Expert Systems With Applications 210 (2022) 118461

Contents lists available at ScienceDirect

Expert Systems With Applications

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

GeoClust: Feature engineering based framework for location-sensitive disaster event detection using AHP-TOPSIS

Monika Rani [1,*], Sakshi Kaushal [2]

University Institute of Engineering & Technology, Panjab University, Chandigarh, India

ARTICLE INFO

Keywords:
Location-sensitive disaster event detection
Feature engineering
Multiple-criteria decision making (MCDM)
AHP-TOPSIS
Context-free and context-based feature sets

ABSTRACT

Disaster event detection aims to identify events like terrorist attacks, fire incidents, stampede incidents, building collapse, *etc.*, reported in the online news articles or social media. Place of occurrence of disaster event is a significant feature associated with events for location-sensitive disaster event detection. Efficient feature selection and their augmentation with location information can contribute towards the evolution of traditional approaches and their adoption for location-sensitive disaster event detection leading to improvement in the overall process as a whole. Since the evaluation of event detection techniques deliberates various intrinsic and extrinsic performance metrics, the decision-making for the selection of feature sets is treated as a Multiple-Criteria Decision Making (MCDM) problem. This paper proposes a framework, *GeoClust*, that is based on feature engineering of traditional textual features in order to enhance their capability for improved location-sensitive disaster event detection. The framework augments context-free and context-based textual feature sets with feature sets of place of occurrence of the events and evaluates their performance using unsupervised machine learning algorithms for various performance metrics. Finally, the best feature set is selected using AHP-TOPSIS technique of MCDM in order to tune the system for automatic and efficient location-sensitive disaster event detection in real-time. Extensive set of experiments have been performed in order to evaluate the framework on a dataset of online news articles reporting disaster events about terrorist attacks, fire incidents, stampede incidents, building collapse and maoist attacks happened at different locations in India. The results show that the location-augmented feature sets significantly improve performance of location-sensitive disaster event detection as compared with traditional feature sets. The results also demonstrate that the context-based feature sets with location-augmentation are ranked higher than the context-free feature sets in MCDM analysis.

1. Introduction

With the advent of growing digitized information, online event detection has been an active area of research for the last two decades. Event detection involves association of terms from structured or unstructured text of online documents. In literature, event detection has been carried out from various sources of online information, e.g., social networking websites, forums, posts on review sites, micro blog posts, travel narratives, life stories, historic articles, online news articles and huge collaborations on Wikipedia. Among these, online news articles have been considered as the most authentic sources of information about events happening around the world. Since the news articles describe similar events in analogous terms, event detection and distinguishing similar events happening at different places becomes crucial. Location-sensitive event detection distinguishes similar types of events that have happened at different locations e.g. *"building collapse in Delhi"* is a different event from *"building collapse in Kolkata"*. Alencar *et al.* specified that, for event detection, establishing relationships among documents and places of occurrence of events reported in the documents is an essential and critical task (Odon De Alencar et al., 2010). As each event is associated with the place of its occurrence, use of location information of events can simplify the overall process of location-sensitive event detection, in particular, and research in this domain can contribute towards evolution of traditional approaches to event detection and improve retrieval accuracy.

Since data on the web is multiplying dramatically and most of this

* Corresponding author.
E-mail addresses: monikaubs@pu.ac.in (M. Rani), sakshi@pu.ac.in (S. Kaushal).
[1] ORCID ID: 0000-0002-8431-4300.
[2] ORCID ID: 0000-0002-1902-3486.

https://doi.org/10.1016/j.eswa.2022.118461
Received 21 December 2021; Received in revised form 16 July 2022; Accepted 5 August 2022
Available online 9 August 2022

data is in unstructured form without any explicit meaning or machine-readable semantics, automatic interpretation of unstructured data and conversion to some structured format/summary is desirable. Thus, machine learning algorithms are adopted for automatic processing of huge amounts of data and event detection. Due to non-availability of comprehensive training datasets, unsupervised machine learning algorithms are commonly applied for event detection from online web. These algorithms are capable of learning the patterns and associations among data automatically without the explicit need of specific programming. But the performance of machine learning algorithms relies highly on features extracted from documents. Hence, efficient feature selection is the foremost crucial step for effective implementation of event detection based on machine learning algorithms. The performance of unsupervised machine learning algorithms can be evaluated for various metrics *viz.* Homogeneity, Completeness, V-Measure, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient, Calinski-Harabasz Index, Davies-Bouldin Index and Clustering Time. Thus the selection of the most efficient feature set for event detection can be achieved from Multiple-Criteria Decision Making (MCDM).

This paper proposes a framework based on feature engineering for location-sensitive disaster event detection, *namely, GeoClust*, that distinguishes similar kinds of disaster events that have happened at different locations. Due to non-availability of comprehensive labelled training datasets for location-sensitive events and diversity of text in online news articles, the proposed framework derives the intelligence of learning from the features extracted from unstructured data automatically by implementing unsupervised machine learning algorithms. The context-based feature sets are extracted from the unstructured text in order to learn the context of the words in the dataset automatically. The traditional context-free and context-based features are augmented with feature sets of place of occurrence of events. Extensive set of experiments and analytic analysis is conducted on a dataset of online news articles reporting 72 disaster events about terrorist attacks, fire incidents, stampede incidents, building collapse and maoist attacks happened at different locations in India. The performance of traditional context-free and context-based textual feature sets augmented with feature sets of place of occurrence of events is evaluated for location-sensitive event detection using unsupervised machine learning algorithms. Finally, the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) (Hwang & Yoon, 1981) has been utilized for ranking the feature sets with weight assignment to each performance metric using Analytic Hierarchy Process (AHP) (Saaty, 1984). This research aims at analyzing the applicability of context-free, context-based features sets and impact of augmentation of location features for location-sensitive disaster event detection keeping in view to answer the following research questions:

RQ1. Are textual features efficient enough for location-sensitive disaster event detection?
RQ2. Does integration of location features can improve the retrieval accuracy for location-sensitive disaster event-detection.
RQ3. How feature selection impacts the retrieval accuracy of the overall event detection process?

The key highlights of the research work are as follows:

(i) A novel feature engineering based framework is proposed for location-sensitive disaster event detection that distinguishes the same kind of events but happened at different places.
(ii) The framework extracts state-of-the-art context-free and context-based feature sets from a dataset of online news articles reporting terrorist attacks, fire incidents, stampede incidents, building collapse and maoist attacks.
(iii) The extracted feature sets are augmented with feature vectors of place of occurrence of events reported in the news articles. Though location feature has been widely used with context-free feature sets in literature, use of place of occurrence and augmentation of context-based feature sets with place of occurrence feature set signifies the framework as unique in one of its kind.
(iv) At preliminary step, AHP-TOPSIS is applied to select from widely-known unsupervised machine learning algorithms on the basis of their performance with unigram feature sets extracted from news articles.
(v) The traditional feature sets and their corresponding location-augmented versions are evaluated for location-sensitive disaster event detection using the selected algorithm. The results of performance metrics are ranked using AHP-TOPSIS in order to obtain the most efficient feature set for location-sensitive event detection.

Remainder of this paper is structured as follows.

Section 2 discusses previous research and studies focusing on either location-sensitive event detection or using location for event detection tasks. Section 3 outlines the workflow of the proposed framework and briefs preliminary concepts used in it. Section 4 is devoted to detailed experimental analysis and results discussion. Section 5 gives practical applications of the proposed framework for location-sensitive disaster event detection. Finally, Section 6 concludes the paper and lists the scope of future work and improvements in the area.

2. Related work

This paper analyses performance of various traditional context-free and context-based feature sets and their location augmented versions using unsupervised machine learning algorithms thereby selecting the most efficient one for location-sensitive disaster event detection. The extraction and establishing association of location of an event is a crucial step when dealing with location-sensitive event detection. Accordingly, a brief outline of the existing works, that had focused on location-sensitive event detection or utilized location for event detection, is given in the next subsection followed by the discussion of state-of-the-art textual features pertinent to event detection. Table 1 gives a brief analysis of these studies in terms of dataset and feature sets used, type of feature set, how location has been utilized, method applied for event detection and area of event detection.

2.1. Location-Sensitive event detection

An event is defined as a real-time happening of something associated with a specific location and point in time. With the increasing influence of location-sensitive event detection, researchers used location as a potential feature for event detection. The location-sensitive event detection system targets the “where” question in the basic five ‘W’ information-gathering questions (who, what, when, where, why). Location features act as a refinement for the same kind of events happening at different locations, thereby, improving accuracy and efficiency in storytelling and event tracking activities. While accomplishing to identify location associated with the event happened, meanwhile, it achieves the first step towards event detection. On the basis of method of utilizing location, the existing research on location-sensitive event detection can be mapped into following three categories:

2.1.1. Formation of Location-based clusters

One standard approach for event detection is by forming clusters of documents that are similar in keywords or whose term vectors are closely related. Apart from creating clusters of words, documents are geographically clustered first and then term-based features are used for further event detection. For instance, Visheratin *et al.* carried out local, city-level and country-level event detection by creating convolutional quad-trees of spatial distribution of data (Visheratin et al., 2018). Authors divided data into 576 datasets based on a unique timestamp and

Table 1
Analysis of studies using location for event detection.

Authors	Dataset used	Feature Set used	Type of feature Set	Usage of location	Applied method for event detection	Area of event detection
(Smith, 2002)	Perseus Digital Library	Statistics of occurrence of location and time at sentence and paragraph level	Location, Time	Establishing association between event and location	Spatio-temporal association of events, Ranking events with statistical measures	Spatio-temporal event detection
(Kumaran & Allan, 2004)	TDT3, TDT4 News Corpus	Tf-idf, Named-entities	Context-free, Location	Location as a feature	BoosTexter Classifier, Cosine similarity	New event detection
(Li et al., 2005)	TDT4 News Corpus, News Articles	Bag-of-Words, Named-entities	Context-free, Location	Location as a feature	Expectation Maximization (EM) algorithm	Event detection
(Cybulska & Vossen, 2010)	Srebrenica Corpus	Named-entities	Context-free, Location, Time	Establishing association between event and location	Kyoto platform	Spatio-temporal event detection
(Pan & Mitra, 2011)	Subset of TDT3 News Corpus, Reuters News Corpus	Bag-of-Words, Location, Time	Context-free, Location, Time	Location-based cluster formation	Topic modelling, K-means clustering algorithm	Spatio-temporal event detection
(Bsoul et al., 2013)	Bernama News of Crimes	Tf-idf, Named-entities, Noun, Verb	Context-free, Location	Location as a feature	Affinity propagation algorithm	Crime pattern detection
(Heravi et al., 2014)	–	Location, Frequency of burst keywords, Text vector	Context-free, Location	Location-based cluster formation	Natural language processing, Ontology-based IE	Location-sensitive event detection
(Cheng & Wicks, 2014)	Tweets for 2013 London Helicopter Crash.	Bag-of-Words, Location, Date	Context-free, Location, Time	Location-based cluster formation	Space-time scan statistics, Latent dirichlet allocation	Spatio-temporal clustering of events
(Ceroni et al., 2015)	Webpages for events reported in Wikipedia	Named-entities, Statistics of named-entities	Context- free, Location	Location as a feature	Feature Extraction, Support vector machine	Event detection
(Liu et al., 2016)	Civil Unrest Dataset, Tweet Dataset on Earthquake and Influenza Outbreaks	Word vectors, Vocabulary, Location	Context-free, Location	Location as a feature	Scan statistic, Subgraph detection and Topic modelling	Spatial event detection
(Edouard et al., 2017)	First Story Detection (FSD), EVENT2012 Tweet Corpus	Tf-idf, Named-entities	Context-free, Location	Location as a feature	Graph partitioning, Page rank algorithm	Event detection
(Robindro et al., 2017)	BBC News Articles	Tf-idf, Location	Context-free, Location	Location as a feature	K-means clustering, Recommender Model	Personalized Event Detection
(Alsaedi et al., 2017)	Tweets for Riots in England August 2011, Middle East 2015	Temporal Tf-idf,	Context-free, Location, Time	Location as a feature	Naïve Bayes Classifier, Online Clustering	Spatio-temporal event detection
(Valentin et al., 2018)	442 News Reports on Disease Outbreaks	Thematic features, Location, Date	Context-free, Location, Time	Location as a feature	Matrix fusion, Cosine similarity	Spatio-temporal disease surveillance
(Visheratin et al., 2018)	Geo-tagged Posts from Instagram	Time, Location, Statistics of posts	Context-free, Location	Location-based cluster formation	Convolutional neural networks with quadtree	Spatio-temporal event detection
(Huang et al., 2018)	Tweets from four College Cities in the U.S	Term frequency, Location, Time	Context-free, Location, Time	Location-based cluster formation	ST-DBSCAN, Latent dirichlet allocation	Spatio-temporal event detection
(Zhang et al., 2018)	Geo-tagged Tweets in New York and Los Angeles for four months	Tf-idf, Location, Time	Context-free, Location, Time	Location-based cluster formation	Geographical Keyword Graph random walk	Spatio-temporal event detection
(Wang et al., 2019)	Mongolian News	Tf-idf, Location, Time	Context-free, Location, Time	Location as a feature	Bi-LSTM + CRF model, Similarity calculation	Spatio-temporal event detection
(Rasouli et al., 2019)	Persian News Webpages. Log Files	*N*-grams, Named-entities, Click Frequency	Context-free, Location	Location as a feature	Kleinberg's burst event detection, KeyGraph	Event detection
(Liu et al., 2020)	Chinese News, 20Newsgroup English Dataset	Tf-idf, Named-entities	Context-free, Location	Location as a feature	Keyword Co-occurrence graph, Clustering	Event Detection & Organization
(Bendimerad et al., 2021)	Tweets collected New York (NYC), Los Angeles (LA) and London	Tf-idf	Context-free, Location, Time	Establishing association between event and location	Pattern discovery, Keyword Graph random walk	Spatio-temporal event detection
(Yasmeen et al., 2021)	Geo-tagged Tweets of Melbourne users, Geo-tagged Flickr Photos for Melbourne, London, New York and Paris	Location, Time	Location, Time	Location-based cluster formation	Spatial quadtree, Poisson distribution	Spatio-temporal event detection
(Choi et al., 2021)	Tweets for Fires, Typhoons, COVID-19, Car Accidents, Earthquakes, Floods *etc.*	Tf-idf, Location	Context-free, location	Establishing association between event and location	Keyword Graph Clustering	Location-sensitive event detection

built a convolutional tree of data of particular location at a point in time. The cells in geo-grids which were above a threshold value were selected for further graph-based event detection. Heravi *et al.* created geo-location clusters of news streams from various sources (Heravi et al., 2014). The geo-clusters were analysed for burst event detection in order to detect breaking news to help journalists. Pan and Mitra performed topic modelling of documents and then separated topics in time dimension (Pan & Mitra, 2011). Finally, they created geo-spatial clusters of events using the K-means algorithm based on geographic distance of the documents. In another study, location-based clusters of tweets were formed by applying space–time scan statistics and further tweets in each cluster were classified into groups of topics (Cheng & Wicks, 2014).

Huang *et al.* formed spatio-temporal clusters of tweets and applied Latent Dirichlet Allocation (LDA) for topic modelling from the word frequencies (Huang et al., 2018). GeoBurst+ system as illustrated in (C. Zhang et al., 2018) detected local events from real-time geo-tagged tweet streams. The authors first formed the geo-topical clusters of geo-tagged streams and later on updated them with more and more tweets

posted in real-time. They observed semantic relationships among the keywords in the tweets posted by creating a keyword co-occurrence graph and applying a random walk on the graph. Yasmeen *et al.* used a rectangular spatial grid to cluster geo-tagged tweets posted for that region with a sliding window of time (Yasmeen et al., 2021). However, they didn't utilize text features to learn semantics of the text for event detection.

2.1.2. Establishing association between event and location

Another straightforward approach for location-sensitive event detection is identifying relations among event participants and locations. Cybulska and Vossen defined an event model in order to identify relation between event descriptions in terms of actor participants, time and location (Cybulska & Vossen, 2010). In another research by David A. Smith (Smith, 2002), for the purpose of detection of historical events for the Perseus Digital Library Project, the author established association among time-location pairs at sentence and paragraph level in text documents. They applied various association measures on a contingency table containing statistics about these associations.

Bendimerad *et al.* proved that non-location aware methods are not efficient for the detection of location-sensitive events (Bendimerad et al., 2021). The authors created co-occurrence graphs of terms present in geo-located tweets and located events by identifying the patterns for the pairs of location, time and a set of terms. Finally, an event summary was created for the events that were above a specified minimum threshold. Due to non-availability of geo-tagged posts, Choi *et al.* associated the extracted locations of tweets with their keyword graph and performed keyword graph clustering to detect local events (Choi et al., 2021).

2.1.3. Location as a feature

Besides above approaches, a prominent machine learning approach for location-sensitive event detection is extracting location features from structured and unstructured content of text and employing these to train supervised or unsupervised classifiers to carry out event detection. *Li et al.* represented events in news articles in the form of tuples < person, locations, keywords, time> (Li et al., 2005). They built generative models of news events and learnt model parameters through Expectation Maximization (EM) algorithm and deployed the system for retrospective event detection from news. Ceroni *et al.* devised a supervised event detection model learnt from named-entity features occurring in free text, *namely,* person, location, organization name and objects (Ceroni et al., 2015). Basic statistics of these features were used to describe an event as unique. In order to improve the event detection process, events thus identified were validated at document and corpus level. Alsaedi *et al.* used clustering methods with feature selection to detect disruptive events from real-time tweets (Alsaedi et al., 2017). At first, the authors separated event related posts from non-event posts using a naïve bayes classifier followed by clustering of documents based on textual, spatial and temporal features.

The location feature has been utilised for event detection in other innovative ways also. Kumaran and Allan highlighted the utility of location and individual features in new event detection by implementing a document model with multiple document representation and similarity identification (Kumaran & Allan, 2004). Edouard *et al.* extracted events using graphs with person, organization and location information to detect whether a new event detected is duplicate and merged the duplicate events with the previously detected events (Edouard et al., 2017).

Robindro *et al.* developed a news recommendation system based on the fact that the users are more interested in local news (Robindro et al., 2017). The authors used K-means clustering to group similar news documents from different sources and then applied a news recommender model integrated with location information so as to recommend local news to the user. Valentin *et al.* utilized location and time features of news articles along with thematic features in order to build a disease surveillance system (Valentin et al., 2018). Wang *et al.* used text and named-entity features to develop a new event detection system from news articles (Wang et al., 2019). They applied a Bi-LSTM and CRF model to extract named-entities from the main content of the news article and calculated similarity among named-entities in order to carry out new event detection.

Rasouli *et al.* developed a graph-based event detection framework that employed burst keywords and click frequency to identify a set of news documents containing a specific term (Rasouli et al., 2019). In the final stage, they extracted people and places information for accurate event detection. Liu *et al.* detected events from news stories and organized stories to form a story forest (Liu et al., 2020). Authors identified named-entities, tf-idf of terms and semantic keywords related to events from the articles and built a keyword co-occurrence graph. They performed topic clustering on the graph using a community detection algorithm to detect topics in the documents followed by event clustering through a trained SVM classifier.

Most of the above studies relied on datasets from social media and focused on current event detection. As per a recent study, only 0.85–3% of daily tweets are geo-tagged, which constrains the performance evaluation of location-sensitive event detection on tweets (Li et al., 2017; Sloan & Morgan, 2015). Since social media posts are not reliable sources of information, hence, the proposed study uses a dataset of news articles being the most authenticated and self-contained source of information.

Despite the significance of feature extraction in machine learning algorithms, the existing studies hadn't evaluated performance and impact of different feature sets for location-sensitive event detection. Moreover, context-based feature sets *viz.* Word2Vec and Doc2Vec that learn semantics and associations among terms automatically; and their location-integrated versions have not been considered and evaluated in the previous studies. Though in above discussed studies, location information has been used with context-free feature sets, use of place of occurrence and its augmentation with context-based feature sets had remained unexplored. This paper aims at a) exploring performance of various context-free and context-based feature sets, b) impact of augmentation of the context-free and context-based feature sets with feature sets of place of occurrence of events c) selection of optimal feature set using AHP-TOPSIS for location-sensitive disaster event detection.

2.2. Feature extraction for event detection

Textual features can be broadly categorized as context-free and context-based features (Fig. 1). Context-free features account for "what is written" while context-based features deal with "what is written in what context". A brief outline of the feature sets implemented in the proposed framework is presented in the following subsection.

2.2.1. Context-free features

Context –free features assume all words in the document as independent of each other. They do not preserve word co-occurrence statistics or word ordering in the document, *i.e.,* in what context a word has been written. Despite being an important intricacy of document representation, these features loose context information that is embedded in the document. Tf-idf and *n*-gram are widely applied context-free feature sets for event detection. Several studies have applied tf-idf and *n*-gram to extract feature sets for event detection. In addition, researchers have applied other context-free features such as various named-entities extracted from text for event detection tasks.

Kumaran and Allan experimented with multiple document representations consisting of a) tf-idf vectors of all term features, b) only named-entity features and c) only non-named-entity features contained in the documents (Kumaran & Allan, 2004). Li *et al.* used a mixture of unigram feature sets to represent keywords, person and location entities in the text (Li et al., 2005). Cybulska and Vossen applied semantic classification of terms in order to identify and annotate terms associated with participants, time and location and used statistics of these terms to

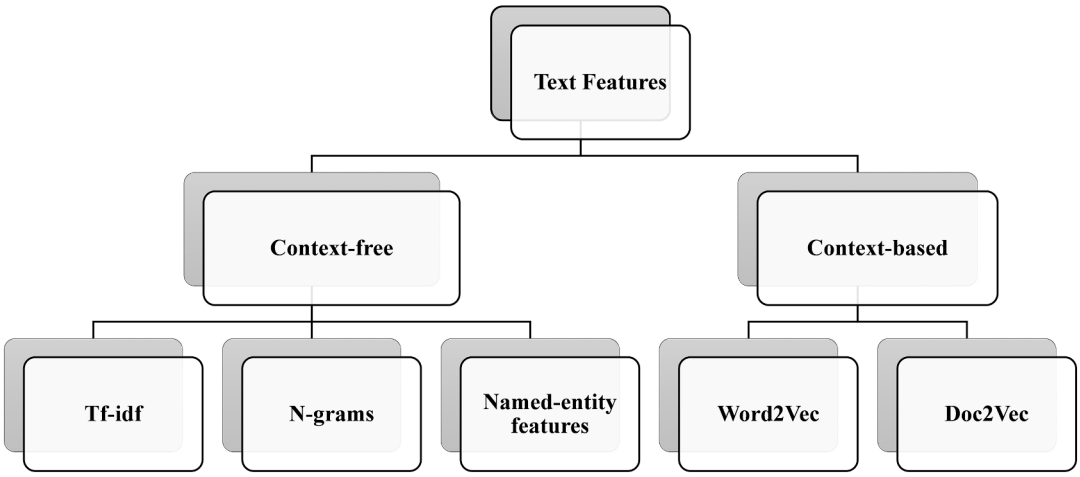

Fig. 1. Traditional text features applicable for event detection.

determine the relations among terms and events (Cybulska & Vossen, 2010). Bsoul *et al.* extracted named entities, nouns and verbs in order to identify location, names, topics associated with crime events and used tf-idf vectors for feature set representation (Bsoul et al., 2013). Heravi *et al.* recorded frequency of burst keywords in location-based clusters of the documents and explored text vector similarity and co-occurrence of location, named-entities and events in order to detect location-sensitive events (Heravi et al., 2014). Cheng and Wicks used bag-of-words feature sets and identified topics in spatio-temporal clusters of tweets (Cheng & Wicks, 2014). Y. Liu *et al.* applied locations and word vectors of documents for spatial graph formulation and topic distribution over a subgraph (Y. Liu et al., 2016). Edouard *et al.* utilized tf-idf vectors and named-entities for graph-based event extraction from tweets (Edouard et al., 2017). Rasouli *et al.* extracted *n*-grams from title and summary of news articles and utilized named-entities contained in the text so as to improve event detection accuracy (Rasouli et al., 2019).

An event can be represented by tuples < person, location, organization, time > from the named-entity information pieces embedded in the text. David A. Smith utilized date and place entities present in sentences and computed statistics of sentences that contain the same < date, place > pairs (Smith, 2002). Pan and Mitra represented documents using bag-of-words feature sets of terms, locations and timestamps contained in the document (Pan & Mitra, 2011). Ceroni *et al.* expressed events as a tuple < keywords representing named-entities, timespan> (Ceroni et al., 2015). Such a representation of events is capable of uniquely detecting events from a text collection.

2.2.2. *Context-based features*

Context-based features represent text of a document in a latent d-dimensional vector space on the basis of underlying context. In this vector space, the vectors which are close together have similar meanings, and vectors, which are far apart, have different meanings. Mikolov *et al.* and Le *et al.* proposed two context-based embedding models Word2Vec and Doc2Vec (Le & Mikolov, 2014; Mikolov et al., 2013). The models measure syntactic and semantic relationships among words and their surrounding context and, thus, are valuable in text mining processes based on word and context learning. For the purpose of event detection, Word2vec and Doc2vec have attracted a lot of attention from researchers recently.

Hu *et al.* utilized Word2Vec and Doc2Vec embeddings for text representation of news streams in order to eliminate the limitations possessed by tf-idf feature sets (Hu et al., 2017). Jang *et al.* applied Word2Vec Skip-Gram and CBOW models with convolutional neural networks for the classification of news articles and tweet streams (Jang et al., 2019). Repp and Ramampiaro used Word2Vec, Paragraph2Vec and Glove embedding models so as to identify news events from clusters of tweets (Repp & Ramampiaro, 2018). Despite above studies using the context-based feature sets, integration of location with these embedding models has remained unexplored so far.

In studies discussed in Section 2.2.1 and 2.2.2, traditional feature sets have been widely applied for event detection nevertheless they have pros and cons being two sides of the same coin leading to a subjective trade-off. For instance, term-frequency based feature vectors are easy and fast to extract, but with increase in vocabulary set, they will grow in size and take more response time. The vocabulary set in case of the news articles is quite big which impacts the length of the tf-idf features. As the size of content varies from one article to another, the predetermined vector size is more desirable. Secondly, the term-frequency feature sets may not be able to distinguish similar types of events but happened at different locations which necessitates the use of the location feature. Moreover, the context-free feature sets loose context information present in the text content. Though extraction of context-based feature sets is complex, their positive side is that they learn from the context information of the terms in text content that helps the context-based models for analyzing term associations present in various documents. In addition, the size of the feature vector can be fixed a priori, no matter how long the article is. Due to this trade-off, this paper aims at analyzing the effect of feature sets on the performance of the location-sensitive disaster event detection which forms the basis of the research question, *RQ3*. The applicability and efficiency of context-free, context-based feature sets and the impact of their augmentation with feature sets of place of occurrence of events on the performance of location-sensitive disaster event detection is evaluated through a series of experiments and analytical analysis using AHP-TOPSIS as given in Section 4.

3. Proposed work

In this section, the proposed framework based on feature engineering for location-sensitive disaster event detection, *namely,* GeoClust has been discussed that augments state-of-the-art textual features with place of occurrence of events and employs selection of unsupervised machine learning algorithms and feature sets by applying AHP-TOPSIS. The objective of this work is to increase efficiency of text features using location information so as to improve overall performance of the location-sensitive disaster event detection. Fig. 2 illustrates the schematic workflow of the proposed framework and Algorithm 1 gives the step-by-step process flow. The location-sensitive disaster event detection pipeline broadly consists of four steps: Data collection & pre-processing, Feature Engineering, Event Detection and Multiple-Criteria Decision Making (MCDM).

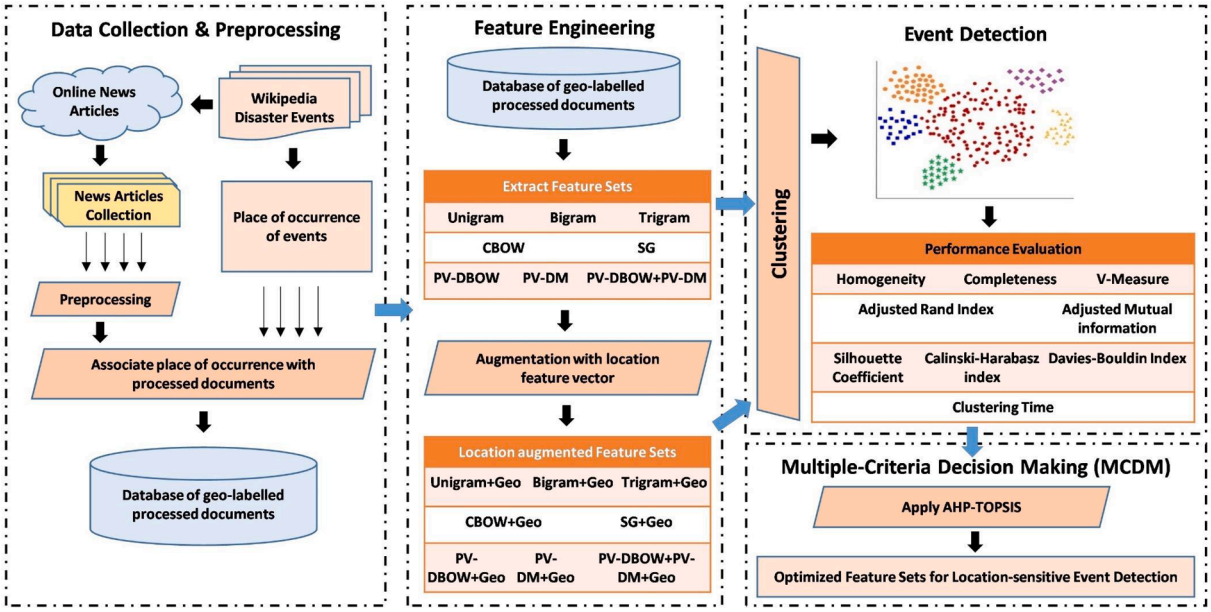

Fig. 2. Schematic representation of proposed framework.

3.1. Data collection & pre-processing

The major bottleneck of location-sensitive disaster event detection is the unavailability of benchmark labelled dataset which distinguishes the same kind of events that have happened at different places. Hence, a dataset of online news articles is acquired for disaster events that have happened in India since 2010 as listed in Wikipedia (Wikipedia, 2021a, 2021b, 2021c, 2021d, 2021e). As information about the place of occurrence of the events is available on Wikipedia, the news articles are geo-labelled at the time of data collection. The dataset is then processed in order to remove HTML tags, special characters, stop words and punctuation symbols followed by lower case conversion, stemming and lemmatization. Though the framework has been evaluated with disaster events, the same is applicable for any kind of location-sensitive events.

Algorithm 1: GeoClust : Location-sensitive disaster event detection

1)	Initialize: E → Disaster events in Wikipedia A: {G_1, G_2, ……., G_k} → Set of Clustering Algorithms C: {C_1, C_2, ……., C_n} → Set of Performance evaluation metrics
2)	*For each event* $E_i \in E$*:* a Initialize D: {D_1, D_2, ….., D_n} → Set of online news articles
3)	*For each* $D_i \in D$*:* a *Geo*(D_i) ⇔ *Geo-Label with place of occurrence of event* $E_i \Leftrightarrow D_i$ b *Pre-process* c *Extract feature sets* $F(D_i)$ *for Text and Title*
4)	*Create location augmented feature sets with location of* $E_i \Leftrightarrow D_i \in D$*:* $F_{geo}(D_i) \Leftrightarrow \sum [F(D_i) + tfidf(Geo(D_i))]$
5)	*For Unigram Feature Set* $F \in F(D_i)$*:* a V_{mxn}*: PerformanceEvaluation* (A_i, F, C_j) $\forall A_i \in A$, $C_j \in C$ b $G \rightarrow$ *Best performing algorithm* ⇔ *AHP-TOPSIS(Subset(A), C,* V_{mxn}*)*
6)	*For each feature set* $F \in \{F(D_i), F_{geo}(D_i)\}$*:* a V_{mxn}*: PerformanceEvaluation* (G, F_i, C_j) $\forall C_j \in C$ b *FS → Most efficient Feature Set* ⇔ *AHP-TOPSIS(F, Subset(C),* V_{mxn}*)*

3.2. Feature engineering

In this step, the features are extracted from text content in order to analyse the associations among the terms in the dataset. For feature extraction, **context-free feature sets** based on term-frequency vectors, *namely,* Unigram, Bigram and Trigram; and **context-based feature sets** based on word embedding-based models, *namely,* Word2Vec and Doc2Vec have been implemented. These feature sets will help to analyse the importance of underlying context of terms for the process of event detection. Certainly, this is not an exhaustive list of textual feature sets, but all these feature sets have proved to be efficient and have been widely applied for text mining tasks as they are domain-independent and generic in nature. Inevitably, location is a significant feature for location-sensitive event detection. Hence, all these feature sets are integrated with place of occurrence vectors of the events in the dataset. Following techniques have been applied for the feature extraction from the news articles:

a) Term frequency–inverse document frequency (Tf-idf): Tf-idf is a quantitative measure for the significance of a word to a text document in the corpus. Term frequency (tf) is a count of frequency of a term in the document. However, it doesn't distinguish meaningful terms from common words/ stop words. Inverse data frequency (idf) is used to determine rare words across all documents in the corpus and decrease the weight of frequent terms so as to differentiate documents. Text documents are represented in tf-idf vector model as per Eq. (1):

$$tfidf(t,d,D) = tf(t,d) * idf(t,D) \tag{1}$$

$$where,\ tf(t,d) = \frac{f_d(t)}{\max_{w \in d} f_d(w)}$$

$$and,\ idf(t,D) = \log\left(\frac{|D|}{|\{d \in D : t \in d\}|}\right)$$

b) *N*-gram: The more conventional way is to build a dictionary of groups of '*n*' consecutive words. The *n*-gram model uses a sliding window, usually overlapping, with the window size of '*n*' specifying the size of the group of words to consider. For a sequence of *N* words in text, the *n*-gram model approximates the probability of subsequent word w_n derived from the probability of preceding *n-1* words using the Eq. (2):

$$P(w_n|w_{1:n-1}) \approx P(w_n|w_{n-N+1:n-1}) \tag{2}$$

The approach retains word ordering information upto a window size of '*n*'. In this study, feature sets have been extracted for three types of *n*-grams *viz.* Unigram, Bigram and Trigram.

c) Word2Vec: Word2Vec embedding model comprises a two-layer neural network that learns relationships among words of text documents and represents these in vectors of hundreds of dimensions framed through context-based prediction by trained neural networks (Mikolov et al., 2013). Word vectors of similar meaning words are close in vector space and word vectors of different meaning words are distant apart.

Word2Vec learns to predict words based on its context. Theoretically, given a series of training words $\{w_1, w_2, w_3, \ldots, w_T\}$, Word2Vec model targets to maximize the average log probability as in Eq. (3):

$$\frac{1}{T}\sum_{t=k}^{T-k}\log p\left(w_t|w_{t-k},\ldots\ldots,w_{t+k}\right) \tag{3}$$

And predicts words using a multiclass classifier, such as softmax as in Eq. (4):

$$p(w_t|w_{t-k},\ldots\ldots,w_{t+k}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}} \tag{4}$$

The Word2Vec model has two variants: Continuous Bag-of-Words (CBOW) and Skip-Gram (SG) (Mikolov et al., 2013). The CBOW model links the distributed representations of adjacent words in order to predict the word in the center. On the other hand, the Skip-Gram model uses the distributed representation of the input word to predict the words in surroundings.

d) Doc2Vec: Doc2Vec model was proposed by Le and Mikolov for representation of paragraphs as an extension to Word2Vec embedding model (Le & Mikolov, 2014). Similar to Word2Vec, Doc2Vec has two variants: PV-DBOW and PV-DM. PV-DBOW model is equivalent to Skip-Gram model and PV-DM model corresponds to the CBOW model. The Doc2Vec model represents each paragraph/ document as a unique vector in a column of matrix D and each word as a unique vector in column of matrix W. The model predicts the next word by averaging or concatenating context-based word vectors and full paragraph's / document's doc-vector. These vectors are fed to supervised or unsupervised machine learning algorithms for further text analysis.

e) Location-augmented feature sets: In this study, location has been used as a significant feature in addition to other textual features. In order to boost the importance of location feature and evaluate its effect, information about place of occurrence of event has been integrated with the text feature sets using Eq. (5):

$$F_{geo}(D_i) = \sum_{i=1}^{n} F(D_i) + tfidf(Geo(D_i)) \tag{5}$$

where $F(D_i) \in \{tfidf(D_i), n\text{-}gram(D_i), word2vec(D_i), doc2vec(D_i)\}$ and $Geo(D_i)$ is the place of occurrence of event reported in the document D_i.

3.3. Event detection

Since text on the web is mostly unstructured, unlabeled and diverse in nature, the proposed framework adopts unsupervised machine learning algorithms for event detection. Unsupervised machine learning algorithms do not require training datasets. These algorithms perform clustering of data by intelligently learning associations among data and target grouping of data points in specific classes such that data points in the same class have identical properties and are close to each other in terms of distance among their feature vectors. It results in the assignment of documents associated with one event to one class. Moreover, for real-time implementation of location-sensitive event detection, unsupervised machine learning algorithms increase the scalability of the system and its adaptation to the variety of content available on the web without the need of comprehensive and domain-specific labelled training datasets. The unsupervised approaches initiate categorization of initially available content in event classes and are capable of predicting the event class of new content that leads to precise growth of content in an event class over the time. Thus, by associating recently published news articles to the previous events classes, the event class of recent news articles can be predicted, thereby, detecting the event reported in these news articles.

In literature, several unsupervised machine learning algorithms *viz.* Affinity Propagation (Frey & Dueck, 2007), Birch (T. Zhang et al., 1996), Agglomerative Clustering (Day & Edelsbrunner, 1984), K-means (MacQueen, 1967), Spectral Clustering (Pothen et al., 1990), MiniBatch K-means (Sculley, 2010), OPTICS (Ankerst et al., 1999) and DBSCAN (Ester et al., 1996) have been applied for event detection. In this research, all these algorithms have been evaluated for their performance with unigram feature sets and the best performing algorithm is selected using analysis through AHP-TOPSIS (*Procedure 1*). The performance of all feature sets for event detection with the selected algorithm is evaluated for a set of performance metrics *viz.* Homogeneity, Completeness, V-Measure, Adjusted Rand Index, Adjusted Mutual information, Silhouette Coefficient, Calinski-Harabasz index, Davies-Bouldin Index and Clustering Time. The most efficient feature set representation is selected by analysing the experimental results for above performance metrics using AHP-TOPSIS (*Procedure 1*).

Procedure 1: *AHP-TOPSIS (A, C, V_{mxn})*

Input:	*A: $\{A_1, A_2, A_3, \ldots\ldots, A_m\} \rightarrow m$ alternatives*
	C: $\{C_1, C_2, C_3, \ldots\ldots, C_n\} \rightarrow n$ criteria
	$V_{mxn} \rightarrow$ Value matrix
Output:	*Ranked alternatives*
1	*For each criteria $C_j \in C$: Assign importance weights on Saaty's scale*
2	*$P_{nxn} \leftarrow$ Pairwise comparison matrix of criteria, s.t. $p_{ij} = 1/p_{ji}$*
3	*For weights assigned to criteria, determine consistency ratio*
4	*For Value matrix V_{mxn}: $N_{mxn} \leftarrow$ Normalised decision matrix*
5	*$M_{mxn} \leftarrow$ AHP Weighted normalized matrix*
6	*$S^+, S^- \leftarrow$ Positive and negative ideal solutions*
7	*$E_j^+, E_j^- \leftarrow$ Euclidean separation measures between alternatives and ideal solutions*
8	*$C_s \leftarrow$ Relative closeness score for each alternative $A_i \in A$*

3.4. Multiple-Criteria decision Making (MCDM)

Multiple-Criteria Decision Making (MCDM) is the process of choosing among a set of alternatives based upon multiple criteria. The performance of unsupervised machine learning algorithms can be evaluated for a number of intrinsic and extrinsic metrics. The selection of the best alternative from these metrics is a problem of the nature of MCDM. The MCDM techniques apply aggregation of scores of multiple evaluating alternatives, with weights assigned by human experts to various decision-making criteria and adopting suitable normalization procedures to put all alternatives in the same range for further ranking. The MCDM problem can be solved through a number of techniques *viz.* VIKOR (Duckstein & Opricovic, 1980), COMET (Sałabun, 2014), AHP (Saaty, 1984), TOPSIS (Hwang & Yoon, 1981) and α -Discounting Method for Multiple Criteria Decision Making (α-D MCDM) (Smarandache, 2010, 2013a, 2013b, 2015). AHP-TOPSIS technique has been implemented in the proposed framework for feature ranking. Since all metrics do not have equal importance to the performance evaluation, weight has been assigned to various criteria through Analytic Hierarchy Process (AHP). The TOPSIS technique ranks the alternatives on the basis of highest proximity from the positive ideal solution and farthermost from the negative ideal solution.

Saaty proposed AHP as a problem solving framework which involves a pairwise comparison of criteria along a scale of importance (Saaty, 1984). AHP has been widely applied in literature for assigning weights to the criteria because of its advantages including ease of application, inclusion of human factor and use of consistency measure (Gyani et al., 2022; Marzouk & Sabbah, 2021; Sotoudeh-anvari, 2022; Zandebasiri & Pourhashemi, 2016). Hwang and Yoon developed the TOPSIS technique in 1981 as a rational method for resolving tangible MCDM problems (Hwang & Yoon, 1981). Subsequently, data scientists had adopted TOPSIS for several computational problems due to its flexibility and ability to rank a set of alternatives based on multiple criteria. The TOPSIS technique ranks alternatives on the basis of highest proximity from the positive ideal solution and farthermost from the negative ideal solution.

Since objective of our work is to evaluate candidate feature sets in terms of multiple performance metrics and rank these as per the

closeness of their clustering performance scores to the ideal solution, the proposed approach is highly driven to adopt Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) as MCDM technique with weights assigned by AHP (*Procedure 1*). Alternatives in our work stands for feature sets for representation of content of news articles and; the performance metrics of clustering evaluation correspond to the criteria set. The steps followed for implementation of Procedure 1 for multi-criteria decision analysis using TOPSIS and weight assignment through AHP are briefly explained as below:

Step 1. Determination of appropriate alternatives and criteria set

The first and primary step of TOPSIS technique is the determination of a) alternative choices to compare for decision and b) a set of deciding criteria on what to deliberate. Suppose there are m alternatives A: $\{A_1, A_2, A_3, \ldots\ldots, A_m\}$ and n criteria C:$\{C_1, C_2, C_3, \ldots\ldots, C_n\}$.

Step 2. Assigning weights using AHP

a) Construction of pairwise comparison matrix: AHP allows decision-makers to define pairwise importance values of criteria on a scale, *namely,* Saaty's scale that measures importance of criteria as per a numerical scale from 1 to 9 as shown in Table 2.

For the construction of a pairwise comparison matrix, all criteria are mapped on the Saaty's scale. Eq. (6) demonstrates the pairwise comparison matrix, P_{nxn}, of n criteria. The value of p_{ij} indicates relative importance of the criteria i *w.r.t* criteria j. If the value of p_{ij} is 1, it indicates that i-*th* criterion is equally important to j-*th* criterion. For all other entries, $p_{ij} = 1/\, p_{ji}$.

$$P_{nxn} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1j} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2j} & \cdots & p_{2n} \\ . & . & . & . & . & . \\ p_{i1} & p_{i2} & \cdots & p_{ij} & \cdots & p_{in} \\ . & . & . & . & . & . \\ p_{n1} & p_{n2} & \cdots & p_{nj} & \cdots & p_{nn} \end{bmatrix} \tag{6}$$

b) Determining criteria weight vector: After building the pairwise comparison matrix, normalized relative matrix is calculated by i) computing geometric mean (G_i) of each i-*th* row of pairwise comparison matrix ii) creating a criteria weight vector (C_i) by normalizing the geometric means of rows using Eqs. (7) and (8):

$$G_i = \sqrt[n]{\prod_{i=1}^{n} p_{ij}} \tag{7}$$

$$C_i = \frac{G_i}{\sum_{i=1}^{n} G_i} \tag{8}$$

c) Determine the consistency of the weights assigned: Though the mapping of criteria importance to Saaty's scale is a subjective decision, AHP allows us to check consistency of this decision. In order to check consistency of the pairwise comparison matrix, the eigenvalue principle is adopted. For the purpose, two additional matrices $C1$ and $C2$ are formed using Eqs. (9) and (10).

$$C1 = M \times C \tag{9}$$

$$C2 = \frac{C1}{C} \tag{10}$$

Prof. Saaty defined Consistency Index (CI) that measures deviation or inconsistency of the criteria weights as given in Eq. (11):

$$CI = \frac{\lambda_{max} - \mathrm{n}}{n-1} \tag{11}$$

In the next step, Consistency Ratio (CR) is computed as the ratio of Consistency Index (CI) and Random Consistency Index (RI) as given by Saaty's table for Random Consistency Index (Table 3). The threshold value for CR is 0.1 and a value less than 0.1 is considered within the justifiable limit.

Step 3. Construction of normalized decision matrix

The next step of TOPSIS is the construction of a normalized decision matrix for all alternatives and criteria. Suppose V_{mxn} is the value matrix where each V_{ij} represents the value of alternative A_i for criteria C_j. Since the range of values for various criteria is non-uniform, and in order to facilitate inter-criteria comparison, the value matrix has to be normalized to produce a normalized decision matrix N_{mxn} using the Eq. (12):

$$N_{ij} = \frac{V_{ij}}{\sqrt{\sum_{i=1}^{m} \left(V_{ij}\right)^2}} \tag{12}$$

Step 4. Obtaining the weighted normalized matrix

Subsequently, a weighted normalized matrix (M_{mxn}) is calculated by multiplying the elements of normalized decision matrix (N) with the criteria weight vector (C) as given in Eq. (13):

$$M_{ij} = C_j \times N_{ij} \tag{13}$$

Step 5. Computing the positive and negative ideal solutions

Depending upon the benefit criterion and cost criterion, the positive and negative ideal solutions: S^+ and S^-, *resp.,* are determined using Eqs. (14) and (15):

$$S^+ = \left[M_1^+, \cdots, M_j^+, \cdots, M_n^+\right] \tag{14}$$

$$S^- = \left[M_1^-, \cdots, M_j^-, \cdots, M_n^-\right] \tag{15}$$

$$\text{where, } M_j^+ = \begin{Bmatrix} max(M_{ij}) & \forall\ C_j \in benefit\ criteria \\ min(M_{ij}) & \forall\ C_j \in cost\ criteria \end{Bmatrix}$$

$$\text{and } M_j^- = \begin{Bmatrix} min(M_{ij}) & \forall\ C_j \in benefit\ criteria \\ max(M_{ij}) & \forall\ C_j \in cost\ criteria \end{Bmatrix}$$

Step 6. Computing the Euclidean distance

In the next step, the Euclidean distance to measure separation between each alternative and the respective positive and negative ideal solution is determined as given in Eqs. (16) and (17):

$$E_j^+ = \sqrt{\sum_{j=1}^{n} \left(M_{ij} - S_j^+\right)^2} \ for\ i = 1, 2, \ldots., m \tag{16}$$

Table 2
Saaty's importance scale.

Definition	Equal importance	Weak or Slight importance	Moderate importance	Above moderate importance	Strong importance	Above strong importance	Very strong importance	High importance	Extreme importance
Importance Intensity	1	2	3	4	5	6	7	8	9

Table 3
Saaty's table for random consistency index.

Number of criteria (n)	1	2	3	4	5	6	7	8	9	10
RI	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49

$$E_j^- = \sqrt{\sum_{j=1}^{n}\left(M_{ij} - S_j^-\right)^2} \; for \; i = 1, 2, ..., m \tag{17}$$

Step 7. Determining the relative closeness measure of the alternatives:

The relative closeness score (C_s) is computed for each alternative *w.r.t.* the ideal solutions using Eq. (18):

$$C_s = \frac{E_j^-}{\left(E_j^+ + E_j^-\right)} \tag{18}$$

Finally, the alternatives are ranked as per descending order of their relative closeness score. The alternative with the highest relative closeness score is closest to the positive ideal solution and is chosen as the best and most preferable alternative.

4. Experimental results and discussion

In order to evaluate the performance of our framework, all experiments in this paper are performed on a dataset collected from online news articles using Newspaper3K API (Ou-Yang, 2018). The unsupervised machine learning algorithms have been implemented in the Scikit library of Python (Pedregosa et al., 2011). All feature sets have been extracted using the Scikit library of Python. For extraction of context-based feature sets, the Gensim library of Python has been used. The location augmentation of feature sets is achieved through concatenation of each extracted feature vector with corresponding place of occurrence vector using Eq. (5).

The experiments and analysis through AHP-TOPSIS are programmed in Python and run in Jupyter notebook IDE on Windows 10 (64-bit) platform. Extensive sets of experiments have been performed to evaluate the performance of different feature sets for location-sensitive disaster event detection.

4.1. Dataset analysis

Experimental analysis of this framework has been carried out on a dataset consisting of disaster articles broadly containing news for terrorist attacks, fire incidents, stampede incidents, building collapse and maoist attacks. The dataset consists of 1954 articles with a maximum of 6145 words/ document and minimum of 51 words/ document. With such diversity of content size in news articles, efficient feature set selection plays a significant role in the event detection process. Table 4a lists the dataset statistics in terms of event category, number of events in each category, number of articles, total number of words in the dataset, maximum number of words/document and minimum number of words/ document and Table 4b gives one instance of the dataset. Fig. 3 shows distribution of dataset for various event categories.

Table 4a
Dataset statistics.

Event Category	No. of events	No. of Articles	Total no. of words	Maximum no. of words / doc	Minimum no. of words /doc
All	72	1954	979,310	6145	51
Terrorist Attack	36	1002	509,220	6079	51
Fire Incident	15	481	260,797	5794	52
Stampede incident	6	179	79,090	2662	52
Building Collapse	8	170	51,796	3199	59
Maoist Attack	7	122	78,407	6145	80

4.2. Performance metrics

In view of unsupervised machine learning algorithms being adopted in this work, following metrics have been chosen for performance evaluation:

a. **Homogeneity (H_{score}):** It is the measure of ratio of members of a single class assigned to a single cluster (Rosenberg & Hirschberg, 2007). A clustering result is perfectly homogenous if all the clusters contain members of a single class. The homogeneity score ranges from 0 to 1. Higher the homogeneity score, better is the performance of the clustering algorithm. Its formulation is expressed mathematically in Eq. (19):

$$H_{score} = \begin{cases} 1 & if \; H(C,K) = 0 \\ 1 - \dfrac{H(C|K)}{H(C)} & else \end{cases} \tag{19}$$

$$where, \; H(C|K) = -\sum_{k=1}^{|K|}\sum_{c=1}^{|C|}\frac{a_{ck}}{N}\log\frac{a_{ck}}{\sum_{c=1}^{|C|}a_{ck}}$$

$$and, \; H(C) = -\sum_{c=1}^{|C|}\frac{\sum_{k=1}^{|K|}a_{ck}}{n}\log\frac{\sum_{k=1}^{|K|}a_{ck}}{n}$$

where n is the number of data points in the dataset, C is the set of classes, K is the set of clusters, and a_{ck} is the number of data points that are members of class c and assigned to cluster k.

b. **Completeness (C_{score}):** It is the measure of ratio of members of a single class assigned to the same cluster (Rosenberg & Hirschberg, 2007). A clustering result is perfectly complete if all the members of a single class are assigned to the same cluster. Completeness score ranges from 0 to 1. Higher the completeness score, better is the performance of the clustering algorithm. Its formulation is expressed mathematically in Eq. (20):

$$C_{score} = \begin{cases} 1 & if \; H(K,C) = 0 \\ 1 - \dfrac{H(K|C)}{H(K)} & else \end{cases} \tag{20}$$

$$where, \; H(K|C) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|}\frac{a_{ck}}{N}\log\frac{a_{ck}}{\sum_{k=1}^{|K|}a_{ck}}$$

$$and, H(K) = -\sum_{k=1}^{|K|}\frac{\sum_{c=1}^{|C|}a_{ck}}{n}\log\frac{\sum_{c=1}^{|C|}a_{ck}}{n}$$

where n is the number of data points in the dataset, C is the set of classes, K is the set of clusters, and a_{ck} is the number of data points members of class c and assigned to cluster k.

Table 4b
Instance of dataset.

Event Class label	Place of Occurrence	Title	Publication Date	Text
Chennai Building collapse 2014	Chennai	Chennai Building Collapse	28–06-2014	An 11-storey building under construction collapsed near Chennai. A fortnight after the collapse, a white board outside Royapettah Government Hospital morgue in Chennai told the sordid tale of 42 victims whose bodies the institution had received ….. (177 more words)
Chennai Building collapse 2014	Chennai	Chennai Building Collapse: Toll Rises to 11, SiX arrested	29–06-2014	The death toll in the multi-storeyed building collapse at Mugalivakkam near Porur here went up to 11 as rescuers continued to sift through the rubble to extricate more persons …… (166 more words)
Chennai Building collapse 2014	Chennai	After Chennai Building Collapse, Civic Officials Rush to Inspect Under Construction Structures	July 03, 2014	Chennai: The Chennai Metropolitan Development Authority has begun a crackdown on under-construction buildings in the city after a multi-storey building collapsed on Saturday, leaving 52 people dead. (145 more words)
Chennai Building collapse 2014	Chennai	Chennai: Experts Inspect Site of Building Collapse	July 18, 2014	Chennai: A team of experts in structural engineering and architecture on Friday inspected the site of a building which collapsed at Porur killing 61 and injuring 27, police said. …. (90 more words)

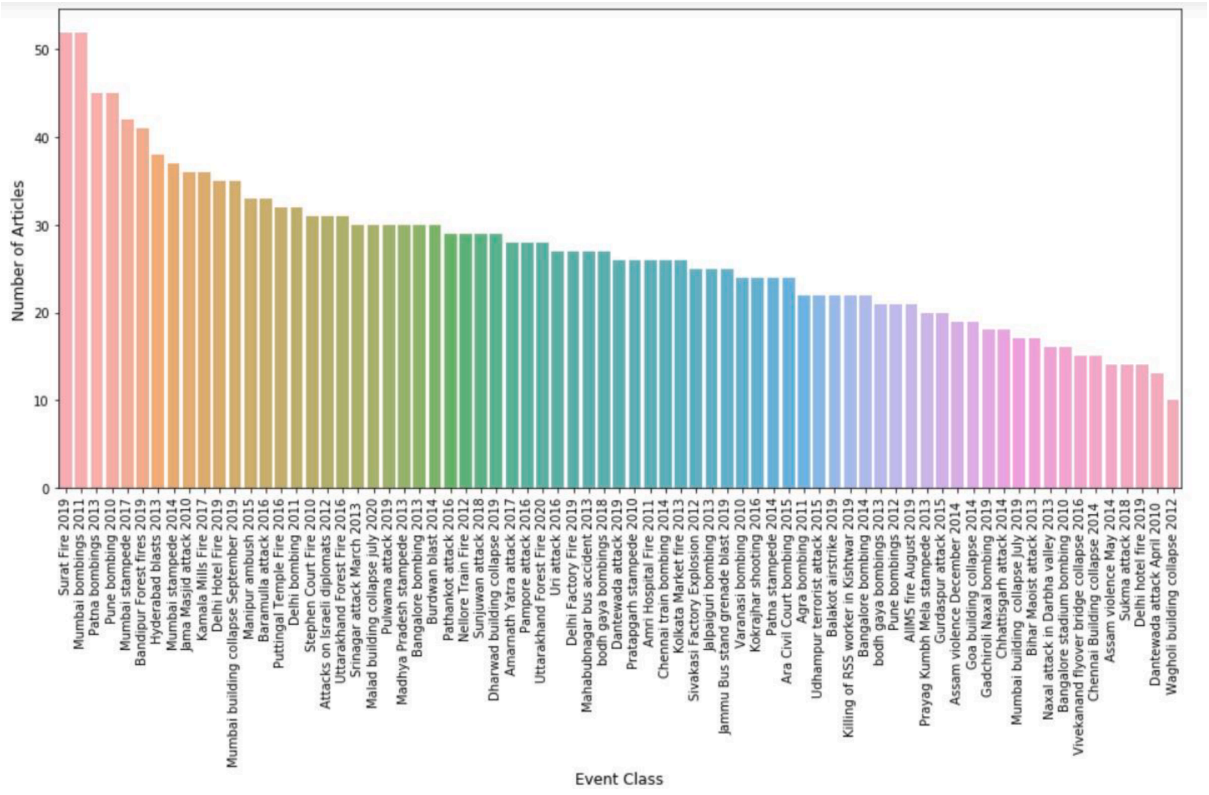

Fig. 3. Analysis of dataset for event categories.

c. **V-Measure (V_β):** It is the measure of success of accomplishing homogeneity and completeness of a clustering algorithm (Rosenberg & Hirschberg, 2007). V-measure is the calculated as weighted harmonic mean of homogeneity score and completeness score as given in Eq. (21):

$$V_\beta = \frac{(1+\beta)*H_{score}*C_{score}}{(\beta*H_{score})+C_{score}} \tag{21}$$

If $\beta > 1$, completeness is given more weightage than homogeneity, if β *less than 1*, homogeneity is preferred over completeness.

d. **Adjusted Rand Index (ARI):** It is a measure of similarity between two clustering in terms of pairs of sample counts that are allocated to same or different clusters in the predicted clustering and true clustering (Hubert & Arabie, 1985). The rand index score is calculated as the ratio of the number of pairs in agreement to the number of total pairs. The rand index score is then "adjusted for chance" to compute adjusted rand index (ARI) as given in Eqs. (22) and (23):

$$RI = \frac{Number\ of\ pairs\ in\ agreement}{Number\ of\ total\ pairs} \tag{22}$$

$$ARI = \frac{RI - RI_{expected}}{max(RI) - RI_{expected}} \tag{23}$$

The ARI score ranges from 0 to 1 with 0 indicating random labelling and 1 for the perfect match.

e. **Adjusted Mutual information (AMI):** It is a measure of information sharing between two clustering in order to depict how information knowledge of one clustering can reduce uncertainty about other clustering (Vinh et al., 2010). Its formulation is as expressed mathematically as in Eqs. (24) and (25):

$$MI_{AB} = \sum_{i=1}^{|A|}\sum_{j=1}^{|B|}\frac{|A_i \cap B_j|}{N}\log\frac{N|A_i \cap B_j|}{|A_i||B_j|} \tag{24}$$

$$AMI_{AB} = \frac{MI_{AB} - Expected[MI_{AB}]}{mean\left(\left(-\sum_{i=1}^{|A|}\frac{|A_i|}{N}\log\frac{|A_i|}{N}\right),\left(-\sum_{j=1}^{|B|}\frac{|B_j|}{N}\log\frac{|B_j|}{N}\right)\right)} \tag{25}$$

where N is the number of data points in the dataset, A and B are two clustering.

f. **Silhouette Coefficient (SC):** It is a measure of distance between data points in neighbouring clusters, *i.e.,* separation distance between clusters (Rousseeuw, 1987). The silhouette score ranges from −1 to 1 where score $\simeq -1$ indicates wrong assignment and $\simeq 1$ indicates perfect assignment. The formulation of silhouette score is given in Eq. (26):

$$SC(i) = \frac{V(i) - U(i)}{\max\{U(i), V(i)\}} \tag{26}$$

where $U(i)$ measures the average distance of a data point i from the rest of data points in its host cluster U and $V(i)$ measures the smallest average distance of a data point i from all data points in cluster V.

g. **Calinski-Harabasz Index (CH):** It is defined as the ratio of variance within clusters and variance among clusters (Caliński & Harabasz, 1974). Higher the value of the *CH* index, the better is the clustering performance. The formulation of *CH* index is given in Eq. (27):

$$CH = \frac{V(B_k)}{V(W_k)} * \frac{N-k}{k-1} \tag{27}$$

where $V(B_k)$ is the overall variance among clusters, $V(W_k)$ is the overall within cluster variance, N is the number of data points and k is the number of clusters.

h. **Davies-Bouldin Index (DBI):** It **is** a measure of separation between similar clusters in terms of ratio of distances with-in the clusters to the distance between clusters (Davies & Bouldin, 1979). Its minimum value is zero. Lower the Davies-Bouldin index, better is the performance of clustering algorithms. It is calculated mathematically as per Eq. (28):

$$DBI = \frac{1}{k}\sum_{i=1}^{k}\max_{j\neq i}\frac{C_i + C_j}{C_{ij}} \tag{28}$$

where k is the number of clusters, C_i and C_j are the intra-cluster distances with-in data points of *i-th* cluster and *j-th* cluster, C_{ij} is the inter-cluster distance between *i-th* and *j-th* clusters.

4.3. Results and discussion

This section presents experiments and analytic analysis of various feature sets, *namely, n*-gram variants: Unigram (Uni), Bigram (Bi), Trigram (Tri); Word2Vec variants: CBOW (CB), Skip-Gram (SG); Doc2-Vec variants: PV-DBOW (DB), PV-DM (DM), PV-DBOW + PV-DM (DBDM); and their location integrated feature sets *viz.* UniGEO, BiGEO, TriGEO, CBGEO, SGGEO, DBGEO, DMGEO, DBDMGEO for their performance evaluation in location-sensitive disaster event detection.

4.3.1. Determination of alternatives and criteria set

In order to implement AHP-TOPSIS for feature selection, steps discussed in section 3.4 are carried out in order. The first step is the determination of appropriate alternatives and criteria set. All feature sets *viz.* traditional context-free and context-based feature sets and their location-augmented variants form the set of alternatives. In literature, all the performance metrics listed in section 4.2 have been widely exercised for clustering evaluation of different types of problems and datasets. Still, there should be a consensus between Homogeneity, Completeness, V-measure metrics. Since V-measure is the harmonic mean of Homogeneity and Completeness, it represents a good trade-off between the two. Hence among these three metrics, V-measure is selected for further analysis. Thus, criteria set for AHP-TOPSIS analysis consists of Clustering time (T), V-measure (V_β), Adjusted Rand Index (ARI), Adjusted Mutual information (AMI), Silhouette Coefficient (SC), Calinski- Harabasz index (CH) and Davies- Bouldin index (DBI). Among all performance metrics, V-measure, Adjusted Rand Index, Adjusted Mutual information, Silhouette Coefficient and Calinski- Harabasz index are anticipated to be maximum, thus contributing towards benefit criteria. On the contrary, clustering time and Davies- Bouldin index are desired to be minimum, hence forming the cost criteria set. The final alternative and criteria set is shown in Table 5.

4.3.2. Weight assignment to criteria using AHP

Since each criterion may or may not be equally important, weights are assigned to different criteria using AHP. The corpus consists of disaster news articles where events of the same type but happened at different locations and at different points in time belong to different event classes. It is possible that the shape of all clusters may not be spherical. Moreover, the number of classes in our corpus is 72 with small sample sizes, which may affect homogeneity and completeness due to the problem of random labelling. For small sample sizes and large number of clusters, Adjusted Mutual information (AMI) and Adjusted Rand Index (ARI) give more accuracy for performance (Pedregosa et al., 2021). In addition, the Adjusted Rand Index is independent of cluster structure. Hence, ARI has been given the highest importance and AMI has been assigned the second highest importance.

Despite small sizes and uneven shapes of clusters, the goal of a clustering algorithm is to assign all data points of the same class to the same cluster and data points of different classes to different clusters. Hence, the second objective of our research is to attain maximum Homogeneity and Completeness. Since V-measure is the representative for both homogeneity and completeness, it has been assigned the third highest importance. Thirdly, Silhouette Coefficient, Calinski- Harabasz index, Davies- Bouldin index all are based on inter cluster distance and intra-cluster distance, they have been given equal importance. In this era of high computing technology, time of computation is not a big deal due to the availability of very fast computation technologies. Hence, clustering time has been assigned the least importance. Table 6 gives the weights assigned to each criterion based on their importance. More the weight assigned, the more important the criterion is.

For weight assignment to criteria using AHP, weights (W) as in Table 6 are mapped to Saaty's scale for importance of criteria ranging from 1 to 9. Table 7 depicts the pairwise comparison matrix, P_{7x7}

Table 5
Set of alternatives and criteria.

Set of Alternatives	Set of Criteria
Unigram (Uni)	Clustering time (T)
Unigram + GEO (UniGEO)	V-measure (V_β)
Bigram (Bi)	Adjusted Rand Index (ARI)
Bigram + GEO (BiGEO)	Adjusted Mutual information (AMI)
Trigram (Tri)	Silhouette Coefficient (SC)
Trigram + GEO (TriGEO)	Calinski- Harabasz index (CH)
CBOW (CB)	Davies- Bouldin index (DBI)
CBOW + GEO (CBGEO)	
SG (SG)	
SG + GEO (SGGEO)	
PV-DBOW (DB)	
PV-DBOW + GEO (DBGEO)	
PV-DM (DM)	
PV-DM + GEO (DMGEO)	
PV-DBOW + PV-DM (DBDM)	
PV-DBOW + PV-DM + GEO (DBDMGEO)	

Table 6
Weight assigned to criteria as per importance in performance evaluation.

	T	V_β	ARI	AMI	SC	CH	DBI
Weight (W)	1	3	5	4	2	2	2
Normalized weight (w)	5.263	15.789	26.316	21.053	10.526	10.526	10.526

Table 7
P_{7x7}: pairwise comparison matrix of criteria.

	T	V_β	ARI	AMI	SC	CH	DBI
T	1	1/3	1/5	1/4	1/2	1/2	1/2
V_β	3	1	3/5	3/4	3/2	3/2	3/2
ARI	5	5/3	1	5/4	5/2	5/2	5/2
AMI	4	4/3	4/5	1	4/2	4/2	4/2
SC	2	2/3	2/5	2/4	1	1	1
CH	2	2/3	2/5	2/4	1	1	1
DBI	2	2/3	2/5	2/4	1	1	1

constructed such that, for all other entries, $p_{ij} = 1/p_{ji}$. The values in the diagonal of matrix *P* are all 1 because a criterion is equally important to itself. The geometric mean vector *(G)* and criteria weight vector *(C)* vectors are constructed using Eqs. (7) and (8) and the resultant vectors are shown in Table 8.

In order to check consistency of the pairwise comparison matrix through the eigenvalue principle, two additional matrices *C1* and *C*2 are constructed using Eqs. (9) and (10). The value of λ^{max} is calculated by average of all values in vector *C*2. Then the Consistency Index (*CI*) is calculated using Eq. (11) and is obtained as 0.00. The value of Random Consistency Index (*RI*) for n = 7 is 1.32 (Table 3). Hence, the consistency ratio (CR = CI/RI) comes out to be 0.00 which is less than 0.1 and is considered within the acceptable limit. It means that our weight assignment through AHP is consistent throughout.

4.3.3. Selection of clustering algorithm

For the purpose of selection of best performing clustering algorithm, performance of well-known unsupervised machine learning algorithms, *namely,* Affinity Propagation, Birch, Agglomerative Clustering, K-means, Spectral Clustering, MiniBatch K-means, OPTICS, DBSCAN has been evaluated for Unigram feature sets. Fig. 4 shows the performance analysis of these algorithms for various metrics. The highest desirable value and negative points are highlighted. It can be seen that OPTICS clustering algorithm takes the highest clustering time. The value of silhouette coefficient for both OPTICS and DBSCAN algorithm is negative. Table 9 gives the performance scores of the algorithms quantitatively for Number of Clusters (N), Clustering Time (in seconds) (T), Homogeneity (H_{score}), Completeness (C_{score}), V-measure (V_β), Adjusted Rand Index (ARI), Adjusted Mutual information (AMI), Silhouette Coefficient (SC), Calinski- Harabasz index (CH) and Davies- Bouldin index (DBI).

As a rule of thumb, it is assumed that variation in the number of clusters should not be greater than ± 3 %, thereby eliminating Affinity Propagation, MiniBatch K-means, OPTICS and DBSCAN out of the race. In order to select the best performing algorithm from the rest of the algorithms, AHP-TOPSIS has been applied. Table 10–13 shows the output of applying Procedure 1 for the selection process of clustering algorithm.

Table 8
Resultant G and C vectors.

	G	C
T	0.414	0.053
V_β	1.242	0.158
ARI	2.070	0.263
AMI	1.656	0.211
SC	0.828	0.105
CH	0.828	0.105
DBI	0.828	0.105

By arranging the obtained closeness scores in descending order gives us the ranking of different alternatives. Since Agglomerative Clustering (AG) achieves the highest closeness score, it is chosen as the best performing clustering algorithm and has been applied for further evaluating all other feature sets.

4.3.4. Selection of feature set

In this section, the process for the selection of the most efficient feature set is explained. The agglomerative clustering algorithm is applied to cluster news articles in predefined classes. Table 14 illustrates performance analysis of all above feature sets for clustering in terms of Clustering Time (in seconds) (T), Homogeneity (H_{score}), Completeness (C_{score}), V-measure (V_β), Adjusted Rand Index (ARI), Adjusted Mutual information (AMI), Silhouette Coefficient (SC), Calinski- Harabasz index (CH) and Davies- Bouldin index (DBI).

Following observations have been made from the performance results in Table 14:

a) With the integration of location features, the clustering performance of each feature set improves significantly. However, an increase in clustering time is observed due to the fact that with the integration of the location vector, the size of the resultant vector increases.
b) Context-based feature sets are fast in clustering because their vector size is very small as compared to tf-idf vectors.
c) The BiGEO feature set outperforms other feature sets for Homogeneity, Completeness, V-measure, Mutual information. However, it is costly in terms of time taken.
d) The SG feature set is the fastest for clustering results, but its results for other parameters are not satisfactory.
e) The performance of SG + GEO feature set in terms of Silhouette Coefficient, Calinski- Harabasz index is better than other feature sets.
f) The DBOW + GEO feature set performs better than other feature sets for Adjusted Rand Index and Davies- Bouldin index.

As per above observations, though Bigram + GEO feature set outperforms other feature sets for maximum number of parameters, its performance for other metrics is not satisfactory. Moreover, it is not justified to consider all metrics equally likely. Nevertheless, on the basis of this preliminary analysis, it is hard to select feature sets that are efficient in all perspectives of clustering. In order to overcome this dilemma, further deeper analysis of results has been carried out using AHP-TOPIS as explained in Procedure 1.

With 16 alternatives and 7 criteria, the value matrix V_{16x7} is constructed from performance scores attained (Table 14) by each feature set for all metrics as shown in Table 15.

The resultant normalized decision matrix N_{16x7} and Weighted Normalized matrix M_{16x7} are shown in Tables 16 and 17.

Finally, relative closeness (C_s) is measured for each alternative with the ideal solution by first computing Euclidean distance using Eqs. (16) and (17) and then applying Eq. (18) (See Table 18).

Arranging the obtained relative closeness scores in descending order ranks the alternatives in order of their preference. For the feature sets, ranking order is obtained as follows:

SGGEO > DBGEO > DB > UniGEO > BiGEO > TriGEO > CBGEO > DBDM > DMGEO > DBDMGEO > DM > SG > CB > Uni > Bi > Tri.

From above ranking order, it is concluded that:

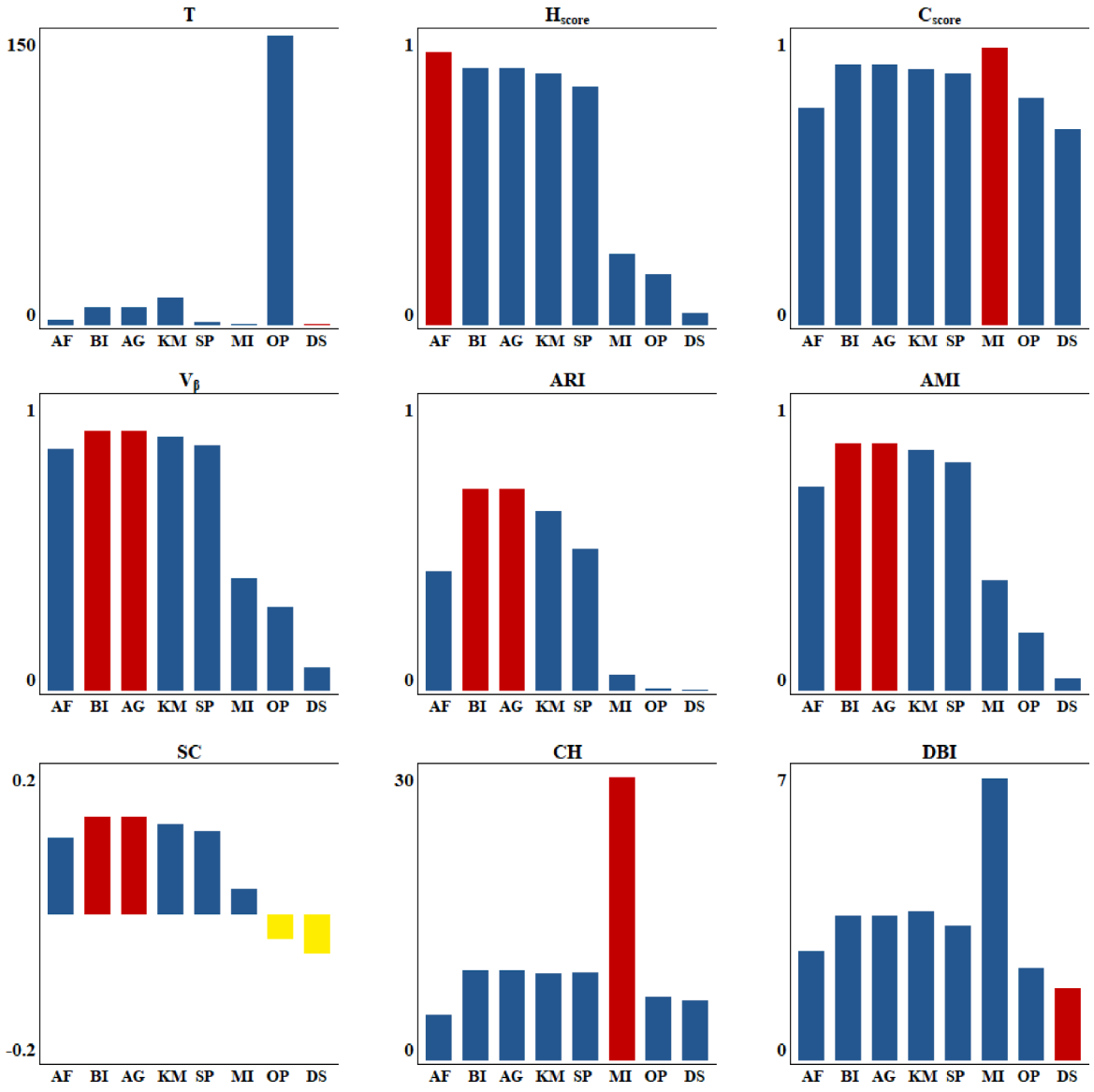

Fig. 4. Performance analysis of different clustering algorithms.

Table 9
Comparative performance analysis of different clustering algorithms for unigram feature set.

Algorithm	N	T	H_{score}	C_{score}	V_β	ARI	AMI	SC	CH	DBI
Affinity Propagation (AF)	237	3.052	0.936	0.743	0.829	0.407	0.698	0.105	4.638	2.615
Birch (BI)	72	9.312	0.882	0.894	0.888	0.689	0.848	0.135	9.251	3.477
Agglomerative Clustering (AG)	72	8.908	0.882	0.894	0.888	0.689	0.848	0.135	9.251	3.477
K-means (KM)	72	14.048	0.863	0.878	0.871	0.613	0.824	0.123	8.942	3.58
Spectral Clustering (SP)	72	1.851	0.816	0.863	0.839	0.484	0.784	0.115	9.089	3.23
MiniBatch K-means (MI)	3	0.246	0.243	0.95	0.387	0.053	0.378	0.035	29.07	6.749
OPTICS (OP)	28	148.878	0.175	0.779	0.285	0.007	0.198	−0.034	6.553	2.22
DBSCAN (DS)	11	0.23	0.043	0.673	0.08	0.001	0.042	−0.054	6.176	1.749

Table 10
V_{mxn}: value matrix.

Algorithm	T	V_β	ARI	AMI	SC	CH	DBI
BI	9.312	0.888	0.689	0.848	0.135	9.251	3.477
AG	8.908	0.888	0.689	0.848	0.135	9.251	3.477
KM	14.048	0.871	0.613	0.824	0.123	8.942	3.580
SP	1.851	0.839	0.484	0.784	0.115	9.089	3.230

Table 11
N_{4x7}: normalized decision matrix.

Algorithm	T	V_β	ARI	AMI	SC	CH	DBI
BI	0.486	0.509	0.552	0.513	0.53	0.506	0.505
AG	0.465	0.509	0.552	0.513	0.53	0.506	0.505
KM	0.733	0.5	0.491	0.499	0.483	0.489	0.52
SP	0.097	0.481	0.388	0.474	0.452	0.498	0.469

Table 12
M_{4x7}: weighted normalized matrix.

Algorithm	T	V_β	ARI	AMI	SC	CH	DBI
BI	0.026	0.080	0.145	0.108	0.056	0.053	0.053
AG	0.024	0.080	0.145	0.108	0.056	0.053	0.053
KM	0.039	0.079	0.129	0.105	0.051	0.052	0.055
SP	0.005	0.076	0.102	0.100	0.048	0.052	0.049

Table 13
Euclidean distance measures (E^+ & E^-) and closeness score (C_s).

Algorithm	E^+	E^-	C_s
BI	0.0209	0.0469	0.6920
AG	0.0198	0.0472	0.7047
KM	0.0381	0.0280	0.4237
SP	0.0450	0.0340	0.4305

a) Word2Vec-SG and Doc2Vec-DB augmented with place of occurrence are the two best performing feature sets for the location-sensitive event detection.
b) The location-augmented feature sets perform better than their corresponding basic versions.
c) The performance of context-based feature sets with location-augmentation surpass the performance of context-free textual feature sets and location-augmented versions.

After having gained significant insight and discussion, all the research questions raised in the introduction section are answered in terms of the vital outcomes of the proposed work.

RQ1. Are textual features efficient enough for location-sensitive event detection?

As revealed by results (Table 14), there is no doubt that textual features have significant potential in the field of location-sensitive event detection. In fact, the clustering performance with major feature sets was by and large indiscernible on the macro level. However, analytical analysis reveals that the performance of context-based feature sets with location-augmentation is better than performance of context-free textual feature sets.

RQ2. Does integration of location features can improve the retrieval accuracy for location-sensitive event-detection.

As supported by ranking order of feature sets returned by AHP-TOPSIS, it is clear that performance of all location-augmented feature sets is better than their corresponding basic versions for location-sensitive disaster event-detection.

RQ3. How feature selection impacts the retrieval accuracy of the overall event detection process?

In literature, traditional feature sets have been widely utilized for accomplishing the task of event detection. However, performance evaluation of various feature sets and their corresponding location augmented versions have remained unexplored. As discussed in Section

Table 14
Comparative performance analysis of different feature sets.

Feature Set	T	H_{score}	C_{score}	V_β	ARI	AMI	SC	CH	DBI
Uni	8.908	0.882	0.894	0.888	0.689	0.848	0.135	9.251	3.477
UniGEO	8.251	0.959	0.964	0.962	0.870	0.948	0.353	45.804	2.137
Bi	17.456	0.887	0.903	0.895	0.693	0.857	0.126	8.158	3.634
BiGEO	17.932	0.961	0.969	0.965	0.877	0.953	0.350	42.828	2.163
Tri	20.383	0.873	0.894	0.883	0.637	0.842	0.125	8.118	3.586
TriGEO	20.945	0.959	0.968	0.963	0.865	0.950	0.350	42.637	2.119
CB	0.230	0.728	0.740	0.734	0.386	0.638	0.202	53.472	1.919
CBGEO	0.286	0.806	0.819	0.812	0.538	0.745	0.263	53.056	1.838
SG	0.212	0.777	0.794	0.785	0.490	0.709	0.242	46.546	1.894
SGGEO	0.286	0.939	0.948	0.943	0.803	0.923	0.551	115.522	1.281
DB	0.751	0.939	0.944	0.942	0.839	0.920	0.498	86.385	1.138
DBGEO	0.824	0.959	0.960	0.959	0.901	0.944	0.526	90.761	1.104
DM	0.707	0.922	0.934	0.928	0.790	0.902	0.078	22.172	2.034
DMGEO	0.783	0.923	0.934	0.928	0.794	0.902	0.074	22.240	2.030
DBDM	1.593	0.926	0.936	0.931	0.808	0.906	0.070	22.927	2.021
DBDMGEO	1.635	0.928	0.936	0.932	0.813	0.907	0.051	22.953	2.047

Table 15
V_{16x7}: Value Matrix of Performance Scores.

Feature Set	T	V_β	ARI	AMI	SC	CH	DBI
Uni	8.908	0.888	0.689	0.848	0.135	9.251	3.477
UniGEO	8.251	0.962	0.87	0.948	0.353	45.804	2.137
Bi	17.456	0.895	0.693	0.857	0.126	8.158	3.634
BiGEO	17.932	0.965	0.877	0.953	0.350	42.828	2.163
Tri	20.383	0.883	0.637	0.842	0.125	8.118	3.586
TriGEO	20.945	0.963	0.865	0.950	0.350	42.637	2.119
CB	0.230	0.734	0.386	0.638	0.202	53.472	1.919
CBGEO	0.286	0.812	0.538	0.745	0.263	53.056	1.838
SG	0.212	0.785	0.490	0.709	0.242	46.546	1.894
SGGEO	0.286	0.943	0.803	0.923	0.551	115.522	1.281
DB	0.751	0.942	0.839	0.920	0.498	86.385	1.138
DBGEO	0.824	0.959	0.901	0.944	0.526	90.761	1.104
DM	0.707	0.928	0.790	0.902	0.078	22.172	2.034
DMGEO	0.783	0.928	0.794	0.902	0.074	22.24	2.030
DBDM	1.593	0.931	0.808	0.906	0.070	22.927	2.021
DBDMGEO	1.635	0.932	0.813	0.907	0.051	22.953	2.047

Table 16
N_{16x7}: Normalized Decision Matrix.

Feature Set	T	V_β	ARI	AMI	SC	CH	DBI
Uni	0.220	0.245	0.229	0.243	0.112	0.044	0.381
UniGEO	0.204	0.265	0.289	0.271	0.294	0.216	0.234
Bi	0.432	0.247	0.230	0.245	0.105	0.039	0.398
BiGEO	0.443	0.266	0.292	0.273	0.292	0.202	0.237
Tri	0.504	0.244	0.212	0.241	0.104	0.038	0.393
TriGEO	0.518	0.266	0.288	0.272	0.292	0.201	0.232
CB	0.006	0.203	0.128	0.183	0.169	0.252	0.210
CBGEO	0.007	0.224	0.179	0.213	0.220	0.250	0.201
SG	0.005	0.217	0.163	0.203	0.202	0.220	0.208
SGGEO	0.007	0.260	0.267	0.264	0.460	0.545	0.140
DB	0.019	0.260	0.279	0.263	0.416	0.408	0.125
DBGEO	0.020	0.265	0.300	0.270	0.439	0.428	0.121
DM	0.017	0.256	0.263	0.258	0.065	0.105	0.223
DMGEO	0.019	0.256	0.264	0.258	0.062	0.105	0.223
DBDM	0.039	0.257	0.269	0.259	0.058	0.108	0.222
DBDMGEO	0.040	0.257	0.270	0.260	0.043	0.108	0.224

2.2 and also indicated by our results, performance of various textual features for location-sensitive disaster event detection is not the same. In addition, location-augmentation improves efficiency of all traditional feature sets significantly. Hence, it is necessary to apply feature engineering and select the optimal feature set at an initial stage of the location-sensitive disaster event detection process so as to improve overall retrieval accuracy.

5. Practical applicability of the proposed framework

Location-sensitive event detection is beneficial in conducting fine-grained analysis in real-world applications such as *disaster surveillance systems for monitoring of disaster events at local and global level.* The online web is flooded with news about a disaster event with the moment of its occurrence. Subsequently, the number of posts reporting that event decrease rapidly. In order to gain more insight about the causes and to monitor the successive developments or after effects of the event, detecting the online content that reports about these events in future and associating the same to existing event class becomes much more significant. This study has focused on the detection of location-sensitive disaster events from retrospective news articles. For practical application of the location-sensitive disaster event detection in real-time online news articles, selection of efficient feature sets at the initial stage plays an important role. The proposed framework presents efficient feature sets and an unsupervised machine learning algorithm specific to the requirements of location-sensitive disaster event detection. The system is capable of distinguishing similar types of events that have happened at different locations from the immense and diverse content of news articles. Implementation of event detection using unsupervised machine learning algorithms enables the system to automatically interpret unstructured content of articles and eliminates the requirement of comprehensive and domain-specific training datasets. The use of context-based feature sets removes the limitation of vocabulary size, thereby, making the framework scalable to real-time online content. The location-augmented feature sets further improve the accuracy of the location-sensitive event detection. Efficient representation of text and important terms in the content at an initial stage will help initiate precise growth of similar content in real-time event classes leading to overall improved accuracy for future location-sensitive disaster event detection in online content.

6. Conclusion and future work

In this paper, GeoClust, a framework based on feature engineering for location-sensitive disaster event detection has been proposed which is based on selection of traditional and location-augmented feature sets by applying unsupervised machine learning algorithms and AHP-TOPSIS. The core inspiration behind the study was to facilitate location-sensitive event detection for news articles for disaster events like terrorist attacks, fire incidents, stampede incidents, building collapse and maoist attacks. Though the framework has been verified with a dataset of disaster events, it can be used for any kind of location-sensitive event detection. On the basis of extensive experiments and evaluation through a set of performance metrics, it has been found that the Skip-Gram model augmented with location features outperforms all other traditional and location augmented feature sets. The proposed framework provides evidence with the fact that selection of appropriate

Table 18
Euclidean Distance Measures (E^+ & E^-) and Closeness Score (C_s).

Feature Set	E^+	E^-	C_s
Uni	0.0735	0.0348	0.3214
UniGEO	0.0420	0.0622	0.5970
Bi	0.0773	0.0318	0.2912
BiGEO	0.0480	0.0600	0.5557
Tri	0.0797	0.0268	0.2518
TriGEO	0.0500	0.0591	0.5418
CB	0.0669	0.0425	0.3884
CBGEO	0.0537	0.0472	0.4677
SG	0.0597	0.0434	0.4210
SGGEO	0.0091	0.0891	0.9074
DB	0.0163	0.0807	0.8318
DBGEO	0.0125	0.0863	0.8732
DM	0.0640	0.0516	0.4463
DMGEO	0.0642	0.0518	0.4469
DBDM	0.0640	0.0523	0.4498
DBDMGEO	0.0651	0.0525	0.4466

Table 17
M_{16x7}: Weighted Normalized Matrix.

Feature Set	T	V_β	ARI	AMI	SC	CH	DBI
Uni	0.0116	0.0387	0.0603	0.0511	0.0118	0.0046	0.0401
UniGEO	0.0107	0.0419	0.0761	0.0571	0.0310	0.0228	0.0247
Bi	0.0227	0.0390	0.0606	0.0517	0.0110	0.0041	0.0419
BiGEO	0.0234	0.0421	0.0768	0.0574	0.0308	0.0213	0.0250
Tri	0.0265	0.0385	0.0557	0.0508	0.0110	0.0040	0.0414
TriGEO	0.0273	0.0420	0.0757	0.0572	0.0307	0.0212	0.0245
CB	0.0003	0.0320	0.0338	0.0385	0.0178	0.0266	0.0221
CBGEO	0.0004	0.0354	0.0470	0.0449	0.0231	0.0264	0.0212
SG	0.0003	0.0342	0.0429	0.0428	0.0213	0.0231	0.0219
SGGEO	0.0004	0.0411	0.0703	0.0556	0.0484	0.0574	0.0148
DB	0.0010	0.0410	0.0734	0.0555	0.0437	0.0429	0.0131
DBGEO	0.0011	0.0418	0.0789	0.0569	0.0462	0.0451	0.0127
DM	0.0009	0.0404	0.0691	0.0543	0.0069	0.0110	0.0235
DMGEO	0.0010	0.0405	0.0695	0.0544	0.0065	0.0110	0.0234
DBDM	0.0021	0.0406	0.0707	0.0546	0.0061	0.0114	0.0233
DBDMGEO	0.0021	0.0406	0.0712	0.0547	0.0045	0.0114	0.0236

algorithm and feature set yields substantial improvement in the retrieval accuracy of the location-sensitive event detection system as a whole. While the findings made in this study are quite significant, the context-based models of location and text remain largely unexplored. As a consequence, the most natural course of future extension is to work towards modelling context of location features embedded in the text. The proposed framework is capable of detecting location-sensitive disaster events in news articles published over time by associating them to previous event classes. However, it requires further extension and evaluation for new event detection. In addition, as a future study, the applicability of the framework for social media posts and other techniques of MCDM such as α -Discounting Method for Multi Criteria Decision Making (α-D MCDM) can also be evaluated for feature set selection from datasets for location-sensitive event detection.

CRediT authorship contribution statement

Monika Rani: Conceptualization, Methodology, Software, Data curation, Writing – original draft, Visualization, Investigation, Validation, Writing – review & editing. **Sakshi Kaushal:** Conceptualization, Methodology, Formal analysis, Supervision, Writing – review & editing.

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

Data will be made available on request.

References

Alsaedi, N., Burnap, P., & Rana, O. (2017). Can we predict a riot? Disruptive event detection using twitter. *ACM Transactions on Internet Technology, 17*(2), Article 18. https://doi.org/10.1145/2996183

Ankerst, M., Breunig, M. M., Kriegel, H. P., & Sander, J. (1999). OPTICS: Ordering points to identify the clustering structure. *ACM SIGMOD International Conference on Management of Data, 28*(2), 49–60. https://doi.org/10.1145/304181.304187

Bendimerad, A., Plantevit, M., Robardet, C., & Amer-Yahia, S. (2021). User-driven geolocated event detection in social media. *IEEE Transactions on Knowledge and Data Engineering, 33*(2), 796–809. https://doi.org/10.1109/TKDE.2019.2931340

Bsoul, Q., Salim, J., & Zakaria, L. Q. (2013). An intelligent document clustering approach to detect crime patterns. *Procedia Technology, 11*(Iceei), 1181–1187. https://doi.org/10.1016/j.protcy.2013.12.311

Caliński, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics, 3*(1), 1–27. https://doi.org/10.1080/03610927408827101

Ceroni, A., Gadiraju, U., & Fisichella, M. (2015). Improving event detection by automatically assessing validity of event occurrence in text. In *International Conference on Information and Knowledge Management*. https://doi.org/10.1145/2806416.2806624

Cheng, T., & Wicks, T. (2014). Event detection using twitter: A spatio-temporal approach. *PLoS ONE, 9*(6), 1–10. https://doi.org/10.1371/journal.pone.0097807

Choi, D., Park, S., Ham, D., Lim, H., Bok, K., & Yoo, J. (2021). Local event detection scheme by analyzing relevant documents in social networks. *Applied Sciences, 11*, 1–18. https://doi.org/10.3390/app11020577

Cybulska, A., & Vossen, P. (2010). Event models for historical perspectives: Determining relations between high and low level events in text, based on the classification of time, location and participants. *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, 3355–3362.

Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1(2)*, 224–227. https://doi.org/10.1109/TPAMI.1979.4766909

Day, W. H. E., & Edelsbrunner, H. (1984). Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification, 1*(1), 7–24. https://doi.org/10.1007/BF01890115

Duckstein, L., & Opricovic, S. (1980). Multiobjective optimization in river basin development. *Water Resources Research, 16*(1), 14–20. https://doi.org/10.1029/WR016i001p00014

Edouard, A., Cabrio, E., Tonelli, S., & Le-Thanh, N. (2017). Graph-based event extraction from twitter. *International Conference Recent Advances in Natural Language Processing, RANLP, 2017-Septe*, 222–230. 10.26615/978-954-452-049-6-031.

Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (pp. 226–231).

Frey, B. J., & Dueck, D. (2007). Clustering by passing messages between data points. *Science, 315*(5814), 972–976. https://doi.org/10.1126/science.1136800

Gyani, J., Ahmed, A., & Haq, M. A. (2022). MCDM and various prioritization methods in AHP for CSS: A comprehensive review. *IEEE Access, 10*, 33492–33511. https://doi.org/10.1109/ACCESS.2022.3161742

Heravi, B. R., Morrison, D., Khare, P., & Marchand-Maillet, S. (2014). Where is the news breaking? Towards a location-based event detection framework for journalists. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8326 LNCS*(PART 2), 192–204. 10.1007/978-3-319-04117-9_18.

Hu, L., Zhang, B., Hou, L., & Li, J. (2017). Adaptive online event detection in news streams. *Knowledge-Based Systems, 138*, 105–112. https://doi.org/10.1016/j.knosys.2017.09.039

Huang, Y., Li, Y., & Shan, J. (2018). Spatial-temporal event detection from geo-tagged tweets. *ISPRS International Journal of Geo-Information*, *7*(4), Article 150. 10.3390/ijgi7040150.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*(1), 193–218. https://doi.org/10.1007/BF01908075

Hwang, C.-L., & Yoon, K. (1981). *Methods for Multiple Attribute Decision Making*. 58–191. 10.1007/978-3-642-48318-9_3.

Jang, B., Kim, I., & Kim, J. W. (2019). Word2vec convolutional neural networks for classification of news articles and tweets. *PLoS ONE, 14*(8), 1–20. https://doi.org/10.1371/journal.pone.0220976

Kumaran, G., & Allan, J. (2004). Text classification and named entities for new event detection. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 297–304). https://doi.org/10.1145/1008992.1009044

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. *Proceedings of the 31st International Conference on Machine Learning, ICML 2014*, *32*, 1188–1196.

Li, Y., Li, Q., & Shan, J. (2017). Discover patterns and mobility of twitter users-a study of four US college cities. *International Journal of Geo-Information, 6*, Article 42. https://doi.org/10.3390/ijgi6020042

Li, Z., Wang, B., Li, M., & Ma, W. Y. (2005). A probabilistic model for retrospective news event detection. In *SIGIR 2005 - Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 106–113). https://doi.org/10.1145/1076034.1076055

Liu, B., Han, F. X., Niu, D., Kong, L., Lai, K., & Xu, Y. (2020). Story forest: Extracting events and telling stories from breaking news. *ACM Transactions on Knowledge Discovery from Data, 14*(3), Article 31. https://doi.org/10.1145/3377939

Liu, Y., Zhou, B., Chen, F., & Cheung, D. W. (2016). Graph topic scan statistic for spatial event detection. In *International Conference on Information and Knowledge Management*. https://doi.org/10.1145/2983323.2983744

MacQueen, J. (1967). *Some methods for classification and analysis of multivariate observations*. 281–297.

Marzouk, M., & Sabbah, M. (2021). AHP-TOPSIS social sustainability approach for selecting supplier in construction supply chain. *Cleaner Environmental Systems, 2*, 1–9. https://doi.org/10.1016/j.cesys.2021.100034

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *Proceeding of the International Conference on Learning Representations (ICLR 2013)*, 1–12.

Odon De Alencar, R., Davis, C. A., & Gonçalves, M. A. (2010). Geographical classification of documents using evidence from Wikipedia. *Proceedings of the 6th Workshop on Geographic Information Retrieval, GIR'10*, 1–8. 10.1145/1722080.1722096.

Ou-Yang, L. (2018). *Newspaper3k 0.2.8*. https://pypi.org/project/newspaper3k/.

Pan, C. C., & Mitra, P. (2011). Event detection with spatial latent Dirichlet allocation. *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, 349–358. 10.1145/1998076.1998141.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2021). *Clustering*. Retrieved From. https://scikit-learn.org/stable/modules/clustering.html#clustering-performance-evaluation.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research, 12*, 2825–2830.

Pothen, A., Simon, H. D., & Liou, K. P. (1990). Partitioning sparse matrices with eigenvectors of graphs. *SIAM Journal on Matrix Analysis and Applications, 11*(3), 430–452. https://doi.org/10.1137/0611030

Rasouli, E., Zarifzadeh, S., & Rafsanjani, A. J. (2019). WebKey: A graph-based method for event detection in web news. *Journal of Intelligent Information Systems, 54*, 585–604. https://doi.org/10.1007/s10844-019-00576-7

Repp, Ø., & Ramampiaro, H. (2018). Extracting news events from microblogs. *ArXiv, 1–17*. https://doi.org/10.1080/09720510.2018.1486273

Robindro, K., Nilakanta, K., Naorem, D., & Singh, N. G. (2017). An unsupervised content based news personalization using geolocation information. *Proceeding – IEEE International Conference on Computing, Communication and Automation, ICCCA, 2017*, 128–132. https://doi.org/10.1109/CCAA.2017.8229785

Rosenberg, A., & Hirschberg, J. (2007). V-Measure: A conditional entropy-based external cluster evaluation measure. *EMNLP-CoNLL 2007 - Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June*, 410–420.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics, 20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Saaty, T. L. (1984). The analytic hierarchy process: decision making in complex environments. *Quantitative Assessment in Arms Control, 285–308*. https://doi.org/10.1007/978-1-4613-2805-6_12

Sałabun, W. (2014). The characteristic objects method: A new distance-based approach to multicriteria decision-making problems. *Journal of Multi-Criteria Decision Analysis, 22*, 37–50. https://doi.org/10.1002/mcda.1525

Sculley, D. (2010). Web-scale k-means clustering. *Proceedings of the 19th International Conference on World Wide Web, WWW '10*, 1177–1178. 10.1145/1772690.1772862.

Sloan, L., & Morgan, J. (2015). Who tweets with their location? Understanding the relationship between demographic characteristics and the use of geoservices and geotagging on twitter. *PLoS ONE, 10*(11), 1–15. https://doi.org/10.1371/journal.pone.0142209

Smarandache, F. (2010). α-Discounting Method for Multi-Criteria Decision Making (α-D MCDM). *Review of the Air Force Academy/The Scientific Informative Review, 2*, 29–42.

Smarandache, F. (2013a). Interval α-Discounting Method for MCDM. In *Proceedings of the Annual Symposium of the Institute of Solid Mechanics and Session of the Commission of Acoustics (The XXIVth SISOM)* (pp. 27–32).

Smarandache, F. (2013b). Three Non-linear α-Discounting MCDM-Method Examples. *Proceedings of The 2013 International Conference on Advanced Mechatronic Systems (ICAMechS 2013)*, 174–176.

Smarandache, F. (2015). *α-Discounting Method for Multi-Criteria Decision Making (α-D MCDM)*. Romania & Educational Publisher.

Smith, D. A. (2002). Detecting events with date and place information in unstructured text. In *Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries* (pp. 191–196). https://doi.org/10.1145/544220.544260

Sotoudeh-anvari, A. (2022). The applications of MCDM methods in COVID-19 pandemic : A state of the art review. *Applied Soft Computing, 126*, 1–40. https://doi.org/10.1016/j.asoc.2022.109238

Valentin, S., Lancelot, R., & Roche, M. (2018). How to combine spatio-temporal and thematic features in online news for enhanced animal disease surveillance? *Procedia Computer Science, 126*, 490–497. https://doi.org/10.1016/j.procS.2018.07.283

Vinh, N. X., Epps, J., & Bailey, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research, 11*, 2837–2854.

Visheratin, A. A., Mukhina, K. D., Visheratina, A. K., Nasonov, D., & Boukhanovsky, A. V. (2018). Multiscale event detection using convolutional quadtrees and adaptive geogrids. *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Analytics for Local Events and News, LENS 2018*. 10.1145/3282866.3282867.

Wang, S., Bao, F., & Gao, G. (2019). Research on new event detection methods for mongolian news. In *Proceedings of the 2019 International Conference on Asian Language Processing*. https://doi.org/10.1109/IALP48816.2019.9037708

Wikipedia. (2021a). *Category:Building collapses in India.* Retrieved From. https://en.wikipedia.org/wiki/Category:Building_collapses_in_India.

Wikipedia. (2021b). *Category:Fires in India*. Retrieved From. https://en.wikipedia.org/wiki/Category:Fires_in_India.

Wikipedia. (2021c). *Category:Human stampedes in India*. Retrieved From. https://en.wikipedia.org/wiki/Category:Human_stampedes_in_India.

Wikipedia. (2021d). *List of terrorist incidents in India*. Retrieved From. https://en.wikipedia.org/wiki/List_of_terrorist_incidents_in_India.

Wikipedia. (2021e). *Naxalite–Maoist insurgency*. Retrieved From. https://en.wikipedia.org/wiki/Naxalite–Maoist_insurgency.

Yasmeen, G., Karunasekera, S., Aaron, H., & Kwan, H. L. (2021). Real-time spatio-temporal event detection on geotagged social media. *Journal of Big Data, 8*, Article 91. https://doi.org/10.1186/s40537-021-00482-2

Zandebasiri, M., & Pourhashemi, M. (2016). The place of AHP method among the multi-criteria decision making methods in forest management. *International Journal of Applied Operational Research, 6*(2), 75–89.

Zhang, C., Lei, D., Yuan, Q., Zhuang, H., Kaplan, L., Wang, S., & Han, J. (2018). GeoBurst +: Effective and real-time local event detection in geo-tagged tweet streams. *ACM Transactions on Intelligent Systems and Technology, 9*(3), Article 34. https://doi.org/10.1145/3066166

Zhang, T., Ramakrishnan, R., & Livny, M. (1996). Birch. *ACM SIGMOD Record, 25*(2), 103–114. https://doi.org/10.1145/235968.233324