Data Lab @ TXST
MediaEval Fake News Submission

Data Science class research project: Andrew Magill, Maria Tomasso

Master thesis pytwanalysis: Lia Nogueira de Moura

Undergraduate research: Mirna Elizondo

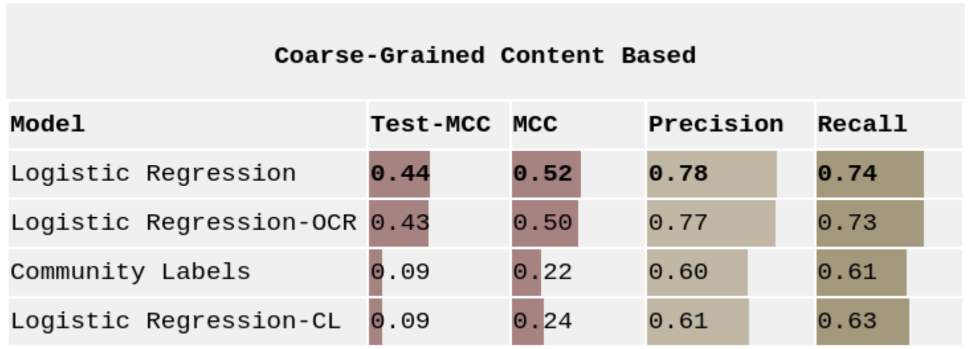
Advisor and presenter: Dr. Jelena Tešić

TEXAS STATE UNIVERSITY

The rising STAR *of Texas*

Text-based Approach (Andrew, Lia, Mirna)

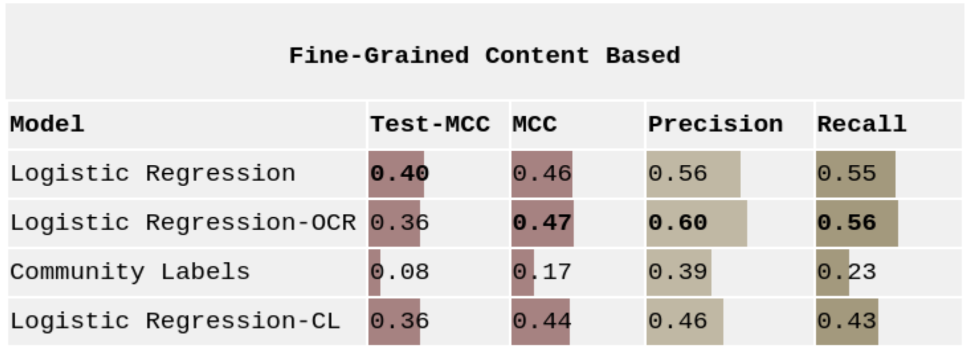
- Coarse-grained classification



Coarse-Grained Content Based

Model	Test-MCC	MCC	Precision	Recall
Logistic Regression	**0.44**	**0.52**	**0.78**	**0.74**
Logistic Regression-OCR	0.43	0.50	0.77	0.73
Community Labels	0.09	0.22	0.60	0.61
Logistic Regression-CL	0.09	0.24	0.61	0.63

- Fine-grained classification



Fine-Grained Content Based

Model	Test-MCC	MCC	Precision	Recall
Logistic Regression	**0.40**	0.46	0.56	0.55
Logistic Regression-OCR	0.36	**0.47**	**0.60**	**0.56**
Community Labels	0.08	0.17	0.39	0.23
Logistic Regression-CL	0.36	0.44	0.46	0.43

- Lexical analysis pipeline: https://github.com/DataLab12/fakenews
- Community analysis pipeline: (submission error)
  - https://pypi.org/project/pytwanalysis/
  - **<u>Dataset:</u>~8 Million tweets related to #Coronovavirus, #Covid19, or #Covid-19 between Mar 2020 and Sep 2020**

TEXAS STATE UNIVERSITY
The rising STAR of Texas

DataLab12.github.io

Structure-based approach (Maria)

❖ Extract feature vectors from each graph, and apply various modeling techniques

Approach failed to capture meaningful patterns

❖ GIGO: augmenting the structural data using community-based approach

❖ Incorporate the structural data features to text-based approaches presented

Model	Run ID	Test MCC	Class	MCC	Acc.	Prec.	Recall
Naïve Bayes	**101**	**0.012**	**multi**	0.176	0.691	0.469	0.423
LDA SVD	**102**	**0.012**	**multi**	0.06	0.707	0.387	0.356
LDA LSQR with shrinkage	**103**	**-0.014**	**multi**	0.057	0.711	0.398	0.351
Decision Tree, gini, no max depth	**104**	**-0.029**	**multi**	0.048	0.572	0.356	0.356
LDA LSQR without shrinkage	**105**	**0.012**	**multi**	0.049	0.702	0.375	0.354
Naïve Bayes	**111**	**-0.001**	**coarse**	0.158	0.859	0.581	0.576
LDA SVD	**112**	**0.033**	**coarse**	0.07	0.896	0.579	0.516
LDA LSQR without shrinkage	**113**	**0.033**	**coarse**	0.07	0.896	0.579	0.516
Decision Tree, gini, no max depth	**114**	**0.007**	**coarse**	0.015	0.809	0.507	0.508
Decision Tree, entropy, no max depth	**115**	**-0.048**	**coarse**	0.017	0.811	0.508	0.509

DataLab12.github.io

Next Steps

❖Publishing 8m dataset and research findings using pytw analysis

❖More complex structure analysis with new dataset

https://datalab12.github.io/work/fake-news.html

❖In-depth student presentation video

❖Project details and source code links

Contact: jtesic@txstate.edu

Text-Based Misinformation Detection Lexical Analysis Pipeline

Andrew Magill, M.Sc.

Computer Science

Texas State University

amagill@tacc.utexas.edu

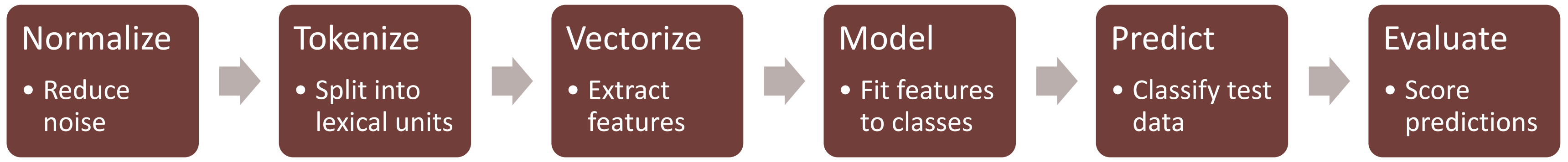
TEXAS STATE UNIVERSITY

The rising STAR *of Texas*

Text-Based Misinformation Detection

- Task Description
  - Given a collection of tweets containing terms related to COVID-19, determine which tweets promote a 5G conspiracy, another conspiracy, or no conspiracy
  - Classification may be two class coarse-grained (5G or not), or four class fine-grained (5G, Other Conspiracy, No Conspiracy, Indeterminate)
  - Submissions may consider media embedded in tweets and include auxiliary data sources
- Approach
  - We have taken a lexical analysis approach to identify writing styles unique to each class using a pipeline that includes a *Bag-Of-Words* vectorizer and Logistic Regression classifier
  - We compare the performance of 7 other classification models, and determine which preprocessing methods produce the most predictive features
  - We extend our approach in two ways: augmenting content with text mined from images embedded in tweets, and considering predicted community membership determined by activity of users in a much larger dataset
- Tools
  - We wrote our pipeline in Python using Jupyter Notebooks with modules for data manipulation (Pandas, NumPy), natural language processing (NLTK), optical character recognition (OpenCV and pytesseract), machine learning (scikit-learn), and visualization (Matplotlib)

Lexical Analysis Pipeline



Normalize

- We found the following preprocessing techniques to positively impact predictive performance: transforming text to lowercase, removing *stopwords* (frequently used words that are common to all classes), preserving URLs, normalizing specific terms
- Other techniques that did not improve performance: preserving and uniquely encoding emojis, stemming and lemmatization

Tokenize:

- We found a tokenization pattern that selects terms from connected alpha-numeric characters produced and splits on all other characters produced the most effective features

Vectorize:

- Features are extracted into a simple Bag-Of-Words that represent occurrences of terms in documents (tweets) as numerical vectors.

Model:

- Logistic Regression and Multi Layer Perception classifiers consistently performed the best in our analysis, almost tying. We opted for the former in our pipeline as it is a much more efficient algorithm

Predict:

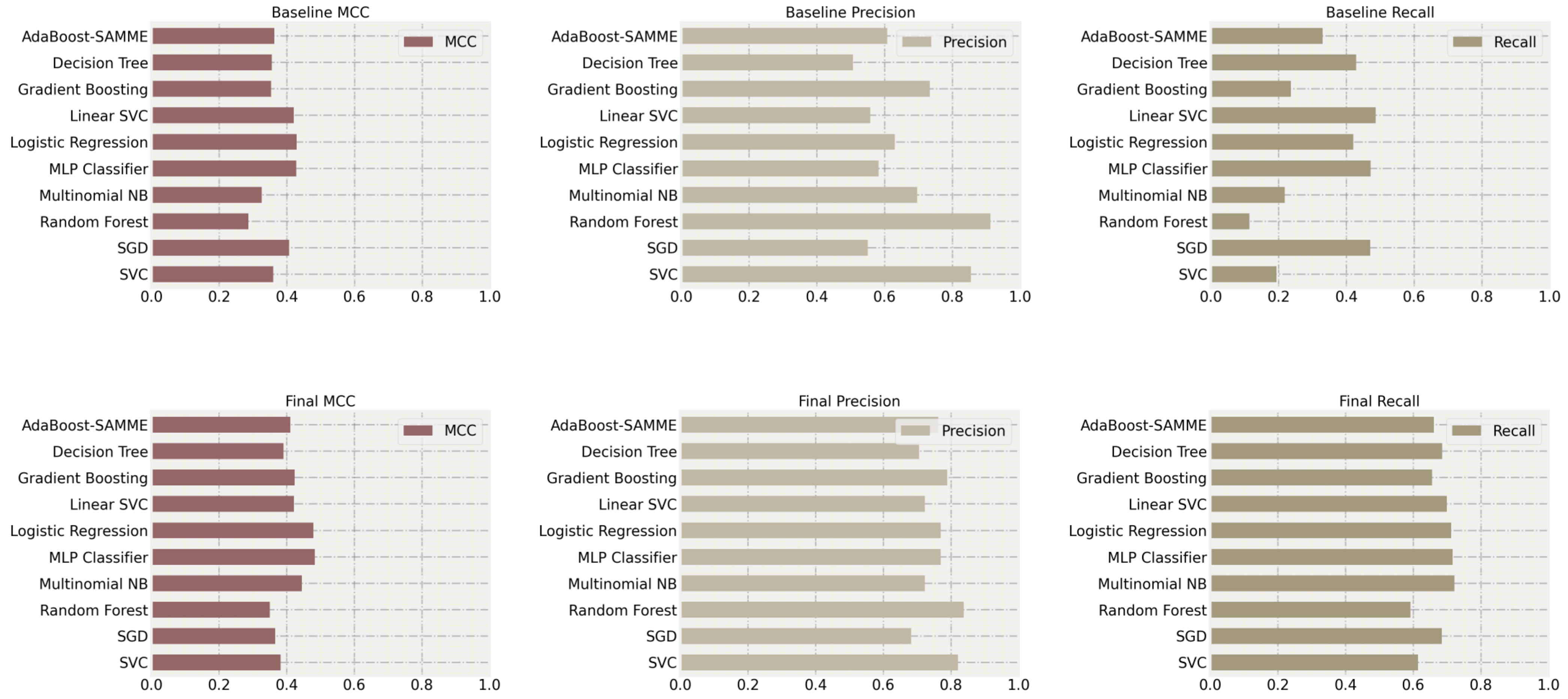
- After fitting our extracted features to classes with our model we make both coarse-grained two class (5G or not), and fine-grained four class predictions (5G, Other Conspiracy, No Conspiracy, Indeterminate)

Evaluate:

- We evaluate our predictions against ground truth using precision, recall, and MCC metrics

DataLab12.github.io

Lexical Analysis Pipeline: Model Performance



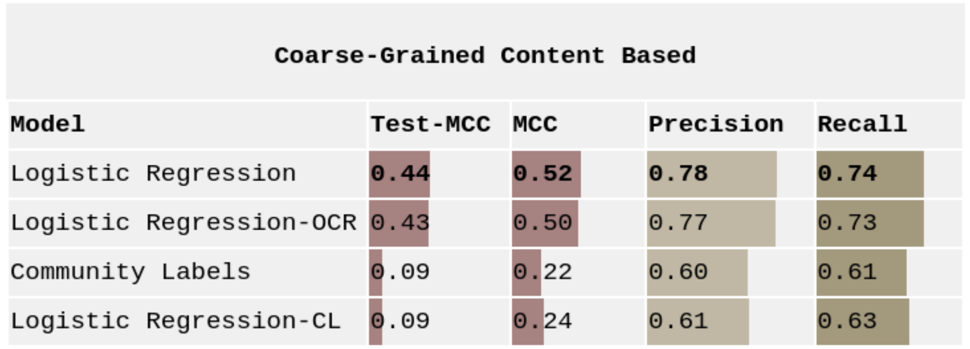
TEXAS STATE UNIVERSITY

The rising STAR of Texas

DataLab12.github.io

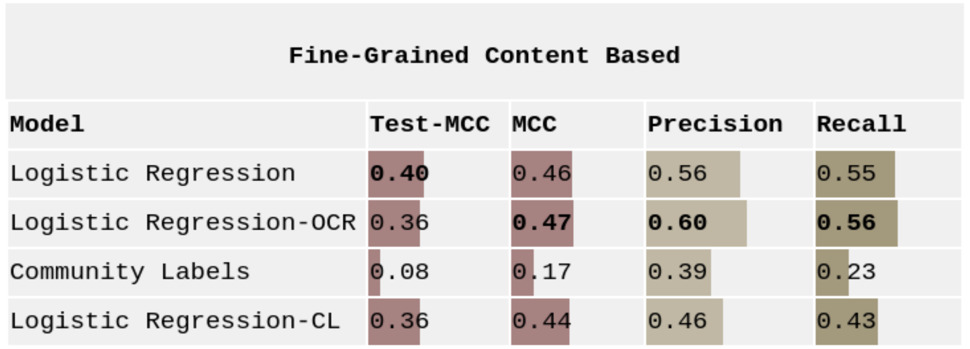
Lexical Analysis Pipeline: Results

- ❖Coarse-grained classification
- ❖Fine-grained classification



Coarse-Grained Content Based

Model	**Test-MCC**	**MCC**	**Precision**	**Recall**
Logistic Regression	**0.44**	**0.52**	**0.78**	**0.74**
Logistic Regression-OCR	0.43	0.50	0.77	0.73
Community Labels	0.09	0.22	0.60	0.61
Logistic Regression-CL	0.09	0.24	0.61	0.63



Fine-Grained Content Based

Model	**Test-MCC**	**MCC**	**Precision**	**Recall**
Logistic Regression	**0.40**	0.46	0.56	0.55
Logistic Regression-OCR	0.36	**0.47**	**0.60**	**0.56**
Community Labels	0.08	0.17	0.39	0.23
Logistic Regression-CL	0.36	0.44	0.46	0.43

- ❖Results of DL-TXST submission on test set, and corresponding MCC, Precision, Recall on development set.
- ❖Unfortunate error resulted in poor performance of the community label predictions in the test set for both the coarse-grained and fine-grained predictions

DataLab12.github.io

Lexical Analysis Pipeline: Conclusion

- ❖ We discovered that our preprocessing techniques rarely resulted in improvements in both precision and recall, we had to balance the benefit of normalizing our content with the potential loss of distinctiveness of each class, opting often to preserve distinctive features.
- ❖ We conclude that the most important steps we took were those that reduced noise, for instance, removing *stopwords* that are not likely to be predictive, and as it turns out, emojis
- ❖ The performance of the MLP matching logistic regression implies that DNN models may achieve greater results
- ❖ Although incorporating predictions of community membership did not improve our results in our own tests, we believe that this approach may produce better results with further work

Text-Based Misinformation Detection Community Analysis Pipeline

Lia Nogueira de Moura, M.Sc.

Computer Science

Texas State University

lia.lnm@gmail.com

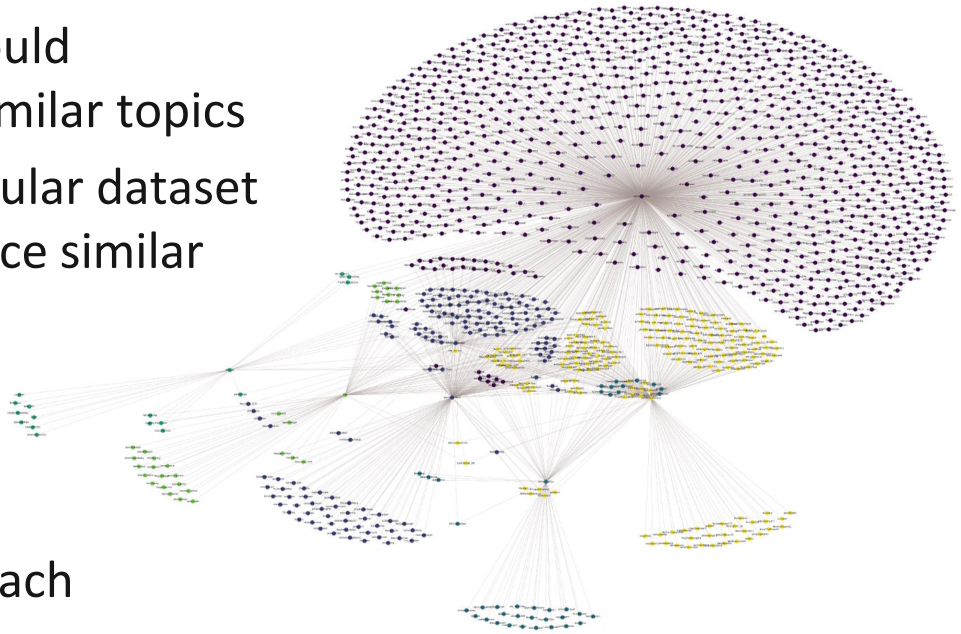
TEXAS STATE UNIVERSITY

The rising STAR of Texas

Community Analysis Pipeline

❖ **Idea**:

- Users that belong to the same community could potentially share similar ideals and discuss similar topics
- Users that are well connected within a particular dataset *(5G, non, and other)* would most likely produce similar content not included in the given datasets



❖ **Approach**:

- Group tweets into communities
- Use the community assignments to classify each community into *(5G, non, or other)*

❖ **Tools**:

- *pytwanalysis* package / MongoDB / Louvain Community / networkX

Community Analysis Pipeline

Step 1 – Network Creation:

- Network Types
  - User connections network: *Vertex=User; Edge=Retweet, Quotes, Mentions, or Replies*
  - Hashtag Network: *Vertex=Hashtag; Edge = two hashtags used in the same tweet*
- Auxiliary Data
  - Size: ~8 Million tweets
  - Content: tweets related to #Coronovavirus, #Covid19, or #Covid-19
  - Period: between March, 2020 and September, 2020
- 3 networks created:
  - 1) User connections (provided data)
  - 2) Hashtag connections (provided data)
  - 3) User connections (8M + provided data)

TEXAS STATE UNIVERSITY

The rising STAR of Texas

DataLab12.github.io

Community Analysis Pipeline

Step 2 – User's Degree:

- The degree of connectivity of each user in the *User Connections* network was extracted separately for each of the different datasets *(5G, non, or other)*

Step 3 – Community Detection:

- Communities found for each network using the Louvain Community method.
- Each tweet was assigned 3 community labels, one from each network

Step 4 – Classification for each community:

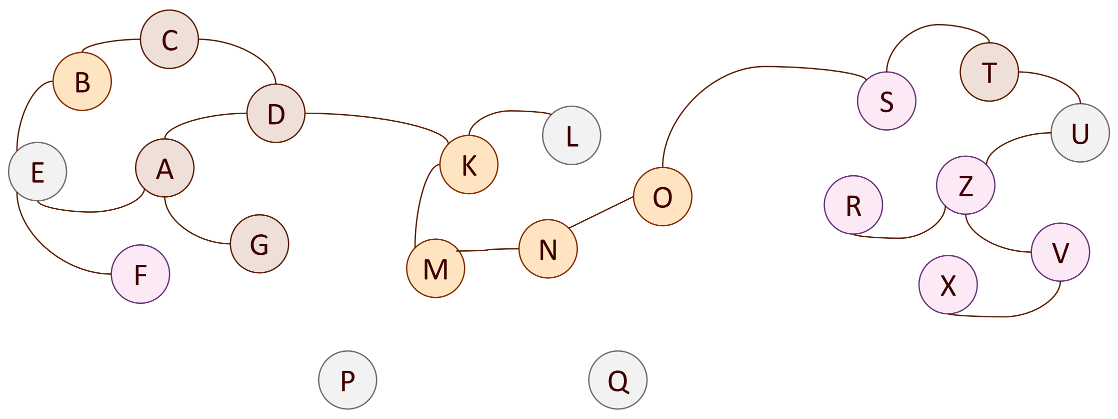
- Each community was classified with one of the labels *(5G, non, or other)*
- The classification was done based on the majority of the tweets in that community originating from one the three datasets *(5G, non, or other)*
- A combination of the community labels from the three networks was used
- Degree of the user was used as fall back if no community was found

Step 5 – Unknows:

- Tweets that did not belong to any community were classified as unknowns

Community Analysis Pipeline

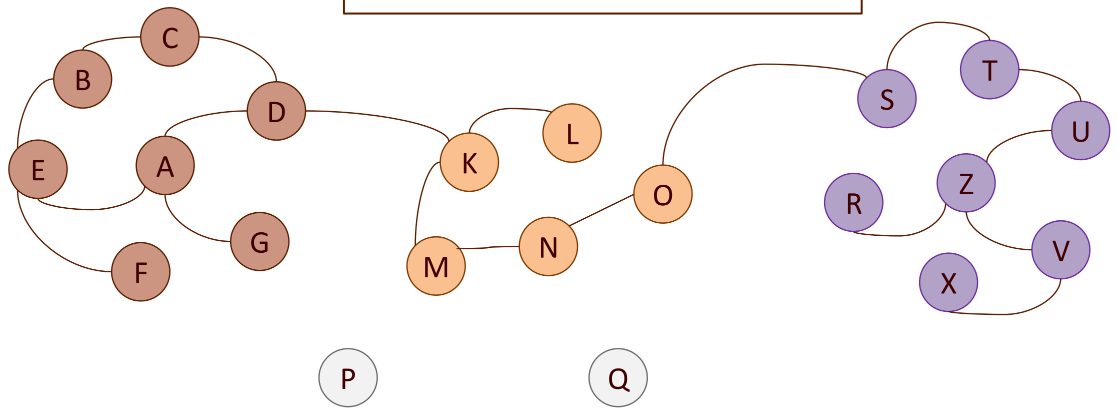
Sample network – using all datasets



Datasets

● 5G_conspiracy
● non_conspiracy
● other
● test

Community Assignment



Community Classification

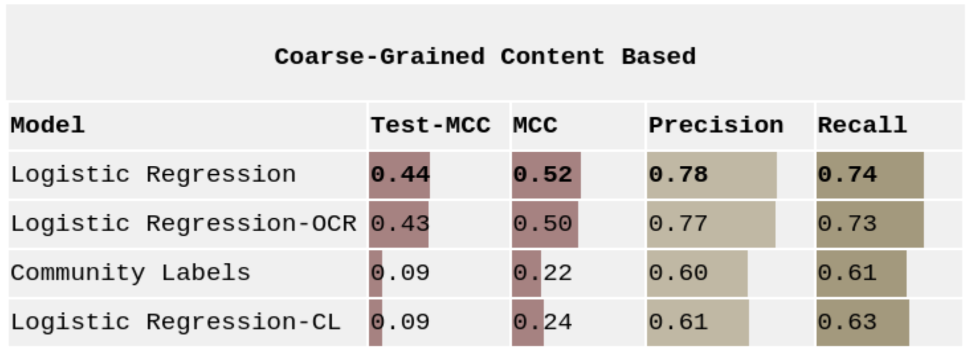
● 5G_conspiracy
● non_conspiracy
● other
● unknow

TEXAS STATE UNIVERSITY
The rising STAR of Texas

DataLab12.github.io

Community Analysis Pipeline: Results

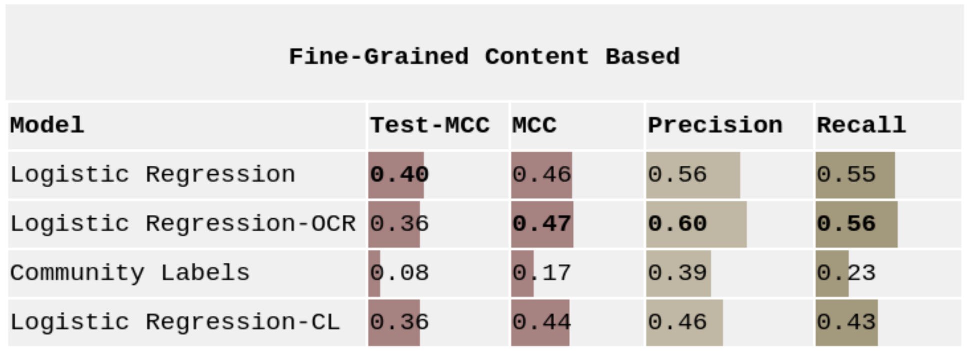
❖ Coarse-grained classification



Coarse-Grained Content Based

Model	Test-MCC	MCC	Precision	Recall
Logistic Regression	**0.44**	**0.52**	**0.78**	**0.74**
Logistic Regression-OCR	0.43	0.50	0.77	0.73
Community Labels	0.09	0.22	0.60	0.61
Logistic Regression-CL	0.09	0.24	0.61	0.63

❖ Fine-grained classification



Fine-Grained Content Based

Model	Test-MCC	MCC	Precision	Recall
Logistic Regression	**0.40**	0.46	0.56	0.55
Logistic Regression-OCR	0.36	**0.47**	**0.60**	**0.56**
Community Labels	0.08	0.17	0.39	0.23
Logistic Regression-CL	0.36	0.44	0.46	0.43

❖ Results of DL-TXST submission on test set, and corresponding MCC, Precision, Recall on development set.

❖ Unfortunate error resulted in poor performance of the community label predictions in the test set for both the coarse-grained and fine-grained predictions

TEXAS STATE UNIVERSITY

The rising STAR of Texas

DataLab12.github.io

Community Analysis Pipeline: Next Steps

Tweets that are isolated from the network degrade the performance of the approach

Large number of tweets without edges in the networks

❖ no retweets, replies, quotes, mentions, or hashtags

Many tweets were classified as unknown

❖ Need for classifying the unknowns

Community Analysis can be used as a feature

❖ Works well when users are well connected

❖ Submission error: Test the correct performance on test set

Structure-Based Misinformation Detection

Maria Tomasso, M.Sc.
Computer Science
Texas State University
met48@txstate.edu

TEXAS STATE UNIVERSITY

The rising STAR of Texas

Structure-Based Misinformation Detection

❖ **Problem Statement**:

- Conspiracy theories about COVID-19 have circulated online since the initial outbreak in December 2019
- Misinformation can affect compliance with public health measures such as mask-wearing and stay-at-home orders
- **Our goal is to build a model that can identify tweets promoting the COVID-19/5G conspiracy theory using text-based and structure-based data**

❖ **Approach**:

- Extract feature vectors from each graph
- Apply Naïve Bayes, decision tree, and linear discriminant analysis models to the labeled feature vectors to build a classifier

❖ **Tools**:

- *R - igraph*
- *Python – Pandas, scikit-learn*

TEXAS STATE UNIVERSITY
The rising STAR of Texas

DataLab12.github.io

Structure-Based Misinformation Detection

❖ **Feature vector for each graph:**

– n_nodes, n_edges, diameter, mean_distance, edge_density, reciprocity, transitivity_global, transitivity_localaverage, triangles, mean_in_degree, max_in_degree, min_in_degree, mean_out_degree , max_out_degree, min_out_degree , mean_total_degree, max_total_degree, min_total_degree

❖ **Decision tree modeling:**

– 80:20 split for tuning criterion and max depth hyper-parameters

– Tested 'Gini' or 'entropy' criteria with max_depth 3, when used

– Trained fine and coarse labels, produced 12 runs

Structure-Based Misinformation Detection

Linear Discriminant Analysis Modeling

- ❖80:20 split used for train/test data
- ❖3 solving methods were tested:
  - Singular value decomposition
  - Least squares
  - Least squares with shrinkage
- ❖Models were trained for coarse and fine runs, produced total of 9 runs

Naïve Bayes Modeling

- ❖80:20 split used for train/test data
- ❖Models were trained on coarse and fine labels for a total of 3 runs

Structure-Based Misinformation Detection

- Results: Structure based analysis presented failed to capture any meaningful pattern

Model	Run ID	Test MCC	Class	MCC	Acc.	Prec.	Recall
Naïve Bayes	**101**	**0.012**	**multi**	0.176	0.691	0.469	0.423
LDA SVD	**102**	**0.012**	**multi**	0.06	0.707	0.387	0.356
LDA LSQR with shrinkage	**103**	**-0.014**	**multi**	0.057	0.711	0.398	0.351
Decision Tree, gini, no max depth	**104**	**-0.029**	**multi**	0.048	0.572	0.356	0.356
LDA LSQR without shrinkage	**105**	**0.012**	**multi**	0.049	0.702	0.375	0.354
Naïve Bayes	**111**	**-0.001**	**coarse**	0.158	0.859	0.581	0.576
LDA SVD	**112**	**0.033**	**coarse**	0.07	0.896	0.579	0.516
LDA LSQR without shrinkage	**113**	**0.033**	**coarse**	0.07	0.896	0.579	0.516
Decision Tree, gini, no max depth	**114**	**0.007**	**coarse**	0.015	0.809	0.507	0.508
Decision Tree, entropy, no max depth	**115**	**-0.048**	**coarse**	0.017	0.811	0.508	0.509

- Next steps:
  - augmenting the structural data using community-based approach
  - incorporating the structural data features to text-based approaches presented

TEXAS STATE UNIVERSITY
The rising STAR of Texas

DataLab12.github.io