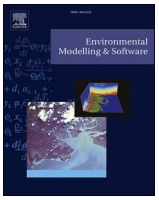
Environmental Modelling and Software 167 (2023) 105734

Contents lists available at ScienceDirect

Environmental Modelling and Software

journal homepage: www.elsevier.com/locate/envsoft



FF-IR: An information retrieval system for flash flood events developed by integrating public-domain data and machine learning

Rohan Singh Wilkho [a,*], Nasir G. Gharaibeh [a], Shi Chang [a], Lei Zou [b]

[a] Zachry Department of Civil and Environmental Engineering, Texas A&M University, College Station, TX, 77840, USA
[b] Department of Geography, Texas A&M University, College Station, TX, 77840, USA

ARTICLE INFO

Handling Editor: Daniel P Ames

Keywords:
Flash flood
Information retrieval
Storm events data
Machine learning

ABSTRACT

Structured databases on flash flood (FF) events have limited information and lack emerging data (e.g., visual media). The web is rich with information that can bridge this gap. However, search engines return long lists of webpages cluttered with commercial and irrelevant information. To address this challenge, we developed a FF information retrieval (IR) system (FF-IR). The system uses machine learning (ML) models in novel ways to automate and enhance this IR process. FF-IR consists of three steps: (1) creates event-specific search queries from the publicly available Storm Events dataset and directs them to Google to collect candidate webpages; (2) transforms the candidate webpages to relevance features; and (3) classifies each candidate webpage as relevant or non-relevant using our ML models. FF-IR outperforms direct Google searches by over 100%, measured by the F2-score. Natural hazard researchers and practitioners can use FF-IR to facilitate FF risk assessments and mitigation planning.

1. Introduction

Information on past flash flood (FF) events can enhance understanding of the causes and impacts of these events through the development of predictive models, case studies, and lessons learned; and consequently, facilitate better risk assessments and more effective preparedness and mitigation strategies (Sarker et al., 2020; Terti et al., 2019; Yu et al., 2018). While existing structured databases contain valuable information on these events, they lack emerging data forms (e.g., visual media) and details (e.g., post-flood illnesses and health risks, disruptions to infrastructure services, human injury types). In this paper, we suggest that the web can be leveraged to bridge this information gap (Illingworth, 2001; Tanner et al., 2009). However, conventional search engines (such as Google) are not optimized for domain-specific searches, owing to the Internet's growing size and the commercial aspects of the search engines (Google Interference, 2019; Lewandowski, 2012). In studying hurricane recovery information, Zheng et al. (2013) found that new technologies are needed for extracting information from the web and delivering that information without redundancy and irrelevance.

To help address this bottleneck, we developed a flash flood information retrieval (FF-IR) system to automate and improve the process of retrieving webpages that contain relevant information about FF events. IR is broadly defined as the process through which computer systems lead users to information sources (webpages, documents) that enable them to fulfill their information needs (Manning et al., 2012). In conventional IR systems (e.g., web search engines), the query entered by the user is the information need, and the ranked webpage list returned by the IR is the information retrieved. A traditional IR system typically operates in three steps (Manning et al., 2012): crawling, indexing, and ranking. Crawling is the process of discovering webpages that exist on the web. Indexing is the process of adding new webpages to the IR system's index (i.e., a database of webpages already crawled by the IR system). Conventional IR systems (such as Google's search engine) typically maintain an index of trillions of publicly accessible webpages and return a subset of webpages from the index upon receiving a query (Lashkari et al., 2017). Ranking is the process of ranking the webpages in the index as per their relevance to a user's query. Different search engines employ different ranking algorithms to identify webpages satisfying the information need (Google Search 2019). Previous IR studies have used Machine Learning (ML) in different ways. For example, Dehghani et al., (2017) and Pang et al., (2017) leveraged neural networks' abilities to learn abstract representation from data to rank

* Corresponding author.
E-mail addresses: rohanswilkho_93@tamu.edu (R.S. Wilkho), ngharaibeh@civil.tamu.edu (N.G. Gharaibeh), changs18@tamu.edu (S. Chang), lzou@tamu.edu (L. Zou).

https://doi.org/10.1016/j.envsoft.2023.105734
Received 11 October 2022; Received in revised form 19 May 2023; Accepted 22 May 2023
Available online 5 June 2023

documents that are likely to be relevant to a user's query, Guo et al., (2022) and Wu et al., (2022) employ pre-training of webpage document object model tree structures and webpage hyperlinks, respectively to enhance retrieval performances, and Lin (2021) use ML to learn scoring functions to maximize the similarity scores between queries and relevant documents. FF-IR, on the other hand, uses ML to enhance the delineation between relevant and non-relevant search results.

Unlike conventional IR steps, FF-IR (described in this paper) integrates publicly available and regularly updated National Oceanic and Atmospheric Administration Storm Events (SE) data and ML models to retrieve webpages containing relevant information about FF events specifically. Upon receiving past FF event(s) of interest as user input, FF-IR constructs event-specific search queries using information from the SE data. FF-IR then directs these targeted search queries to a conventional search engine (e.g., Google) to collect candidate webpages and transforms these webpages into numerical features. Then, it employs a trained ML algorithm to classify the candidate webpages into two categories (relevant and non-relevant). After the classification, FF-IR outputs the webpages containing relevant information for the FF event of interest (e.g., news stories and published reports, among others). The term 'relevant' is used here to mean information specifically related to the event(s) of interest, covering diverse topics such as physical damages, economic losses, human harm, rescue operations, hydrometeorology, federal or state declarations, community resilience, public health notices, and mitigation measures, among others. The novelty of the FF-IR is threefold:

(1) The FF-IR fills a gap in the literature about harvesting information from the internet (webpages and web-documents) about flash flood events, and more broadly natural hazard events. Previous studies have focused on retrieving information from social media platforms to establish better communication channels between stakeholders during active disaster situations and to gain insights from social media posts (Romero and Becker, 2019; Ullah et al., 2021; Zheng et al., 2013). The FF-IR is not limited to social media content. Instead, we harvest relevant information about FF events from webpages and web-documents available in newspaper websites, blogs, governmental websites, academic websites, and other types of websites.
(2) It uses the SE data to form targeted search queries. This approach is advantageous because (a) it eliminates the crawling, indexing, and ranking processes used in conventional IR systems, which are time-consuming and memory-intensive (Dean, 2009), and (b) it increases the accuracy of the search through better search queries. The SE data is publicly available and is regularly updated by NOAA (typically on monthly basis) to add new flash flood events.
(3) It advances the application of ML in the natural hazard and disaster domain by providing a newly constructed ML model to retrieve webpages containing relevant information about FF events. ML has been usually used in the past within the context of natural hazards and disasters in primarily four ways: (1) prediction of future events and their impacts (e.g., Hosseini et al., 2020; Khanmohammadi et al., 2022), (2) damage assessment (e.g., Khajwal et al., 2022; Hao and Wang, 2021), (3) mapping the susceptibilities of regions to natural hazards and disasters (e.g., Gudiyangada Nachappa et al., 2020 and Zhao et al., 2019), and (4) extraction of real-time actionable information from social media tweets (e.g., Barker and Macleod, 2019 and Donratanapat et al., 2020).

We focus on FF events because they occur frequently, and therefore are widely reported in the web. Furthermore, flash flooding is among the most lethal and destructive natural disasters (Ashley and Ashley, 2008). During 2000–2019, the National Weather Service (NWS) records show that FFs have resulted in approximately 70%, 72%, and 72% of all flood-related fatalities, injuries, and damages in the USA, respectively (NOAA Storm Events Database, 2021). Increasing urbanization and climate change are likely to result in worse flash flooding in the future (Milly et al., 2002). Despite the safety and economic risks associated with FF, detailed information on past events is scarce in structured databases. Terti et al. (2019) called for data "with more details and at finer resolutions to better capture local temporal and spatial complexities associated with human losses from flash flooding." These observations highlight the need to harness the full potential of the web to improve access to information that can inform FF mitigation and preparedness planning (Ogie and Verstaevel, 2020). FF-IR responds to these calls.

As a result, researchers and practitioners can use FF-IR to conveniently retrieve detailed information regarding past FF events. The retrieved information can then be used to conduct various analyses, such as risk assessment (Sarker et al., 2020), causal modeling (Ramanan and Natarajan, 2020), identifying vulnerable communities (Kontokosta and Malik, 2018), and predicting future events (Anbarasan et al., 2020).

The rest of the paper is organized as follows. Section 2 provides the literature review for this study. Section 3 discusses the public-domain seed data (SE data). Section 4 discusses FF-IR's architecture and design. Section 5 presents the experimental results to select the best-performing ML model for the IR system and compares conventional search engines and FF-IR. Section 6 provides a summary and conclusions. Finally, section 7 identifies ongoing and future works.

2. Literature review

2.1. Challenges in web information retrieval

Recognizing the limitations of conventional web search processes, the IR community has used ML to improve the ranking process of retrieved webpages. Traditional models like the Boolean model [Hiemstra, 2009], Vector Space Model [Hiemstra, 2009], BM25 [Robertson and Zaragoza, 2009], and PageRank [Brin and Page, 1998] did not use ML. However, with advances in ML modeling techniques, increased data availability, and improved computing power, recent studies have applied ML and deep learning techniques to build more effective ranking systems. The influential study "Learning-to-Rank" [Liu, 2010] sparked interest in the community to use ML. For instance, Yilmaz et al., (2019) applied the then state-of-the-art language model, i.e., Bidirectional Encoder Representation from Transformers (BERT), to enhance the ranking results of retrieved short social media posts and newspaper articles. Nogueira et al., (2019) developed a multi-stage ranking model using BERT to ensure document quality in the ranking process while providing comparable results. Esteva et al., (2021) also used BERT in a multi-stage ranking system for a domain-specific search engine built to retrieve relevant COVID-19-related documents. Furthermore, Chekalina & Panchenko [2022] combined decision trees and BERT to establish a new baseline for ranking performance in comparative argument retrieval.

Despite the advances in the ranking process, the sheer amount of information on the web presents a challenge for IR systems, as the ranked lists of webpages often contain non-relevant results. This is especially concerning for domain-specific IR systems, where the users expect the retrieved webpages to be highly relevant to their search query. To address this issue, IR researchers have explored combining webpage ranking with classification as an additional step to filter out non-relevant webpages [Hashemi, 2020]. For example, in the domain of biomedicine, Liu et al., (2021) developed an advanced IR system that included a bidirectional gated recurrent unit-attention model for classifying webpages in the ranked list returned by search engines and further re-ranking the classified webpages. Similarly, in legal case retrieval tasks, Hudzina et al., (2021) and Shao et al., (2020) used ML-based binary classification models to improve the ranking performance of retrieving legal cases. Wang et al., (2021) improved the accuracy of a question-answering task (a subfield of IR) by re-ranking

retrieved results using BERT. In this study, we take a more practical approach to filtering out non-relevant webpages. We implement a classification task over the candidate webpages (obtained from Google search) and do not re-rank them.

2.2. Current web IR systems for disaster management

Web IR systems in the context of disaster management have focused on retrieving information from social media. For instance, Barker and Macleod, 2019 and Donratanapat et al., (2020) developed ML pipelines to extract data from Twitter, enabling the identification of vulnerable communities and infrastructure systems in flood-prone areas. Ullah et al., (2021) and Loynes et al., (2022) focused on retrieving relevant tweets to support emergency officials in rescue operations during disasters. These IR systems primarily aim to retrieve real-time information and establish effective communication channels among stakeholders in active disaster situations. In contrast, FF-IR is centered around retrieving webpages containing information relevant to past FF events of interest more rapidly and accurately than currently possible using conventional internet search methods.

3. SE data

This section describes NOAA's public-domain SE data, which is used in this study as seed data for FF-IR. The SE data contain information on reported weather/storm events (floods, hurricanes, tornadoes, among others) from January 1950 to the present. In total, the SE data contains information on approximately 37,500 FF events from 2010 to 2019. The SE data is available in the public domain and is maintained by NOAA's National Weather Service (NWS). It is updated monthly; however, the published data lags 90–120 days behind the current date. NOAA curates the information in the SE data from multiple sources, including NWS, media, law enforcement, government agencies, emergency managers, private businesses, and individuals (NOAA Storm Events Database, 2021). Information recorded in the SE data about these events is limited to the following:

a) Event date and approximate location of the affected area,
b) Impact: number of human injuries, number of human fatalities, and cost estimates of damage to crops and property,
c) Episode narrative: a brief text description of the event meteorology, and
d) Event narrative: a brief text description of the event's impact and the surrounding conditions, such as road conditions.

In this study, we use the above information as seed for retrieving additional types of information from the web, including:

a) Detailed impacts, such as flood-related illnesses and health risks, details about infrastructure damage, disruptions to infrastructure service, human injury types and locations
b) Meteorological conditions, such as rainfall intensity
c) Physical characteristics of affected sites, such as land cover, soil topology, and built environment
d) Socioeconomic and demographic characteristics of affected communities, population density, income, race, and ethnicity
e) Geographic information, such as affected census tracts and census block
f) Visual information, such as videos and photos

The relevant webpages (e.g., news stories and published reports) retrieved from FF-IR can be used to extract this information, which is essential for understanding the flash flooding phenomena and mitigating its impacts.

4. FF-IR architecture and design

FF-IR's architecture consists of five major components, as shown in Fig. 1. The first component, "*Information Need*", takes the flash flood event(s) of interest from the user as input. The "*Targeted Search Query Formation*" component uses the SE seed data to form targeted search queries about the event of interest. The "*Candidate Webpage Collection*" component collects candidate webpages for each formed targeted search query. The "*Relevance Feature Generation*" component generates relevance features for each candidate webpage for the event of interest. Finally, the "*Classification*" component uses the generated relevance features as the input for a trained Binary Classification Machine Learning (BCML) model to classify each candidate webpage as either relevant or non-relevant. The BCML model was trained and tested on a domain-specific dataset explicitly developed for this study. The following subsections describe each component in detail.

4.1. Component 1: information need

A conventional IR system requires an information need as input provided by the user (e.g. query in Google search). In this FF-IR, the information need is a past flash flood event. The user specifies the location of interest (county and state), then FF-IR identifies all flash flood events that occurred in the entered county and state, as per the SE data. The user can select any event(s) identified by FF-IR.

4.2. Components 2 & 3: Targeted Search Query Formation and Candidate Webpage Collection

FF-IR forms targeted search queries by extracting informative sentences from the episode and event narratives in the SE data. This system identifies the informative sentences using two criteria: sentences containing (1) keywords and (2) capitalized words (except the first word in a sentence) indicating proper nouns. The keywords are used to identify informative sentences specific to flash flooding (Table 1). These sentences may contain information such as the rainfall intensity and location of fatalities. The system also extracts sentences containing capitalized words because such words may indicate the names of roads, neighborhoods, landmarks, yielding specific information about the subject flash flood event.

Fig. 2 contains an example showing how sentences are extracted from SE narratives to form search queries for a FF event. The underlined and bold words in Fig. 2 are the keywords used to extract informative sentences from the event and episode narratives for the particular FF event. The targeted queries were formed using a rolling window of one, two, and three sentences. Therefore, the number of targeted queries is a function of the number of sentences in the SE narratives. A search query length of two sentences resulted in the highest number of relevant webpages returned by Google. For a 2-sentence query, the number of targeted queries formed is N-1, where N is the number of sentences in the SE narratives. For example, nine targeted queries are formed for a narrative that has 10 sentences. The extracted sentences are used collectively with the event's county, state, and date to form the targeted search queries. FF-IR forms an additional generic query based on the date, state, and county information because all SE data records do not contain narrative descriptions. Thus, in the example presented in Fig. 2, the last query in the "Constructed queries" box is the generic query. FF-IR directs each formed query to the Google Search Engine (using the Python-based library "googlesearch").

For each formed query, FF-IR collects the top 10 ranked webpages. Fetching the top 10 ranked webpages for each search query was determined using a trial and error process. It was found that for most FF events, the relevant webpages were within the top ten results returned by Google. For FF events that have little internet coverage, this choice was intuitive. However, it may not be intuitive for events that have extensive internet coverage. We found that these events tend to have

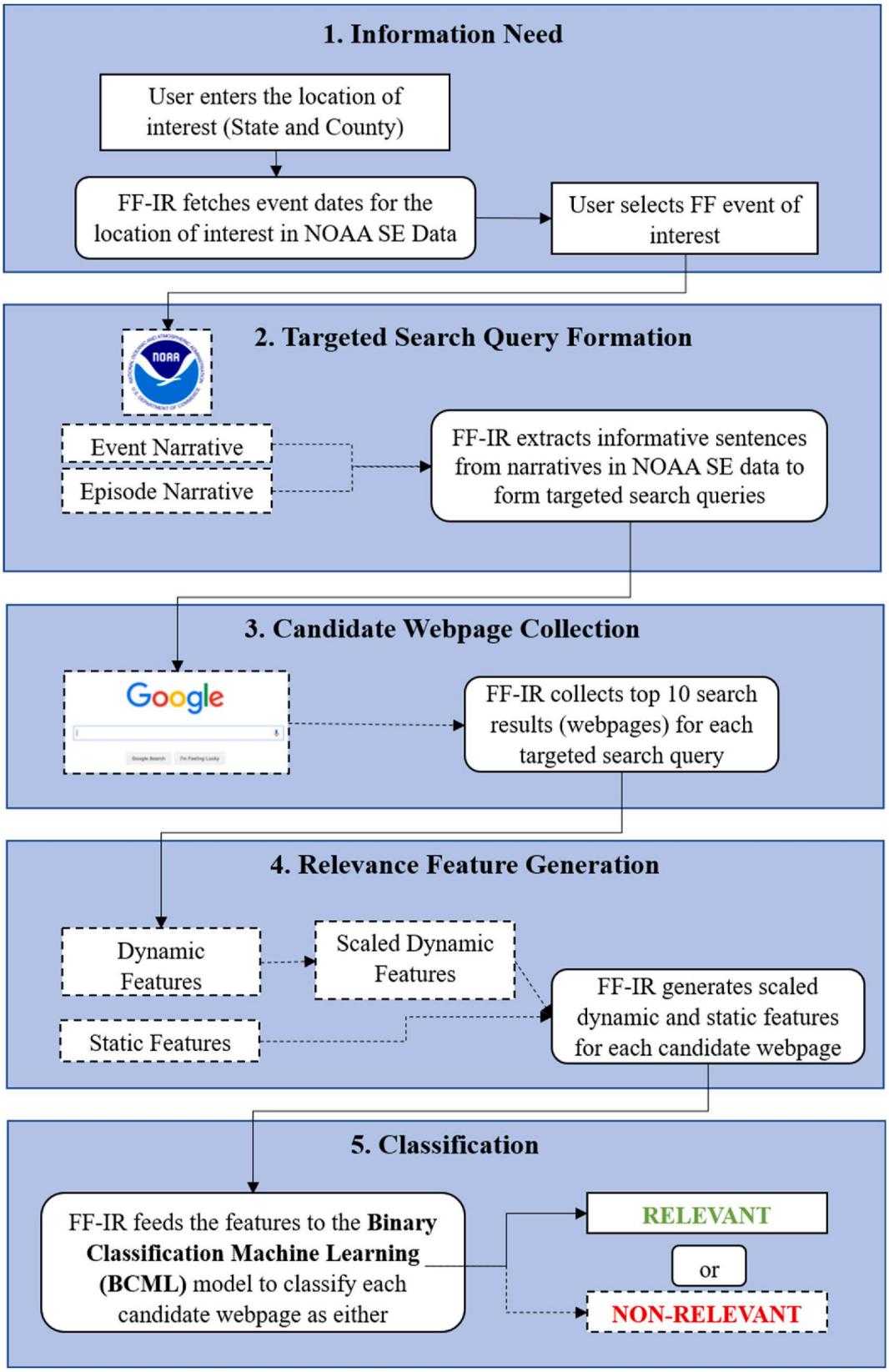


Fig. 1. FF-IR architecture.

longer SE narratives as well. Therefore, for these events, FF-IR forms more targeted queries with greater content heterogeneity among them. This content heterogeneity helps concentrate the relevant webpages in the top 10 results from Google. Hence, for the example in Fig. 2, this system collects 40 webpages in total for the four constructed queries. After removing duplicate webpages (if any), the unique list formed is the candidate webpage list for the subject event. The selection of the top 10 results as a threshold was determined based on two considerations: (a)

Table 1
Keywords used to extract informative sentences from narratives in the SE dataset.

Issue of Interest	Keywords
Injury	injury, injuries, injured, injuring, hospital, rescue, trapped, hypothermia, evacuated, accident, bruises
Fatality	deadly, died, drowned, drown, drowning(s), fatality, submerged, swept, dead, killed, perished, deceased, body, recovered, lives, life, fatalities, accident, hypothermia, lost
Rainfall amount/ intensity	feet, inch
Damage	damage
Flood type	flash

relevant webpages were rarely found outside the top 10 results, and (b) computational complexity needed to be maintained at a manageable level.

4.3. Component 4: relevance feature generation

FF-IR transforms each candidate webpage into numerical values, called relevance features. These features are similarity scores for the candidate webpages, computed by comparing the webpage content to the existing information regarding the event(s) of interest in the SE data. We can broadly classify these features into two categories:

a) Dynamic features: these are similarity scores for a webpage relative to the other candidate webpages for the same flash flood event.
b) Static features: these are similarity scores for a webpage independent of the other candidate webpages for the same flash flood event.

Wilkho et al. (2023) contain pseudocodes for dynamic and static relevance feature generation.

4.3.1. Dynamic features

Dynamic features are quantitative values representing the degree of similarity between the webpage text and the narratives in the SE data. In addition to the similarity between the entire webpage text and the SE narratives, the dynamic features also include similarity scores between the individual text passages within the candidate webpage and the SE narratives. These similarity scores are computed using different natural language processing (NLP) techniques, such as latent semantic analysis (LSA) (Landauer et al., 1998) and word embeddings (word2vec (Mikolov et al., 2013) and doc2vec (Le and Mikolov, 2014)).

For retrieving the most relevant passage(s) from webpage text, FF-IR considers a webpage to be a set of passages (Equation (1)).

$$wp = \{p_1, p_2, p_3, \ldots. p_n\} \tag{1}$$

where wp = webpage,

p_i = passage i in webpage wp. A passage is a contiguous section of the text in the webpage.

And for each candidate webpage, the system constructs summaries by extracting the passages most similar to the SE narratives (i.e., those having the highest similarity score with the SE narrative under consideration).

$$summarized\ wp = \{p_a, p_b\} : sim(p_a, N)\ \&\ sim(p_b, N) > sim(p_i, N) \forall i = 1\ to\ n, i \neq a, b \tag{2}$$

where *summarized wp* = webpage summary,

p_a, p_b = passages having the highest cosine similarity score among all the passages p_i, *(i = 1 to n)* of a webpage *wp* when computed against the SE narratives,

N = SE narrative under consideration,

n = Total number of passages in the webpage under consideration,

sim (a, b) = Cosine similarity as per Fig. 3.

In order to compute the dynamic features, this system transforms the webpage texts/summaries/passages and the SE narratives into numerical vectors (vectors a and b) and calculates the cosine similarity (*sim(a, b)*) (Fig. 3). The cosine similarity ranges from −1 to +1, where −1 denotes no similarity and +1 denotes the highest similarity. The system considers the maximum among the two similarity scores for events having both narratives in the SE data (i.e., the event and episode narratives).

Table 2 contains the complete list of dynamic features used in this study. Each score serves a different purpose in this methodology, as follows:

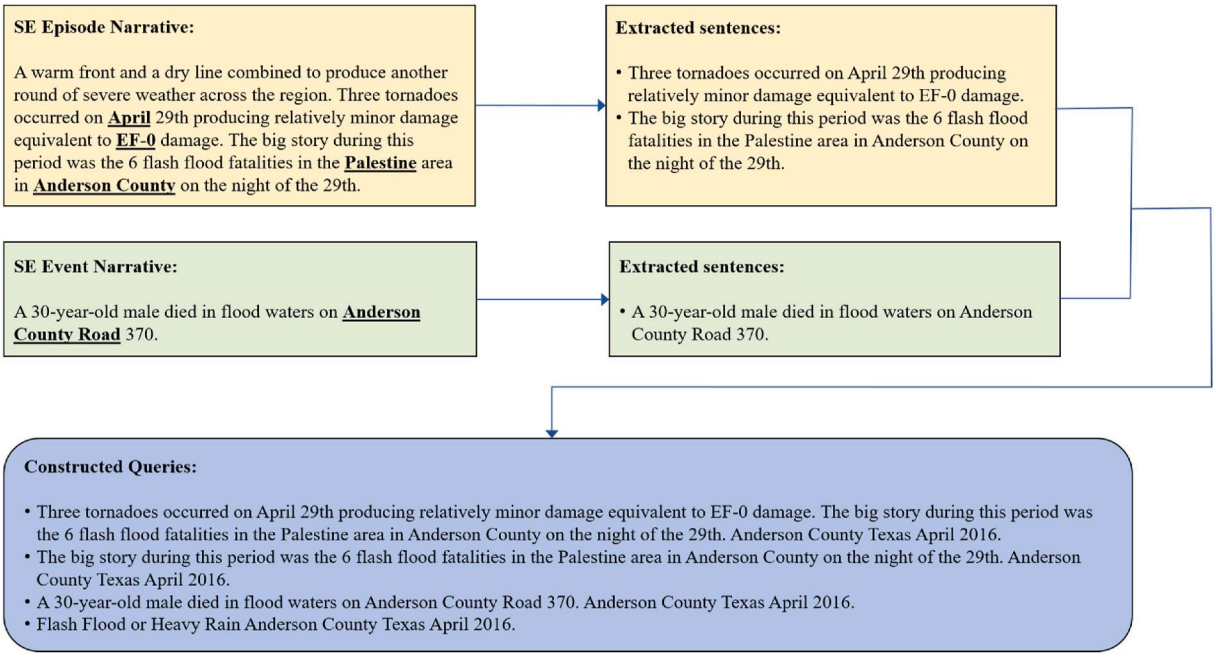


Fig. 2. Example showing the formation of a targeted query for a flash flood event from the SE data.

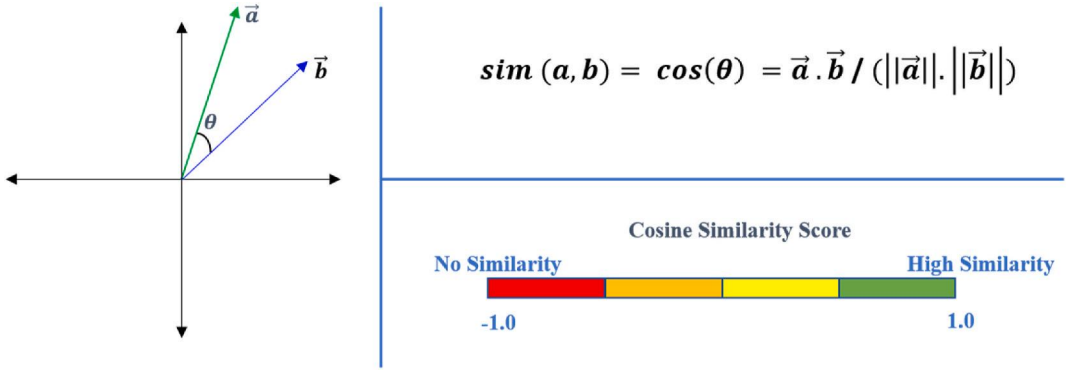


Fig. 3. Cosine similarity score for two vectors.

Table 2
List of dynamic features computed to represent a webpage.

Feature Name	Similarity computed between	Text-to-vector transformation method
Lsa-sim	Entire webpage and SE narratives	Latent semantic analysis
Passage-1-y-x	Webpage summary and SE narratives	For webpage passages - word2vec network For webpage summaries – latent semantic analysis
Passage-2-y-x	Webpage summary and SE narratives	For webpage passages - doc2vec network For webpage summaries – Latent semantic analysis
Passage-3-y	Webpage individual passages and SE narratives	Latent semantic analysis

Note: y = number of words in a passage (250, 350 or 450), and x = webpage length based on number of passages (1, 25, 35, 45). x = 1 indicates summary length is one passage. And, x = 25, 35, and 45 indicate summary length is 25%, 35%, and 45% of the total number of passages in the webpage.

a) The *Lsa-sim*feature is used as the similarity measure between the entire webpage text and SE narratives.
b) The *Passage-1-y-x* and *Passage-2-y-x* features are used as the similarity measures between the webpage summary and SE narratives. However, *Passage-1-y-x* employs the word2vec network, whereas *Passage-2-y-x* employs the doc2vec network to construct the summaries. We trained both word2vec and doc2vec models on the webpage texts. The final *Passage-1-y-x* and *Passage-2-y-x*scores are the similarity measures between the constructed summaries and the SE narratives.
c) The *Passage-3-y* feature is used as the similarity measure between the webpage's individual passages and the SE narratives, where *Passage-3-y*is the maximum score across all passages in the webpage.

The y and x in Passage-1-y-x, Passage-2-y-x, and Passage-3-y represents the number of words used to form a passage and the webpage summary length, respectively. We could not arrive at optimal values for summary length and passage length that apply to all kinds of webpages. This is because some webpages contain very long descriptions of multiple events, whereas some might contain small descriptions (such as an emergency declaration statement). The webpages in the first category will require a small passage to be relevant to the FF event of interest. In contrast, for webpages in the second category, we would need a large passage to capture its relevance to the FF event. Therefore, the summary length parameter in FF-IR is varied as one passage, 25%, 35%, and 45% of the total number of passages in the webpage under evaluation. Similarly, the passage length is varied as 250, 350, and 450 words. FF-IR uses these values every time it evaluates a webpage.

After generating the dynamic features, the system converts them to a 0–1 scale, as follows:

$$dynamic\_feature_i^{'} = \left(dynamic\_feature_i - dynamic\_feature_{min}\right) \Big/ \left(dynamic\_feature_{max} - dynamic\_feature_{min}\right) \quad (3)$$

where.

dynamic_feature$_{max}$ is the maximum value of the dynamic feature among all *'num'* candidate webpages,

dynamic_feature$_{min}$ is the minimum value of the dynamic feature among all *'num'* candidate webpages,

dynamic_feature$_i$ is the dynamic feature value for *i*th *webpage*, where i ranges from 1 to *num*,

dynamic_feature$_i^{'}$ is the scaled dynamic feature value for *i*th *webpage*, where i ranges from 1 to *num*,

num = number of candidate webpages for a flash flood event of interest.

4.3.2. Static features

Static features are qualitative and quantitative values, indicating the similarity between the webpage and the event information in the SE data, as follows:

a) *Check-date*, *Check-county*, and *Check-loc* are qualitative static features (binary-0/1). Check-date is used as a similarity measure between the webpage publishing date and the event date. Check-county and Check-loc are used as similarity measures between the location information in the webpage and event location.
b) *BERT-title*, *USE-title*, *Narrative-keyword-count*, and *Narrative-word-avg* are the quantitative static features (numerical values on a continuous scale). BERT-title and USE-title are used as similarity measures between webpage titles and individual SE narrative sentences. We considered the maximum similarity score as the ultimate value. Narrative-keyword-count and Narrative-word-avg are used as similarity measures between the number of common keywords (capitalized words) and words in SE narratives, webpage texts, and titles, respectively.

Table 3 contains a complete list of the static features used in this study.

4.4. Component 5: classification - BCML model

4.4.1. Training-testing dataset

For training and testing (TT) the BCML model, we developed a TT dataset by forming customized search queries, directing them to Google search, collecting the top 10 ranked webpages (i.e., candidate webpages), and generating the relevance features (as described in the previous sub-section) for the candidate webpages. The TT dataset was generated for 500 flash flood events that occurred in the US between

Table 3
List of static features computed for each candidate webpage.

Feature Name	Similarity computed between	Possible Values
Check-date	Webpage publishing date and event date	1 if the webpage was published in the same month or the next month and same year as the event date, 0 otherwise
Check-county	Webpage text and event county	1 if the webpage text contains the county's name where the event occurred, 0 otherwise
Check-loc	Webpage text and event county & state	1 if the webpage text contains the county and state names where the event occurred, 0 otherwise
BERT-title	Webpage title and SE narrative sentences	Cosine similarity between webpage title vector and narrative sentence vectors. Text-to-vector transformed by BERT Sentence Transformer (Reimers and Gurevych, 2020).
USE-title	Webpage title and SE narrative sentences	Cosine similarity between webpage title vector and narrative sentence vectors. Text-to-vector transformed by Universal Sentence Encoder (Cer et al., 2018).
Narrative-keyword-count	Webpage text & title and SE narrative keywords	Log of the keywords count from SE narratives in the webpage text and title.
Narrative-word-avg	Webpage text & title and SE narrative words	The ratio of the number of narrative words in the webpage text and title to the total number of words in the webpage text and title.

2010 and 2019. This sample of events was chosen to (1) optimize the time and effort in manually annotating candidate webpages, and (2) keep the sample size sufficiently large to avoid over-fitting (Roh et al., 2018). The 500 events were selected using stratified random sampling to ensure that: (1) the TT dataset is not dominated by events that did not result in human harm, (2) the TT dataset included FF events from all states in the US, and (3) the TT dataset contains events across all years in the study period. The final TT dataset contained 325 events that did not result in human harm and 175 that did.

The TT dataset contained 14,420 webpages (for 500 events). The elements of this dataset are:

a) *Rows*: each row represents a webpage; the dataset contained 14,420 rows (i.e., 14,420 webpages collected for 500 events).
b) *Columns*: each column (except the first two) is a relevance feature representing a webpage; the BCML model classifies a webpage based on these features. In total, the TT dataset contained 35 relevance features.
c) *Label*: manual annotation representing each webpage as relevant (1) or non-relevant (0). Domain experts visited all the 14,420 webpages, read them and then annotated them as either containing relevant information for the event(s) of interest (1) or not (0).

Table 4 shows an abridged example of the developed TT dataset. Wilkho et al. (2022) contains the full TT dataset.

4.4.2. Imbalanced dataset and evaluation metric

The TT dataset contained 14,420 webpages, composed of 1564 relevant and 12,856 non-relevant webpages (i.e., a relevant to non-relevant ratio of approximately 1:8). This class imbalance ratio is typical in IR applications (Nallapati, 2004). However, this class imbalance necessitates applying data-sampling (DS) techniques, such as the synthetic minority oversampling technique (SMOTE) (Chawla et al., 2002), edited nearest neighbors (ENN) (Wilson, 1972), among others, to train the ML models.

Because the TT dataset is imbalanced and recall (i.e., the proportion of relevant webpages classified as relevant) is more critical for the current study than precision (i.e., the proportion of relevant webpages in those classified as relevant), we use the F2-score as the guiding metric for evaluating the performance of the ML models (Fig. 4). Previous studies suggest that the F2 score is most suitable for cases such as the present (Gu et al., 2009). To perform a comprehensive evaluation, we also assess the final model performance based on accuracy, precision, and recall scores. All metrics (accuracy, precision, recall, and F2 score) range from 0 to 1, where a higher value indicates a better performance.

4.4.3. BCML model selection

We analyzed the performance of multiple tree-based ML techniques (Decision Tree (Breiman 2017), AdaBoost (Freund and Schapire, 1996), Gradient Boost (Friedman, 2001) and Random Forest (Breiman 2017)) and DS techniques (SMOTE (Chawla et al., 2002), Borderline SMOTE (Han et al., 2005), SVM-SMOTE (Nguyen et al., 2011), ADASYN (He et al., 2008), TomekLinks (Tomek, 1976), ENN (Wilson, 1972), CNN (Hart, 1968), SMOTETomek (Batista et al., 2004), SMOTEENN (Batista et al., 2003)) combinations in this study. We considered only tree-based techniques for this study because they are: (1) easier to interpret and visualize than other ML models, (2) do not require scaling or normalization of features, (3) can handle mixed feature types (qualitative and quantitative, like in this study), and (4) have lower computational complexity (Gatnar, 2002). These traits are not requirements; however, they are advantageous for the usability and scalability of FF-IR.

To determine the feature subset most informative of the labels for each ML technique, we performed feature selection using recursive feature elimination with cross-validation (RFECV) (Guyon et al., 2013).

The steps followed in choosing the best performing ML model and DS technique combination are outlined below (and illustrated in Fig. 5):

a) Dataset Division: We experimented with three split ratios (i.e., 70:30, 75:25, and 80:20) to divide the TT dataset into training and testing sets by a random stratified split while maintaining the same class imbalance as in the TT dataset.
b) Hyperparameter Tuning: For each split ratio, we used the randomized search algorithm for hyperparameter tuning of different ML and DS model combinations. To avoid over-fitting, we selected the hyperparameter-tuned model combinations which gave the maximum average 10-fold cross-validation F2-scores.

Table 4
Abridged example of the TT dataset.

Webpage	Label	LSA-sim	Passage-1-500-01	Passage-2-300-45	Passage-3-25	Check-Date	Check-loc	Check-county	BERT-title
1	1	0.804	0.7680	0.7319	0.5383	0	0	0	0.7372
2	0	0.738	0.3648	0.7057	0.0520	0	0	0	0.4813
3	0	0.974	0.8339	0.9788	0.8224	0	0	0	0.8494
4	1	0.804	0.7680	0.7319	0.5383	0	0	0	0.7477
5	0	0.962	0.6965	0.9508	0.7621	0	0	0	0.6088
6	0	0.963	0.5751	0.9668	0.7810	0	0	0	0.8291
7	0	0.963	0.7000	0.9720	0.7288	0	0	0	0.7677
8	0	0.943	0.8056	0.9577	0.7289	0	0	0	0.7140
9	0	0.914	0.7585	0.9277	0.6620	0	0	0	0.7744
10	1	0.957	0.8412	0.9547	0.8733	1	0	0	0.8641

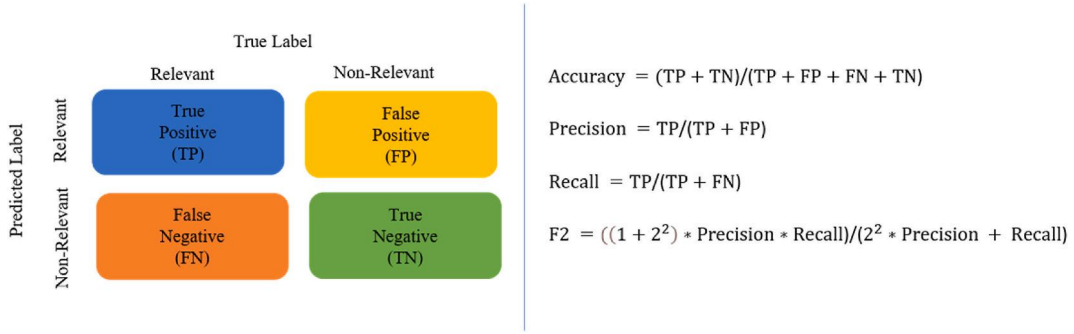


Fig. 4. BCML model evaluation metrics.

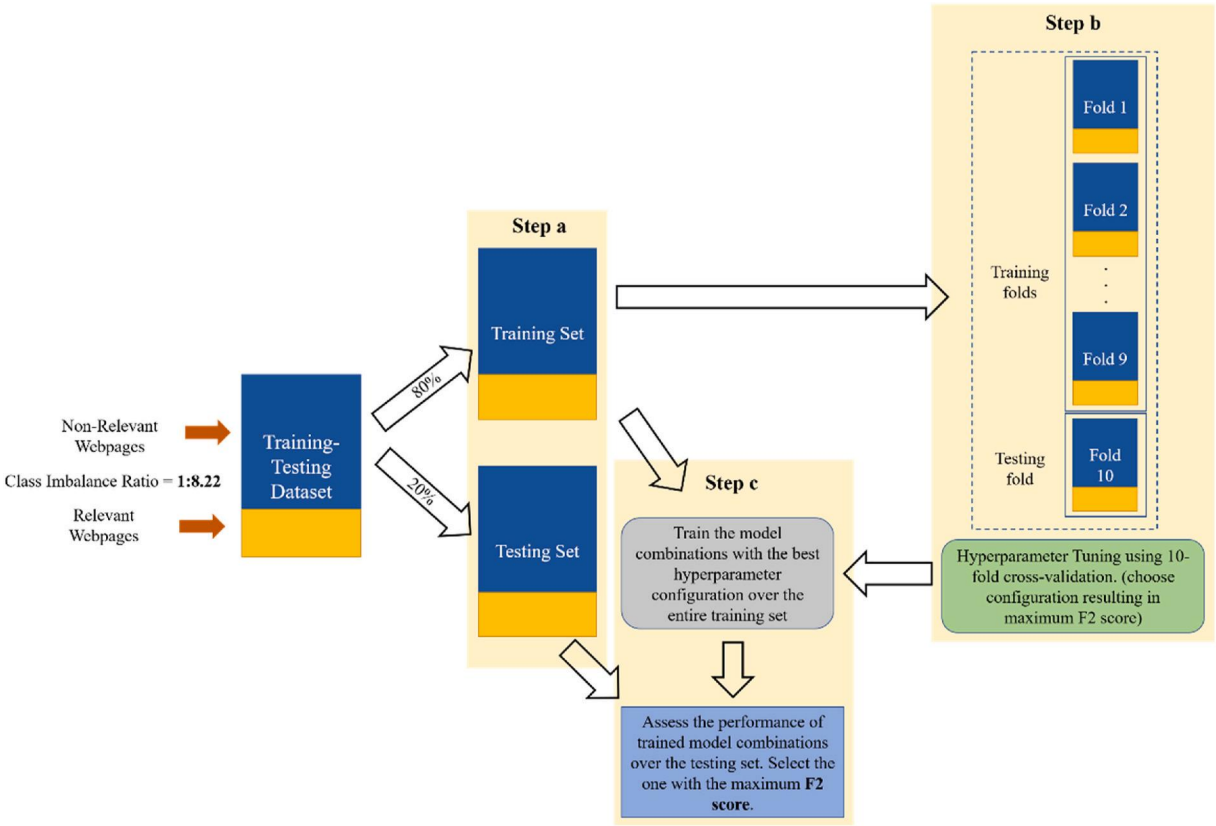


Fig. 5. BCML model selection procedure (We follow the same procedure for the other two splits).

c) Final Testing: Then, for each split ratio, we trained the selected model combinations initialized their tuned hyperparameter configurations over the entire training set and observed the F2-scores for the testing set. Finally, we selected the model with the best F2-score for the testing set.

While performing the cross-validation, we performed DS only on the training folds; and for the last step, we performed DS only on the training set. To ensure accurate results, we did not perform DS on the testing set/folds. We also maintained the same class imbalance ratio across the different stages.

5. Results

5.1. BCML model selection

We use a baseline model to put the performance of the ML models in perspective. This baseline model represents the unfiltered top 10 Google search results. For both the ML models and the baseline model, each performance metric (in Tables 5 and 6) is computed considering multiple search queries and multiple FF locations in the validation and testing sets.

Table 5 compares the average 10-fold cross-validation F2-scores for the different hyperparameter-tuned model combinations considered in this study. For brevity, we show only the average 10-fold cross-validation F2-scores for the 80:20 split. We follow the same procedure for the other two split ratios.

As shown in Table 5, the Gradient Boost + SVMSMOTE, Random Forest + SVMSMOTE, Random Forest + ENN, and Random Forest + SMOTETomek combinations outperform other combinations based on the average 10-fold cross-validation F2-scores. Therefore, we selected these combinations for the final testing (Step b in Fig. 5). Table 5 also highlights that the DS techniques help improve the performance of all ML models considered in this study. The average 10-fold cross-

Table 5
Average 10-fold cross-validation F2-scores (0–1 range) for different hyperparameter-tuned model combinations for the 80:20 split ratio.

DS Technique	Nm	Minority Class Oversampling (Min-O)				Majority Class Undersampling (Maj-U)				Min-O + Maj-U	
		SM	B-SM	SVM-SM	A-SYN	TL	ENN	CNN	NM	SM + T	SM + ENN
Baseline	0.38	0.38	0.38	0.38	0.38	0.38	0.38	0.38	0.38	0.38	0.38
RF	0.69	0.73	0.73	**0.74**	0.73	0.70	**0.74**	0.72	0.70	**0.74**	0.73
GB	0.68	0.72	0.71	**0.74**	0.71	0.73	0.71	0.71	0.67	0.72	0.71
DT	0.61	0.65	0.63	0.67	0.64	0.63	0.72	0.56	0.48	0.65	0.64
AB	0.60	0.70	0.69	0.71	0.68	0.62	0.72	0.63	0.56	0.70	0.68

Note: Abbreviations - (*Rows*) RF: Random Forest; GB: Gradient Boost; DT: Decision Tree; AB: Adaboost; (*Columns*) Nm: normal (without any data sampling); SM: Synthetic Minority Oversampling Technique (SMOTE); B-SM: Borderline SMOTE; SVM-SM: Support Vector Machine SMOTE; A-SYN: Adaptive Synthetic Sampling; TL: Tomek Links; ENN: Edited Nearest Neighbors; CNN: Condensed Nearest Neighbors; SM + T: SMOTE + Tomek Links; SM + ENN: Smote + ENN.

Table 6
BCML model selection.

Split Ratio	Model Combination	Accuracy	Precision	Recall	F2-Score
–	Baseline	0.108	0.108	1.000	0.377
80:20	Random Forest + SVMSMOTE	0.928	0.627	0.824	**0.775**
70:30	Random Forest + SVMSMOTE	0.921	0.600	0.833	0.773
70:30	Random Forest + SMOTEENN	0.928	0.628	0.819	0.772
75:25	Random Forest + SMOTE	0.928	0.629	0.817	0.770
75:25	Random Forest + SMOTEENN	0.931	0.642	0.811	0.770

Note: For the definition of these metrics, refer to Fig. 4. All metrics are expressed in a 0–1 range.

validation F2-scores for a particular ML model with any DS technique is higher than the corresponding F2-score for the same model without DS (column 'Nm' in Table 5).

Table 6 contains the final results of the BCML model selection. We compared the performances of model combinations selected from the average 10-fold cross-validation F2-scores comparisons (Table 5) for all split ratios when trained over the entire training set and tested over the testing set. For brevity, we show only the top 5 performing model combinations.

As evident from Table 6, we selected the Random Forest + SVMSMOTE combination when trained over an 80:20 training:testing split as the BCML model since it performed the best per the F2-score (0.775) over the testing set. The 'Baseline' achieved an F2-score of 0.377; hence, the proposed method improved the performance by **105.57%**. While FF-IR outperforms conventional search engines, the F2-score may not be viewed as exceptionally high. We attribute this to the fact that not all events have the same amount of information available in the web. This gives rise to circumstances where webpages returned for customized queries by Google search (for events with less information available) may not be relevant to the event of interest but be relevant to some other event. Such situations give rise to exceptions where FF-IR may need to classify a webpage as relevant for one flash flood event but non-relevant for all other events. Improved feature engineering that better captures the semantic relationship between events and their relevant webpages can help overcome this limitation.

5.2. Comparing Google Search and FF-IR results

Table 7 describes FF-IR's performance for ten FF events chosen randomly from events outside of the TT dataset. Out of the ten events, six caused no human harm (first six rows in Table 7), and the remaining four did (fatality or injury). We can draw two key observations from Table 7. First, FF-IR correctly identified most relevant webpages. This is evident by comparing columns 5 and 6 of Table 7 (No. of relevant webpages vs. No. of relevant webpages returned by FF-IR). Second, FF-IR filtered out the vast majority of non-relevant webpages returned by Google. This is evident in column 7 of Table 7 (No. of non-relevant webpages returned by FF-IR). The exception was an event that occurred on August 14, 2018 in Cochise County, Arizona. In this case, FF-IR retrieved four non-relevant webpages because they contain information about another FF event that occurred in the same county in the same month and year. This incorrect prediction is likely caused by the static feature "check_date". Overall, these results demonstrate FF-IR's ability to filter out non-relevant webpages and its utility in

Table 7
FF-IR performance statistics.

County	State	Date	No. of google returned webpages	No. of relevant webpages	No. of relevant webpages returned by FF-IR	No. of non-relevant webpages returned by FF-IR
Angelina	AZ	10.08.2018	31	3	3	0
Christian	MO	07/05/2015	21	3	1	0
Venango	PA	07/19/2019	16	1	1	0
Rowan	KY	08/31/2013	13	0	0	0
Sunflower	MS	03/31/2016	46	7	5	0
Tioga	NY	07/25/2018	30	1	0	0
Cochise	AZ	08/14/2018	37	7	7	4
Faulkner	AR	05/01/2011	80	2	1	1
Grant	NM	09/30/2017	53	2	2	0
Riverside	CA	02/14/2019	61	21	15	0

enhancing internet search.

FF-IR is designed to retrieve webpages containing relevant information about past flash flood (FF) events. Compared to conventional search engines, FF-IR system offers several advantages. First, it automates the retrieval process, which is crucial considering the large number of FF events reported in the USA every year. A manual search of webpages for all these events using conventional search engines is impractical, making the FF-IR's automated approach more efficient. Second, the system employs targeted search queries generated from a trusted database, reducing the user's effort in forming precise queries. Third, the IR system enhances the segregation of search results by verifying the relevance of webpages returned by Google search. This improved segregation is supported by FF-IR's higher F2-score and accuracy. Fourth, FF-IR does not require constant index updates as it uses Google's maintained index. Consequently, it reduces the effort needed to provide up-to-date search results. Collectively, these features demonstrate the effectiveness of the proposed IR system compared to traditional internet search engines in the flash flooding domain.

6. Summary and conclusions

This paper describes the development, verification, and utility of a domain-specific IR system for flash flood events in the United States. FF-IR incorporates a newly constructed ML model and automated search queries to enhance the retrieval of information about past flash flood events from webpages and web-documents. The emergence of artificial intelligence (AI)-based systems for internet search (like ChatGPT) highlights the importance of developing better internet search capabilities. FF-IR is a step forward toward AI-based search engines for the natural hazards and disasters domain. It uses a publicly available dataset (the SE data) to form targeted search queries in an automated manner, eliminating the trial and error searches used in conventional search engines. The targeted search queries are directed to Google to collect candidate webpages, avoiding further crawling, indexing, and ranking. A new BCML model was trained to classify each candidate webpage as either relevant or non-relevant. The BCML model was trained and tested on a domain-specific dataset explicitly developed for this study. FF-IR outperforms direct Google searches about FF events by over 100%, measured by the F2-score.

The main limitation of FF-IR lies in its reliance on the SE data to create customized search queries. A possible way to overcome this limitation could be to incorporate additional public data from different sources and levels (e.g., national, state, county, town/city) into this IR system to make it less dependent on the SE data. Another limitation is the use of a single search engine (Google) to fetch candidate webpages. Collecting candidate webpages from additional search engines like Microsoft Bing and DuckDuckGo, among others, can help address this limitation. Despite these limitations, FF-IR's multiple advantages make it a breakthrough in retrieving relevant information from the internet about past disaster events.

7. Ongoing and future work

Our ongoing work includes the dissemination of FF-IR to the public through a user-friendly web application. The users of the web application would be researchers and practitioners in the natural hazards community, as well as members of the general public who are interested in finding information about past FF events in the US. Future works could focus on addressing the limitations of FF-IR (identified earlier) and improve its performance. First, future studies could examine the computation time required by FF-IR and compare it to that required by existing conventional search engines. Such comparisons should take into consideration the differences in hardware systems that host conventional search engines and those that host FF-IR. Second, future work could explore the possibility of incorporating additional publicly available data, beyond the SE data, and sourcing candidate webpages from additional conventional search engines (such as Bing and DuckDuckGo). Third, FF-IR could be extended to include other natural hazards and disasters, such as earthquakes, heatwaves, and wildfires. Fourth, the current user-interface (under development) allows for search based on location (i.e., county and state) and keywords in the SE narratives. This search capability can be enhanced in future works to retrieve information on events with certain characteristics, without specifying locations. Finally, future studies could investigate new deep learning, NLP, and IR techniques as they emerge in the future to enhance FF-IR's performance continuously. However, these improvements will warrant the development of a new and more comprehensive TT dataset.

Software availability

Name: FF-IR (Flash Flood-Information Retrieval).
Developers: Rohan Singh Wilkho, Nasir G. Gharaibeh.
Contact: rohanswilkho_93@tamu.edu.
Year first available: 2023.
Hardware requirements: Any device with a web browser and Internet connection.
Availability: Will be made available as an open-access URL.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have linked the data used in the attached manuscript

Acknowledgments

Funding: This material is based upon work supported by the National Science Foundation under Grant No. 1931301. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Computing Resources: Experiments for this study were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing.

References

Anbarasan, M., Muthu, B.A., Sivaparthipan, C.B., Sundarasekar, R., Kadry, S., Krishnamoorthy, S., Samuel, R.D.J., Dasel, A.A., 2020. Detection of flood disaster system based on IoT, big data and convolutional deep neural network. Comput. Commun. 150, 150–157. https://doi.org/10.1016/j.comcom.2019.11.022.

Ashley, S.T., Ashley, W.S., 2008. Flood fatalities in the United States. J. Appl. Meteorol. Climatol. 47 (3), 805–818. https://doi.org/10.1175/2007JAMC1611.1.

Barker, J.L.P., Macleod, C.J.A., 2019. Development of a national-scale real-time Twitter data mining pipeline for social geodata on the potential impacts of flooding on communities. Environ. Model. Software 115, 213–227. https://doi.org/10.1016/j.envsoft.2018.11.013. November *2018*.

Batista, G.E., Bazzan, A.L., Monard, M.C., 2003. Balancing Training Data for Automated Annotation of Keywords: a Case Study. InWOB, pp. 10–18.

Batista, G.E.A.P.A., Prati, R.C., Monard, M.C., 2004. A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explorations Newsletter 6 (1), 20–29. https://doi.org/10.1145/1007730.1007735.

Breiman, L., 2017. Classification and Regression Trees. Routledge. https://doi.org/10.1201/9781315139470.

Brin, S., Page, L., 1998. The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst. 30 (1–7), 107–117. https://doi.org/10.1016/S0169-7552(98)00110-X.

Cer, D., Yang, Y., Kong, S.Y., Hua, N., Limtiaco, N., John, R.S., Kurzweil, R., 2018. Universal sentence encoder for English. In: Proceedings of the 2018 Conference *on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 169–174. https://doi.org/10.18653/v1/d18-2029. *Brussels, Belg*ium.

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. 16, 321–357. https://doi.org/10.1613/jair.953.

Chekalina, V., Panchenko, A., 2022. Retrieving comparative arguments using deep language models. CEUR Workshop Proceedings 3180, 3032–3040.
Dean, J., 2009. Challenges in building large-scale information retrieval systems. In: Keynote of the 2nd ACM Internat*ional Conference On Web Search And Data Mining (WSDM)*, vol. 10. https://doi.org/10.1145/1498759.1498761. *No. 1498759.1498761)*, Barcelona, Spain.
Dehghani, M., Zamani, H., Severyn, A., Kamps, J., Bruce Croft, W., 2017. Neural ranking models withweak supervision. In: SIGIR 2017 - Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 65–74. https://doi.org/10.1145/3077136.3080832.
Donratanapat, N., Samadi, S., Vidal, J.M., Sadeghi Tabas, S., 2020. A national scale big data analytics pipeline to assess the potential impacts of flooding on critical infrastructures and communities. Environ. Model. Software 133, 104828. https://doi.org/10.1016/j.envsoft.2020.104828.
Esteva, A., Kale, A., Paulus, R., Hashimoto, K., Yin, W., Radev, D., Socher, R., 2021. COVID-19 information retrieval with deep-learning based semantic search, question answering, and abstractive summarization. Npj Digital Medicine 4 (1). https://doi.org/10.1038/s41746-021-00437-0.
Freund, Y., Schapire, R.E., 1996. Experiments with a new boosting algorithm. In icml 96, 148–156. https://dl.acm.org/doi/10.5555/3091696.3091715.
Friedman, J.H., 2001. Greedy function approximation: a gradient boosting machine. Ann. Stat. 29 (5), 1189–1232. https://doi.org/10.1214/aos/1013203451.
Gatnar, E., 2002. Tree-based models in statistics: three decades of research. In: Jajuga, K., Sokołowski, A., Bock, H.H. (Eds.), Classification, Clustering, and Data Analysis. Springer Berlin Heidelberg, pp. 399–407. https://doi.org/10.1007/978-3-642-56181-8_44.
Google Interference, 2019. How Google interferes with its search algorithms and changes your results. Wall St. J. https://www.wsj.com/articles/how-google-interferes-with-its-search-algorithms-and-changes-your-results-11573823753 (accessed May 23, 2021).
Google Search, 2019. How Search Algorithms *Work. Google, 2–4*. Retrieved from. https://www.google.com/intl/en_uk/search/howsearchworks/algorithms/. May 23, 2021.
Gu, Q., Zhu, L., Cai, Z., 2009. Evaluation measures of the classification performance of imbalanced data sets. In: Cai, Z., Li, Z., Kang, Z., Liu, Y. (Eds.), Computational Intelligence and Intelligent Systems. Springer Berlin Heidelberg, pp. 461–471. https://doi.org/10.1002/bit.24634.
Gudiyangada Nachappa, T., Tavakkoli Piralilou, S., Gholamnia, K., Ghorbanzadeh, O., Rahmati, O., Blaschke, T., 2020. Flood susceptibility mapping with machine learning, multi-criteria decision analysis and ensemble using Dempster Shafer Theory. J. Hydrol. 590, 125275 https://doi.org/10.1016/j.jhydrol.2020.125275. November 2019.
Guo, Y., Ma, Z., Mao, J., Qian, H., Zhang, X., Jiang, H., Cao, Z., Dou, Z., 2022. Webformer: pre-training with web pages for information retrieval. In: SIGIR 2022 - Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1502–1512. https://doi.org/10.1145/3477495.3532086.
Guyon, I., Weston, J., Barnhill, S., Labs, T., Bank, R., 2013. Tracking cellulase behaviors. Biotechnol. Bioeng. 110 (1) https://doi.org/10.1002/bit.24634.
Han, H., Wang, W.-Y., Mao, B.-H., 2005. In: Huang, D.S., Zhang, X.P., Huang, G.B. (Eds.), Borderline-SMOTE: A New Over-sampling Method in Imbalanced Data Sets Learning, In *Advances In Intelligent Computing*. Springer Berlin Heidelberg, pp. 878–887. https://doi.org/10.1007/11538059_91.
Hao, H., Wang, Y., 2021. Assessing disaster impact in real time: data-driven system integrating humans, hazards, and the built environment. J. Comput. Civ. Eng. 35 (5), 1–17. https://doi.org/10.1061/(asce)cp.1943-5487.0000970.
Hart, P., 1968. The condensed nearest neighbor rule (corresp.). IEEE Trans. Inf. Theor. 14 (3), 515–516. https://doi.org/10.1109/TIT.1968.1054155.
Hashemi, M., 2020. Web page classification: a survey of perspectives, gaps, and future directions. Multimed. Tool. Appl. 79 (17–18), 11921–11945. https://doi.org/10.1007/s11042-019-08373-8.
He, H., Bai, Y., Garcia, E.A., Li, S., 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE International Joi*nt Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. *IEEE*, Hong Kong, China, pp. 1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969.
Hiemstra, D., 2009. Information retrieval models. In: Göker, A., Davies, J. (Eds.), Information Retrieval: Searching in the 21st Century. Wiley, pp. 1–19. http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470027622,descCd-description.html.
Hosseini, F.S., Choubin, B., Mosavi, A., Nabipour, N., Shamshirband, S., Darabi, H., Haghighi, A.T., 2020. Flash-flood hazard assessment using ensembles and Bayesian-based machine learning models: application of the simulated annealing feature selection method. Sci. Total Environ. 711, 135161 https://doi.org/10.1016/j.scitotenv.2019.135161.
Hudzina, J., Madan, K., Chinnappa, D., Harmouche, J., Bretz, H., Vold, A., Schilder, F., 2021. Information extraction/entailment of common law and civil code. In: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12758 LNAI. Springer International Publishing. https://doi.org/10.1007/978-3-030-79942-7_17.
Illingworth, N., 2001. The Internet matters: exploring the use of the Internet as a research tool. Socio. Res. Online 6 (2), 79–90. https://doi.org/10.5153/sro.600.
Khajwal, A.B., Cheng, C.S., Noshadravan, A., 2022. Post-disaster damage classification based on deep multi-view image fusion. Comput. Aided Civ. Infrastruct. Eng. https://doi.org/10.1111/mice.12890.
Khanmohammadi, S., Arashpour, M., Golafshani, E.M., Cruz, M.G., Rajabifard, A., Bai, Y., 2022. Prediction of wildfire rate of spread in grasslands using machine learning methods. Environ. Model. Software 156 (May), 105507. https://doi.org/10.1016/j.envsoft.2022.105507.
Kontokosta, C.E., Malik, A., 2018. The Resilience to Emergencies and Disasters Index: applying big data to benchmark and validate neighborhood resilience capacity. Sustain. Cities Soc. 36, 272–285. https://doi.org/10.1016/j.scs.2017.10.025. October 2017.
Landauer, T.K., Foltz, P.W., Laham, D., 1998. An introduction to latent semantic analysis. Discourse Process 25 (2–3), 259–284. https://doi.org/10.1080/01638539809545028.
Lashkari, F., Ensan, F., Bagheri, E., Ghorbani, A.A., 2017. Efficient indexing for semantic search. Expert Syst. Appl. 73, 92–114. https://doi.org/10.1016/j.eswa.2016.12.033.
Le, Q., Mikolov, T., 2014. Distributed representations of sentences and documents. In: International Conference on *Machine Learning*. PMLR, *Bei*jing, China, pp. 1188–1196. https://arxiv.org/abs/1405.4053.
Lewandowski, D., 2012. Credibility in web search engines. In: Online Credibility and Digital Ethos: Evaluating Computer-Mediated Communication, vols. 131–146. https://doi.org/10.4018/978-1-4666-2663-8.ch008.
Lin, J., 2021. A proposed conceptual framework for a representational approach to information retrieval. ACM SIGIR Forum 55 (2), 1–29. https://doi.org/10.1145/3527546.3527552.
Liu, T.-Y., 2010. Learning to Rank for Information Retrieval. https://doi.org/10.1145/1835449.1835676.
Liu, Z., Feng, J., Yang, Z., Wang, L., 2021. Document retrieval for precision medicine using a deep learning ensemble method. JMIR Medical Informatics 9 (6), e28272. https://doi.org/10.2196/28272.
Loynes, C., Ouenniche, J., De Smedt, J., 2022. The detection and location estimation of disasters using Twitter and the identification of Non-Governmental Organisations using crowdsourcing. Ann. Oper. Res. 308, 339–371. https://doi.org/10.1007/s10479-020-03684-8.
Manning, C.D., Raghavan, P., Schutze, H., 2012. Web Search Basics. *Introduction to Information Retrieval*. https://doi.org/10.1017/cbo9780511809071.020. *C. 385–404*.
Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013. Efficient estimation of word representations in vector space. In: 1st International Conference *on Learning Representations, ICLR 2013 - Workshop Track Proceedings*. *Scottsdale, A*rizona, USA, pp. 1–12. https://arxiv.org/abs/1301.3781.
Milly, P.C.D., Wetherald, R.T., Dunne, K.A., Delworth, T.L., 2002. Increasing risk of great floods in a changing climate. Nature 415 (6871), 514–517. https://doi.org/10.1038/415514a.
Nallapati, R., 2004. Discriminative models for information retrieval. In: Proceedings of the 27th Ann*ual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 64–71. https://doi.org/10.1145/1008992.1009006. *Sheffield, U*nited Kingdom.
Nguyen, H.M., Cooper, E., Kamei, K., 2011. Borderline over-sampling for imbalanced data classification. Int. J. Knowl. Eng. Soft Data Paradigms 3, 4–21. https://doi.org/10.1504/IJKESDP.2011.039875.
NOAA Storm Events Database, 2021. Search Results for All U.S. States and Areas, Event Types: Flash Flood [Available from: https://www.ncdc.noaa.gov/stormevents/, 2022-August-23.
Nogueira, R., Yang, W., Cho, K., Lin, J., 2019. Multi-Stage Document Ranking with BERT. http://arxiv.org/abs/1910.14424.
Ogie, R.I., Verstaevel, N., 2020. Disaster informatics: an overview. Progress in Disaster Science 7, 100111. https://doi.org/10.1016/j.pdisas.2020.100111.
Pang, L., Lan, Y., Guo, J., Xu, J., Xu, J., Cheng, X., 2017. DeepRank: a new deep architecture for relevance ranking in information retrieval. International Conference on Information and Knowledge Management, Proceedings, Part F131841, 257–266. https://doi.org/10.1145/3132847.3132914.
Ramanan, N., Natarajan, S., 2020. Causal learning from predictive modeling for observational data. Frontiers in Big Data 3 (October), 535976. https://doi.org/10.3389/fdata.2020.535976.
Reimers, N., Gurevych, I., 2020. Sentence-BERT: sentence embeddings using siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, pp. 3982–3992. https://doi.org/10.18653/v1/d19-1410. Hong Kong, China.
Robertson, S., Zaragoza, H., 2009. The probabilistic relevance framework: BM25 and beyond. In Foundations and Trends in Information Retrieval 3 (4). https://doi.org/10.1561/1500000019.
Roh, Y., Heo, G., Whang, S.E., 2018. A survey on data collection for machine learning: a big data – AI integration perspective. In: IEEE Transactions on Knowledge and Data Engineering, pp. 1–20. https://doi.org/10.1109/TKDE.2019.2946162.
Romero, S., Becker, K., 2019. A framework for event classification in tweets based on hybrid semantic enrichment. Expert Syst. Appl. 118, 522–538. https://doi.org/10.1016/j.eswa.2018.10.028.
Sarker, M.N.I., Peng, Y., Yiran, C., Shouse, R.C., 2020. Disaster resilience through big data: way to environmental sustainability. Int. J. Disaster Risk Reduc. 51 (August), 101769 https://doi.org/10.1016/j.ijdrr.2020.101769.
Shao, Y., Liu, B., Mao, J., Liu, Y., Zhang, M., Ma, S., 2020. THUIR@COLIEE-2020: Leveraging Semantic Understanding and Exact Matching for Legal Case Retrieval and Entailment, p. 61732008. http://arxiv.org/abs/2012.13102.
Tanner, A., Friedman, D.B., Koskan, A., Barr, D., 2009. Disaster communication on the internet: a focus on mobilizing information. J. Health Commun. 14 (8), 741–755. https://doi.org/10.1080/10810730903295542.
Terti, G., Ruin, I., Gourley, J.J., Kirstetter, P., Flamig, Z., Blanchet, J., Arthur, A., Anquetin, S., 2019. Toward probabilistic prediction of flash flood human impacts. Risk Anal. 39 (1), 140–161. https://doi.org/10.1111/risa.12921.

Tomek, I., 1976. An experiment with the nearest-neighbor rule. In: IEEE Transactions on Systems, *Man, and Cybernetics*, pp. 448–452. https://doi.org/10.1109/TSMC.1976.4309523. SMC-6, 6.

Ullah, I., Khan, S., Imran, M., Lee, Y.K., 2021. RweetMiner: automatic identification and categorization of help requests on twitter during disasters. Expert Syst. Appl. 176 (February), 114787 https://doi.org/10.1016/j.eswa.2021.114787.

Wang, Z., Ng, P., Nallapati, R., Xiang, B., 2021. Retrieval, re-ranking and multi-task learning for knowledge-base question answering. EACL 2021 - 16th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference 347–357. https://doi.org/10.18653/v1/2021.eacl-main.26.

Wilkho, R., Gharaibeh, N., Chang, S., Zou, L., 2022. Flash Flood Information Retrieval System (ML Dataset).” *DesignSafe-CI*. https://doi.org/10.17603/ds2-rwg3-v337.

Wilkho, R., Gharaibeh, N., Chang, S., Zou, L., 2023. Flash Flood Information Retrieval System (Pseudocodes). https://doi.org/10.17603/ds2-ed3t-b759. *DesignSafe-CI*.

Wilson, D.L., 1972. Asymptotic properties of nearest neighbor rules using edited data. IEEE Transactions on Systems, Man, and Cybernetics SMC-2 (3), 408–421. https://doi.org/10.1109/TSMC.1972.4309137.

Wu, J., Zhang, X., Zhu, Y., Liu, Z., Guo, Z., Fei, Z., Lai, R., Wu, Y., Cao, Z., Dou, Z., 2022. Pre-training for Information Retrieval: Are Hyperlinks Fully Explored? http://arxiv.org/abs/2209.06583.

Yilmaz, Z.A., Wang, S., Yang, W., Zhang, H., Lin, J., 2019. Applying BERT to document retrieval with birch. EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Proceedings of System Demonstrations 19. https://doi.org/10.18653/v1/d19-3004. –24.

Yu, M., Yang, C., Li, Y., 2018. Big data in natural disaster management: a review. Geosciences 8 (5). https://doi.org/10.3390/geosciences8050165.

Zhao, F., Meng, X., Zhang, Y., Chen, G., Su, X., Yue, D., 2019. Landslide susceptibility mapping of karakorum highway combined with the application of SBAS-InSAR technology. Sensors 19 (12), 2685. https://doi.org/10.3390/s19122685.

Zheng, L., Shen, C., Tang, L., Zeng, C., Li, T., Luis, S., Chen, S.C., 2013. Data mining meets the needs of disaster information management. IEEE Transactions on Human-Machine Systems 43 (5), 451–464. https://doi.org/10.1109/THMS.2013.2281762.