NLP course

EX-FAKE

FAKE NEWS EXPLAINED

Interim Presentation

Presented by Lihi Nofar,Tomer Portal, Aviv Elbaz

DESCRIPTION

<u>Input</u>: News article consisting of title, body text, author, subject, and date
<u>Output</u>: Binary label – Fake or Real
<u>NLP Task:</u> Text classification with an <u>explainability component</u> – identifying what contributed to the decision

Data and evaluation
<u>Source:</u> Fake and Real News Dataset – Kaggle
≈ 40,000 news articles with
Option to analyze title vs. body separately
Identifying "evidence" in the text that suggests forgery (dramatic language, manipulative wording).

PRIOR ART

Source / Title	Approach / Model	Data	Metrics	Results
CSI: A Hybrid Deep Model for Fake News Detection. Ruchansky et al. (CIKM 2017)	Hybrid deep architecture with three modules: • Capture: RNN to model temporal patterns in article text & user responses • Score: Feed-forward NN learning user/source credibility • Integrate: Final classification	Two real-world social-media datasets (Twitter & Weibo)	Accuracy (vs. seven SOTA detectors)	Outperforms seven state-of-the-art fake-news detectors
EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. Wang et al. (KDD 2018)	End-to-end adversarial framework: • Feature Extractor: Multi-modal embeddings (textual, visual, social) • Fake News Detector: Classifier on extracted features • Event Discriminator: Learns event-invariant representations	Multimedia datasets from Weibo & Twitter	Accuracy (benchmark comparison)	Achieves best overall accuracy across two benchmark datasets
Early Detection of Fake News on Social Media Through Propagation Path Classification. Liu & Wu (AAAI 2018)	Propagation-path time-series classification: • Construction: Transform spread into multivariate series of user features • RNN Branch: Captures temporal dynamics • CNN Branch: Captures local variations • Classifier: Combines RNN & CNN outputs	Twitter15, Twitter16, and Sina Weibo datasets	Accuracy 5 min after spread begins	85% accuracy on Twitter; 92% on Weibo within five minutes

STEPS

1. Raw Data: the original data is collected from the Fake and Real News Dataset on Kaggle and stored in the data/raw/ directory.

2. Preprocessing: in this stage, the text is cleaned, stopwords are removed, and the text is lemmatized.

3. Feature Extraction: the processed text is converted into numerical representations using methods like TF-IDF or word embeddings.

4. Model Training: the model is built and trained on the processed data to learn how to distinguish between real and fake news.

5. Performance Evaluation: the model is evaluated using metrics such as Accuracy, Precision, Recall, and F1-score.

6. Explainability: tools like SHAP or LIME are used to explain the model's decisions, such as which words contributed most to the classification.

7. Synthetic Data Generation: generative models are used to create fake news samples, which can help expand the dataset and improve model training.

EXPLORATION & BASELINE

Data:

- From Kaggle: Fake and Real News Dataset
- ≈ 40,000 articles

- Columns: title, text, author, date, subject

- Difference in average length between real and fake articles

Preliminary analysis:

1. Text Length Analysis: fake articles are shorter (mean = 450 words) ,real articles are longer (mean = 780 words)

2. Subject Distribution: fake News: dominant in "politics" (45%) and "health" (30%), real News: spread across "technology" (25%), "world news" (35%), "business" (20%)

3. Temporal Trends: fake news peaks during election months (October–November)

INSIGHTS

- **Key Observations:** fake articles use shorter, emotional language (exclamation marks, clickbait titles). Real articles contain more named entities (organizations, locations)

- **Baseline Limitations:** struggles with sophisticated fake news mimicking journalistic style

- **Next Steps:** implement RoBERTa fine-tuning for contextual understanding

THANKS FOR
LISTENING

-