Sentiment based Graph Learning for Fake News Detection and Analysis

Saptarashmi Bandhyopadyay,
Laxmaan Balaji,
Manasi Dinesh Nair
Team 12

December 11, 2019

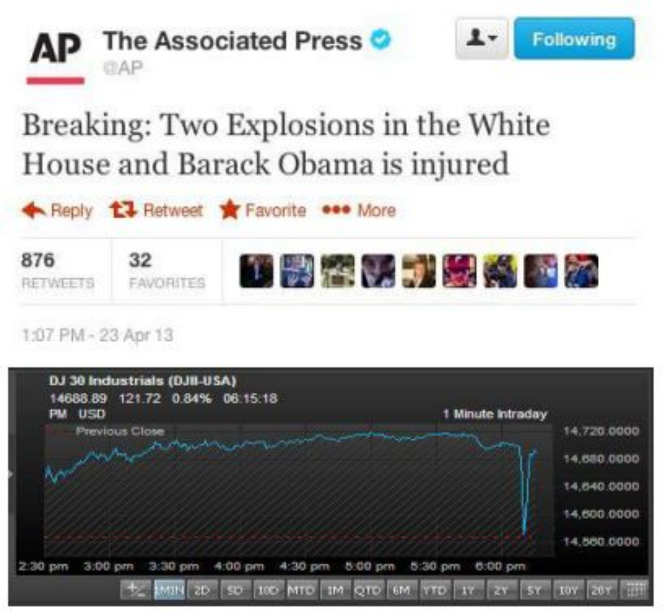
Outline

- Overview on Fake News Detection
- Literature Survey
- Introduction to Graph Neural Networks
- Sentiment based Multi-Graphs from Fake News
- Initial Learning Framework
- Planned Extensions
- Conclusion

Motivation for Fake News Detection

Fake news is now viewed as one of the greatest threats to modern society

1. Political aspect:
   a. 2016 U.S. Presidential Election
      i. 8,711,000 for top 20 frequently-discussed **FAKE** election stories
      ii. 7,367,000 for top 20 frequently-discussed **TRUE** election stories
2. Economic aspect:
   a. "Barack Obama injured in explosion" cost $130B loss in 2013



What is fake news?

1. Serious fabrications:

   e.g., celebrity gossip, news about false or non-existing events

2. Hoaxes:

   e.g., fabricated news with the intention to be picked up

3. Satire:

   e.g., humorous news containing irony and absurdity

Related works

1. Social-text-based methods:
    a. User demographics, user social structure, etc.

2. Propagation-based methods:
    a. Fake news spread is like infectious disease

Related works

3. Content-based methods:
    a. Linguistic approach
        i. Linguistic behaviors are involuntary and hard to fake
        ii. Drawback: fails to detect sophisticated fake news
    b. Fact-checking approach
        i. Automated verification of propositions to asses truthfulness
        ii. Drawback: requires external verification

Dataset

Two novel datasets: crowdsourcing, online articles

a. Crowdsourcing (FakeNewsAMT Dataset)
   i. Covering 6 domains: sports, business, entertainment, politics, technology, education
   ii. Real news comes from mainstream media; e.g., CNN, NYTimes.
   Dataset collected 40 news in each of the six domains, for a total of 240 legitimate news.
   To ensure the veracity of the news, dataset conducted manual fact-checking on the news content, which included verifying the news source and cross-referencing information among several sources
   iii. Fake version of legitimate news comes from crowdsourcing via Amazon Mechanical Turk and manual filters. Collected 240 fake versions.

Dataset (continues)

2 Web Dataset (celebrity Dataset)

i. News about public figures obtained directly from web, collected from online magazines.
ii. The data was collected in pairs, with one article being legitimate and the other fake.
iii. Collected a total of 500 news articles, with an even distribution for fake and legitimate news
iv. Verification of information via gossip-checking websites

The datasets addresses the shortcomings of Fact-checking a*<u>pproach</u>*

Existing Support Vector Machine (SVM) Approach

- The literature uses a linear SVM classifier
- Evaluations are conducted using five-fold cross-validation, with accuracy, precision, recall, and F-score as performance metrics
- The datasets contain an even distribution between fake and real news items. Hence a random baseline of 50% is used as a reference value.

Features in the Existing Approach

1. (Pérez-Rosas et al., 2018) paper in COLING 2018 has considered the following linguistic features for learning:

➔ Ngrams - Obtained from Bag of Words for every news article

➔ Punctuations - 12 types of punctuations like periods, commas, etc

➔ Psycholinguistic features - Extracted under 3 feature sets

➔ Readability - Comprising of content features and readability metrics

➔ Syntax - Derived from Production Rules based on Context-Free Grammar

Takeaways from the Existing Approach

- For the FakeNewsAMT dataset, the best performing classifiers are the ones that rely on stylistic features(punctuation and readability)

- The classifiers build with the Celebrity dataset show the best performance when using the psycholinguistic features.

- The results in the literature shows that when using all the features on the two datasets, the SVM approach achieve the best accuracies, with 0.74 and 0.76 for the FakeNewsAMT and the Celebrity datasets respectively

More Approaches to Fake News Detection

Literature	Learning Framework	Performance	Dataset
V. Perez-Rosas, et al., 2018	SVM	Accuracy: 76%	Across 7 domains by crowd-sourcing and news websites
F. Monti, et al.,2019	CNN	AUC: 93%	News from Twitter
A. Hanselowski, et al.,2017	LSTM	F1m: 0.609	Fake News Challenge Dataset

Introduction to Random Forest Classification

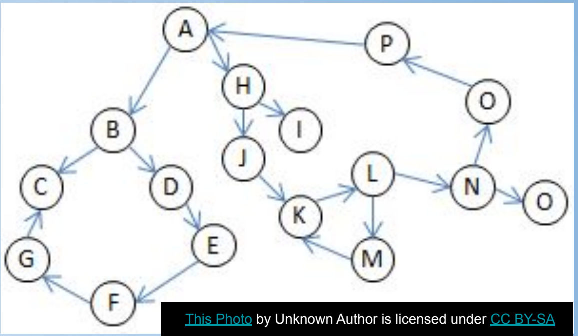
- Ensemble-based classification
- Goal is that the meta-estimator fits a certain number of decision tree classifiers on different sub-samples of the data
- Sub-samples are selected with replacement from the Dataset
- This approach improves classification and addresses overfitting

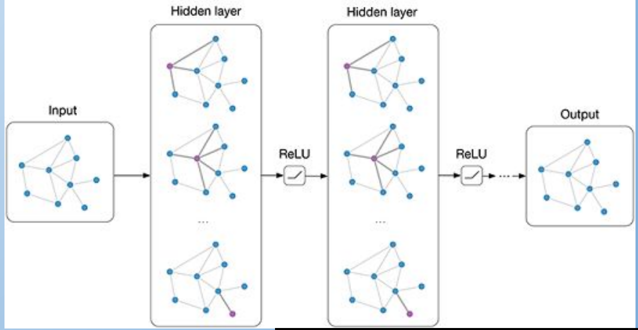
Graph Neural Networks – An Introduction

- A graph is typically denoted as G(V,E) where V is the set of vertices and E is the set of edges connecting vertices in V.

- A Graph Neural Network operates on input data structured as a graph. It can process cycles, directed edges and relationships between nodes (and edges).

- A graph neural net converges to a stable equilibrium by the process of diffusion. All nodes in the neural net exchange information and update themselves accordingly until there are no more updates.





Src: Applications of GNN's (click on image)

Sentiment in the News Article

- Hypothesis: Fake news creators is to confound and shock the readers.
- Words in fake news generate more extreme sentiments with an alternate reality.
- Existing work focuses on sentiment of users, which may vary from person to person.
- We focus on the sentiment conveyed by the article, to capture the malicious intent of the content creator.

Dataset - Preprocessing

- **VADER** (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media.
- It predicts 4 sentiments for each input:
  - Positive - the proportion of positive sentiment text in the sentence
  - Negative - the proportion of negative sentiment text in the sentence
  - Neutral - the proportion of neutral sentiment text in the sentence
  - Compound - compound < -0.05 implies negative sentiment, compound > 0.05 implies positive sentiment, and -0.05 ≤ compound ≤ 0.05 is neutral sentiment

Dataset - Preprocessing

- The content from each article is converted into a list of tokenized sentences using NLTK's Punkt Tokenizer.
- VADER is used to predict the sentiment for each sentence. The sentiment for each sentence has 4 components: [positive, negative, neutral,compound]
- The sentiment scores for each article are calculated as the mean of the sentiment scores for all sentences in the article.

Dataset Representation

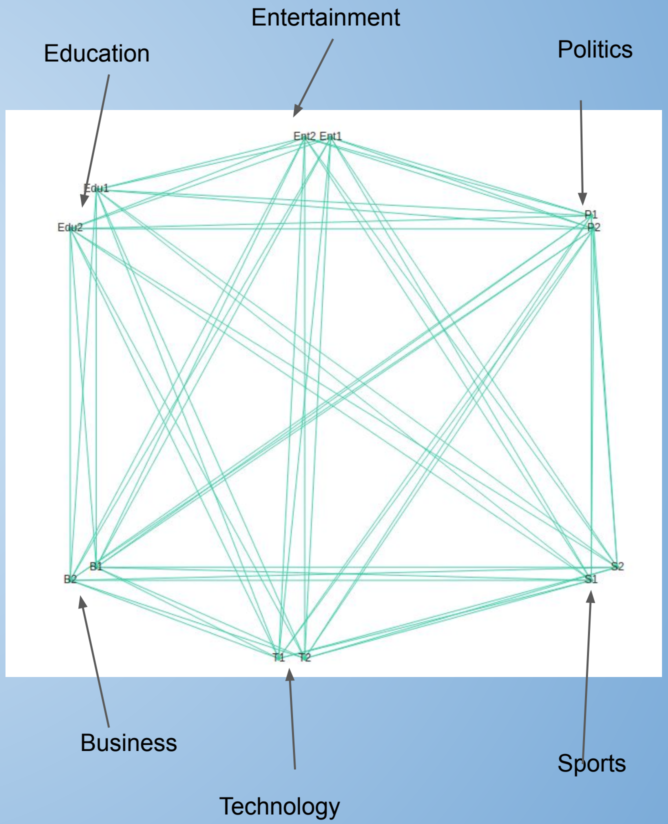
The resulting data, post processing , looks like:

Article #	Positive	Negative	Neutral	Compound	Label
148	0.047	0.303	0.650	-0.811	fake
361	0.204	0.031	0.765	0.699	real

There are 478 more such rows in the dataset

Our Graph Dataset Representation

- Represented the articles as vertices, and established edges between them as shown:
- Each article in a domain is connected to **all other articles to the 5 remaining domains**. i.e. This is a fully connected graph **without intra-domain connections.**
- Node Embeddings are generated using the nearest neighbors approach.



Definitions for the Graph Dataset

- An edge list is used to represent the graph with the edge connectivity as shown on the last slide

- We connected across domains to analyze the impact of propagation of sentiments across legitimate and fake news across domains
  - Sentiments generated from political news articles (say, election results) can impact the sentiments generated from business news articles.

- The four sentiment scores for every sentence in each legitimate news and fake news article have been added to represent the positive, negative, neutral and compound sentiment of the entire news article.

- Each node is represented by the 4 composite sentiment scores

- There are 480 nodes and 59200 edges

SVM and Random Forest - Our Approach

- We re-trained the dataset with a linear SVM classifier and Random Forest classifier for comparative analysis with GNN model

- Scikit-learn packages were used

- The dataset has been randomly shuffled after 5-fold cross validation

- 1000 trees, each having a maximum depth of 4, are considered in the random forest approach

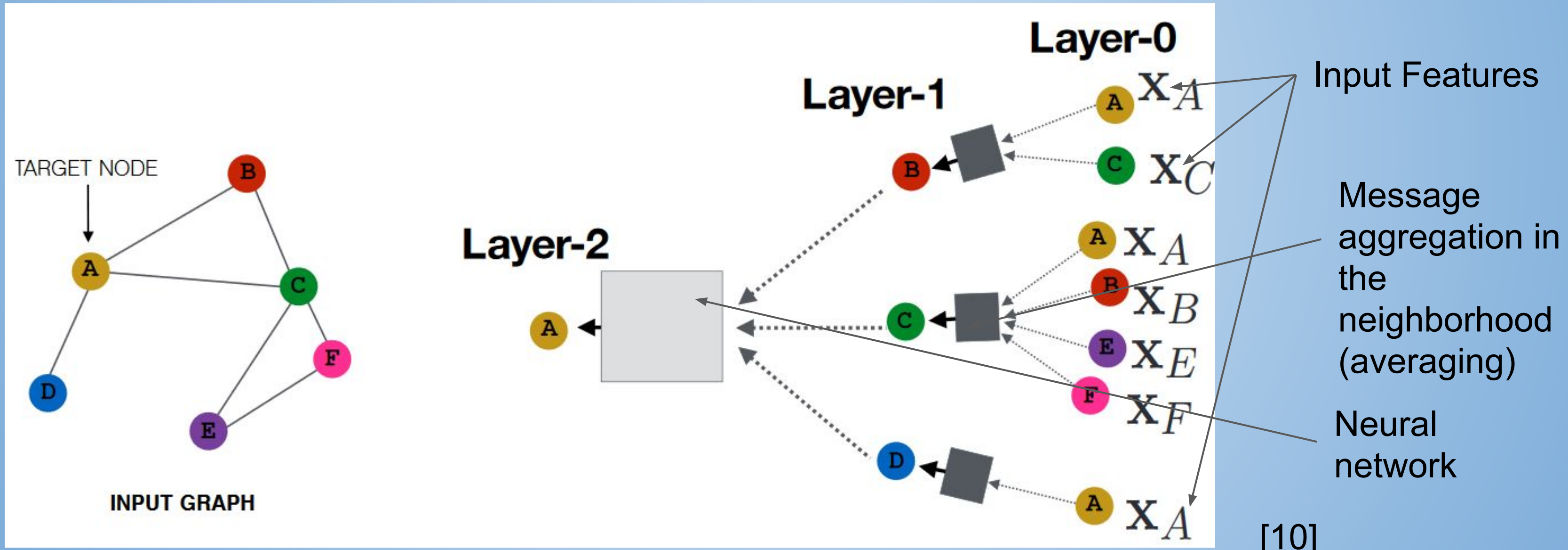
- Accuracy, Precision and F1-scores have been computed along with the confusion matrix

GNN Learning Framework

- Phase 1: Node embedding:-
    - Learning based on inverse of shortest path between 2 nodes in the neighborhood
- Graph is input with a nxn adjacency matrix where n is the number of nodes and nxm feature matrix where m is the number of features
- Embedding vectors generated for each node
- Phase 2: Node classification with LSTM:-
    - Use node embedding and average binary features of the neighboring nodes using distance of learned embedding vectors

Generation of Node Embeddings

- Structural properties of the graph can be used for node embedding generation
- We will follow neighborhood aggregation techniques to generate node embeddings.



[10]

Nearest Neighbors Approach

- Nearest neighbors algorithm is an unsupervised algorithm that identifies the top k neighbors of a given data point as defined by a distance metric (L1, L2.. etc)
- We are considering the top 5 neighbors
- The Scikit-learn (sklearn) package for this algorithm is sklearn.neighbors.NearestNeighbors
- The dimension of the dense embedding is 100
- Early stopping is implemented on validation loss to avoid overfitting

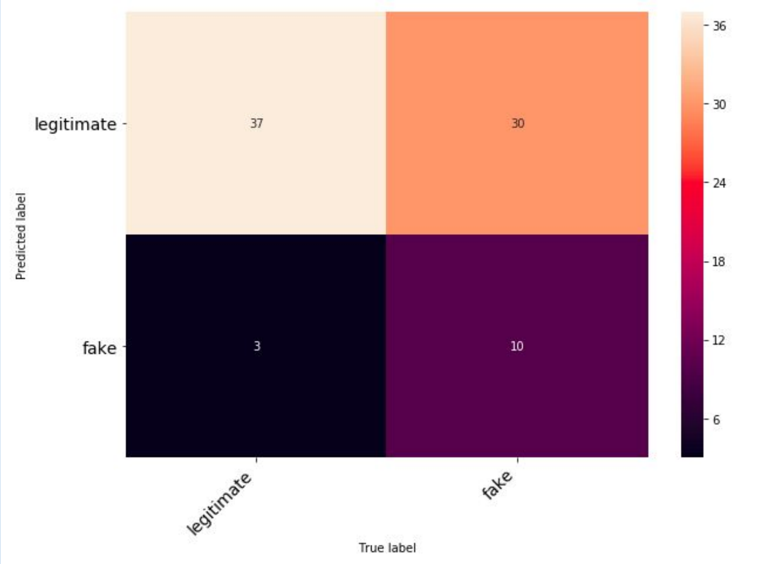
LSTM based Recurrent Neural Network

- We have designed a Long-Short Term Memory based Recurrent Neural Network (RNN) by stacking the node embeddings to a Sequential model
- Early stopping is implemented on validation loss to avoid overfitting
- Our parameters for the RNN in Tensorflow are:
    - 2 layer unidirectional RNN
    - 100 hidden units
    - Initial Learning rate of 0.001
    - Learning rate decay of 0.0001
    - L1 regularization
    - 0.2 drop-out rate
    - Adam Optimizer
    - 30 epochs

Random Forests - Metrics

The performance of the RF classifier was found to be:

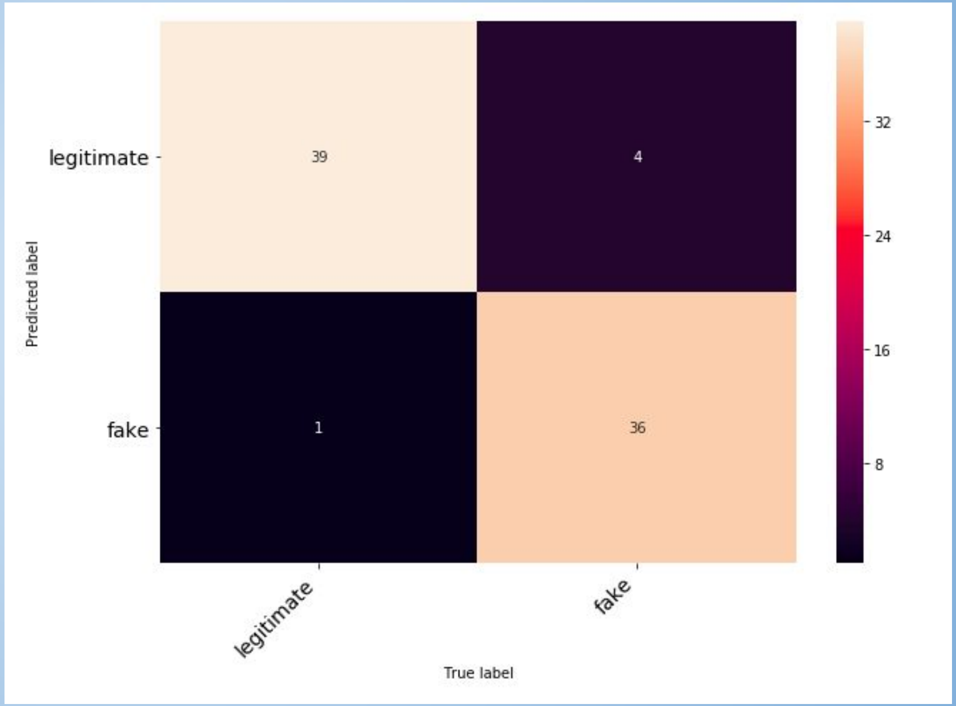
Accuracy	58.75
Precision	55.22%
F1 Score	69.16%



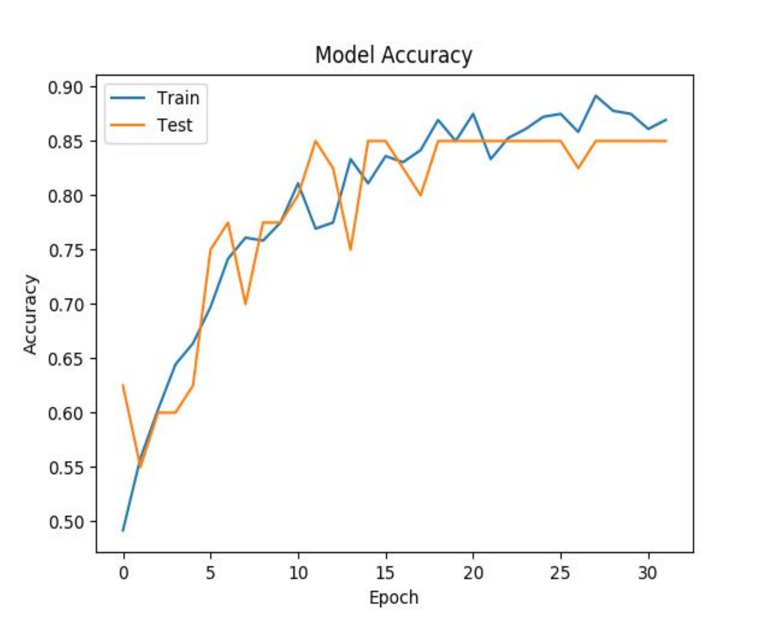
GNN Metrics

The GNN achieves :

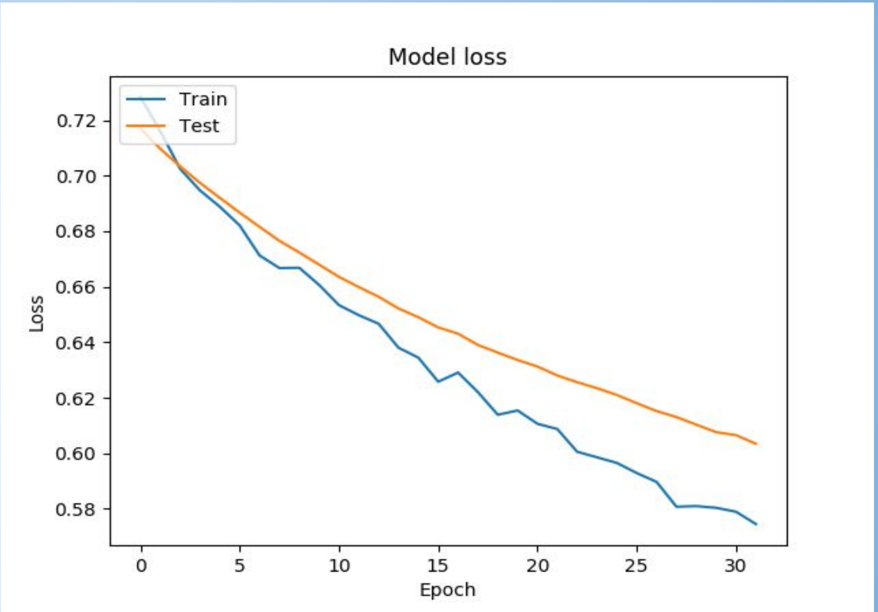
Accuracy	93.75%
Precision	90.7%
F1 Score	93.96%



GNN Training vs. Test Accuracy and Loss Plots



Accuracy vs Number of Epochs



Loss vs Number of Epochs

Classification Results

5-fold cross validation has been implemented like in the paper [1].

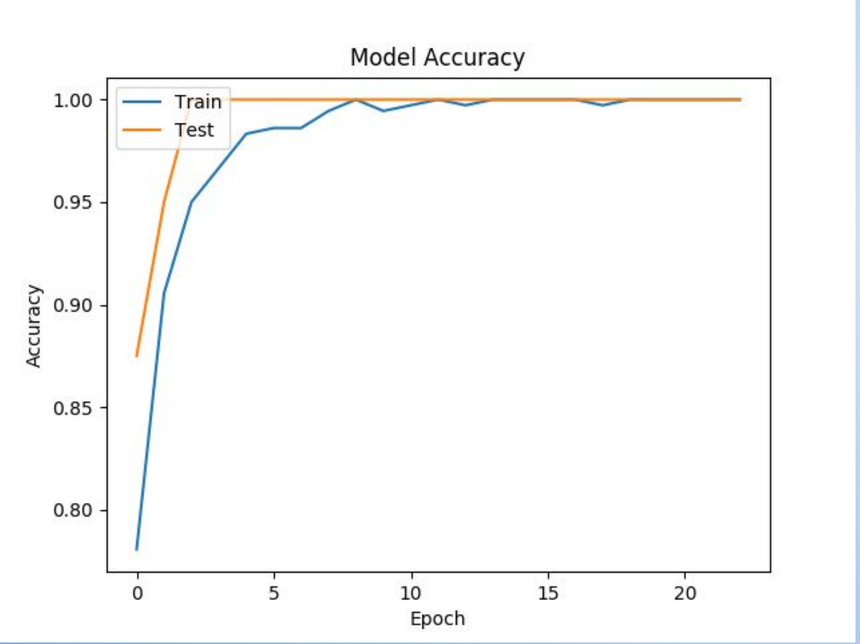
Method	**Accuracy(%)**	**Precision(%)**	**F1 Score(%)**
SVM	52.5	51.56	63.46
Random Forest	58.75	55.22	69.16
Graph Neural Net	93.75	90.7	93.96
SVM in (Pérez-Rosas et al., 2018)	74	(not provided)	74

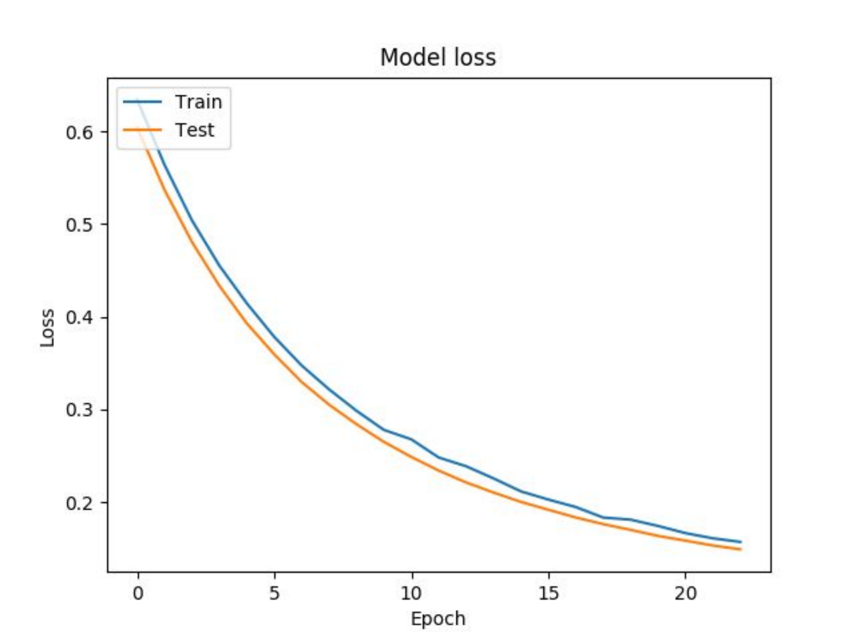
Dataset Split for Cross-Domain Analysis

- All 80 nodes (news articles) from a domain (say politics) constitute the Testing dataset
- Remaining 400 nodes from the 4 other domains are split 9:1 in training and validation dataset
- Training dataset has 360 nodes and Validation dataset has 40 nodes
- The above 3 steps have been executed for each domain as the testing dataset and the remaining 5 domains as the training dataset

Accuracy and Loss Plots - Business Domain

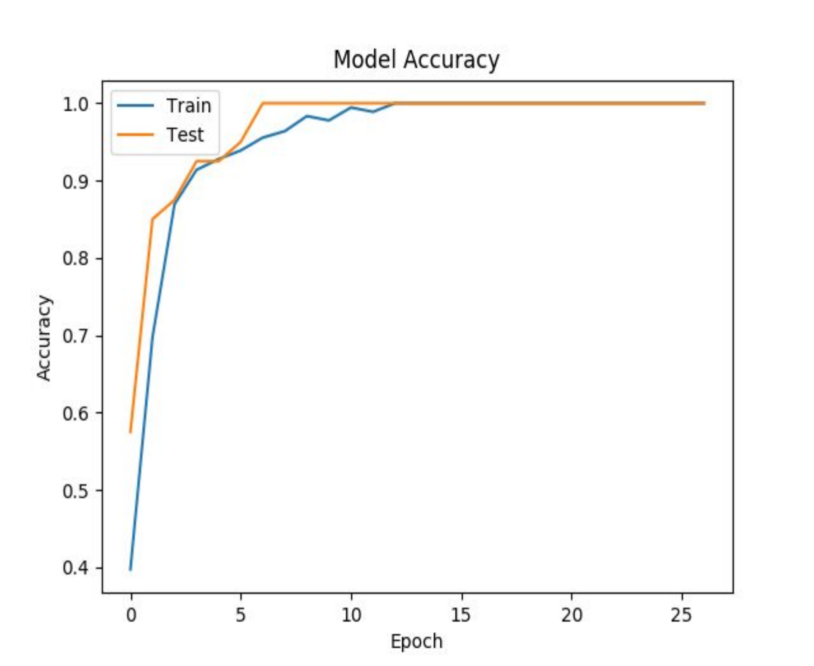
Business news articles are in the test dataset

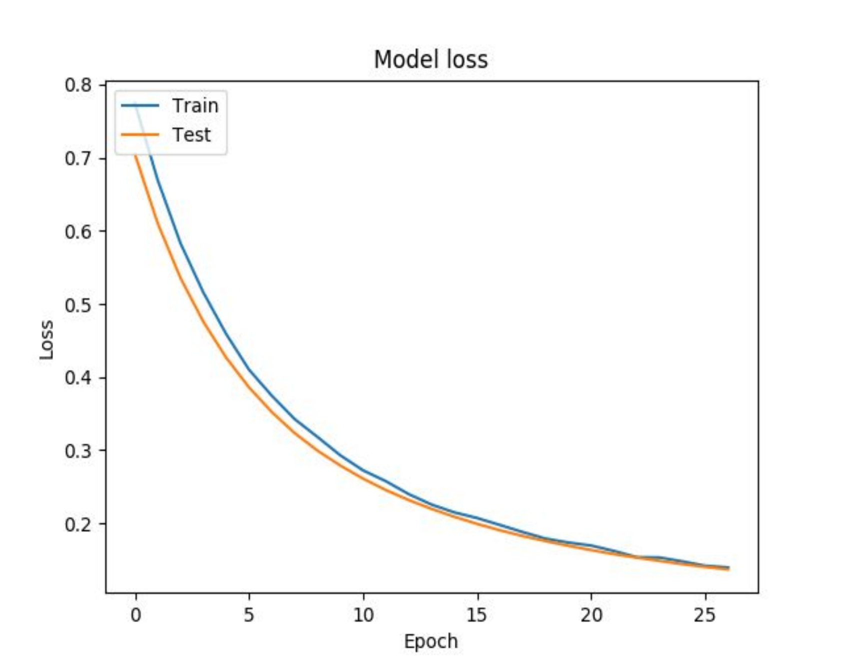




Accuracy and Loss Plots - Politics Domain

News articles from the domain of politics are in the test dataset





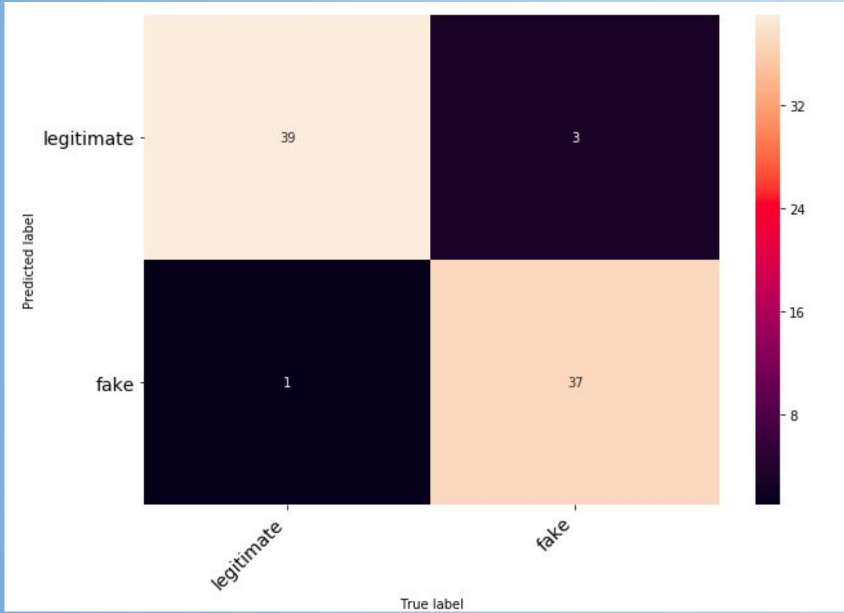
Cross-Domain Accuracy, Precision and F1 Score

Performance of GNN by Domain

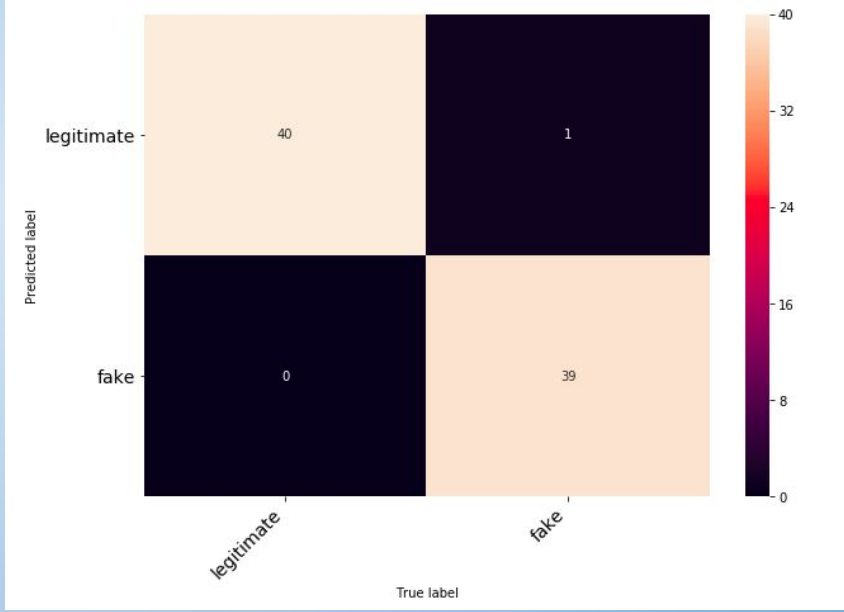
Domain	System Accuracy(%)	SOTA Accuracy (%)	System Precision(%)	System F1 Score(%)
Business	97.5	85	95.23	97.56
Sports	88.75	81	82.98	89.66
Technology	91.25	75	88.37	91.57
Education	98.75	84	97.56	98.77
Entertainment	96.25	75	95.12	96.3
Politics	95	75	92.86	95.12

Confusion Matrices

Some example confusion matrices of GNN by Domain



Politics



Education

Conclusion

- Sentiment based graph creation to capture cross-domain dependence of news articles
- Classification accuracy with GNN and sentiments is 93.75% which is higher than the state-of-the-art (SOTA) accuracy of 74%
- The classification accuracy with GNN is significantly higher with news articles from each of the 6 domains as test datasets than SOTA accuracy
- Future Work: Knowledge aware Graph Learning with extraction of time period and location as features from the news articles to improve the accuracy of fake news detection

References

1. Pérez-Rosas, Verónica, et al. "Automatic Detection of Fake News." *Proceedings of the 27th International Conference on Computational Linguistics*. 2018.

2. https://www.fake-news-tutorial.com/

3. Hanselowski, Andreas, et al. "A Retrospective Analysis of the Fake News Challenge Stance-Detection Task." *Proceedings of the 27th International Conference on Computational Linguistics*. 2018.

4. Monti, Federico, et al. "Fake News Detection on Social Media using Geometric Deep Learning." *arXiv preprint arXiv:1902.06673* (2019).

5. Ruchansky, Natali, Sungyong Seo, and Yan Liu. "Csi: A hybrid deep model for fake news detection." *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2017.

References

6.https://towardsdatascience.com/a-gentle-introduction-to-graph-neural-network-basics-deepwalk-and-graphsage-db5d540d50b3

7. Scarselli, Franco, et. al., “The Graph Neural Network Model”,IEEE Transactions on Neural Networks, Volume 20 Issue 1, January 2009 Pages 61-80

8. Monti, Frederico, et. al., “Geometric Matrix Completion with Recurrent Multi-Graph Neural Networks”, 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

9. Velickovic, Peter, et. al., “Graph Attention Networks”, ICLR 2018

10. SNAP Stanford Tutorial on Graph Neural Networks
http://snap.stanford.edu/proj/embeddings-www/files/nrltutorial-part2-gnns.pdf

References

11. Hajiramezanali, Ehsan, et al. "Variational Graph Recurrent Neural Networks." *arXiv preprint arXiv:1908.09710* (2019)

12. Taheri, Aynaz, Kevin Gimpel, and Tanya Berger-Wolf. "Learning graph representations with recurrent neural network autoencoders." *KDD Deep Learning Day* (2018)

13. Jain, Ashesh, et al. "Structural-RNN: Deep learning on spatio-temporal graphs." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016

14. Hutto, Clayton J., and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text." *Eighth international AAAI conference on weblogs and social media*. 2014.

15. Breiman, Leo. "Random forests." *Machine learning* 45.1 (2001): 5-32.

Thank you!