# Network Analysis of Dramatic Texts

Session 1: Building Your First Network

## Introduction

Today you'll transform a play into a network—a mathematical structure that reveals patterns in how characters relate to one another. The key insight is that *the model you build is an argument, not a neutral representation*. Every choice you make about what counts as a connection shapes what the network can reveal.

By the end of this session, you'll have a visualization of your play's social world and initial measurements of which characters matter—and in what ways.

## Part 1: Predictions and Modeling Choices (15 minutes)

Before you extract any data, think through your play and commit to some predictions. You'll revisit these at the end.

### Your Predictions

> *Write your responses in the spaces below. Be specific—name characters and explain your reasoning briefly.*

**1. Who do you predict will be the most central character in your play—the one connected to the most other characters? Why?**

**2. Who do you predict will be a "bridge" figure—someone who connects groups of characters who wouldn't otherwise be connected to each other?**

**3. Are there characters you expect to be present in many scenes but not speak much? Or characters who speak a lot but only to a few people?**

**4. Do you expect to see distinct clusters or groups of characters? If so, what separates them (class, family, location, plot)?**

### Key Concept: Co-presence vs. Dialogue

Today we're building a **co-presence network**: two characters are connected if they appear in the same scene together. This captures who *could* interact—the shared dramatic space. In Session 2, you'll build a **dialogue network** based on who actually speaks to whom, and compare what each reveals.

## Part 2: Extracting Scene Data with Gemini (25 minutes)

You'll now use Gemini to extract structured data from your play. The goal is to create a matrix showing which characters appear in which scenes.

### Step 1: Upload and Initial Prompt

1. Upload your play text to Gemini.
2. Use the following prompt, adapting the bracketed sections:

**PROMPT 1: Scene-by-character matrix**

> I'm analyzing this play to build a character co-presence network. For each scene, list which characters are present (either speaking or explicitly indicated as present in stage directions).
>
> Please format your output as a table with scenes as rows and characters as columns, with "1" if the character is present in that scene and "0" if not. Include all named characters who appear in at least two scenes.
>
> Start with Act 1, then continue through the entire play.

### Step 2: Spot-Check the Output

**This is critical.** Gemini will make mistakes. Before proceeding, verify the output against your text:

- Pick 2-3 scenes you know well. Does the matrix match who's actually present?
- Are any major characters missing entirely?
- Did Gemini include characters who are only mentioned but not present?
- How did it handle characters who enter or exit mid-scene?

If you find errors, use a follow-up prompt:

**PROMPT 1b: Clarification**

> For [Act X, Scene Y], can you clarify which characters are physically present on stage versus merely mentioned? I only want characters who are actually in the scene, including those who enter partway through.

### Record Your Verification

*Document your spot-check below. What did you verify, and what (if anything) did you need to correct?*

## Part 3: Converting to an Edge List (15 minutes)

Network analysis software needs data in a specific format: an **edge list** where each row represents a connection between two characters. You'll now transform your scene matrix into this format.

**PROMPT 2: Generate edge list**

> Based on the scene-by-character matrix you created, generate an edge list for a co-presence network. Two characters share an edge if they appear in at least one scene together.
>
> Format as CSV with three columns: Source, Target, Weight
>
> The weight should be the number of scenes in which both characters appear together.
>
> Example format:

```
Source,Target,Weight
Hamlet,Horatio,8
Hamlet,Claudius,6
```

**Save this output carefully**—you'll paste it into your Google Colab notebook in the next step.

## Part 4: Visualization in Google Colab (20 minutes)

Now you'll generate Python code to visualize your network. Ask Gemini to write the code, then run it in Google Colab.

#### PROMPT 3: Generate visualization code

> Generate Python code using NetworkX and Matplotlib to visualize the co-presence network from my edge list. The code should:
>
> 1. Read the edge list CSV data (I'll paste it as a string variable)
> 2. Create an undirected weighted graph
> 3. Use a force-directed layout (spring layout)
> 4. Size nodes by degree centrality (more connections = larger node)
> 5. Make edge thickness proportional to weight (scenes shared)
> 6. Label all nodes with character names
> 7. Print a table of degree centrality values sorted from highest to lowest
>
> Make the figure large enough to read (at least 12x12 inches).

### Running the Code

1. Open Google Colab (colab.research.google.com)
2. Create a new notebook
3. Paste Gemini's code into a cell
4. Replace the placeholder edge list data with your actual CSV data
5. Run the cell (Shift+Enter)

**Troubleshooting:** If you get errors, paste the error message back to Gemini and ask it to fix the code. Common issues include missing import statements or formatting problems with the CSV data.

## Part 5: First Analysis (10 minutes)

With your visualization in front of you, answer the following questions. These will form the basis for your write-up.

### Analysis Checkpoint 1

**1. Who has the highest degree centrality (most connections)?**

Does this match your prediction from Part 1? Why or why not?

**2. Look at the visual structure of the network. Do you see any clusters—groups of characters who connect primarily to each other?**
If so, what seems to define these groups (family, class, subplot, location)?

**3. Who appears to bridge different clusters?**
Look for characters positioned between groups. Does this match your prediction?

**4. What surprises you about this visualization?**
Is any character more or less central than you expected? Are any connections surprising?

**5. What can't you see in this network?**
What information about the play is lost when you represent it this way?

## Before You Leave

- Save your Google Colab notebook (File → Save a copy in Drive)
- Save or screenshot your network visualization
- Keep your edge list CSV data accessible
- Keep your Gemini conversation—you'll build on it in Session 2

### Preview: Session 2

Next session, you'll build a **dialogue network** (who speaks to whom), make it **directed** (who initiates), calculate additional centrality measures (betweenness, eigenvector), and compare what different network representations reveal about your play.

## Quick Reference: What Degree Centrality Tells You

| | |
|---|---|
| **High degree centrality** | Character shares scenes with many others. Could indicate protagonists, servants who move between groups, or messengers. |
| **Low degree centrality** | Character appears with few others. Could indicate isolated figures, characters confined to one subplot, or minor roles. |
| **Weighted degree (strength)** | Sum of edge weights. Distinguishes between a character who appears once with ten people vs. ten times with one person. |

**Remember:** Degree centrality measures *breadth* of connection in the co-presence network—who moves through many social spaces in the play. In Session 2, you'll learn measures that capture different kinds of importance: bridging (betweenness) and proximity to power (eigenvector).