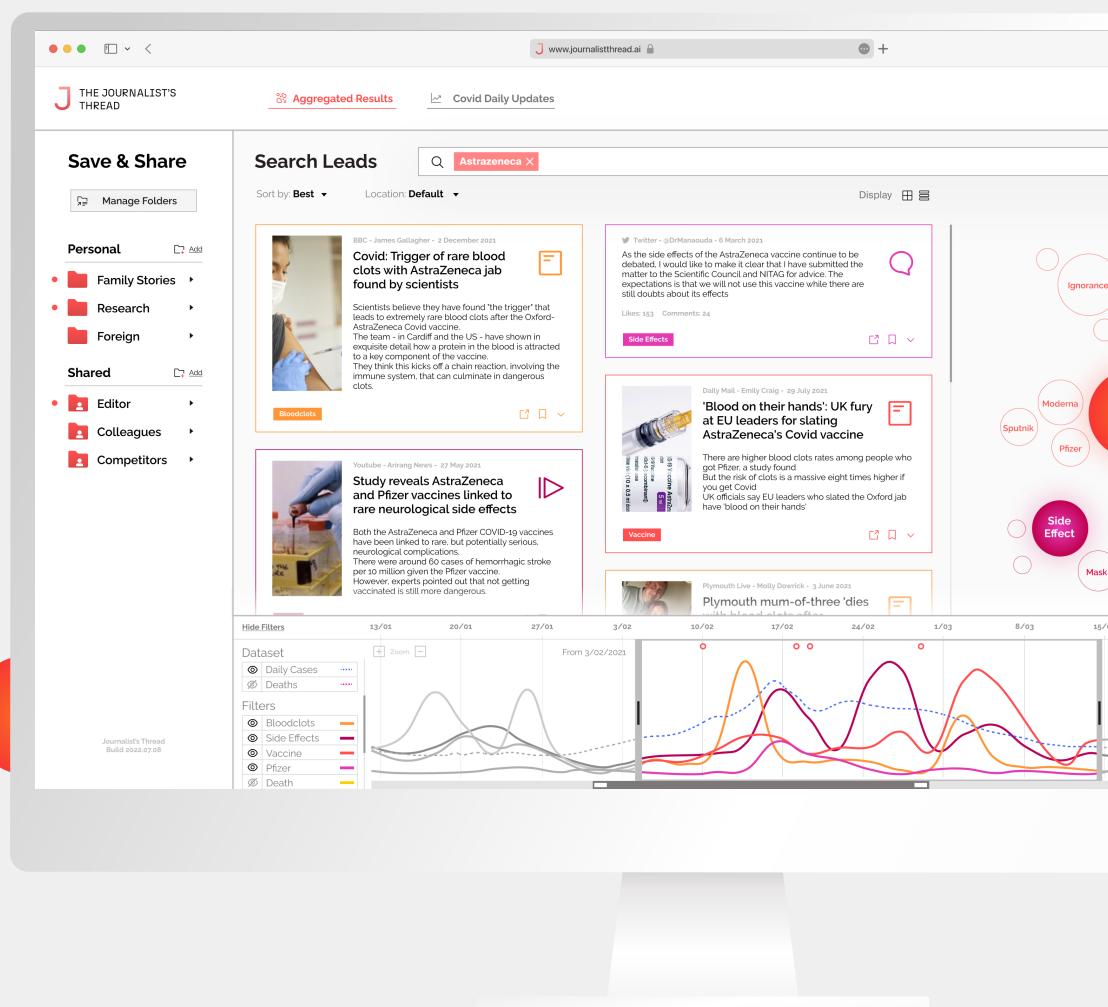


THE JOURNALIST'S THREAD

REQUIREMENT AND DESIGN PAPER



TEAM

ELIE BARAKAT
MARCO DE CRISTOFARO
MATTEO PAOLI
MANUEL REALE
ANDREA SIMEONE



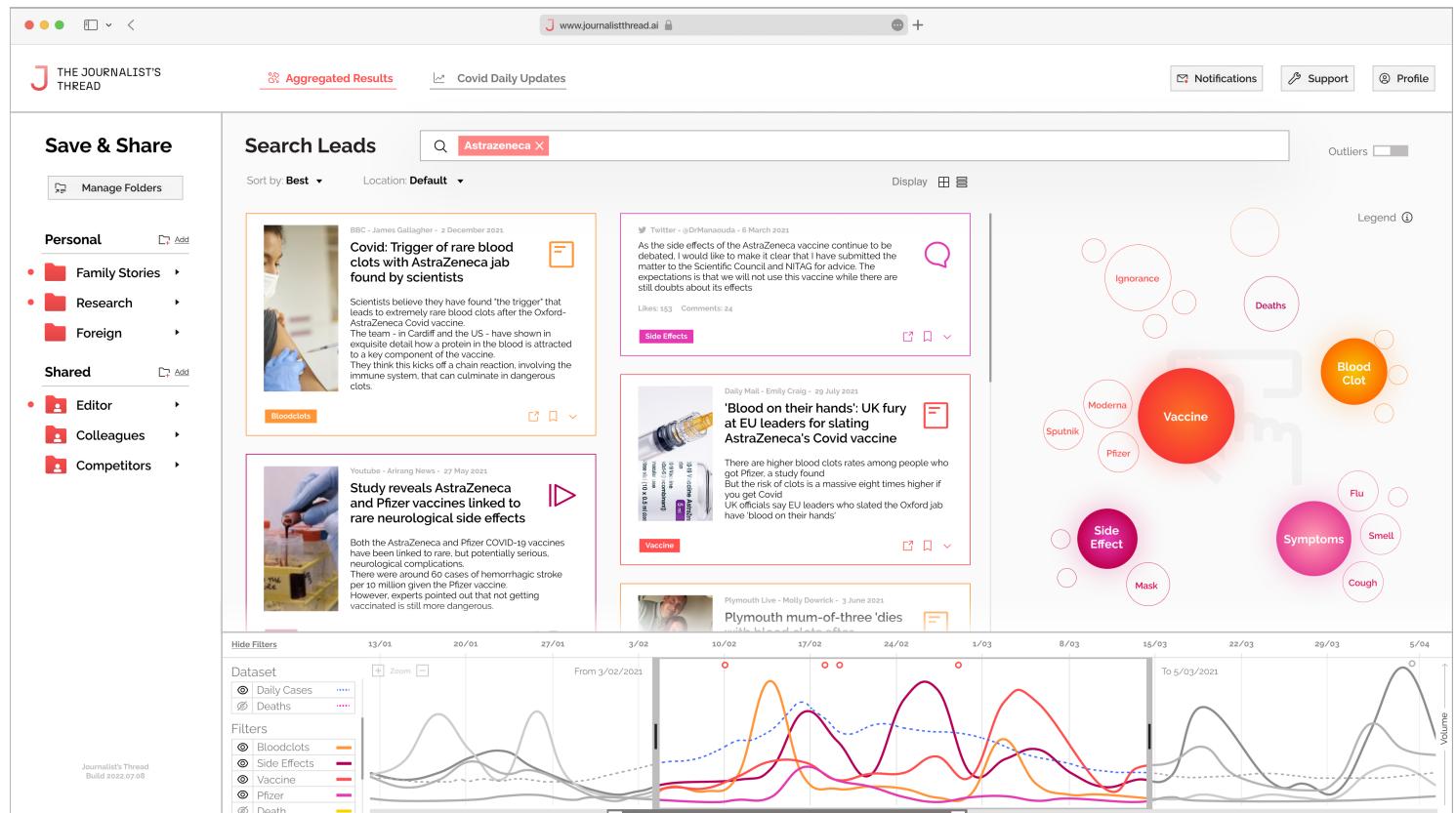
SCOPE OF THE PROJECT

The Journalist's Thread is a web platform that gathers and clusters cross-media data from several sources to optimize the navigation of large data sets, it acts as a supporting tool for **decision making** for journalists and media researchers during the newsgathering and data collection processes.

The system employs different machine learning techniques to organize and cluster data scraped from different media such as articles, social media posts and audiovisual content, which will be then visualized in the dashboard.

This tool will help researching specific topics starting from a query inserted by the user, this could be an object, a name of a person, a place or a general topic; as a result of the search the user will be shown:

- An aggregated view of the main topics related to the searched word
- How the cluster size changed through time which would represent the media coverage of each cluster through time
- Information regarding each piece of content included in the results, such as a summary of its content, the sentiment analysis of the text, keywords, but also basic information about it like author, source, image, etc...



IMG The aggregated results page of the platform, showing the clusters on the right, a timeline at the bottom and the single pieces of content in the middle

GOAL OF THE SYSTEM

In order to make the proposed platform work, the system will need to gather all the data required, which will be achieved by scraping articles, social media posts, audiovisual content and papers from the web; and then process this data to cater to the different features offered by the platform listed below:

- Clustering: the system will have to group the different pieces of content into clusters. Each cluster will represent a specific topic covered by the included articles, social media posts, and audiovisual content. As a by-product of this process it will also have to detect outliers: pieces of content that doesn't belong to the main topics
- Keyword extraction: the system will have to detect recurring and relevant keywords in the collected datasets which will be then used to help the research and filtering for the user.
- Sentiment Analysis: the system will have to detect the sentiment analysis of the articles starting from their text
- Summarization: the system will offer a condensed version of the content of each piece of content to help the user quickly understand the content of them

DATASETS

For each medium the scraping and processing method required is different and requires different tools, however they could all be concatenated to create a single dataset where the different machine learning tasks would work only the textual part of the pieces of content, where most of the relevant information lies.

For each medium there would be some parameters to decide which piece of content would be scraped, for example for social media there could be a minimum engagement to be selected and for articles a source reliability and relevance index.

Below there are listed possible processes and datasets for the main categories of content.

ARTICLES



Example of dataset

Date	URL	Author	Title	Text	Engagement
02/12/2021	www.bbc.com/news/health-60259302 ...	Fergus Walsh	Covid: Trigger of rare blood clots in...	Researchers in Cardiff and US...	15367

AUDIOVISUAL



Example of dataset

Date	URL	Author	Title	Text	Engagement
02/12/2021	www.youtube.com/watch?v=TigikMU3 ...	Good Morning America	Vaccine effectiveness against virus...	Researchers in Cardiff and US...	654

REDDIT POSTS



Example of dataset

Date	URL	Author	Title	Text	Subreddit	Filter
02/12/2021	www.bbc.com/news/60259302 ...	@UKHSA	Who funded the research behind...	With funds coming from...	r/Covid19	Top

TWEETS



Example of dataset

Date	URL	Author	Text	Location	Retweets	Followers
15/03/2021	www.twitter.com/UKHSA/73432	@UKHSA	Look at this...	UK	97	524.000

SCENARIOS

n°	Scenario
1	The user must be able to install our product and connect to a centralized database where the algorithm processes the data
2	The user must be able to choose some favorite sites / newspapers or favorite Twitter users that are more visible when part of the results processed by the search
3	The user must be able to limit the search to a specific geographical area
4	The user must be able to see a list of trending topics on twitter and reddit
5	The user must be able to perform a search starting from one of the trending topics extracted from twitter or reddit
6	The user must be able to search starting from a typed word
7	The user must be able to choose a specific time frame to limit the search
8	The user must be able to choose to view only one type of medium among those analyzed (articles, papers, videos, tweets)
9	The user must be able to use a keyword that emerged from the results as a prompt to filter the search starting from it, excluding or including the results that include it
10	The user must be able to see the variation in the amount of content search results on a timeline
11	The user must be able to see the sentiment analysis of every single piece of content resulted from the search
12	The user must be able to see the outliers that emerge from the clustering of the articles
13	The user must be able to save a specific proposed result (in the same way as saving a file in a folder) and share it in the platform
14	The user must be able to export the search made to pdf

FUNCTIONAL REQUIREMENTS

SCRAPING

- Scraping tweets depending on a specific parameter (number of likes, number of followers, country...)
- Scaping trending topics from twitter
- Scaping trending topics from reddit
- Scraping online articles depending on reliability of the source
- Scraping online papers from academics archives
- Scraping videos from Youtube (and their transcription) based on keyword or channel

CLUSTERING

- Unsupervised clustering of the sources analyzed (articles, papers, videos, tweets)
- Supervised clustering of sources analyzed (articles, papers, videos, tweets) based on specific parameters chosen (frequency of a specific word, links to websites...) to train the algorithm
- Outlier Detection

KEYWORDS

- Keyword Detection of the sources analyzed (articles, papers, videos, tweets)

SENTIMENT ANALYSIS

- Sentiment Analysis of the sources analyzed (articles, papers, videos, tweets) based on online comments and feedbacks on them left online

ASSOCIATION

- Association between the sources analyzed (articles, papers, videos, tweets) to highlight correlations (authors, date...)

SUMMARIZATION

- Summarization of long articles into small paragraphs

FROM DATA TO UI

The scheme below shows the information flow, it starts from the source then shows the processes it goes through and finally how it will appear in the UI

Source	Raw Data	ML Tasks	UI Elements
	<ul style="list-style-type: none"> ● Title ● Text ● Source ● Link ● Timestamp ● Author ● Social Media Metrics 	<ul style="list-style-type: none"> • Extractive Summarization • Clusterization • Sentiment Analysis • Keyword Extraction 	
<div style="background-color: red; color: white; padding: 5px; text-align: center;">NEWSPAPER ARTICLE</div> <div style="background-color: red; color: white; padding: 5px; text-align: center;">SOCIAL MEDIA CONTENT</div> <div style="background-color: red; color: white; padding: 5px; text-align: center;">AUDIOVISUAL CONTENT</div>	● Text	<ul style="list-style-type: none"> • Clusterization • Association • Keyword Extraction 	
	● Text ● Date	<ul style="list-style-type: none"> • Association • Clusterization 	

SCHEME: Connection between the primary source used and the final UI element shown in the dashboard

LINKS

WEBSITE

<https://manuelreale.github.io/The-Journalist-s-Thread/>

UI PROTOTYPE

<https://www.figma.com/proto/riEbx80DbEO4Wn7B7gpcc1/The-Journalist's-Thread-Prototype?page-id=232%3A17870&node-id=232%3A18957&viewport=2035%2C-320%2C0.27&scaling=scale-down&starting-point-node-id=232%3A18957>

VIDEO

<https://www.youtube.com/watch?v=QicPmUmtnJk>