**Dataset Extraction**

**Fake news**

Before cleaning

| Source | Number of articles |
| --- | --- |
| OneIndia | 203 |
| Prajavani | 96 |
| Factly | 439 |
| Asianet | 1992 |
| **Total** | **2730** |

After cleaning

| Source | Number of articles |
| --- | --- |
| OneIndia | 203 |
| Prajavani | 74 |
| Factly | 438 |
| Asianet | 285 |
| **Total** | **1000** |

**True news**

Before cleaning

| Source | Number of articles |
| --- | --- |
| news18 | 2016 |
| Prajavani | 2014 |
| PublicTV | 2000 |
| Asianet | 2009 |
| **Total** | **8039** |

After cleaning

| Source | Number of articles |
| --- | --- |
| news18 | 2011 |
| Prajavani | 597 |
| PublicTV | 2000 |
| Asianet | 1082 |
| **Total** | **5690** |

**Train - test split : Across news sources**

Split ratio = 0.2

Train

| Source | Real | Fake |
| --- | --- | --- |
| news18 | 52 | - |
| Prajavani | 497 | 59 |
| PublicTV | 50 | - |
| Asianet | 901 | 228 |
| Factly | - | 350 |
| OneIndia | - | 162 |

Test

| Source | Real | Fake |
| --- | --- | --- |
| news18 | 10 | - |
| Prajavani | 100 | 15 |
| PublicTV | 10 | - |
| Asianet | 180 | 57 |
| Factly | - | 88 |
| OneIndia | - | 41 |

**TRAIN-TEST SPLIT - across real and fake news**

Train

Fake : Real = 1:1.875

| Real | Fake | Total |
| --- | --- | --- |
| 1500 | 800 | 2300 |

Test

Fake: Real = 1:1.5

| Real | Fake | Total |
| --- | --- | --- |
| 300 | 200 | 500 |

**Modeling aspect:**

| **Model** | **Accuracy** | **Precision** | **Recall** | **f1-score** | **TPR** |
| --- | --- | --- | --- | --- | --- |
| GPT-2 | 0.91 | 0.94 | 0.82 | 0.88 |  |
| mBert + CNN | 0.88 | 0.89 | 0.88 | 0.88 |  |
| mBert | 0.86 | 0.85 | 0.86 | 0.85 |  |
| Distil-mBERT | 0.85 | 0.85 | 0.84 | 0.85 |  |
| IndicBert + CNN | 0.84 | 0.85 | 0.84 | 0.84 |  |
| Fasttext + CNN | 0.82 | 0.82 | 0.82 | 0.81 |  |
| Fasttext + LSTM | 0.82 | 0.82 | 0.82 | 0.82 |  |
| LSTM | 0.81 | 0.81 | 0.81 | 0.81 |  |
| CNN | 0.81 | 0.81 | 0.81 | 0.81 |  |
| SBert + CNN | 0.81 | 0.82 | 0.81 | 0.80 |  |
| Indicft + LSTM | 0.76 | 0.80 | 0.76 | 0.76 |  |
| Bert + CNN | 0.76 | 0.80 | 0.76 | 0.73 |  |
| Ensemble model(LR, SVM) | 0.76 | 0.78 | 0.76 | 0.74 |  |
| Logistic Regression | 0.75 | 0.79 | 0.75 | 0.73 |  |
| SVM | 0.75 | 0.75 | 0.75 | 0.73 |  |
| IndicBert + LSTM | 0.72 | 0.75 | 0.72 | 0.72 |  |
| RoBerta + LSTM | 0.72 | 0.75 | 0.72 | 0.72 |  |
| Random Forest | 0.72 | 0.71 | 0.72 | 0.71 |  |
| Passive-aggressive classifier | 0.71 | 0.71 | 0.71 | 0.71 |  |
| XGBoost | 0.71 | 0.73 | 0.71 | 0.68 |  |
| Decision Tree | 0.66 | 0.67 | 0.66 | 0.66 |  |
| mBart + CNN | 0.62 | 0.40 | 0.62 | 0.5 |  |
| Naive Bayes | 0.61 | 0.68 | 0.61 | 0.60 |  |
| KNN | 0.43 | 0.60 | 0.43 | 0.31 |  |

**Outline:**

**7 pages**

1. Abstract (150-200 words)

3-4 sentences each

* 1. What is the problem statement?
  2. What is the solution we are providing?
  3. What are the results obtained?

1. Introduction (3-4 paragraphs < 1000 words)

Basic background

* 1. Why fake news detection in kannada?
  2. What is the motivation?
  3. Contribution in brief

1. Related work
   1. Consider english, other low resource languages
   2. 5-6 papers, one paragraph for each paper
   3. Highlight positive and negative aspects of each paper
   4. Last paragraph - research gaps in our work (no fake news detection system for kannada etc.)
2. Proposed approach

Include Pipeline

* 1. Dataset construction and pre-processing
  2. Data visualization - data distribution, word cloud
  3. Word embeddings
  4. Training and testing

1. Results and discussions

Results - Graphs, tables explaining results

Discussion - Interpret results

1. Conclusion and future scope

References