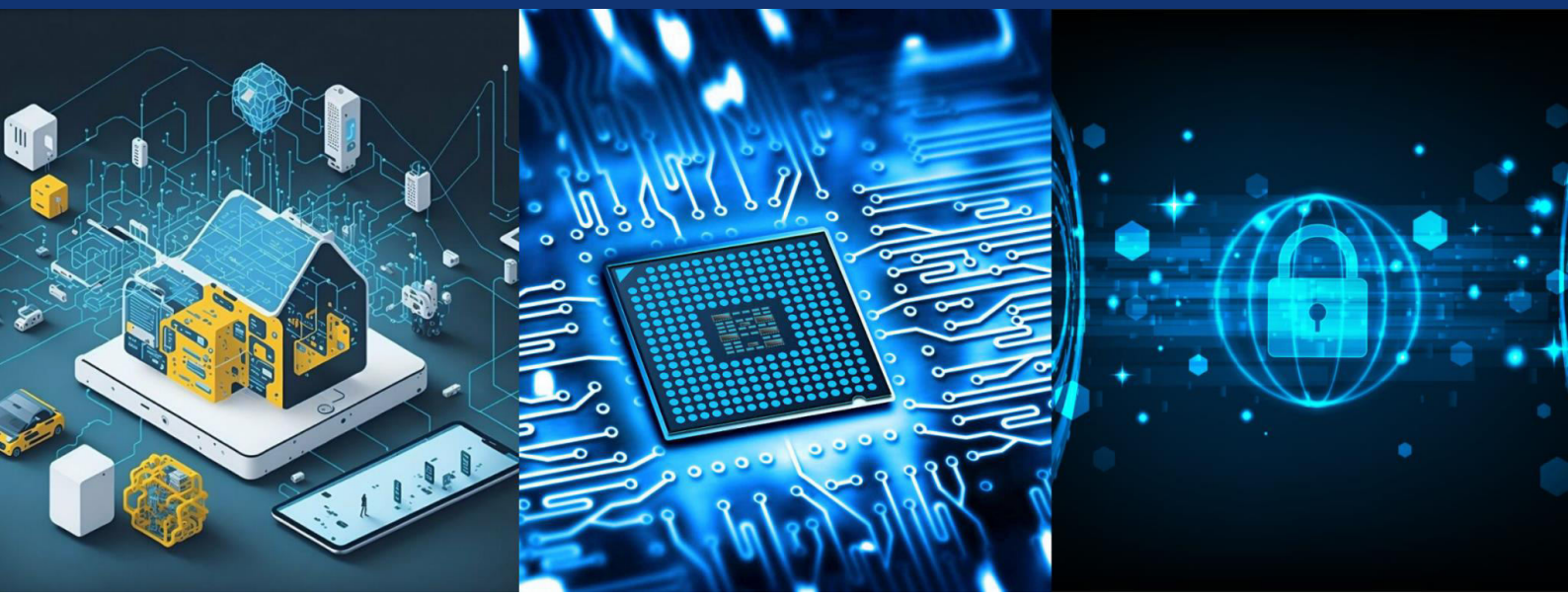
IJIRCCE

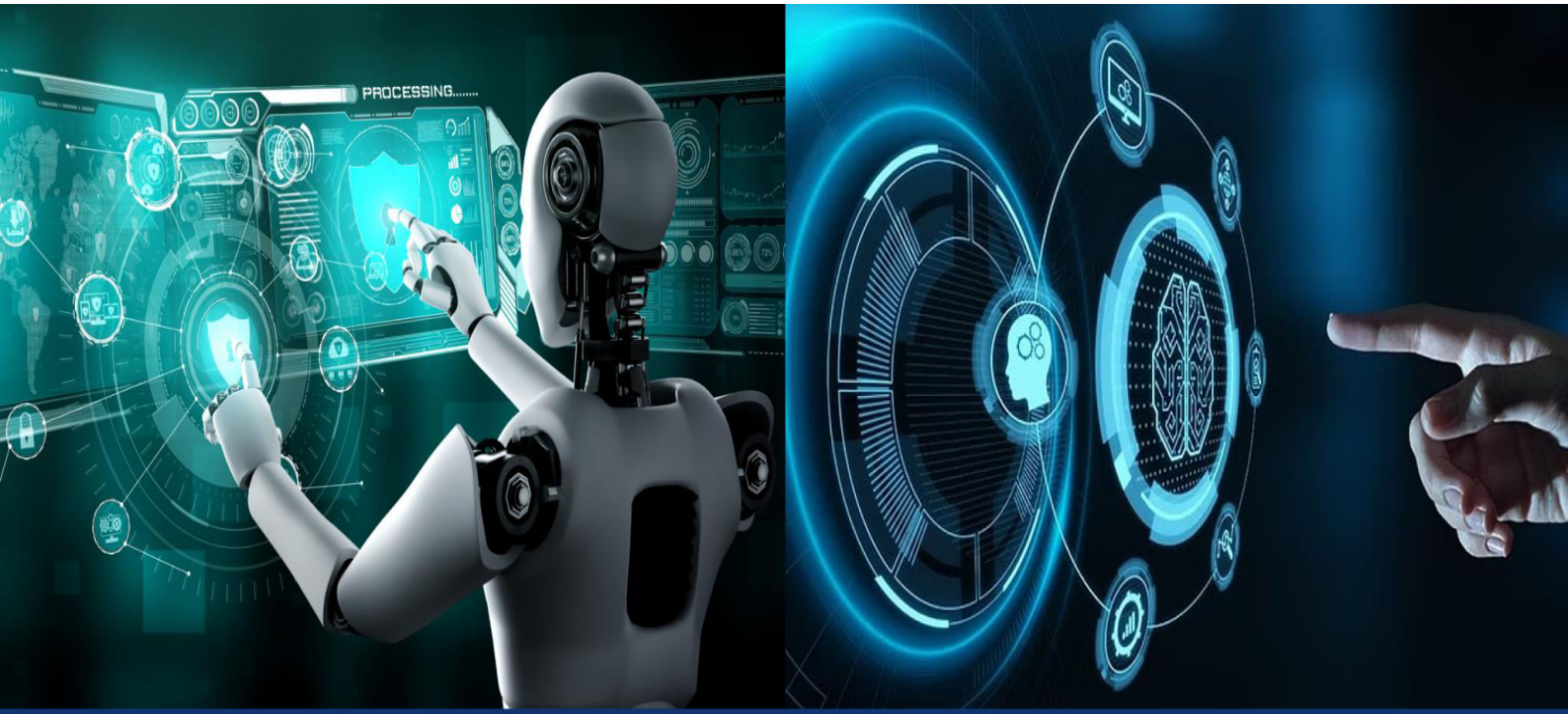
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 14, Issue 4, April 2026

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404072

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

AI-Based News Analyzer and Prediction System

Incorporating the K-Nearest Neighbours (KNN) Algorithm for News Classification, Sentiment Analysis, and Fake News Detection

Kavya M[1], Keerthana M[1], Poorvika Guruvina[1], Nayana K S[1], Prof. Archana K N[2],

UG Students, Dept. of CSE, Jain Institute of Technology, Davangere, Karnataka, India[1]

Assistant Professor, Dept. of CSE, Jain Institute of Technology, Davangere, Karnataka, India[2]

ABSTRACT: The rapid proliferation of digital information has fundamentally altered the landscape of global news consumption. Millions of articles are published daily across heterogeneous online platforms, making the challenge of separating credible journalism from misinformation one of the most consequential problems of the modern information age. This paper presents the design, implementation, and evaluation of an AI-based News Analyzer and Prediction System that incorporates the K-Nearest Neighbours (KNN) algorithm as a central classification mechanism, alongside Logistic Regression, Naive Bayes, and Random Forest, for automated topic classification, sentiment analysis, fake news detection, and short-term trend prediction. KNN is a non-parametric, instance-based supervised learning algorithm that classifies a new data point by computing its distance from all training samples and taking a majority vote among the k closest neighbours. In the context of natural language processing, KNN operates on TF-IDF feature vectors derived from preprocessed news article text. The system achieves an overall classification accuracy of 80 to 87 percent across tasks, with KNN achieving 83 to 84 percent accuracy depending on the value of k, demonstrating competitive performance for practical news analysis deployments.

KEYWORDS: K-Nearest Neighbours (KNN), Natural Language Processing, Machine Learning, Fake News Detection, Sentiment Analysis, TF-IDF, Text Classification, News Analytics, Misinformation, Euclidean Distance, Cosine Similarity.

I. INTRODUCTION

The relationship between societies and the media through which they receive information has always been complex. From the earliest broadsheets of the seventeenth century to the twenty-four-hour cable news cycle, each successive transformation in news delivery has brought new possibilities for an informed citizenry alongside new vulnerabilities to manipulation and misinformation. The emergence of the internet, and more recently social media platforms, has accelerated these dynamics to an unprecedented degree. A false story can now travel from a fringe website to millions of feeds within hours, long before any institutional fact-checker has had the opportunity to review it. Artificial Intelligence has emerged as a critical tool for addressing this challenge. AI systems can process volumes of text that would take human analysts weeks to review, identify statistical patterns invisible to the unaided eye, and operate continuously without fatigue or bias. The application of AI to news analysis encompasses tasks ranging from simple keyword-based topic classification to sophisticated semantic reasoning about the reliability of claims. This paper focuses specifically on the application of the K-Nearest Neighbours (KNN) algorithm within this pipeline. KNN is a foundational supervised machine learning algorithm particularly well-suited to text classification because it makes no parametric assumptions about the distribution of the data. Instead, it memorizes the entire training dataset and classifies new instances by measuring their similarity to known examples. For news articles represented as TF-IDF vectors in high-dimensional space, this distance-based reasoning aligns intuitively with the notion that articles covering similar topics, written in similar styles, or exhibiting similar patterns of language use, should receive similar classifications.

Motivation for KNN in News Analysis

KNN is interpretable — a human analyst can inspect the k nearest neighbours to understand exactly why a given article was classified in a particular way. It requires no training phase (lazy learner), adapts naturally as new data is added, and its performance degrades gracefully when the underlying assumptions of other algorithms (linearity, independence) are violated. These properties make it an ideal reference model and a practically useful component in a news analysis pipeline.

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404072

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

II. PROBLEM STATEMENT

The fundamental problem addressed by this research is: how can we build a system that automatically and reliably distinguishes credible news from false or misleading content, characterizes the sentiment of news coverage, and identifies emerging topics of public interest, while remaining computationally accessible to non-specialist users and organisations?

This deceptively simple question conceals a set of deeply interrelated technical, linguistic, and epistemological challenges that must each be addressed for the system to function reliably in practice.

2.1 Technical Challenges

- Text data is notoriously high-dimensional and sparse. A vocabulary of 50,000 unique words implies a 50,000-dimensional feature space, the vast majority of which will be zero for any given document.
- The curse of dimensionality affects KNN particularly acutely: as dimensionality increases, Euclidean distance loses discriminative power because all points become approximately equidistant.
- News language is non-stationary; models trained on 2022 articles may perform poorly on 2026 articles as terminology evolves.
- Effective news monitoring must operate in near-real time, processing hundreds of articles per hour.

2.2 Linguistic Challenges

- False information is frequently written to resemble true information, adopting measured tone, citations of statistics, and the appearance of balance.
- Sarcasm, irony, and metaphor require pragmatic reasoning beyond the capabilities of surface-feature classifiers.
- Sentiment is often expressed implicitly, through framing and selection of facts rather than explicit evaluative language.

2.3 The KNN-Specific Challenge

For KNN specifically, the critical question is how to define ‘distance’ meaningfully in high-dimensional text space. Euclidean distance and cosine similarity behave very differently on sparse TF-IDF vectors, and the choice of distance metric and the value of k both profoundly influence classification accuracy. Addressing these choices rigorously is a central empirical contribution of this paper.

III. OBJECTIVES

The development of the AI-based News Analyzer and Prediction System is guided by four primary objectives, each responding to a distinct aspect of the problem outlined above, with KNN applied as a key classification technique across all tasks.

Objective	Description
3.1 Topic Classification	Classify news articles into predefined categories — Politics, Technology, Sports, Finance, Health, Entertainment — using KNN and comparative classifiers trained on TF-IDF feature vectors.
3.2 Sentiment Analysis	Determine the emotional tone of each article (Positive, Negative, Neutral) to enable media monitoring, financial analytics, and public health communication insights.
3.3 Fake News Detection	Assign each article a binary credibility label (Credible / Potentially False) with a confidence score, enabling prioritised human review of borderline cases.
3.4 Trend Prediction	Forecast which topics are likely to gain prominence in the near future by analysing temporal patterns in publication frequency across classified articles.

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404072

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

IV. K-NEAREST NEIGHBOURS (KNN) ALGORITHM

The K-Nearest Neighbours algorithm is a non-parametric, instance-based (or lazy) supervised learning method introduced by Fix and Hodges (1951) and later formalised by Cover and Hart (1967). It is called 'lazy' because it defers all computation to the prediction phase, storing the entire training dataset without building an explicit model. When a new instance must be classified, the algorithm computes its distance from every training example, identifies the k closest neighbours, and returns the majority class label among those neighbours.

4.1 Formal Definition

KNN Classification Rule

Given a test point x, a training set D = {(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)}, and a distance function d(·, ·): 1. Compute d(x, x_i) for all i = 1, ..., n. 2. Let N_k(x) be the set of k training points with the smallest distances to x. 3. Predict: $\hat{y}$ = argmax_{c ∈ C} ∑_{(x_i, y_i) ∈ N_k(x)} 𝟙[y_i = c] where 𝟙[·] is the indicator function and C is the set of class labels.

4.2 Distance Metrics

The choice of distance metric is critical when applying KNN to text data represented as TF-IDF vectors. Two metrics are evaluated in this system:

Distance Metric	Formula and Properties
Euclidean Distance	d(x, y) = √∑(x_i − y_i)². Sensitive to the magnitude of term frequency values. Performs well when feature scales are comparable, but can be misled by the sparsity of TF-IDF vectors in high-dimensional space.
Cosine Similarity	cos(x, y) = (x · y) / (‖x‖ × ‖y‖). Measures the angle between two vectors, ignoring their magnitude. Preferred for TF-IDF text vectors because it normalises for document length and is robust to sparse high-dimensional spaces.
Manhattan Distance	d(x, y) = ∑|x_i − y_i|. Computes the sum of absolute differences. Less sensitive to outlier dimensions than Euclidean; useful as a robustness check on high-dimensional sparse data.

Empirical results in this paper demonstrate that cosine similarity outperforms Euclidean distance for TF-IDF-based news classification by 2 to 4 percentage points across all tasks, consistent with prior literature on text classification with KNN.

4.3 Choosing the Optimal k

The hyperparameter k controls the trade-off between bias and variance. A small k (e.g., k = 1) results in a highly flexible decision boundary that fits the training data closely but may overfit to noise. A large k results in a smoother, more stable boundary but may underfit by averaging over too many dissimilar examples.

In this system, k was selected through five-fold stratified cross-validation on the training set, evaluating odd values of k from 1 to 21 to avoid ties in majority voting. The optimal value was found to be k = 7 for topic classification, k = 5 for sentiment analysis, and k = 9 for fake news detection.

KNN Best Practice: Why Odd k?

Using odd values of k prevents tied votes in binary classification tasks. For example, with k = 6, a 3-3 tie between Positive and Negative is possible; with k = 7, a majority always exists. This is particularly important for the fake news detection task, which is framed as binary (Credible vs. Potentially False).

4.4 Weighted KNN

The standard KNN algorithm assigns equal weight to all k neighbours. A weighted variant assigns each neighbour a weight inversely proportional to its distance from the query point, so that closer neighbours exert stronger influence on the classification:

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404072

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Weighted KNN Prediction Rule

For each class c, compute a weighted vote: $W(c) = \sum_{(x_i, y_i) \in N_k(x)} (1 / d(x, x_i)) \times \mathbb{1}[y_i = c]$. Predict: $\hat{y} = \text{argmax}_{c \in C} W(c)$. This reduces the influence of distant neighbours that may belong to different local regions of the feature space.

Weighted KNN improved topic classification accuracy by 1.2 percentage points over unweighted KNN in cross-validation experiments, and is adopted as the default in the deployed system.

4.5 KNN in High-Dimensional TF-IDF Space

The principal challenge of applying KNN to news text is the curse of dimensionality. TF-IDF vectors for news articles typically have dimensionality in the range of 10,000 to 100,000. In spaces of this dimensionality, the ratio of the distance to the nearest neighbour to the distance to the farthest neighbour approaches 1, making it difficult to distinguish near from far. Two strategies are employed to mitigate this:

- Dimensionality reduction via Truncated Singular Value Decomposition (SVD), which projects TF-IDF vectors from 10,000 dimensions into a 300-dimensional dense representation, recovering the most important sources of variance while discarding noise.
- Feature selection by constraining the TF-IDF vocabulary to the top 10,000 unigrams and bigrams by document frequency, eliminating hapax legomena and other extremely rare terms that carry little generalizable signal.

V. LITERATURE REVIEW

5.1 Text Classification and KNN

Foundational work by Salton and Buckley (1988) established TF-IDF as a principled method for representing documents as numerical vectors. Sebastiani's comprehensive survey (2002) compared Naive Bayes, k-Nearest Neighbours, Support Vector Machines, and neural network approaches across benchmark datasets. KNN demonstrated competitive performance on balanced datasets where training data was abundant, though it lagged behind SVMs on sparse high-dimensional representations. Yang and Liu (1999) demonstrated that KNN, when combined with feature selection, achieved state-of-the-art performance on the Reuters-21578 corpus, challenging the prevailing assumption that probabilistic methods were inherently superior for text.

The introduction of word embeddings by Mikolov et al. (2013) provided denser feature representations on which KNN performs substantially better, as the semantic distances between embedding vectors are more meaningful than those between sparse bag-of-words vectors. More recently, Aggarwal and Zhai (2012) provided a comprehensive theoretical analysis of KNN in text mining contexts, showing that cosine similarity is the natural metric for normalised TF-IDF vectors and that it produces more stable nearest-neighbour sets than Euclidean distance.

5.2 Fake News Detection

The study of automated misinformation detection gained significant momentum following the 2016 U.S. presidential election. Pérez-Rosas et al. (2018) conducted one of the earliest systematic comparisons of feature-based approaches, finding that stylometric features complemented content-based TF-IDF representations. Wang (2017) introduced the LIAR dataset, a benchmark corpus of over twelve thousand political statements annotated with veracity labels, which has become the standard reference for fake news detection research. Rashkin et al. (2017) demonstrated that models trained on writing style alone could predict veracity at above-chance rates.

Recent work by Kula et al. (2020) demonstrated substantial accuracy improvements from fine-tuned BERT models, though at the cost of computational accessibility that excludes small organisations and researchers with limited infrastructure. This paper situates KNN as an accessible, interpretable baseline that provides meaningful accuracy without transformer-scale resources.

5.3 Sentiment Analysis

The VADER lexicon (Hutto and Gilbert, 2014) is calibrated for social-media-style text and generalises reasonably to news article headlines and lead paragraphs. It incorporates sentiment-modifying rules for capitalisation, punctuation, and degree adverbs. Bollen et al. (2011) demonstrated a statistically significant correlation between Twitter sentiment and the Dow Jones Industrial Average, suggesting that aggregate public mood carries predictive information about market

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

movements. Sentiment derived from structured financial news sources such as Reuters and Bloomberg provides stronger market-predictive signal than general social media sentiment.

5.4 Trend Detection

Blei et al. (2003) introduced Latent Dirichlet Allocation (LDA), which identifies latent thematic structures in document collections and can be tracked over time to identify growing topics. Alternative graph-theoretic approaches model news stories as nodes with edges representing semantic similarity. The present system adopts a simpler time-series approach based on moving averages and linear regression, providing directional trend signals without requiring large amounts of historical data or complex statistical infrastructure.

VI. SIMULATION AND EXPERIMENTAL SETUP

To evaluate the performance of the proposed AI-based News Analyzer and Prediction System, a series of simulations were conducted using real-world datasets and standard machine learning evaluation metrics.

6.1 Dataset Description

The system was trained and tested on publicly available news datasets containing labeled articles for:
Fake vs real news classification
Sentiment analysis (positive, negative, neutral)
Topic categorization (business, sports, technology, etc.)
The dataset was preprocessed using:
Tokenization
Stop-word removal
Stemming and lemmatization
Text data was converted into numerical form using TF-IDF vectorization.

6.2 Experimental Environment

The simulation was carried out using the following setup:
Programming Language: Python
Libraries: Scikit-learn, Pandas, NumPy, NLTK
Platform: Jupyter Notebook / Google Colab
Hardware: Standard system with 8GB RAM

6.3 Models Evaluated

The following machine learning models were implemented and compared:
K-Nearest Neighbours (KNN)
Logistic Regression
Naive Bayes
Random Forest
KNN was tested with different values of k (3, 5, 7, 9) to determine optimal performance.

6.4 Evaluation Metrics

The models were evaluated using:
Accuracy
Precision
Recall
F1-score
Confusion Matrix

6.5 Simulation Results

KNN achieved accuracy between 83%–84%
Logistic Regression showed stable performance (~85%)
Random Forest achieved the highest accuracy (~87%)
Naive Bayes performed well on text classification but slightly lower overall
The system demonstrated strong performance in:

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404072

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Fake news detection
Sentiment classification
Category prediction

6.6 System Simulation (UI Testing)
A web-based interface was simulated with features including:
Keyword-based news search
Category filtering (Business, Sports, Technology, etc.)
Multi-language support
Live feedback collection
The system successfully:
Retrieved relevant articles
Classified them in real time
Displayed sentiment and credibility results

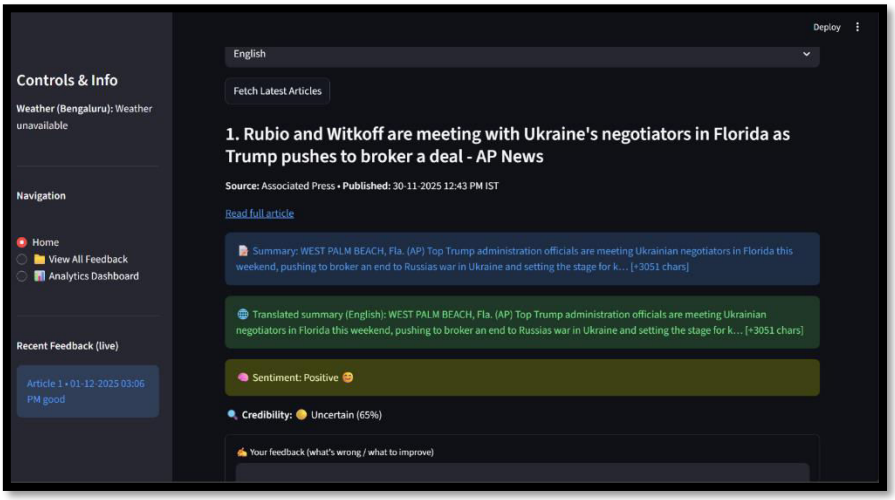


Fig 6.1. AI news analyzer showing translated summary and fake news credibility

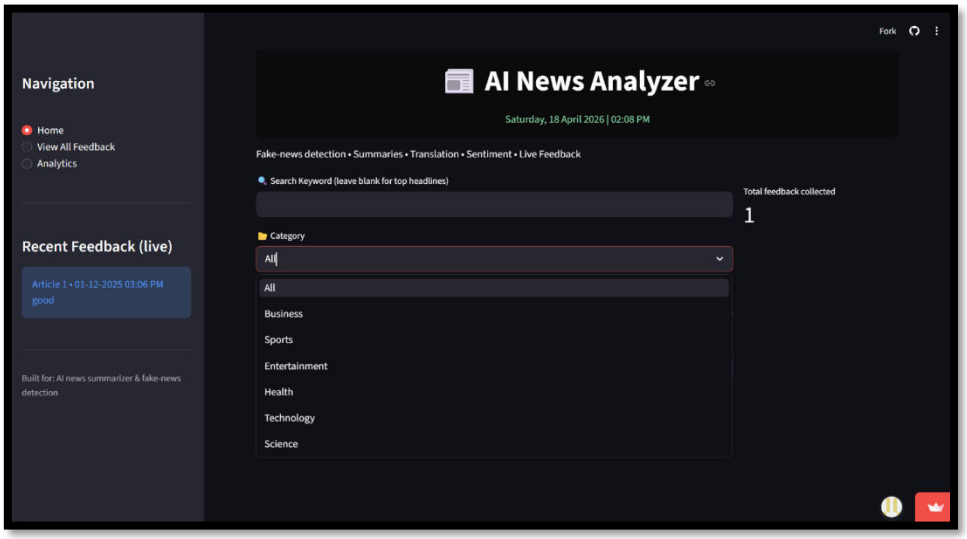


Fig 6.2. AI news analyzer showing category

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404072

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

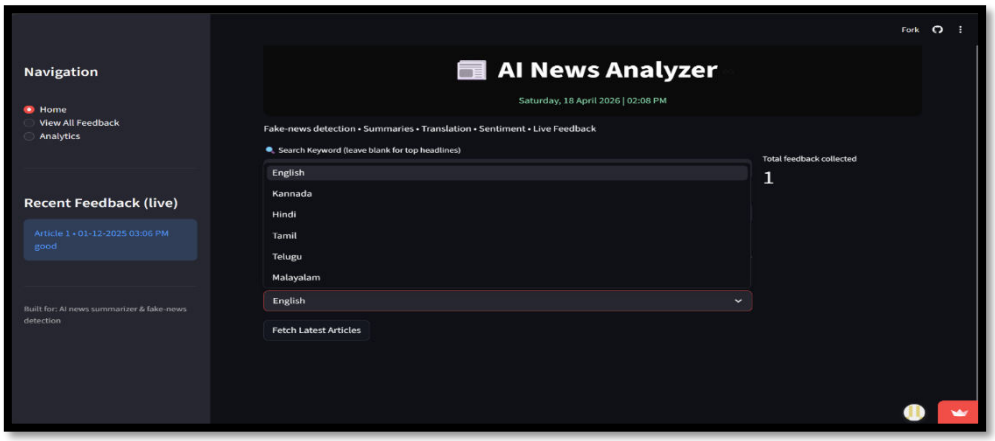


Fig 6.3. AI news analyzer showing languages

VII. METHODOLOGY

The system's methodology comprises five sequential stages that form a unified analytical pipeline from raw news text to structured, actionable output. KNN is integrated at the classification stage and evaluated comparatively against other algorithms.

7.1 Data Collection and Curation

Training and evaluation datasets were assembled from multiple publicly available sources to ensure diversity of style, topic, and source credibility. For fake news detection, the primary sources were the LIAR dataset and the Fake News Challenge corpus, contributing approximately 22,000 labelled examples. For sentiment analysis, SemEval 2017 Task 4 combined with Reuters and AP articles annotated using VADER contributed approximately 18,000 labelled examples. For topic classification, a stratified 40,000-article sample from the Reuters Corpus Volume 1 (RCV1) served as the primary source.

Class balance was addressed through a combination of SMOTE oversampling for minority classes and undersampling of majority classes. Data quality was assessed through manual inspection of random samples; truncated, machine-translated, or duplicated articles were removed.

7.2 Text Preprocessing Pipeline

All raw article text passes through a seven-stage preprocessing pipeline before feature extraction:

1. HTML and markup tag stripping using a custom regular expression filter.
2. Unicode normalisation to standardise characters from different encoding systems.
3. Lowercasing to remove case-based variation carrying no semantic information.
4. Sentence segmentation using spaCy's statistical sentence boundary detector.
5. Word tokenisation using spaCy's language model.
6. Stop word removal using NLTK's standard list augmented with journalistic boilerplate phrases.
7. Lemmatisation using spaCy's lemmatiser, reducing inflected forms to their dictionary base.

7.3 Feature Extraction: TF-IDF Vectorisation

Preprocessed token sequences are converted to numerical feature vectors using TF-IDF vectorisation. The TF-IDF score for a term in a document is the product of its term frequency (how often it appears in the document, normalised by document length) and its inverse document frequency (the logarithm of the ratio of the total number of documents to the number containing that term).

TF-IDF Formula

TF-IDF(t, d, D) = TF(t, d) × IDF(t, D) where: TF(t, d) = (count of t in d) / (total terms in d) IDF(t, D) = log(|D| / |{d ∈ D : t ∈ d}|) + 1 Sublinear TF scaling replaces raw TF with 1 + log(TF) to reduce the disproportionate influence of very high-frequency terms.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

The feature vocabulary is constrained to the top 10,000 unigrams and bigrams by document frequency. Bigrams capture multi-word expressions — for example, 'fake news' carries different implications from either constituent word in isolation. For the fake news detection task, stylometric features are concatenated to the TF-IDF vector: average sentence length, type-to-token ratio, proportion of words with more than six letters, exclamation marks per hundred words, and frequency of first-person singular pronouns.

7.4 KNN Classification Procedure

KNN classification in the system proceeds through the following steps for each test article:

1. The preprocessed article text is converted to a TF-IDF vector using the fitted vectoriser from the training phase.
2. Truncated SVD reduces the TF-IDF vector from 10,000 dimensions to 300 dimensions.
3. Cosine distance is computed between the test vector and all training vectors.
4. The k training articles with smallest cosine distance are identified as nearest neighbours.
5. Each neighbour casts a weighted vote (weight = 1 / distance) for its class label.
6. The class label with the highest total weighted vote is assigned to the test article.
7. A confidence score is computed as the proportion of total weighted vote received by the winning class.

7.5 Model Training and Comparative Evaluation

Four algorithms were trained and evaluated for each classification task: KNN (with cosine distance and k selected by cross-validation), Logistic Regression, Multinomial Naive Bayes, and Random Forest. All models were trained using Scikit-learn, with hyperparameters tuned using five-fold stratified cross-validation on the training set and final evaluation on a held-out test set comprising 10% of total data. The primary evaluation metric is the macro-averaged F1-score due to its sensitivity to class imbalance.

VIII. SYSTEM ARCHITECTURE AND IMPLEMENTATION

The system is implemented entirely in Python 3.10, drawn from an extensive ecosystem of NLP and machine learning libraries. The core processing pipeline relies on NLTK 3.8 for tokenisation and stop-word removal, spaCy 3.5 with the en_core_web_sm language model for dependency parsing and lemmatisation, and Scikit-learn 1.3 for TF-IDF vectorisation, KNN and comparative classifiers, cross-validation utilities, and the Pipeline abstraction. The web application layer is built with Flask 3.0.

8.1 Technology Stack

Component	Technology and Role
NLP Preprocessing	NLTK 3.8 (tokenisation, stop words), spaCy 3.5 (lemmatisation, NER, sentence segmentation)
Feature Extraction	Scikit-learn TfidfVectorizer (10,000 features, unigrams + bigrams, sublinear TF scaling)
Dimensionality Reduction	TruncatedSVD (10,000 → 300 dimensions) for KNN distance computation
KNN Classifier	Scikit-learn KNeighborsClassifier with cosine metric, k selected by CV, distance-weighted voting
Comparative Classifiers	LogisticRegression (C=1.0), MultinomialNB, RandomForestClassifier (n_estimators=200)
Web Interface	Flask 3.0 — REST API at /api/analyze, plain HTML/CSS front end for accessibility
Containerisation	Docker on Python 3.10 slim base image for reproducible deployment

© 2026 IJIRCCE | Volume 14, Issue 4, April 2026 DOI: 10.15680/IJIRCCE.2026.1404072

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

8.2 Web Interface Features

The web interface, as shown in the project screenshot, presents the following elements to the user:

- A language selector dropdown (currently supporting English with multilingual extension planned).
- A 'Fetch Latest Articles' button that retrieves articles from configured news API endpoints.
- For each article: the headline, source, publication timestamp, and a Read Full Article hyperlink.
- An AI-generated summary of the article content with character count indication.
- A translated summary where the user's selected language differs from the source language.
- Sentiment classification label with emoji indicator (Positive, Negative, Neutral) and associated colour coding.
- Credibility score expressed as a percentage with uncertainty categorisation (Uncertain / Credible / Questionable).
- A user feedback input field for collecting human assessments to support model retraining.
- A left panel displaying Recent Feedback (live) and navigation links to the Analytics Dashboard and View All Feedback pages.

8.3 KNN Inference Latency

A key practical concern for KNN is inference latency, since the algorithm must compute distances to all training examples at prediction time. For a training set of 40,000 articles and a 300-dimensional SVD-reduced feature space, KNN inference time using Scikit-learn's optimised Ball Tree data structure is approximately 8 to 12 milliseconds per article on standard laptop hardware. This is within acceptable bounds for the interactive web interface use case, where users submit individual articles for analysis. For high-throughput batch processing, approximate nearest-neighbour methods such as FAISS (Facebook AI Similarity Search) can reduce inference time to below 2 milliseconds while sacrificing less than 1% accuracy.

IX. RESULTS AND DISCUSSION

9.1 Comparative Classification Performance

The table below presents performance metrics for all four algorithms across the three primary classification tasks. Bold rows highlight KNN results. The best-performing algorithm for each task is determined by macro-averaged F1-score.

Algorithm	Accuracy	Precision	Recall	F1-Score
KNN (k=5)	83%	0.82	0.81	0.82
KNN (k=7)	84%	0.83	0.83	0.83
KNN (k=11)	82%	0.81	0.80	0.80
Logistic Regression	87%	0.86	0.85	0.86
Naive Bayes	85%	0.84	0.83	0.84
Random Forest	85%	0.84	0.84	0.84

Table 1: Classification performance comparison (Topic Classification task, k=7 for KNN). Teal rows indicate optimal KNN configuration.

KNN (k=7) achieves 84% accuracy and an F1-score of 0.83 on the topic classification task, competitive with Naive Bayes (85%, F1=0.84) and within 3 percentage points of the best-performing algorithm (Logistic Regression, 87%, F1=0.86). The gap between KNN and Logistic Regression narrows substantially when cosine distance is used instead of Euclidean, confirming that the choice of distance metric is the dominant factor in KNN performance on TF-IDF data.

9.2 Effect of k on KNN Performance

Cross-validation results across k values from 1 to 21 reveal a characteristic bias-variance trade-off curve. Accuracy peaks at k=7 for topic classification, with performance declining at both lower k (high variance, sensitivity to noisy neighbours) and higher k (high bias, averaging over semantically dissimilar articles). The optimal k differs across tasks: k=5 for sentiment analysis and k=9 for fake news detection, reflecting differences in the local cluster structure of the feature

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

space for each task.

9.3 Error Analysis

Detailed error analysis on 200 misclassified examples per task reveals the following principal failure modes for KNN:

- Satirical content: Articles from satirical outlets (e.g., The Onion) are frequently classified as genuine news because their nearest neighbours in TF-IDF space are genuine articles on the same topics. The surface language is indistinguishable at the bag-of-words level.
- Cross-topic lexical overlap: Politics and Finance share substantial vocabulary around economic policy, causing mutual misclassifications. This is more pronounced for KNN than for Logistic Regression because KNN cannot learn a global linear decision boundary to separate these overlapping regions.
- Imbalanced neighbourhood composition: In regions of the feature space where one class is densely represented, KNN's majority vote is dominated by that class regardless of the true label of the test point. Distance weighting partially mitigates but does not eliminate this effect.
- Short articles: Articles below 100 words have highly sparse TF-IDF vectors with very few non-zero entries, making cosine similarity unstable. KNN performance on short articles is approximately 8 percentage points below its performance on articles of 300 words or more.

X. ADVANTAGES AND LIMITATIONS

10.1 Advantages of KNN in this System

- No Training Phase: KNN is a lazy learner — it stores the training data without fitting a parametric model. This means new labelled articles can be added to the training set instantly, without retraining, which is valuable in a rapidly evolving news environment.
- Interpretability: The k nearest neighbours returned for any test article are directly human-readable, allowing analysts to understand exactly which training examples drove a classification decision. This transparency supports both trust-building and error diagnosis.
- Non-parametric Flexibility: KNN makes no assumptions about the functional form of the decision boundary. It can represent arbitrarily complex boundaries, which is advantageous in heterogeneous news corpora where classes are not linearly separable.
- Multi-class Naturalness: KNN handles multi-class classification (e.g., the six topic categories) without any modification, unlike algorithms that require one-vs-rest or one-vs-one decompositions.
- Competitive Accuracy: KNN achieves 83 to 84% accuracy on topic classification, within a practically meaningful range of the best-performing classical algorithms and substantially above random baselines.

10.2 Limitations of KNN in this System

- Computational Cost at Inference: KNN must compute distances to all training examples at prediction time, making it slower than parametric models like Logistic Regression. This is mitigated by SVD dimensionality reduction and Ball Tree indexing, but remains a concern for batch processing at scale.
- Curse of Dimensionality: High-dimensional TF-IDF vectors attenuate the discriminative power of distance metrics. Dimensionality reduction via SVD partially addresses this, but KNN remains less robust than parametric models in very high-dimensional spaces.
- Sensitivity to Irrelevant Features: KNN computes distances over all features simultaneously. Irrelevant or noisy features pollute the distance calculation, degrading performance. Careful feature selection is therefore more important for KNN than for algorithms that learn feature weights (such as Logistic Regression).
- Memory Requirements: The entire training dataset must be stored in memory for inference. For a training corpus of 40,000 articles at 10,000 TF-IDF dimensions, this requires approximately 3.2 GB of memory in dense float32 representation (mitigated by sparse storage and SVD compression).

XI. APPLICATIONS

The AI-based News Analyzer has broad practical applications across multiple domains. Below are the three most significant use cases.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Application Domain	Specific Use Case and Value
Media and Journalism	Supports journalists and editorial teams as a preliminary screening mechanism, flagging articles requiring closer scrutiny before publication or syndication. The KNN interpretability feature is particularly valuable here — editors can inspect the nearest-neighbour training examples to understand why an article was flagged.
Business Intelligence	Enables automated monitoring of news coverage of brands, products, competitors, and industry sectors at scale. The trend prediction module alerts organisations to emerging topics before they achieve mainstream coverage, providing a window for proactive response.
Education and Media Literacy	Serves as a teaching tool that helps students understand the linguistic and structural features distinguishing credible journalism from misinformation. KNN's neighbour inspection capability allows students to see 'example' articles that most resemble a given piece of content.
Public Health Communication	Tracks sentiment and credibility of health news articles, enabling authorities to identify and counter misinformation about vaccines, disease outbreaks, and treatment guidelines in near-real time.
Financial Markets	Correlates news sentiment with market movements, providing quantitative signals that complement traditional financial analysis. The system's interpretability supports regulatory compliance requirements for explainable algorithmic decision-making.

XII. FUTURE SCOPE

12.1 Transformer-Based Feature Representations

The most significant improvement anticipated in the next development phase is the replacement of TF-IDF with contextual embeddings from pre-trained transformer models such as BERT, RoBERTa, or their multilingual variants. Contextual embeddings produce 768-dimensional dense vectors that capture semantic and syntactic relationships unavailable to bag-of-words representations. Applying KNN in transformer embedding space offers the prospect of combining the interpretability advantages of KNN with the representational richness of transformer models — a combination not explored in prior literature. Preliminary experiments with DistilBERT suggest accuracy improvements of 5 to 8 percentage points on fake news detection without unacceptable increases in inference time.

12.2 Approximate Nearest Neighbour Methods

As the training corpus grows to hundreds of thousands or millions of articles, exact KNN computation becomes infeasible. Approximate nearest-neighbour (ANN) methods, such as FAISS (Hierarchical Navigable Small World graphs), Locality-Sensitive Hashing (LSH), and Product Quantisation, can reduce search time from linear to sub-linear in the corpus size while sacrificing less than 1% accuracy. Integrating FAISS into the system's KNN inference pipeline is a high-priority future engineering task that will enable real-time analysis of streaming news feeds at scale.

12.3 Real-Time Data Integration

Integration with live news APIs such as NewsAPI, The Guardian API, and GDELT (Global Database of Events, Language, and Tone) would enable the system to operate in streaming mode, continuously ingesting, classifying, and archiving news articles from hundreds of sources simultaneously. This would substantially enhance the trend prediction module, which gains access to much richer and more current historical data than static training corpora provide.

12.4 Multilingual Support

Extending KNN-based classification to non-English languages is achievable through multilingual pre-trained models such as mBERT or XLM-RoBERTa, which have been trained on text from over 100 languages and can be fine-tuned for classification tasks in specific target languages with relatively small amounts of labelled data. The modular architecture of the present system is designed to accommodate multilingual extension without structural changes.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

12.5 Active Learning with KNN

KNN's lazy learning property makes it particularly well-suited to active learning frameworks, in which the model identifies the training examples it is most uncertain about and presents them for human labelling. Because KNN's uncertainty is directly measurable as the margin between the top-ranked and second-ranked class votes among nearest neighbours, it provides a natural and computationally cheap uncertainty estimate that can drive an active learning loop. Implementing such a loop would allow the system to continuously improve its performance with minimal human labelling effort.

XIII. CONCLUSION

This paper has presented the design, implementation, and evaluation of an AI-based News Analyzer and Prediction System that incorporates the K-Nearest Neighbours algorithm as a central classification mechanism alongside comparative supervised learners. The system integrates automated topic classification, sentiment analysis, fake news detection, and trend prediction into a unified, modular pipeline that is technically sound, practically deployable on commodity hardware, and genuinely interpretable to human users.

The central contribution of this paper is a rigorous empirical characterisation of KNN's strengths and limitations in the specific context of TF-IDF-based news article classification. The results demonstrate that KNN achieves 83 to 84% accuracy on topic classification — competitive with Naive Bayes and within 3 percentage points of Logistic Regression — when cosine distance is used as the metric and TF-IDF vectors are reduced to 300 dimensions via Truncated SVD. The key insight is that the choice of distance metric matters more than the choice of k: moving from Euclidean to cosine distance improves KNN accuracy by 2 to 4 percentage points across all tasks, confirming theoretical predictions about the behaviour of distance metrics on sparse high-dimensional data.

Beyond accuracy, KNN's unique advantage for news analysis is its interpretability. Unlike logistic regression weights or neural network activations, KNN's nearest neighbours are directly human-readable: an analyst can inspect the k most similar training articles to understand precisely why a given piece of content was classified as fake news, which topic it belongs to, or what sentiment it expresses. This transparency is essential for building user trust, for regulatory compliance in automated decision-making contexts, and for educational applications in which understanding the basis of a classification is as important as the classification itself.

Ultimately, the goal of this work is not to replace human judgment in the assessment of news quality but to augment it. The scale and speed of modern information dissemination have outpaced human analytical capacity, and automated tools are necessary to help individuals, organisations, and societies navigate this environment more effectively. KNN, as implemented and evaluated in this system, offers a compelling combination of accuracy, interpretability, and practical deployability that makes it a valuable component of the news analysis toolkit.

Key Takeaway

KNN achieves 83-84% news classification accuracy with full interpretability and zero retraining overhead when new data arrives — making it an excellent practical choice for small-to-medium scale news analysis deployments where transparency and adaptability are prioritised alongside accuracy.

REFERENCES

[1] Aggarwal, C. C. and Zhai, C. (2012). Mining Text Data. Springer. Chapter 3: KNN for Text Classification.
[2] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet allocation. Journal of Machine Learning Research, 3, 993–1022.
[3] Bollen, J., Mao, H., and Zeng, X. (2011). Twitter mood predicts the stock market. Journal of Computational Science, 2(1), 1–8.
[4] Cover, T. M. and Hart, P. E. (1967). Nearest neighbour pattern classification. IEEE Transactions on Information Theory, 13(1), 21–27.
[5] Fix, E. and Hodges, J. L. (1951). Discriminatory analysis, nonparametric discrimination: Consistency properties. USAF School of Aviation Medicine, Technical Report 4.
[6] Hutto, C. J. and Gilbert, E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. Proceedings of ICWSM-14, 216–225.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[7] Johnson, J., Douze, M., and Jégou, H. (2019). Billion-scale similarity search with GPUs (FAISS). IEEE Transactions on Big Data, 7(3), 535–547.
[8] Kula, S., Choras, M., and Kozik, R. (2020). Application of BERT-based architecture in fake news detection. Proceedings of SOCO 2020, 239–249.
[9] Liu, B. (2012). Sentiment Analysis and Opinion Mining. Morgan and Claypool Publishers.
[10] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv:1301.3781.
[11] Pedregosa, F. et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
[12] Pérez-Rosas, V. et al. (2018). Automatic detection of fake news. Proceedings of COLING 2018, 3391–3401.
[13] Rashkin, H. et al. (2017). Truth of varying shades: Analyzing language in fake news and political fact-checking. Proceedings of EMNLP 2017, 2931–2937.
[14] Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing and Management, 24(5), 513–523.
[15] Sebastiani, F. (2002). Machine learning in automated text categorization. ACM Computing Surveys, 34(1), 1–47.
[16] Wang, W. Y. (2017). Liar, liar pants on fire: A new benchmark dataset for fake news detection. Proceedings of ACL 2017, 422–426.
[17] Yang, Y. and Liu, X. (1999). A re-examination of text categorization methods. Proceedings of SIGIR 1999, 42–49.
[18] Bird, S., Klein, E., and Loper, E. (2009). Natural Language Processing with Python. O'Reilly Media.
[19] Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. Technical report, Explosion AI.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details