:

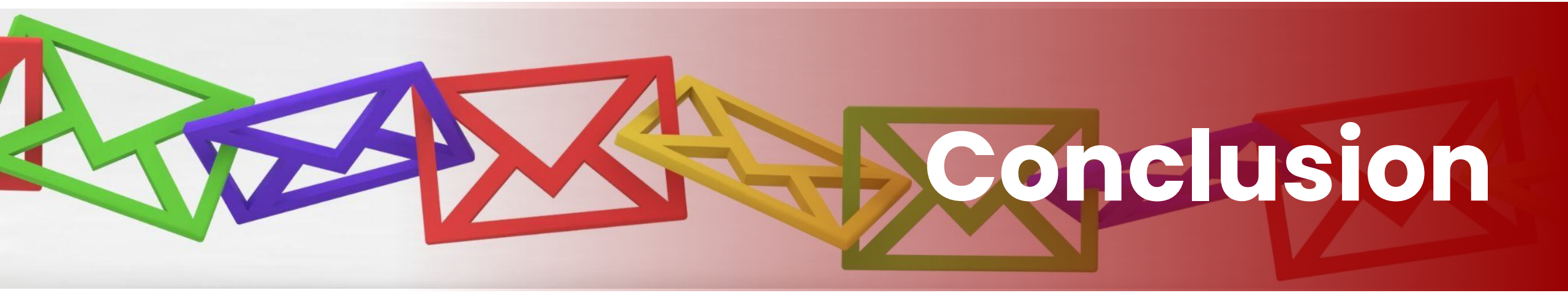
- After implementing the strong and weak baselines, we observed that the strong baseline consistently outperformed the weak baseline.
- This made it apparent that using a general pre trained model and then fine-tuning it with data specific to our email summarization task significantly improved the quality of the results.

Extension 1:

- The summarizations produced by fine tuning GPT 3.5 were much more detailed and captured the main points of the thread; however, the BERT & ROGUE scores were similar to that of the fine-tuned T5 baseline.
- This highlights that 1. GPT is more powerful than T5 and the number of training parameters does make a difference in the final results, and 2. BERT & ROGUE are not the best metrics for evaluating the quality of a summarization relative to a ground truth output.

Extension 2:

- Fine-tuning T5 to SummaQA scores better on most metrics than other models, but some of the summaries that it produces seem to be unintuitive from a human perspective



Conclusion

Main Takeaways

- Models generally did a great job of generating an abstractive summary which contains the gist of each conversation, and fine-tuned models consistently outperformed the weak baseline.
- Models struggled with correctly identifying the senders' intent and the individual sender/recipient's roles.
- Quantitative evaluation metrics might not concur with human readers' subjective judgment of summary quality.

Potential Opportunities for Future Work/Extensions

- Obtaining a larger dataset encompassing 20,000+ training examples through Amazon Mechanical Turk, etc.
- Experimenting with more advanced LLMs including GPT 4.0 and LLaMA2.
- Looking into additional evaluation criteria, such as BARTScore and QuestEval[1]

1- Gao et al. find that few evaluation metrics are excellent in all dimensions and that **BARTScore** (see Yuan et al., 2021) and QA-based metrics, such as **SummaQA** (see Scialom et al., 2019) and **QuestEval** (see Scialom et al., 2021) are comparatively outstanding and worth exploring.

Thank you