Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

Homework 5: What happens in Vegas, stays in Venmo

This report presents a comprehensive analysis of Venmo transaction data, exploring user behavior through three distinct analytical lenses: text analytics, social network analysis, and predictive modeling.

The Venmo dataset includes peer-to-peer transactions with fields like user1, user2, datetime, and description. Each description may contain text, emoji, or both.

```
df.show(3)
```

```
+-------+-------+----------------+-------------------+-----------+-----------+--------------------+
|  user1|  user2|transaction_type|           datetime| description|is_business|            story_id|
+-------+-------+----------------+-------------------+-----------+-----------+--------------------+
|1218774|1528945|         payment|2015-11-27 10:48:19|       Uber|      false|5657c473cd03c9af2...|
|5109483|4782303|         payment|2015-06-17 11:37:04|     Costco|      false|5580f9702b64f70ab...|
```

To classify these messages, we used:

- **Emoji Dictionary**: Categorizes emojis into 7 themes like Food, Travel, Activity, and Utility.

```
+-----+------+----+--------+--------------+------+-------+
|Event|Travel|Food|Activity|Transportation|People|Utility|
+-----+------+----+--------+--------------+------+-------+
|🇦🇺  |🏔    |🍇  |🦑      |🚄            |😀    |⚡     |
|🇫🇷  |🏔    |🍈  |🕯      |🚙            |😃    |💡     |
|🎂   |🌋    |🍉  |🎪      |🚆            |😄    |🔌     |
+-----+------+----+--------+--------------+------+-------+
only showing top 3 rows
```

- **Text Dictionary**: Maps common words to 9 topics including People, Cash, Illegal/Sarcasm, and more.

	People	Food	Event	Activity	Travel	Transportation	Utility	Cash	Illegal/Sarcasm	@dropdown
0	friend	food	birthday	ball	beach	lyft	bill	atm	addiction	NaN
1	friendship	bbq	christmas	boat	place	uber	cable	bank	drug	NaN
2	baby	bean	happy	bar	la	cab	fee	cash	wangs	NaN

These dictionaries enabled us to classify transactions by type, identify emoji-only messages, and build both static and dynamic user spending profiles. Screenshots below show examples of the data and dictionaries used.

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

1. Text Analytics

We began by classifying all 7.1 million transaction descriptions using a custom UDF. Each message was categorized into one of the following types:

- **Emoji-only**: Contains only emoji(s), no text.
- **Text-only**: Contains only alphabetic characters.
- **Mixed**: Contains both emoji and text.
- **Empty/Other**: No valid content.

Breakdown of message types:

```
+----------+-------+
|  msg_type|  count|
+----------+-------+
|     mixed| 716768|
|emoji-only|1858740|
|     other| 104951|
| text-only|4432497|
|     empty|    181|
+----------+-------+
```

To analyze emoji-based messages, we extracted and exploded each emoji per transaction, then joined them with a manually labeled emoji dictionary containing 7 categories: *Food, Activity, Event, Travel, Transportation, Utility, People*.

Corrected percentage of emoji-only messages: 26.13%

This confirms that over a quarter of all Venmo messages use emojis alone — underlining the importance of incorporating emoji semantics in our analysis.

We extracted and exploded emojis from all emoji-only and mixed messages, then matched them to a custom dictionary that maps each emoji to one of seven categories:

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

Top 5 emojis-

emoji	count
🍕	215031
😂	145229
😭	124726
🍷	111157
🎉	94321

Top 3 emoji categories:

1. **Food** – 1.7 million
2. **People** – 787K
3. **Activity** – 650K+

These findings demonstrate the strong cultural presence of food and social activities in Venmo messages.

Spending Behavior Profiling

This section investigates how Venmo users allocate their transactions across various spending categories. We first generate a static profile that summarizes a user's overall transaction distribution, and then develop a dynamic profile to capture how these preferences evolve during the first year of activity on the platform.

Static Spending Profile

The static profile represents the long-term category distribution for each user. After assigning each transaction to one of the nine categories using the emoji and text dictionaries, we computed the proportion of transactions per category for every user.

This involved aggregating transaction counts by user and category, dividing by the user's total categorized transactions, and calculating the category share. The following table provides an example of a static profile for a user with 25 categorized transactions:

user1	category	cat_count	total_txns	category_share
10305628	Food	11	25	0.44
10305628	Activity	10	25	0.4
10305628	Transportation	2	25	0.08
10305628	Travel	1	25	0.04
10305628	Event	1	25	0.04

This approach was applied to over 4.1 million categorized transactions. Some users had more than 1,200 categorized messages, reflecting high engagement levels. The most frequent categories across the user base were food, activity, and people, followed by transportation and cash.

The static profile gives a high-level summary of each user's long-run behavioral tendencies on Venmo.

Dynamic Spending Profiles

To capture temporal evolution in user behavior, we constructed dynamic profiles that track monthly category preferences over a user's first 12 months on Venmo. Only users with at least one year of activity were included to ensure complete tracking.

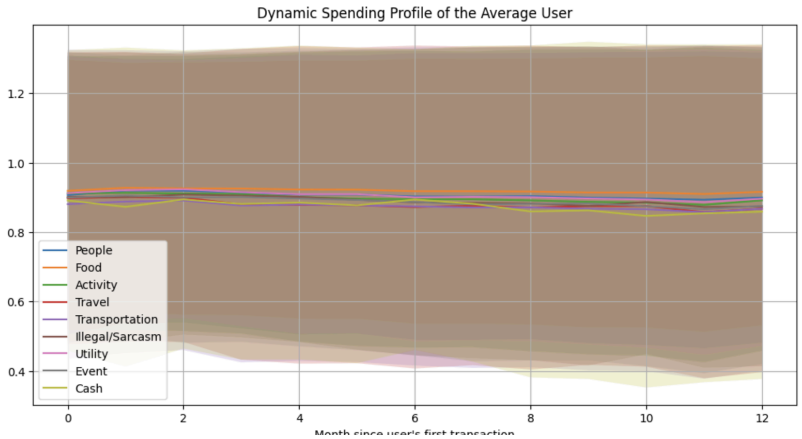
Each transaction was assigned a relative month based on the number of days since the user's first transaction. We then calculated the share of transactions per category for each user-month combination. These were averaged across users to obtain the monthly mean and standard deviation for each category.

This method allowed us to analyze both the average trend and variability in user behavior. For instance:

- Food and activity consistently held the largest shares across all 12 months.
- Cash, event, and illegal/sarcasm categories showed higher fluctuations, especially in the early months.
- By month six, most categories exhibited reduced variation, indicating stabilization in behavior.

These results were visualized using line plots with shaded bands representing ±2 standard deviations, revealing how user behavior evolves from diverse and exploratory to stable and habitual over time.

The dynamic profile adds a crucial temporal dimension to our understanding of user behavior and complements the static profile by showing not just what users do, but when and how their preferences change.



Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

Text Embedding

1. Model Selection and Embedding Generation

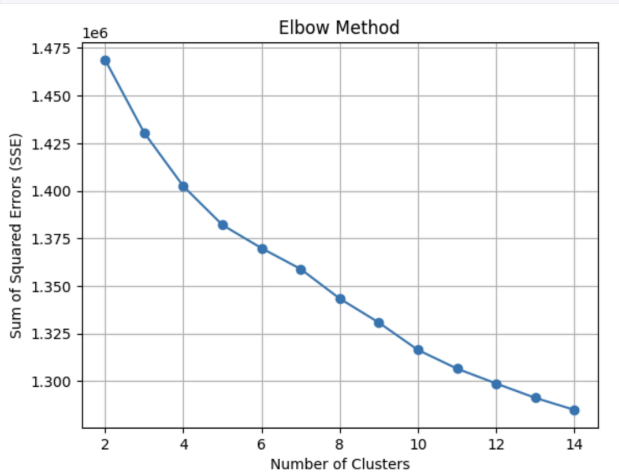
We used the pretrained all-MiniLM-L6-v2 model from the Sentence-Transformers library. This model is optimized for computing sentence embeddings and balances semantic accuracy with computational efficiency. It produces 384-dimensional dense vectors that preserve the contextual meaning of short texts.

The model was applied to a sample of 50,000 cleaned text-only Venmo messages. These were tokenized, padded, and passed through the model in batches. We computed mean-pooled embeddings using the model's final hidden state and attention masks to ensure proper weighting of tokens.

Embeddings were computed in 47 batches of approximately 1,000 messages each, then merged to create a single embedding matrix for clustering.

Clustering with K-Means

To discover themes in the embedding space, we applied K-Means clustering to the 384-dimensional embeddings. We used the elbow method to determine an appropriate number of clusters, settling on k = 10 based on the point of diminishing returns in the sum of squared errors as shown by the figure below.



Each message was assigned a cluster label, and representative samples were extracted using both:

- Messages closest to the cluster centroid (high cosine similarity)
- Most frequent terms via TF-IDF within each cluster

These methods helped us assign manual labels to each cluster based on dominant themes.

Sample Cluster and Count

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

Cluster	Topic Label	Common Keywords and Phrases
0	Essentials & Meals	pizza, thanks, gas, costco, lunch
1	Shared Expenses & Rent	rent, split, bills, utilities, pay
2	Dining & Drinks	sushi, beer, tacos, drinks, happy hour
3	Friends & Fun	movie, bbq, party, hangout, ice cream
4	Utilities & Services	parking, hotel, venmo, fee, cleaning
5	Sarcasm & Memes	drug rug, friendship fee, jokes, comedy

These clusters capture distinct and interpretable themes, some of which—such as sarcasm—are not represented in the rule-based classification.

Comparison with Rule-Based Classification

We evaluated the alignment between our LLM-based clusters and the predefined rule-based categories using the Adjusted Rand Index (ARI). The ARI score was **–0.0016**, indicating near-random alignment between the two systems.

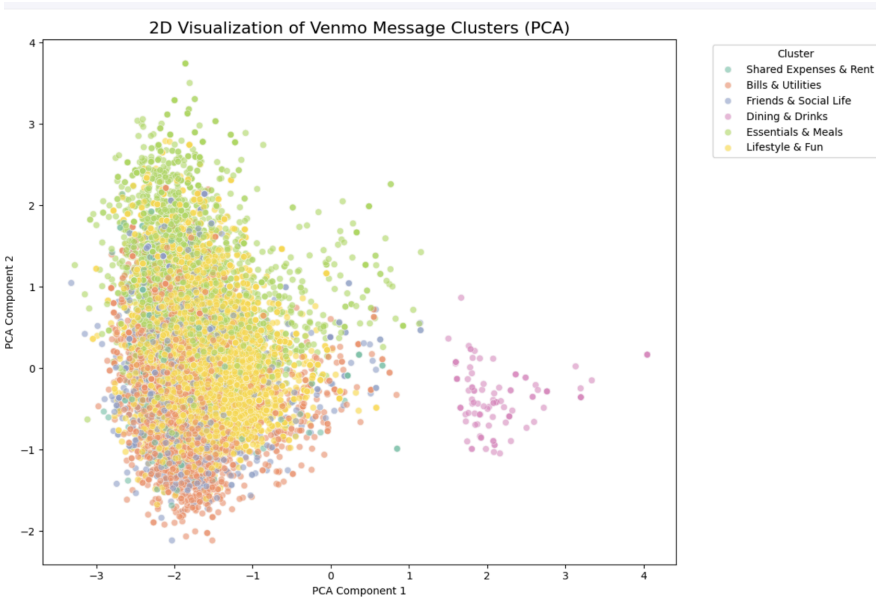
This suggests that while dictionary-based classification is interpretable and controlled, embedding-based clustering captures a different layer of meaning, often driven by context or slang.

Visualization

We applied Principal Component Analysis (PCA) to reduce the 384-dimensional embeddings to 2D and plotted them using Seaborn. The clusters appeared visually separable, supporting the effectiveness of k-means in structuring high-dimensional message data.

Each point represented a message and was color-coded by its assigned cluster. This visualization served as a helpful tool for assessing the cohesiveness and separation of topics in latent space.

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani



Strengths and Limitations

Advantages:

- Captures subtle semantics, even in short or noisy messages
- Uncovers organic patterns (e.g., inside jokes, sarcasm) missed by dictionaries
- Scales easily with new data and domains

Limitations:

- Interpretability is lower compared to rule-based tags
- Requires post-hoc labeling and analysis
- Cluster boundaries can be sensitive to preprocessing and model noise

Social Network Analytics

Venmo transactions reveal a rich, user-to-user network. We constructed an undirected social graph by treating each financial exchange as a bidirectional connection between users. Our analysis includes building the network, identifying direct and indirect connections, calculating clustering coefficients, and measuring user influence via PageRank.

Direct Friends and Friends-of-Friends

We first filtered out transactions where is_business was null to ensure interactions were between individual users. Then, we treated each user1 ↔ user2 pair as a friendship connection.

Direct Friends: A user who has directly transacted with the given user.

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

Friends-of-Friends (FoF): A user who is connected to one of the user's friends but has not directly transacted with the user.

Using PySpark:

- We extracted the undirected edge list from the transaction data.
- Then, we joined the edge list with itself to discover 2-hop connections.
- Direct connections were excluded from the FoF list.

Example results:

User ID	Number of Direct Friends	Number of Friends-of-Friends
10	7	45
361	3	15
1234	9	68

This showed that while some users had rich second-degree networks, others were relatively isolated.

Clustering Coefficient

The clustering coefficient measures how interconnected a user's friends are:

$$\text{Clustering Coefficient} = 2 \cdot \text{Triangles} / \text{Degree} \cdot (\text{Degree} - 1)$$

Steps we used:

- For each user, we formed all pairs of friends.
- We then checked if the pair was connected, forming a triangle.
- Users with fewer than 2 friends had a coefficient of 0.

User ID	Degree	Triangles	Clustering Coefficient
12	9	4	0.22
243	23	6	0.047

Most users had low coefficients, suggesting sparse interconnections even among known peers. Some anomalies (e.g., coefficients >1) may stem from data duplications or unfiltered loops.

PageRank Analysis

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

To assess user influence in the Venmo network, we used PageRank, a global social network metric that accounts for both the number and importance of connections. Our approach began by filtering the transaction data to exclude business accounts and constructed a directed graph from user-to-user transactions. Then to manage scale, we selected the top 500 users by degree centrality and clustering coefficient. Finally by using GraphFrames, we computed PageRank choosing a damping factor of 0.15 over 5 iterations.

Key Results

User ID	PageRank Score
277650	4.50
986090	4.48
871852	4.20

These users have strong transactional ties with other highly connected users, marking them as central to the network's structure.

Insight

PageRank revealed users with the most structural influence, beyond just the number of friends. This can inform user targeting, fraud detection, or influence modeling in peer-to-peer payments.

Summary

This network analysis revealed:

- A small number of users with high connectivity and influence.
- Most users were sparsely connected, with minimal friend overlap.
- Clustering and PageRank provided complementary insights into local and global user structures. This framework lays the groundwork for deeper network-based behavioral modeling.

Predictive Modeling of Venmo User Activity

To forecast user engagement on Venmo, we trained models to predict the total number of messages a user would send within the first 12 months of joining the platform. This helps us understand what types of signals—behavioral or social—best predict future activity.

Model A: Recency-Frequency Only

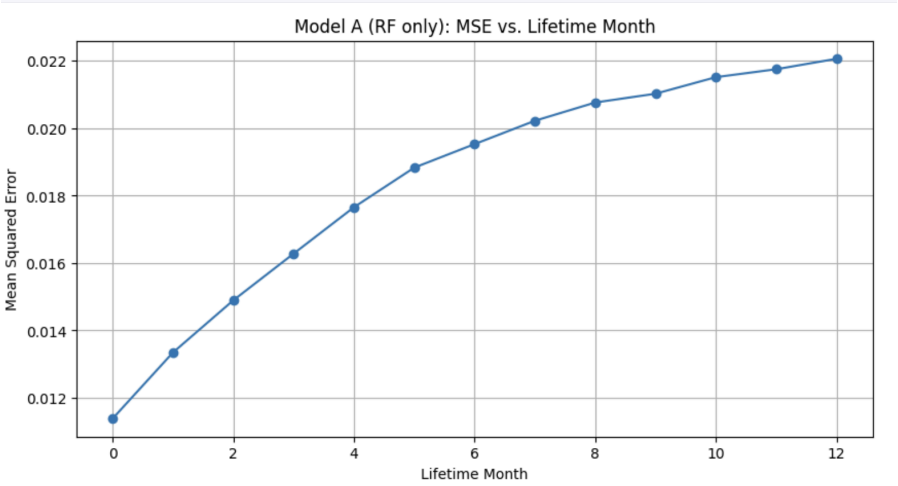
Model A used two core behavioral features:

- Recency: Time since last transaction

- Frequency: Cumulative number of transactions

These features were computed dynamically for each user over their first 12 months.

The following plot shows how Mean Squared Error (MSE) changes across lifetime months:



Interpretation: As more user history becomes available, predictions become more accurate. MSE increases with lifetime month because users with shorter histories are easier to predict. This confirms that recency and frequency are strong early indicators of engagement.

Model B: Behavioral + Social Network Metrics

Objective: Extend Model A by incorporating social network metrics—such as PageRank, degree, and clustering coefficient—into the feature set alongside recency and frequency.

Approach:

- Combined behavioral variables (recency, frequency) with social variables (degree, clustering coefficient, pagerank) using a VectorAssembler.
- Trained a Random Forest Regressor to predict the target variable Y (future 12-month transaction count).
- Evaluated Mean Squared Error (MSE) for each month m = 0 to 12, training on users observed up to that month.

Insights:

- The model showed consistently lower MSE than Model A, especially in the middle months (Month 3 to 9).
- Social metrics improved predictive power by offering a richer view of a user's engagement context—e.g., users with high PageRank or clustering tend to be more embedded in active communities, influencing their future behavior.

- Model B offers a balanced fusion of behavioral patterns and social influence.

Model C: Social Network Metrics Only

Objective: Evaluate whether social features alone—without any behavioral data—can sufficiently predict user engagement.

Approach:

- Used only degree, pagerank, and clustering coefficient as predictors in a Random Forest model.
- Followed the same temporal evaluation structure (MSE by month).

Results:

- Model C underperformed Model B, but still outperformed Model A during early months (e.g., Month 0–2), indicating that social position is an early signal of future activity.
- As the lifetime month increases, the absence of behavioral data makes Model C less competitive—suggesting that historical behavior becomes more critical over time.

Link for Text Analytics and Predictive Modeling-

https://colab.research.google.com/drive/1ew4JFqvFsx_L86MeCt-ZGT0MbbNcBCWk?usp=sharing

Link for Social Networking-
https://colab.research.google.com/drive/1CjyhbkaN0RtO27WQ3Q7X92CVOGShkZjG?usp=sharing

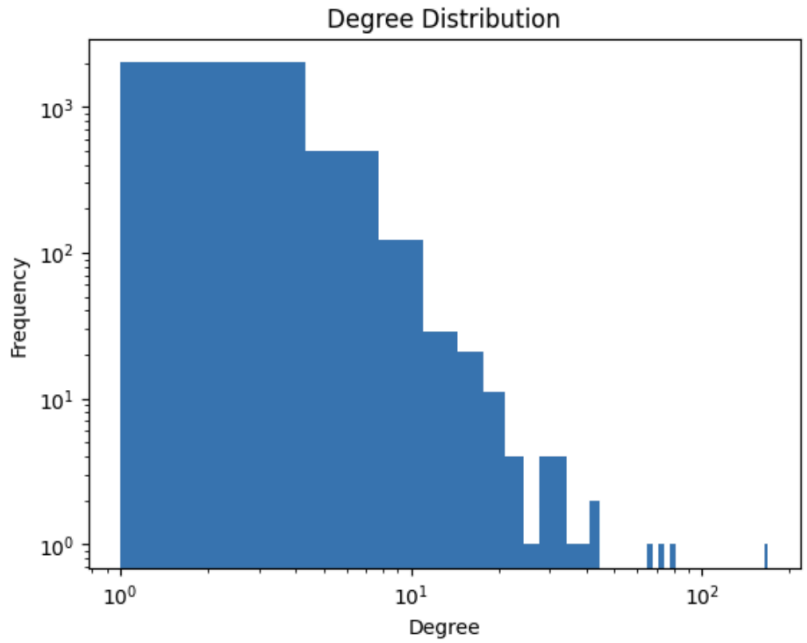
Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

EXERCISE 2 Graph Neural Network in Action

Question 1. Graph Inspection

Part 1.
The number of nodes are 2708
The number of edges are 10556
Feature dimension is 1433
Class distribution: tensor([351, 217, 418, 818, 426, 298, 180])

Part 2.

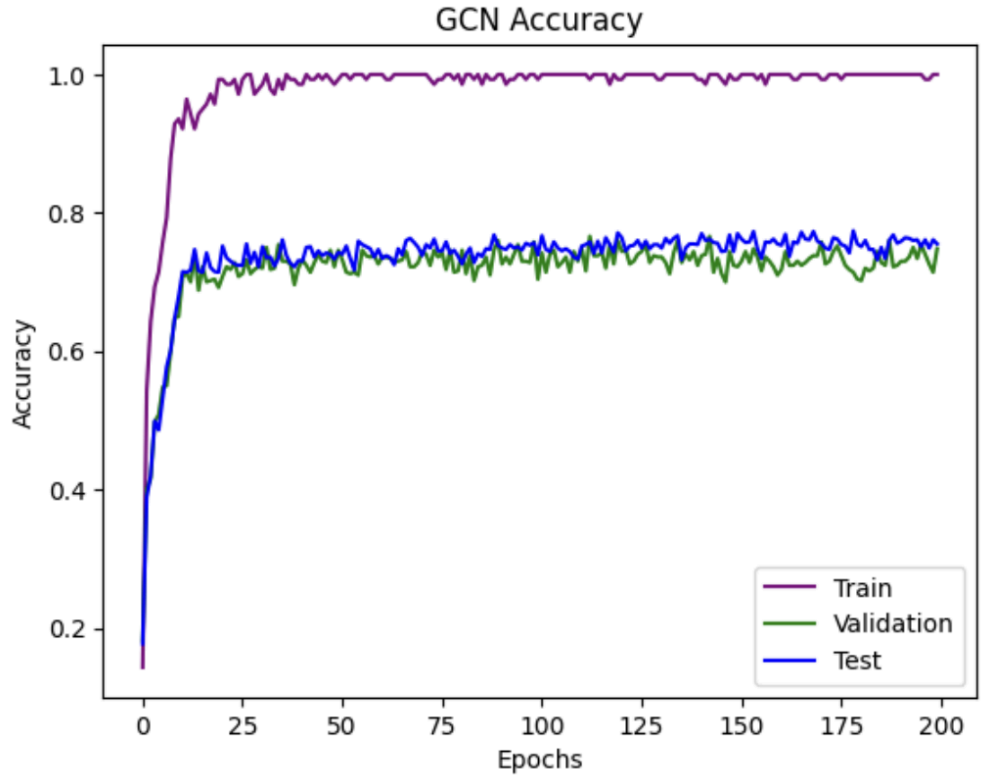


Part 3.
This log- log histogram of degree distribution of the Cora citation network shows a heavy-tailed specifically being similar to power law distribution in which as the number of degrees increases there are fewer nodes. And as we can see in the graph, above 100 the frequency is really low.
This is a common trend seen in academia where very famous and prominent papers have really high citations and are cited more frequently than most papers.

Question 2. Baseline Without Graph

Final Test Accuracy comes out to be 0.4820.

Question 3. Two layer GCN



Part 4.

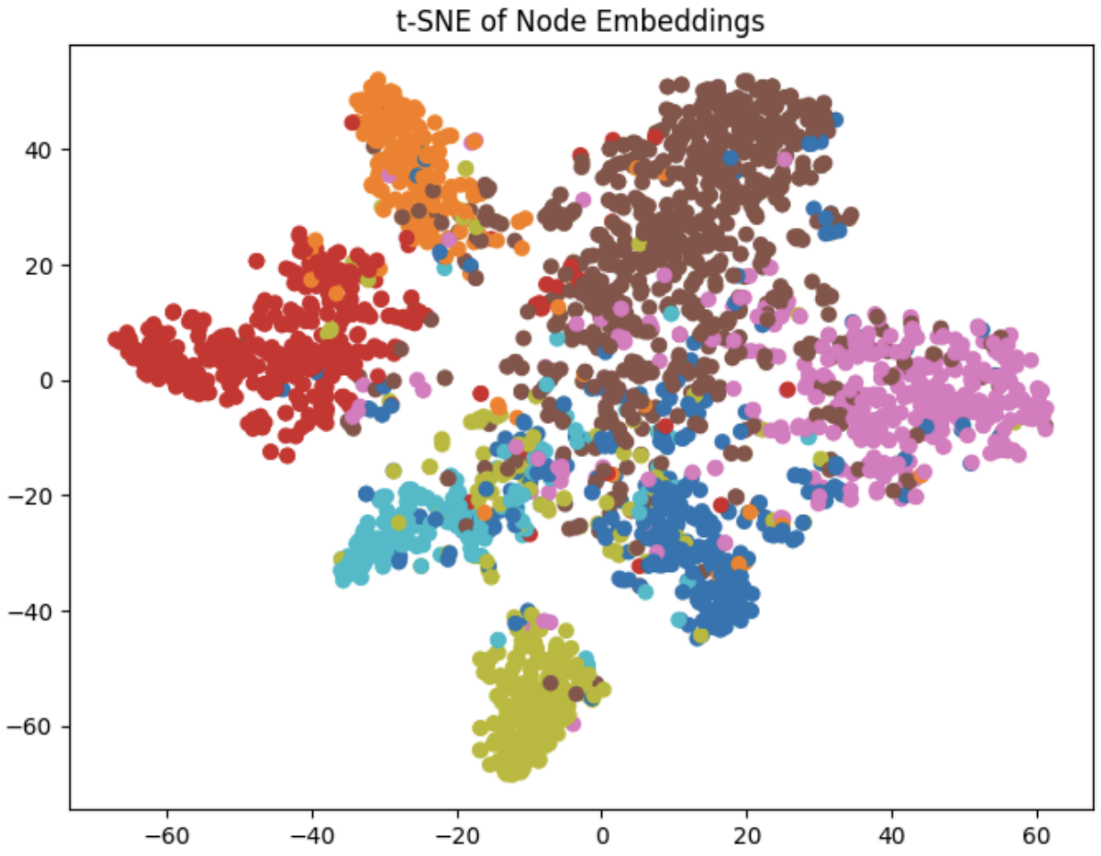
The Graph Convolutional Network (GCN) performs much better than the baseline model without graphs. The test accuracy for the baseline model becomes stagnant around ~48% which is quite low, whereas for GCN, the accuracy for test and validation reaches ~75%. This showcases the usefulness of the structure of graphs, with the presence of nodes and connections the GCN has the ability for complex representations.

Looking at the above plot, we can see that GCN reaches 100% accuracy in training very quickly which may show overfitting behaviour as for the validation and test set, the accuracy becomes stagnant very quickly. Also there is a significant gap between the accuracy of test and training set which also shows overfitting, and suboptimal generalizability. Adding weight decay and dropout has also helped in reducing overfitting, with adjusting and tuning the model, we may be able to achieve higher accuracy.

Question 4.

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

Part 1.



Part 2.

The nodes of the same research field cluster together in the above 2D t-SNE plot. This shows that the 16 dimensions hidden embeddings learned by GCN are able to collect information about the similarities between nodes and connections that are in the same category.

This kind of clustering shows automated feature learning, as compared to manual network metrics, these GCN embeddings are more context based, understand more complex behaviour and learn more directly about node features.

Part 3.

The overall experience working through GCN and understanding how helpful it is in contrast to the logistic regression model gives a clear picture about how GCN offers a more structural and in-depth approach as compared to vectors.

Big Data Analytics: Homework 5
Team: Amber, Sakshi, Karishma, Shivani

Firstly, the 16 dimensional GCN embeddings take into account the neighbourhood and node features to gain information on relationship patterns and similarities. Secondly, these embeddings are task-flexible and the node embeddings are learned directly to perform better at classification.

Whereas in Logistic regression, the relationship between the nodes are not taken into account, and the model is unable to perform wherever there is information regarding similarities and structural information for example- citations.

However, the tradeoff with GCN embeddings is that they are computationally heavy and hence difficult to scale as compared to vectors.

https://colab.research.google.com/drive/1c65cE2jhOHNztFi4ykfwdjIEkfGJcc_-?usp=sharing