

Community Interaction

A dark blue diagonal gradient bar that starts from the bottom left and extends towards the top right, covering the lower half of the slide.

Original Project Description

- Original Paper: Community Interaction and Conflict on the Web
- Prediction of negative mobilizations between subreddits
- Create embeddings of subreddits and use Random Forest to predict negative mobilization

Come look at all the brainwashed idiots in r/Documentaries

Seriously, none of those people are willing to even CONSIDER that our own country orchestrated the 9/11 attacks. They are all 100% certain the “turrists” were behind it all, and all of the smart people who argue it are getting downvoted to the depths of hell. Damn shame. Wish people would do their research. The link is [CROSS-LINK].

This post was created on the subreddit *r/conspiracy* and is an example of Negative Mobilization towards another subreddit (in this case *r/Documentaries*)

Embedding Creation

4 Embeddings in total are created:

1. Text Embeddings of posts using VADER
2. Community embeddings based on bipartite multigraph between users and communities
3. User embeddings
4. Socially primed LSTM embeddings

**Meta-Features
(VADER, LIWC)**

User-Embeddings

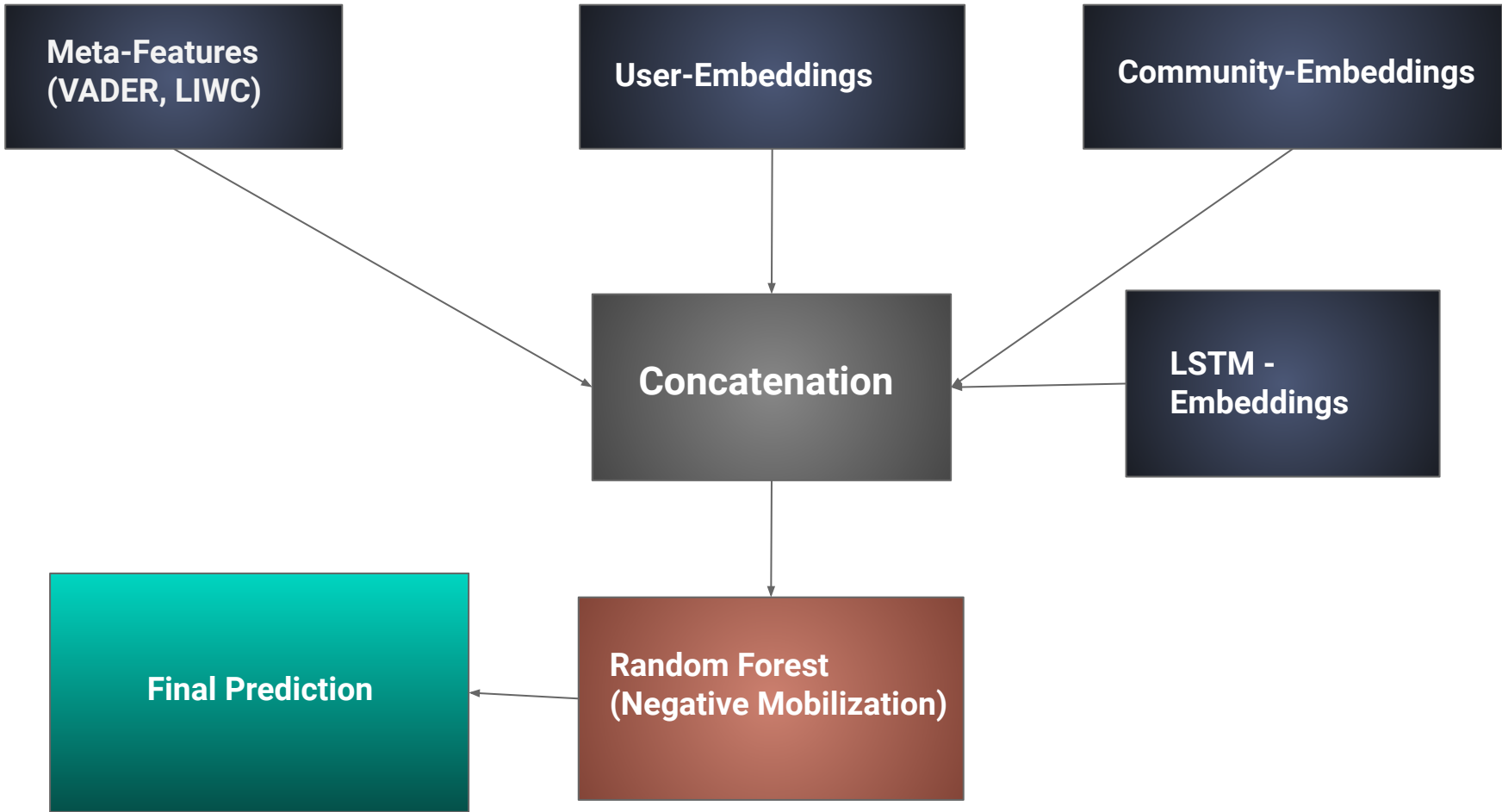
Community-Embeddings

Concatenation

**LSTM -
Embeddings**

Final Prediction

**Random Forest
(Negative Mobilization)**

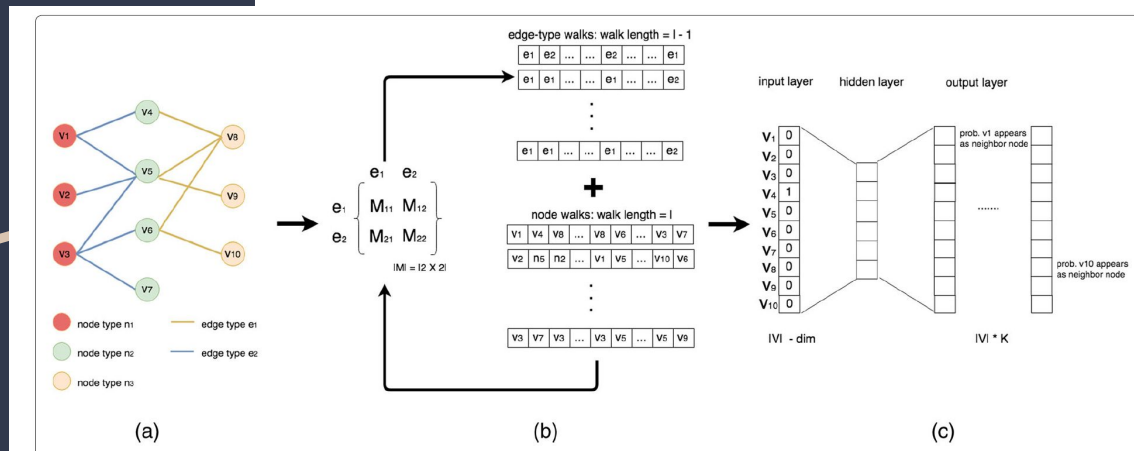


Our Approach – So Far

- Simplify graph
- Create new embeddings with node2vec of subreddits
- Create edge embeddings using edge2vec
- Use embeddings with Random Forest to predict negative mobilizations

Novelties of Edge2Vec

- Being able to represent edge heterogeneity
- an edge-type transition matrix
- Expectation- Maximization model



Results – So far

- Baseline accuracy of Paper: **ROC-AUC = 0.75**
- Ours (Node2Vec): **ROC-AUC = 0.76**
- Ours(Edge2vec): **ROC-AUC = 0.76**
- Strong reduction in overall features needed
2063(benchmark) vs. 306 (ours)
- Exchange of final classifier also increased
performance strongly

Dataset	ROC_AUC	Contains Graph Embedding
Level 1		
Meta-Features	0.6800	No
All embeddings	0.7450	No
LSTM-Embeddings	0.4926	No
Node2Vec-Embeddings	0.7170	Yes
Edge2Vec-Embeddings	0.6784	Yes

Level 1 - Metrics (Single Embeddings)

Dataset	ROC_AUC	Contains Graph Embedding
All embeddings + LSTM-Embeddings	0.7440	No
All embeddings + Node2Vec-Embeddings	0.7450	Yes
Meta-Features + Node2Vec-Embeddings	0.7523	Yes
User embeddings + Edge2Vec-Embeddings	0.70841	Yes

	Dataset	ROC_AUC	Contains Graph Embedding
	All embeddings + Meta-Features + LSTM-Embeddings	0.7501	No
	All embeddings + Meta-Features + Node2Vec-Embeddings	0.7598	Yes
	All embeddings + Meta-Features + Edge2Vec-Embeddings	0.75697	Yes
	User embeddings + Meta-Features + Edge2Vec-Embeddings	0.7342	Yes

Dataset	ROC_AUC	Contains Graph Embedding
All available datasets	0.7499	No

Change of Classifier

- Original Paper used Random Forest for final predictions
- Improvement: Replace Random Forest with Gradient boosting tree
- Final Performance: **ROC-AUC = 0.80** (Lightgbm + *Node2vec* + User-Embeddings + Meta-features)
- Current performance is without tuning the Booster

Future

- Create Edge Classifier using SAGE
- Tune Node2Vec and Edge2Vec
- Possibly restructure Multigraph
- Hyperparameter Search for Lightgbm
- More focus on sensitivity analysis for Embeddings