Cellebrite

## Background

At Cellebrite, we build tools to assist with digital forensic investigations. Our products provide the means to easily structure, view and analyze the unstructured data that is extracted from mobile devices involved in a criminal case.

On our AI team, we develop ML based analytics features that create insights from the raw data in order to minimize the investigator's time to encounter crucial evidence. Chat messages are often a primary source of relevant information to an investigation. The bulk of chats contain irrelevant or mundane information which can slow down the investigative process. Summarizing chats can help to speed up the investigation process.

## Task overview

A notorious criminal known as "Chunky Data Shuffler" has sneaked into our office and got a hold on our chat summarization dataset. Under the cover of the darkness he split the summaries into chunks and shuffled them before dissapearing with a gnarly laugh.

## Goal

Implement an algorithm that:

1. Attributes each summary piece to its corresponding conversation.
2. For each conversation, the attributed summary pieces should be ordered by the dialog's chronological order.

## Data

- dialogues.csv - contains 1400 chat conversations missing summaries
- summary_pieces.csv - contains 3996 shuffled pieces of summaries for chats in dialogues.csv
- reference_dialogues.csv, reference_summaries.csv - contains 610 conversations, their summary chunks and positional indices (with no relation to the data in other CSVs, you could use this data for tuning your algorithm if required)

## Deliverables

- Code / notebooks with algorithm implementation.
- A short write-up on the solution methodology
- A CSV with three columns:
    - ○ "dialog_id" - id of the dialogue
    - ○ "summary_piece" - a summary chunk associated with the dialogue
    - ○ "position_index" – the positional index of the summary chunk relative to other chunks which are attributed to the same dialogue
        - ▪ Example:

| dialog_id | summary_piece | position_index |
|---|---|---|
| 13862975 | Owen is looking for volunteers for an unpaid project in Toronto. | 0 |
| 13862975 | All associated costs are covered. | 1 |
| 13681042 | Tom wants to borrow Alan's car but Alan needs to ask Sally first. | 0 |

- Python method (somewhere in your code) that accepts 3 arguments: paths to the input files (dialogues.csv and summary_pieces.csv) and path to the output results CSV file. We will run this method and evaluate your solution against a test set based on the results you wrote to the given output CSV file. Use the following method signature:

```
def main(dialogues_path: str, summary_pieces_path: str, output_path: str) -> None:
    ...
```

Please reach out to us with any questions. Good luck!