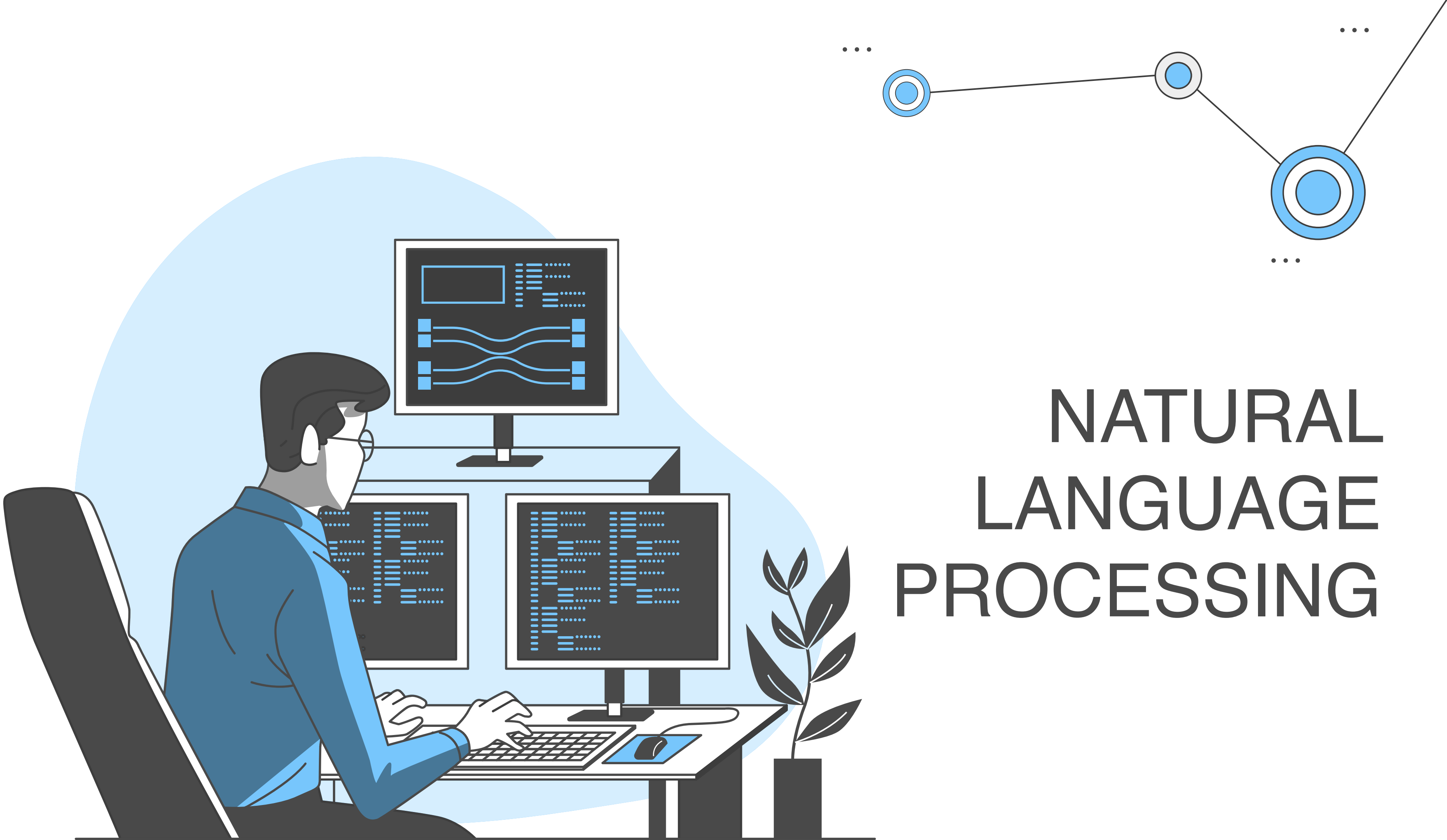


NATURAL LANGUAGE PROCESSING

FAKE NEWS DETECTION FOR LOW RESOURCE LANGUAGE

Objective

Fake news poses a significant threat to societal stability, influencing public opinion, inciting violence, and undermining trust in institutions. In low-resource language communities, the impact can be more severe due to limited access to counteracting resources.

The objective of this project is to develop an effective Natural Language Processing (NLP) model using machine learning (ML) and deep learning (DL) for detecting fake news in a low-resource language. The model aims to enhance the reliability of information dissemination and support the fight against misinformation in regions where linguistic resources are scarce.

DATASET

DFND is a Dravidian fake news dataset for detecting fake news in Dravidian languages, namely Telugu, Kannada, Tamil, and Malayalam. We collected the data from different sources: for real news articles, we scrapped the data from various news websites like Eenadu, Dinamalar, Kannadaprabha, Malayala manorama, etc.; for fake news articles, we scrapped the data from various fact-checking websites like factly, factcrescendo, etc. We collected the data from January 2021 to December 2022. After collecting the data, data preprocessing was performed through our designed script. The DFND dataset is preprocessed. This dataset contains more than 27,000 news articles which consist of 50% fake and 50% real news articles.

Text Vectorization

TF-IDF

A numerical statistic that reflects the importance of a word in a document relative to a collection of documents, balancing term frequency and inverse document frequency.

...

Bag of Words

A simple text representation method that treats each document as an unordered collection of words, disregarding grammar and word order, and represents it by word frequency vectors.

...

Word2Vec

A neural network-based embedding technique that maps words into continuous vector spaces where semantically similar words are positioned closer together.

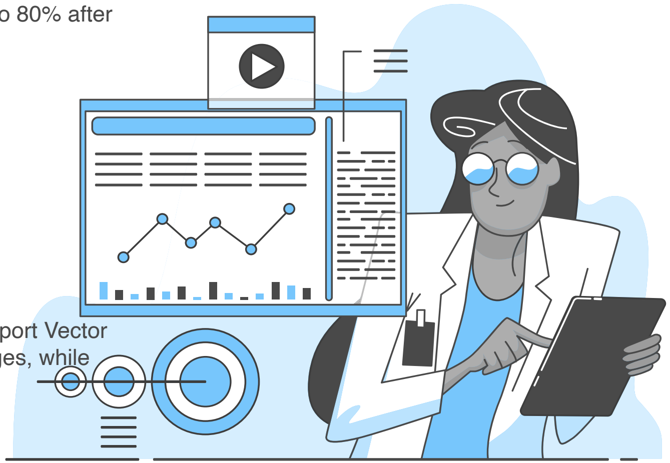
...

LITERATURE REVIEW

01

Fake News Detection using Machine Learning Algorithms

The study evaluated Naive Bayes, Random Forest, and Logistic Regression classifiers for fake news detection. Naive Bayes achieved 60% accuracy, Random Forest achieved 59% accuracy, and Logistic Regression initially had 65% accuracy, which improved to 80% after parameter optimization



02

Fake News Detection Using Machine Learning Approaches

The research paper on fake news detection utilized machine learning models such as Naive Bayes, Neural Network, Support Vector Machine (SVM), and Random Forest. Naive Bayes achieved an impressive accuracy of 96.08% in detecting fake messages, while Neural Network and SVM reached an accuracy of 99.90%

03

Fake News Detection Using Deep Learning Techniques

The paper proposes a fake news detection system utilizing Logistic regression (LR), Naïve bayes (NB), Support vector machine (SVM), Random forest (RF), and deep neural network (DNN) models. The comparison of these models shows that DNN achieved the highest accuracy of 91%, outperforming LR (75%), RF (77%), SVM (79%), and NB (89%)

Model Training & Visualization

01

Machine Learning

- Logistic Regression
- Decision Tree
- Random Forest
- XGBoost

02

Deep Learning

- LSTM
- MLP
- KANs

03

Visualization

- tSNE plot

ML METHODOLOGY

Textual data undergoes thorough preprocessing to standardize format and eliminate irrelevant characters. This cleaning ensures consistency and enhances subsequent analysis. Next, the text is converted into numerical feature vectors using two prevalent techniques: TF-IDF and Bag of Words.

TF-IDF assigns weights to words based on their frequency in the document relative to their frequency across all documents, while Bag of Words represents documents as frequency distributions of words. These methods prepare the data for machine learning algorithms. The subsequent stage involves training and evaluating classical machine learning algorithms such as Logistic Regression, Decision Trees, Random Forest, and XGBoost

DL APPROACH

Data Preparation: Initially, the textual data undergoes tokenization and padding using Keras utilities. This step ensures that the data is formatted appropriately for consumption by deep learning models.

Model Architecture: The architecture of the deep learning model is constructed using TensorFlow and Keras. It comprises a Sequential model consisting of an Embedding layer, an LSTM layer, and a Dense layer. The Embedding layer helps in representing words as dense vectors, capturing semantic similarities. The LSTM layer, a type of recurrent neural network (RNN), is employed to learn temporal dependencies in the sequential data. Finally, the Dense layer serves for classification tasks, providing the output layer of the model.

Model Training: The deep learning model is trained on the tokenized and padded data. During training, the model learns to map input text sequences to corresponding labels, gradually adjusting its internal parameters to minimize the defined loss function. After training, the model's performance is evaluated on a separate test set to assess its generalization ability and effectiveness in making predictions on unseen data.

RESULTS (TAMIL)

model	Accuracy	Precision (fake)	Recall (fake)	F1-score (fake)	Precision (real)	Reca (real)	F1-Score (Real)
LR (TF-IDF)	0.927	0.98	0.86	0.92	0.89	0.98	0.93
DT (TF-IDF)	0.880	0.88	0.87	0.87	0.88	0.89	0.89
RF (TF-IDF)	0.908	0.93	0.87	0.90	0.89	0.94	0.91
LR (BOW)	0.928	0.92	0.94	0.92	0.94	0.92	0.93
DT (BOW)	0.857	0.87	0.82	0.85	0.84	0.89	0.87
RF (BOW)	0.895	0.88	0.91	0.89	0.91	0.88	0.90
XGBOOST (TF-IDF)	0.801	0.93	0.65	0.77	0.95	0.95	0.73
LSTM (DL)	0.869	0.82	0.93	0.87	0.93	0.81	0.87

model	Accuracy	Precision (fake)	Recall (fake)	F1-score (fake)	Precision (real)	Reca (real)	Precision (real)
MLP	0.939	0.98	0.98	0.94	0.87	0.87	0.90

KANs Architecture

```
Column names in the CSV file: Index(['text', 'label'], dtype='object')
train loss: 2.98e-05 | test loss: 1.59e+00 | reg: 9.16e+02 : 100%|█| 20/20 [54:33<00:00, 163.66s/it]
Test Accuracy: 0.8357575757575757
Test F1 Score: 0.8355281788634382
Test Precision: 0.8360270552684346
Test Recall: 0.8357575757575757
Train Accuracy: 0.9894029850746268
Test Accuracy: 0.8357575757575757
```

RESULTS (MALAYALAM)

model	Accuracy	Precision (fake)	Recall (fake)	F1-score (fake)	Precision (real)	Reca (real)	F1-Score (Real)
LR (TF-IDF)	0.928	0.93	0.91	0.92	0.92	0.94	0.93
DT (TF-IDF)	0.885	0.91	0.85	0.88	0.87	0.92	0.89
RF (TF-IDF)	0.895	0.92	0.85	0.89	0.87	0.93	0.90
LR (BOW)	0.907	0.96	0.84	0.90	0.87	0.97	0.92
DT (BOW)	0.892	0.91	0.87	0.89	0.88	0.92	0.90
RF (BOW)	0.898	0.94	0.85	0.89	0.87	0.95	0.91
XGBOOST (TF-IDF)	0.816	0.92	0.67	0.78	0.76	0.95	0.84
LSTM (DL)	0.896	0.90	0.91	0.92	0.91	0.85	0.86

model	Accuracy	Precision (fake)	Recall (fake)	F1-score (fake)	Precision (real)	Reca (real)
MLP	0.905	0.97	0.93	0.94	89	0.90

CONCLUSION

Our methodology combines traditional machine learning and advanced deep learning techniques to effectively detect fake news in Tamil,Malayalam. Through rigorous preprocessing, feature engineering, and model training, we've demonstrated our approach's ability to identify misinformation and promote media literacy. Looking ahead, future research could explore enhancing model robustness through ensemble methods or leveraging emerging technologies like natural language processing for more nuanced analysis. Our results highlight the importance of interdisciplinary approaches in combating fake news and fostering informed decision-making in low-resource language contexts.