TAIC Legacy report Semantic Analysis

Project proposal for 159333 Massey University

James Thompson (21011195) - 1jamesthompson1@gmail.com

31/07/2023

Info

Author: James Thompson
Supervisor: Dr Chris Scogings
Stakeholder: Rob Thompson (Marine Accident Investigator - TAIC)

Project outline

Intro

This project is about using Large Language Models (LLMs) and Natural Language Processing (NLP) techniques to study marine accident reports. This will be done by taking report found on the public Transport Accident Investigation Commission (TAIC) website Investigations | TAIC and extracting the information from them. From there statistical techniques can be employed to study the relationship between causes and outcomes. These reports are the end product of investigations funded by the government into why and how these accidents occurred. To narrow the scope we will only be looking at marine reports from boat crashes.

Note that even though this is a project with benefit for TAIC, as of right now it is not being completed in any official capacity.

Goals / Motivation

The goal is to create a proof of concept that can be used to jump start further work in this area and create a more general engines. This will give the ability for the investigators to understand the long term trends in the causes of the accidents.

It was inspired by Rob's experience using ChatGPT and him wanting to apply it to his work. In particular doing the long and tedious job of reading all the reports they have published.

Method

There are three distinct tools that will be used for this project. Firstly Python will be used for the first step of creating the dataset. OpenAI api will be called by the Python scripts to give me access to prebuilt LLMs to do the reading and interpreting of reports. Lastly R/Rmd will be used to do the final EDA and report creation, not only is it one of the best tools for reproducible reports I am also quite familiar with it.

When time permits trials will be made with fine-tuning / training our own model on all the reports.

The intended structure for the first milestone will look something like this: 1. Python - download and extract text from all of the PDFs 2. Python/openAI - Extract the content section and find which pages are important pages to be “read” 3. openAI - Read the important section of the reports. Return information on the causes of the accident with weighting as well as a generative and extractive summary. 4. Python - Collate all report summaries into a csv file. 5. R/Rmd - Read csv and do some EDA and create a simple report I expect that the structure may change a bit later in development but follow a similar pattern.

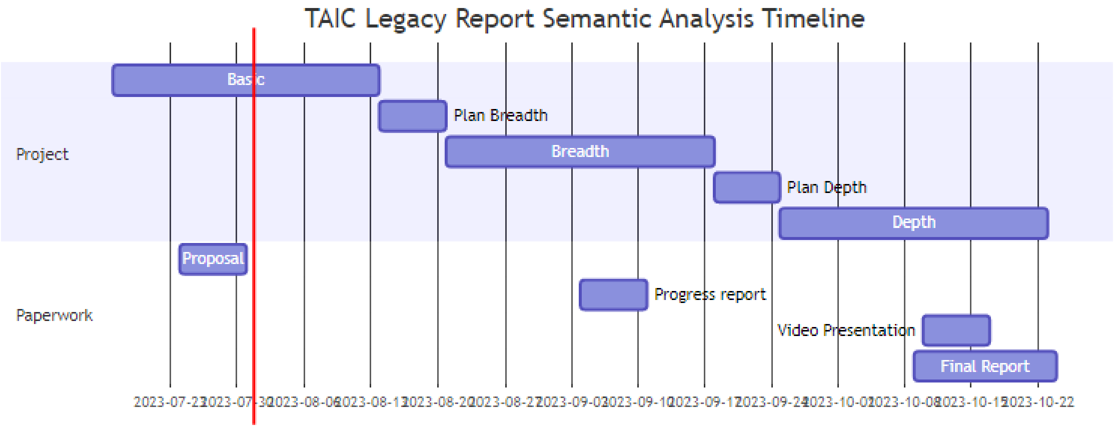
Previous work

A project specifically using an LLM to read and interpret transport accident investigation reports is a new idea that I cant find any evidence for being completed or attempted elsewhere. However there have been some exploration in the ability for LLMs to interpret reports. For example Evaluating Large Language Models for Radiology Natural Language Processing Liu, Zhengliang, et al found that were capable of producing useful interpretations of situations. Other studies such as Qin, C., Zhang, A., Zhang, Z., Chen, J., Yasunaga, M., & Yang, D. (2023). Is ChatGPT a general-purpose natural language processing task solver? demonstrate that using a GPT model to replace human tasks is a plausible proposition.

Timeline

This project has 13 weeks of work to be completed. I plan to work on this in three distinct sprints.

Due to this report being in the third week work has already begun on the basic milestone.



I expect that at the end of each of the 4 week sprints I have a deliverable with a report that can be checked with the stakeholder for their input. Each of these sprints will have a 1 week planning period which will help actually determine what will be completed in the next sprint.

Milestones

Here I will layout the ideas that will be explored in each of the milestones. However as this will be a exploration in what can be done the specific direction will be determined closer to each milestone.

Basic

This will be what is laid out above in the strucutre. It will consist of a basic engine that extracts the causes and a summary of each of the reports. Note that the causes will be 2-6 themes that will be pre determined using subject expertise. It will be collated into one big csv file and then traditional statistics technique can be run on it.

Breadth

This will extend the amount of varaibles mined from each report. This could be the number of fatalities or owner of the ship.

Work will also be looked at in trying to get the causes of the accidents to be discovered by the model itself. This would mean that the engine would discover its own causes for the accident.

Depth

Now that a lot of information will be collected from each of the reports it will be important to start looking at going deeper into the causes of the accidents. Firstly this can be done by splitting the themes into subthemes. Secondly time can be spent looking at training / fine-tuning a model on all the reports. This would let you “speak” directly with someone who has read all the reports.

Due to this being the last milestone I am leaving the specification open to be shaped by the next few weeks of work.

Deliverables

The final deliverable product will be two fold.

Example report

There will be an example report which is used to show the insights that have been garnered from the reports up until this point. It would be making statements like “maintenance is the most prevalent cause being in 80% of accidents” or “fatigue and intoxication are most commonly observed together”. This report will be no longer then a two sided piece of paper.

In particular it will help expand on a report that was manually created by the chief investigator at TAIC by reading a handful of reports. This will help demonstrate the ability of computers and LLMs to assist the investigators in drawing insight from old reports.

Single use engine

This work will all be done as programmatically as possible. Meaning that at the end there should be a python project that on the press of a button will download all the reports from the website, extracts the texts from reports, summarises them and lastly packages it up in csv files. It is these csv files that can be picked up and used to make the example report.

Even though the engine will only be setup to be used on these specific reports the ideas and methods developed in the extracting and summarising stages can be the basis of building a more general engine that could be used across different accident investigation reports.