The *Blair* prototype for social media data mining was developed to understand human behaviour during wildfires. The name, *Blair*, is a short name for **B**ehaviour **L**abelling **AI** for **R**esearch on social media data about wildfire evacuation. An active learning method was implemented in *Blair* to enable the AI in Blair to learn the labelling standard from users in small samples of data. *Blair* needs Dot Net (.NET) Desktop 6 runtime to run[1].

1. Architecture and Workflow

The architecture of Blair is shown in Figure 1.

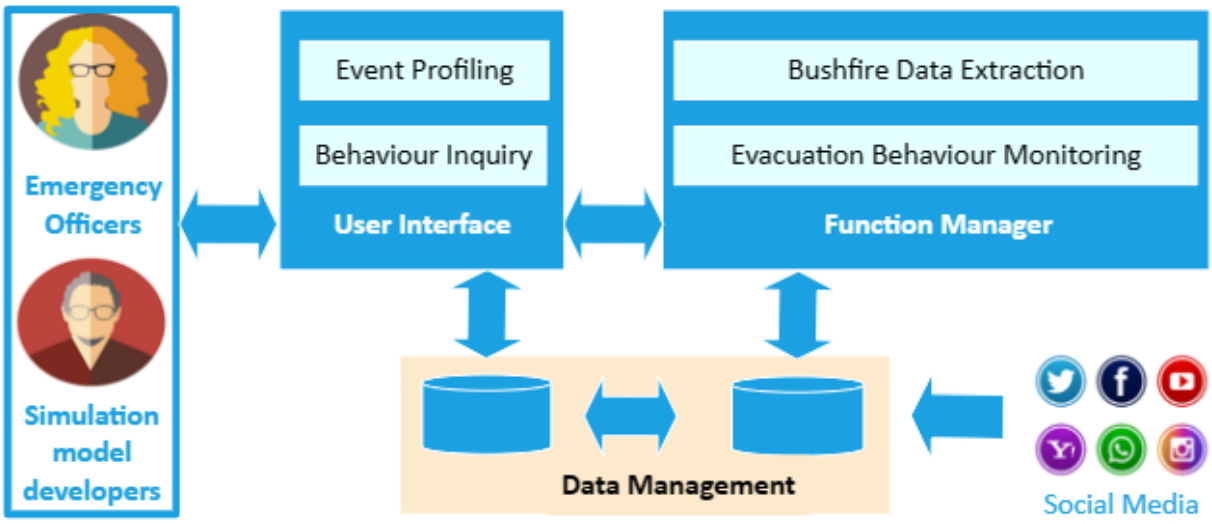


*Figure 1. Architecture of Blair (prototype for **B**ehaviour **L**abelling **AI** for **R**esearch on social media data about wildfire evacuation).*

Blair is designed to extract and mine social media data to study how people evacuate and move during wildfire events. Its design includes:

- Wildfire data extraction: It finds and categorizes data related to wildfires based on various factors, such as evacuation decisions, destination types, and locations.
- Evacuation behaviour monitoring: It provides statistics on evacuation behaviours from the extracted data for monitoring purpose. However, the monitoring is performed on the collected datasets rather than monitoring social platforms in real-time.
- Graphical user interface (GUI): It supports these functions:
    - Event profiling: It summarizes basic information for each dataset.
    - Behaviour inquiry: It allows users to search for behaviours by various combination of conditions, such as conditions on date range, location, evacuation decision, and destination type.

[1] If the user of *Blair* does not Dot Net installed, it can find it on https://dotnet.microsoft.com/en-us/download/dotnet/6.0.

- Data management: The input data of each dataset is stored in a CSV file. The CSV file contains a table that includes these columns: text, post id, user id, time, and geotag. The output data of each dataset is stored in an XLSX file. The output data contains the post id and text of each relevant post as well as its categorisation labels.

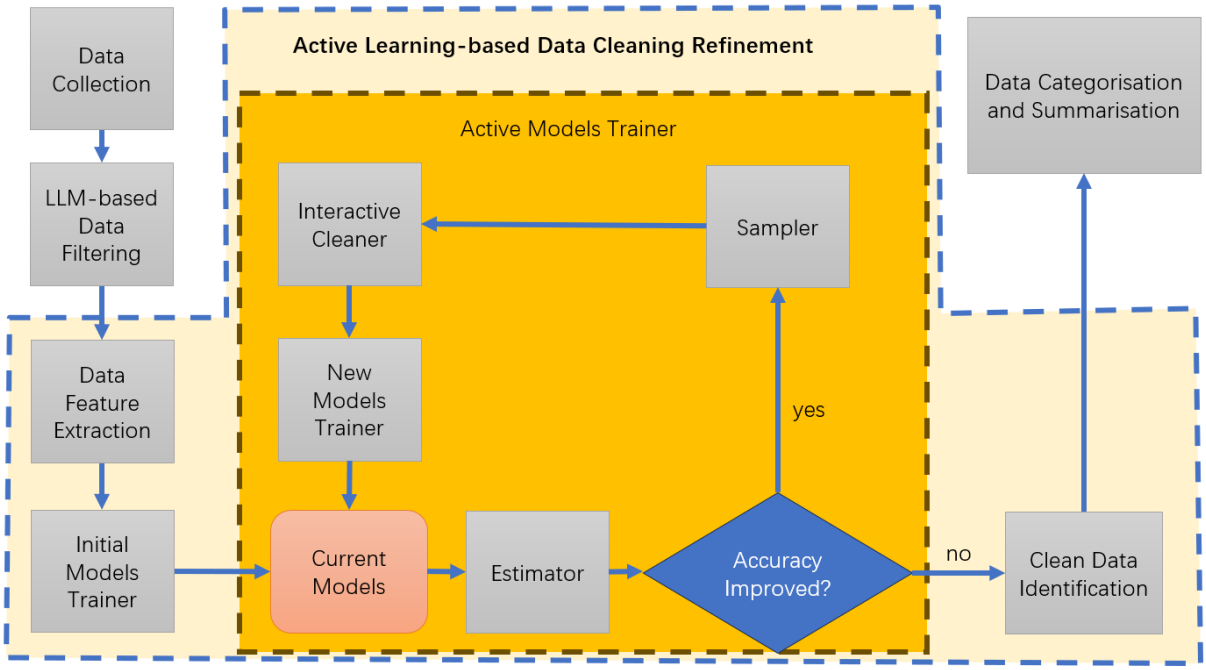


Figure 2. Workflow of active learning-based human behaviour analysis framework for wildfires.

The workflow of Blair is shown in Figure 2, which has the following main steps:

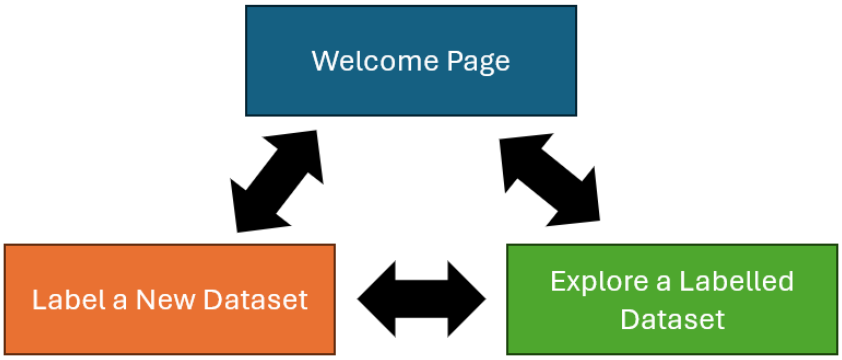
- In **data collection**, we collect data from social media. The input data of Blair is required to be a table stored in a CSV file with essential columns: text, post ID, user ID, time, and geo-tag. The first line of the CSV file must be a header specifying these columns. The other lines are the data, each line a social media post.
- In **LLM-based data filtering**, we use a Large Language Model (LLM) to remove most of the irrelevant posts, thus obtaining the initial results of data cleaning.
- In **active learning-based data cleaning refinement**, the accuracy of the data cleaning results is improved using active learning.
- In **data categorisation and summarisation**, the clean data are categorised and summarised according to the output requirement of *Blair*.

2. Graphic User Interface overview

The Graphic User Interface (GUI) of *Blair* is outlined in Figure 3. The GUI has three pages, as detailed below. The user of *Blair* can navigate to the other two pages from each page.

- **Welcome page**: When a user initiates *Blair*, it presents a welcome page. The welcome page introduces what *Blair* does. It also asks the user to set an API key of Google Gemini Pro (the large language model used in the prototype).

- **Data labelling**: The data labelling for data extraction is provided in the "label a new dataset" page. The labelling is driven by active learning, so the user only needs to label a small amount of data. The large amount of data in the rest of the dataset are labelled automatically by *Blair*.
- **Data exploration**: The last page of *Blair* supports exploring the data by presenting a summary of human behaviour in the data and offering an interactive query on any such behaviour.



- *Figure 3. Navigation among three pages in the GUI of Blair.*

3. Demonstration Scenarios

The use of the GUI of *Blair* is here shown using the 2019 Tick Fire dataset as example. The examples can be used to guide users into the analysis of any dataset.

Phase 1: Start *Blair*

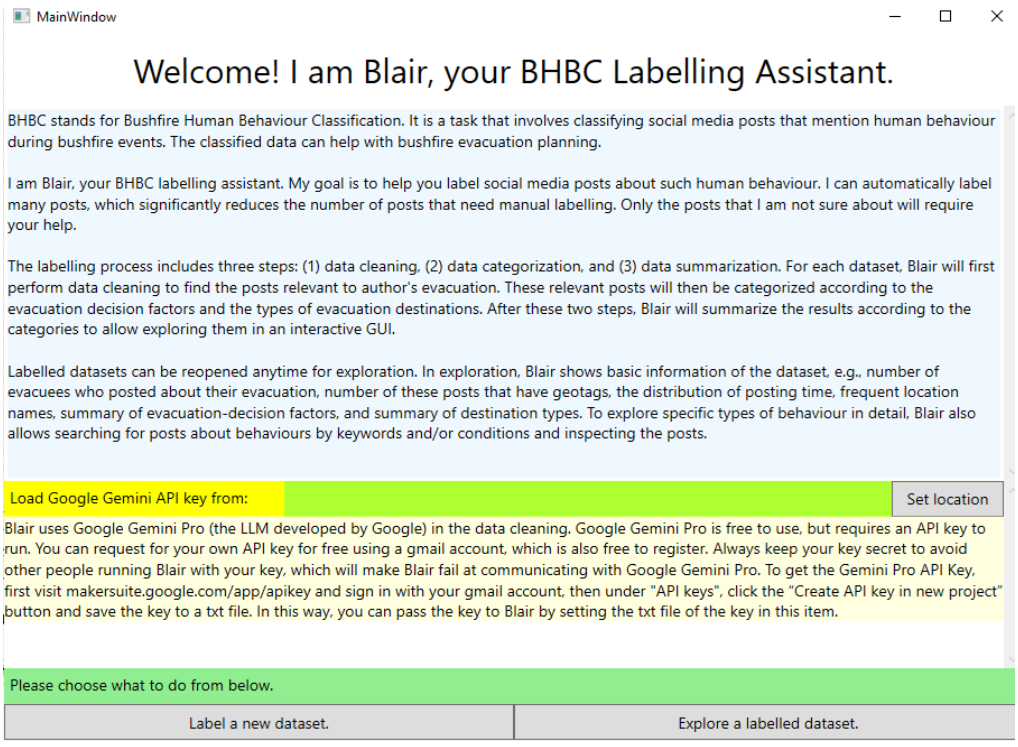


Figure 4 Screenshot of the welcome page of Blair.

When the user starts *Blair*, a welcome page will appear (see Figure 4). The welcome page introduces *Blair* to the user, then asks the user to provide a Google Gemini API Key from a text file. The light-yellow part of the interface explains how to get a free API key for this. At the bottom of the welcome page, there are two buttons leading to the other two pages respectively.

Phase 2: Label a new dataset

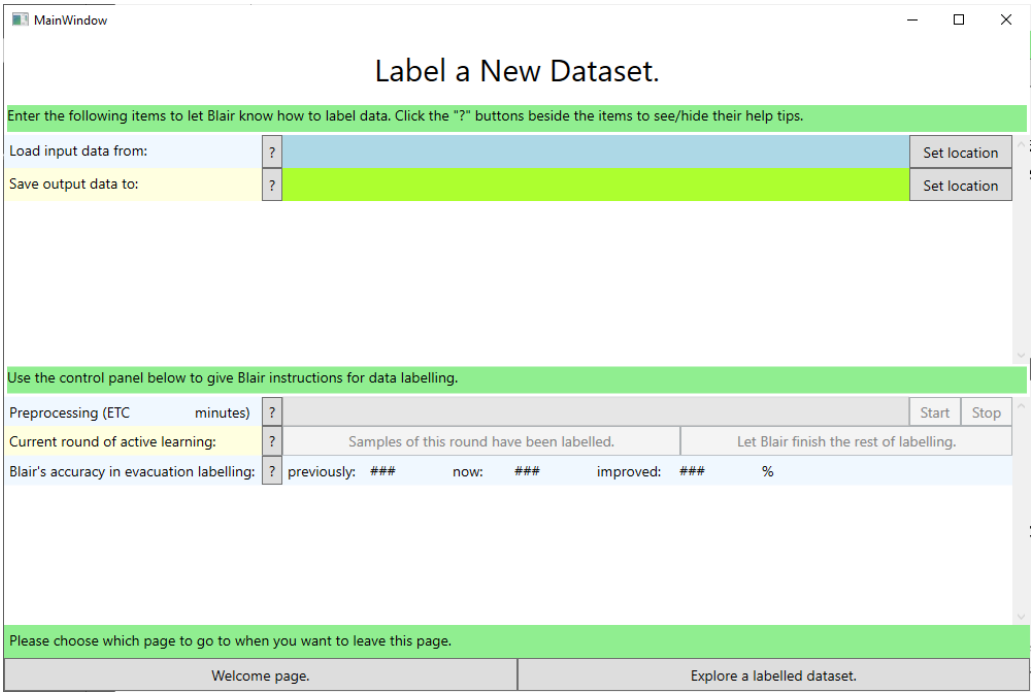


Figure 5. Screenshot of the initial "Label a new dataset" page.

The workflow in this scenario, demonstrated in Figures 5 to 10, is the same as the workflow in Figure 2. It has four steps, where the operations of each step will be detailed after the overview of these steps:

(i) **Set input data** (Figures 5 and 6): The user sets input data on the GUI page. The collection of the input data must follow the earlier explained methodology before setting the data.

(ii) **Start LLM-based data cleaning** (Figure 7): The algorithm performs the data cleaning when prompted by the user clicking a button.

(iii) **Optimise data cleaning based on active learning** (Figures 8 and 9): The operations are consistent with the loop in subsection 4.3.3. Blair works with the user interactively to optimise the data cleaning results for several rounds.

(iv) **Get data-cleaning output** (Figure 10): The operations for the identification of final data-cleaning results are performed. After clicking a button to obtain the output results, the user needs to review a small number of results marked with "?".

The operations of these steps are detailed as follows:

Set input data: As the user enter the "label a new dataset" page, they will see the GUI as shown in Figure 5.

First-time users might not know about what is input data and what is output data. They are probably unfamiliar with other elements on this page as well. The "?" buttons are prepared for them. Clicking a "?" button will show the help tip of the corresponding part. Figure 6 shows what the page will become if we click some of the "?" buttons. Please note that each help tip can be hidden using the same "?" button that reveals it.

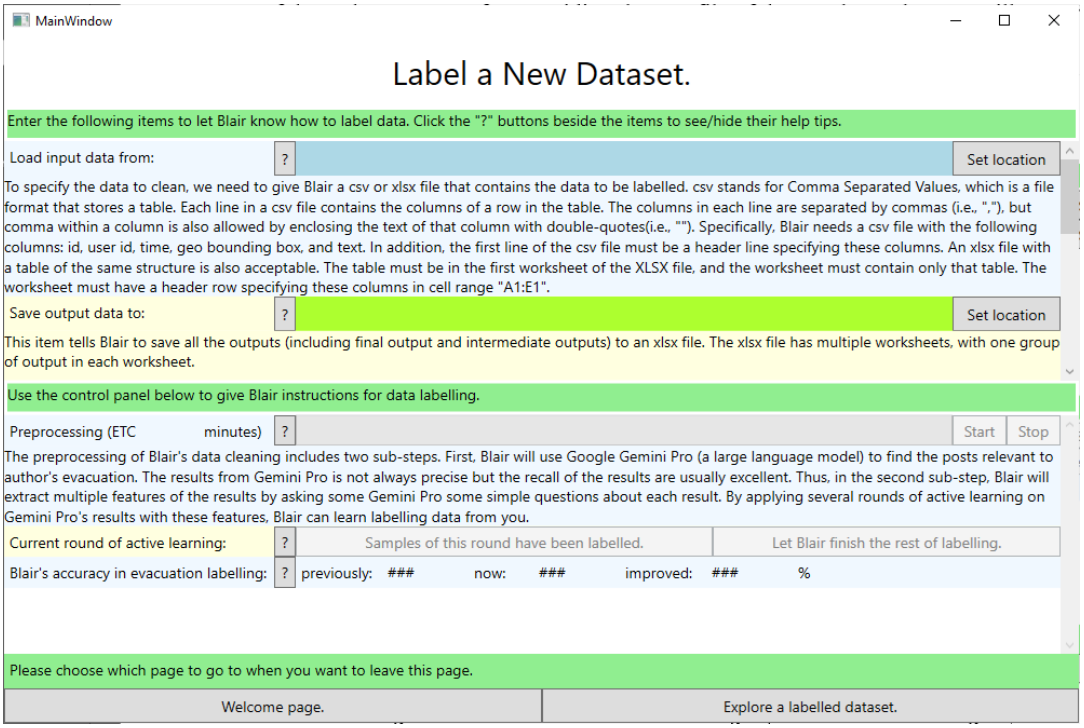


Figure 6. The "Label a New Dataset" page with some of the help tips shown.

Start data cleaning: Once the user sets the input and output files of labelling and clicks the "start" button in the control panel, *Blair* will preprocess the input data and obtain the first-round sample of active learning, as shown in Figure 7. The computational time for preprocessing will depend on the size of input dataset. For example, the Kincade Fire dataset that was used in this project contained 11934 posts. The use of a windows-machine with Intel Core i7-9750H CPU @2.6GHz and 32GB memory resulted in 6 hours for preprocessing during our tests. The preprocessing time is (approximately) linearly proportional to the number of posts in the dataset. The progress bar of preprocessing tells us how much work Blair has finished in this preprocessing task. The ETC (*Estimated time of completion*) in minutes on the left of the progress bar provides an estimate on how much time it needs to finish the rest of preprocessing. The "stop" button on the right side can stop the preprocessing temporarily if, for example, the user wants to close the computer for a while.

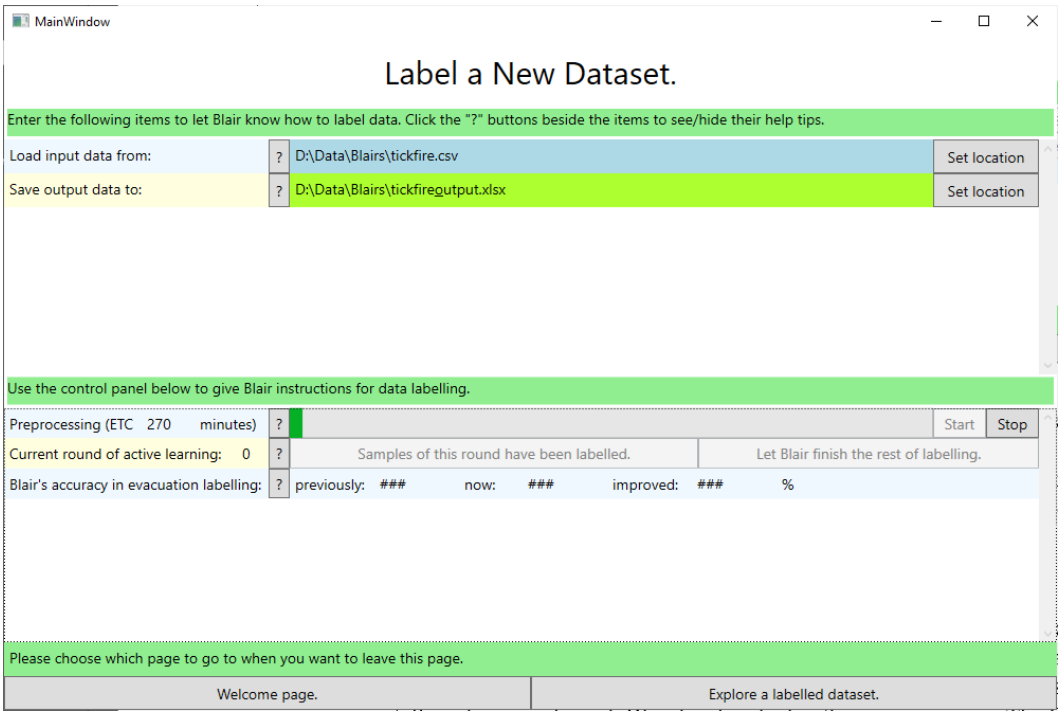


Figure 7. The "label a new dataset" page with the preprocessing task started.

Optimise data cleaning based on active learning: After the preprocessing is finished, the user will find the "start", "stop" buttons disabled and the two active learning buttons below the preprocessing progress bar enabled. As shown in Figure 8, disabled buttons are greyed while enabled buttons are presented in their normal colour. The two active learning buttons, i.e., "samples of this round have been labelled" and "let Blair finish the rest of labelling", are for different purposes. The "samples of this round have been labelled" button is used to tell Blair that user has labelled the samples of the current round in a worksheet named "round x samples" in the output file, where x is the index or number of the current round shown on the control panel of this page. Figure 9 shows the "round 0 samples" worksheet of the output file in Microsoft Excel. Column C are the labels the user manually puts into the worksheet. The user of Blair needs to determine the labels according to the texts in Column B. The texts in column B are the texts of the sampled posts. Round 0 will normally have 100 samples while each of the later rounds has no more than 50 samples. After saving the labels of these samples, the user can click the button "samples of this round have been labelled". Blair will start the next round of active learning and give the user a new round of samples to label, so that it can learn from the user how to label the rest of the data. As shown in Figure 10, the change in the accuracy is shown in the control panel to determine when to end the active learning. If the percentage of improvement is too small (e.g., less than 1%) or negative, the user can click "let Blair finish the rest of labelling" to get a final output. Figure 10 shows that the improvement of accuracy is "-1.1" in round 4 of active learning, so the user can click "let Blair finish the rest of labelling" in this round. Note that there is no need to label round 4 samples before clicking that.

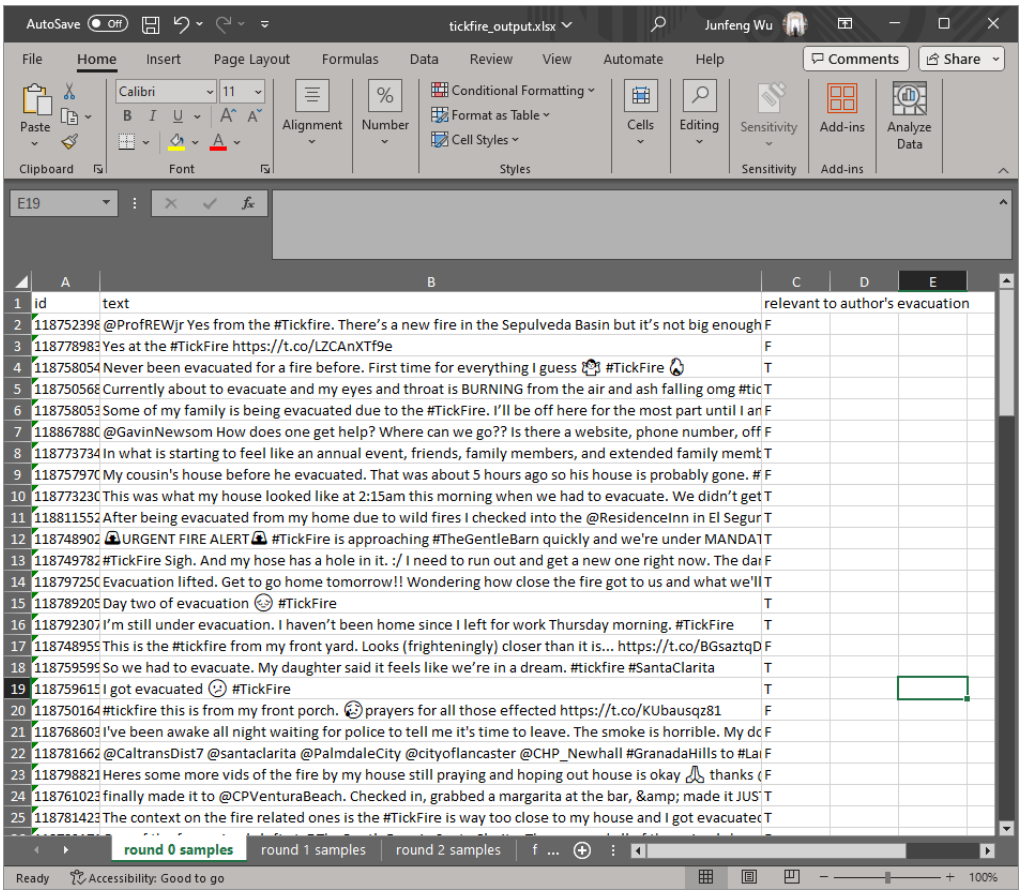


Figure 8. Labelled round 0 samples in the output file.

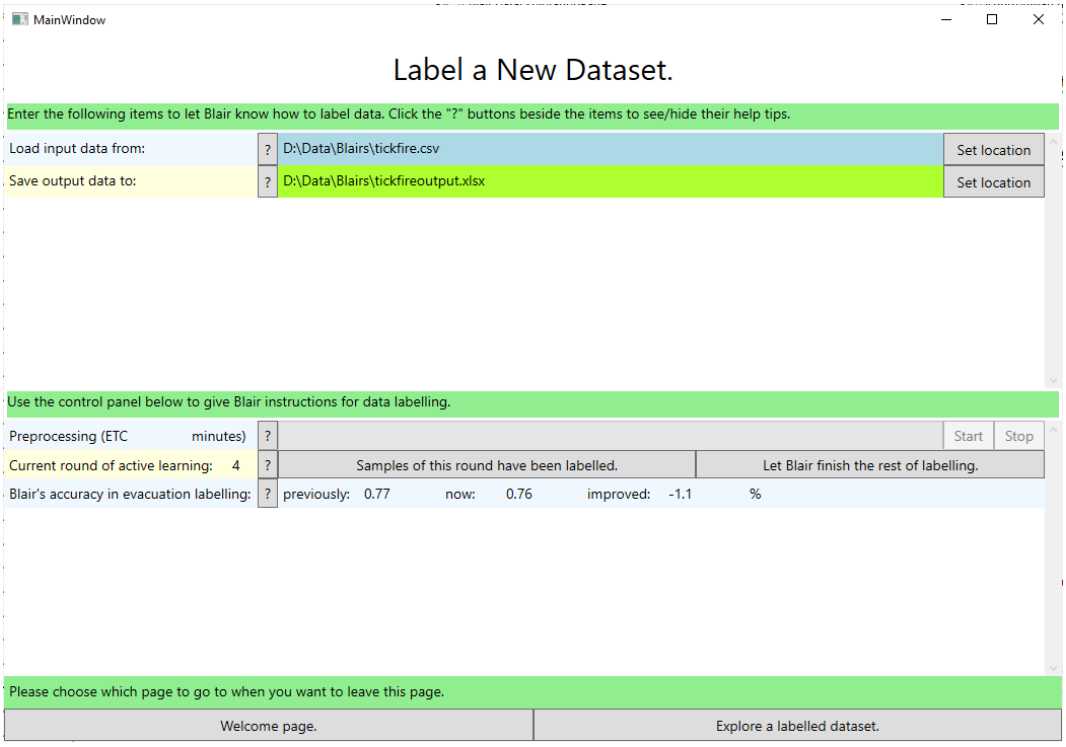


Figure 9. A screenshot showing that the accuracy improvement is too small in this round (less than 1%). Therefore, the user can click the button: "let Blair finish the rest of labelling".

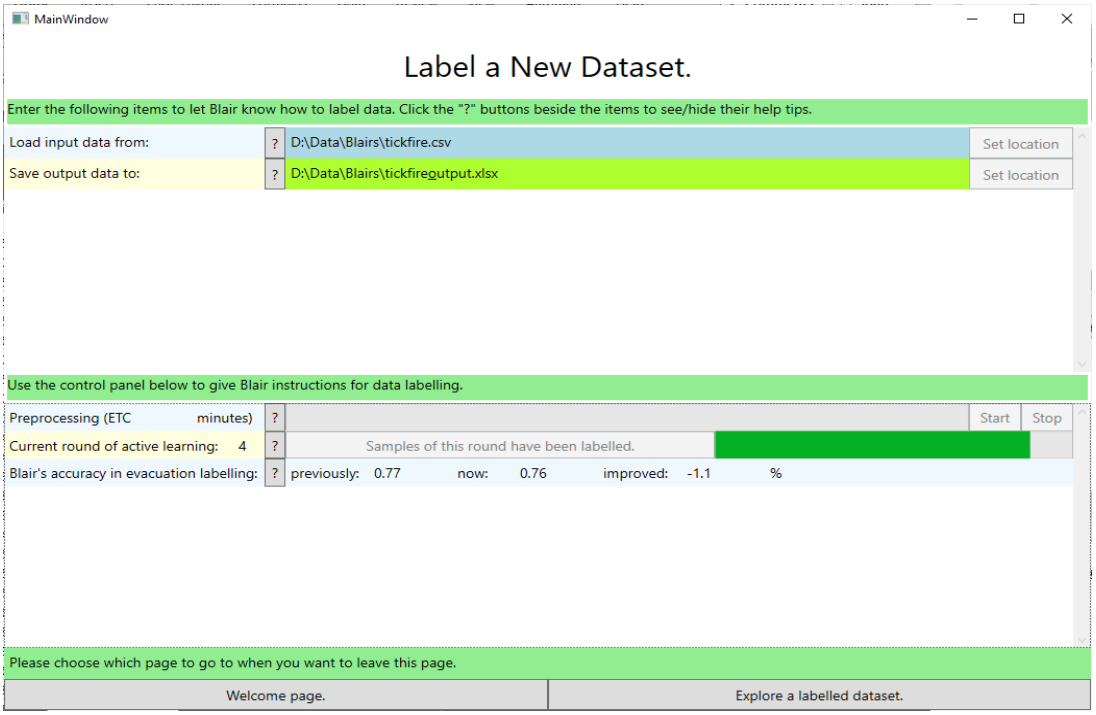


Figure 10. A progress bar appears on button "let Blair finish the rest of labelling" to tell you the progress of final output generation.

Get data-cleaning output: Clicking the "let *Blair* finish the rest of labelling" button allows the user to obtain the output. After clicking the button, Blair will take about 20 minutes to generate the final output. A progress bar on that button will show the progress of output generation, as shown in Figure 10. The user needs to review the results in the "final output" worksheet of the output file and replace some of the "?" labels with "F".

Phase 3: Explore a labelled dataset

On the "explore a labelled dataset" page of *Blair*'s GUI, the user can explore a dataset labelled using the "label a new dataset" page. The exploration has two steps: data summarisation and interactive query of bushfire evacuation behaviour.

Data summarisation: Opening the output data file on the "explore a labelled dataset" page, the user will be provided with a summary of the labelled data as shown in Figure 11. Note that in Figure 11, there is highlighted text reminding the user to review the final output and review some posts in the final output, change the "?" labels to "F" to reject them as irrelevant posts, for better data exploration. The user only needs to review a small number of posts, as normally there is only a small number of those with "?" in column "about author' evacuation" in the spreadsheet of the final output, so it will not take too much of the user's time. In this way, *Blair* obtains user-confirmed results for quality exploration of the dataset.

Interactive inquiry: Switching to the tab "Interactive Inquiry of Evacuation Behaviour" on the same page allows the user to query the posts with specified conditions. Figure

12 shows, as an example, that there are five matched posts for a query with time condition (find posts between "24/Oct/2019 – 26/Oct/2019") and location condition (find posts mentioning "canyon country").

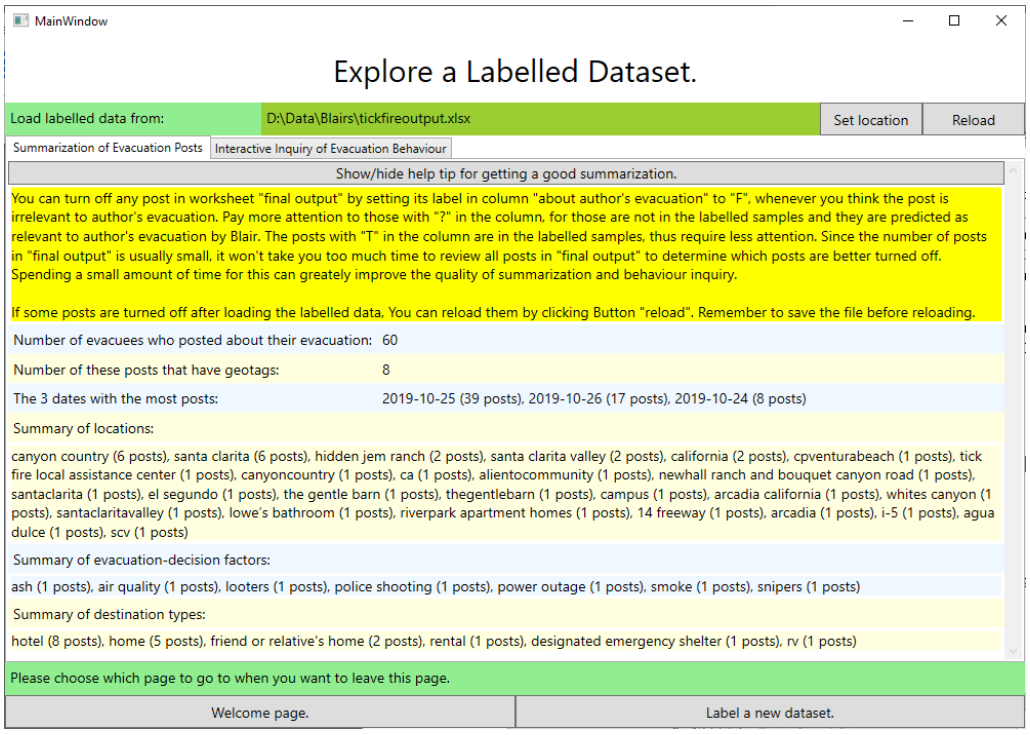


Figure 11. Summarisation of the labelled dataset on page "explore a labelled dataset".

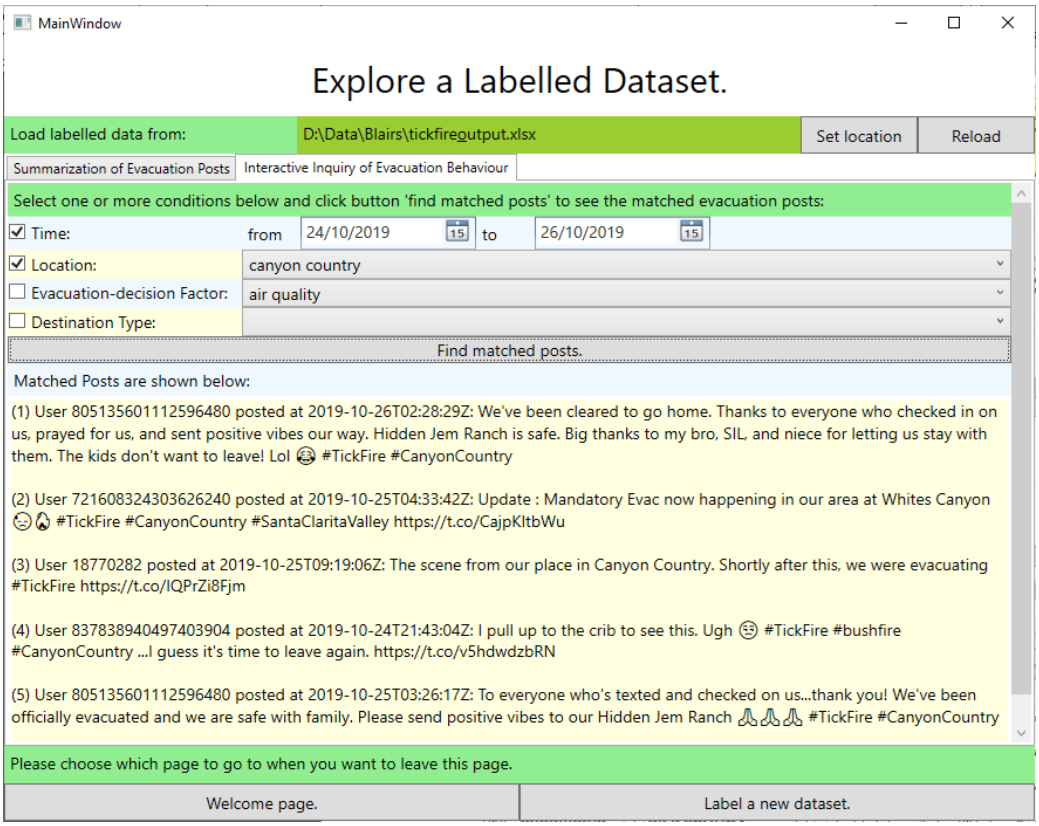


Figure 12. Five posts are matched for a query with time and location conditions.