Deeper analysis and Network statistics

Jinxi_Hu-48528608, Samarth_Grover-38220463

2025-11-09

set up

```r
library(readr)
library(igraph)
library(RColorBrewer)
library(ggplot2)
library(reshape2)
library(scales)
set.seed(48528608)
```

Load and prepare connected nodes data (Jinxi Hu)

```r
# Load the connected nodes and edges data
nodes_connected <- read.csv("data/nodes_connected.csv")
edges_connected <- read.csv("data/edges_connected.csv")

cat("=== DATA LOADING ===\n")
```

```
## === DATA LOADING ===
```

```r
cat("Connected nodes loaded:", nrow(nodes_connected), "\n")
```

```
## Connected nodes loaded: 1582
```

```r
cat("Connected edges loaded:", nrow(edges_connected), "\n\n")
```

```
## Connected edges loaded: 3757
```

```r
# Create the graph from connected nodes only
graph_connected <- graph_from_data_frame(edges_connected,
                                         vertices = nodes_connected,
                                         directed = TRUE)
# Remove multiple edges and self-loops for cleaner analysis
graph_connected <- simplify(graph_connected,
                           remove.multiple = TRUE,
                           remove.loops = TRUE)

cat("Graph created successfully\n")
```

```
## Graph created successfully
```

```r
cat("Final nodes in graph:", vcount(graph_connected), "\n")
```

```
## Final nodes in graph: 1582
```

```r
cat("Final edges in graph:", ecount(graph_connected), "\n\n")
```

```
## Final edges in graph: 3730
```

Basic Graph Analysis (Jinxi Hu)

```r
cat("=== BASIC GRAPH STATISTICS ===\n")
```

```
## === BASIC GRAPH STATISTICS ===
```

```r
cat("Total nodes:", vcount(graph_connected), "\n")
```

```
## Total nodes: 1582
```

```r
cat("Total edges (citations):", ecount(graph_connected), "\n")
```

```
## Total edges (citations): 3730
```

```r
cat("Network density:", round(edge_density(graph_connected), 6), "\n")
```

```
## Network density: 0.001491
```

```r
cat("Is directed:", is_directed(graph_connected), "\n")
```

```
## Is directed: TRUE
```

```r
cat("Is weighted:", is_weighted(graph_connected), "\n\n")
```

```
## Is weighted: FALSE
```

```r
# Calculate degree statistics
all_degrees <- degree(graph_connected, mode = "all")
in_degrees <- degree(graph_connected, mode = "in")
out_degrees <- degree(graph_connected, mode = "out")

cat("=== DEGREE STATISTICS ===\n")
```

```
## === DEGREE STATISTICS ===
```

```r
cat("Average total degree:", round(mean(all_degrees), 2), "\n")
```

```
## Average total degree: 4.72
```

```r
cat("Average in-degree (citations received):", round(mean(in_degrees), 2), "\n")
```

```
## Average in-degree (citations received): 2.36
```

```r
cat("Average out-degree (citations made):", round(mean(out_degrees), 2), "\n\n")
```

```
## Average out-degree (citations made): 2.36
```

```r
cat("Degree range (total):", min(all_degrees), "-", max(all_degrees), "\n")
```

```
## Degree range (total): 0 - 110
```

```r
cat("Degree range (in):", min(in_degrees), "-", max(in_degrees), "\n")
```

```
## Degree range (in): 0 - 104
```

```r
cat("Degree range (out):", min(out_degrees), "-", max(out_degrees), "\n\n")
```

```
## Degree range (out): 0 - 29
```

```r
cat("Standard deviation (total):", round(sd(all_degrees), 2), "\n")
```

```
## Standard deviation (total): 6.74
```

```r
cat("Standard deviation (in):", round(sd(in_degrees), 2), "\n")
```

```
## Standard deviation (in): 6.2
```

```r
cat("Standard deviation (out):", round(sd(out_degrees), 2), "\n\n")
```

```
## Standard deviation (out): 2.72
```

```r
cat("=== COMPONENT ANALYSIS ===\n")
```

```
## === COMPONENT ANALYSIS ===
```

```r
# Analyze weakly connected components
weak_components <- components(graph_connected, mode = "weak")
cat("Number of weakly connected components:", weak_components$no, "\n")
```

```
## Number of weakly connected components: 58
```

```
cat("Size of largest weak component:", max(weak_components$csize), "\n")
```

```
## Size of largest weak component: 1433
```

```
cat("Proportion of nodes in largest weak component:",
    round(max(weak_components$csize) / vcount(graph_connected) * 100, 2), "%\n\n")
```

```
## Proportion of nodes in largest weak component: 90.58 %
```

```
# Analyze strongly connected components
strong_components <- components(graph_connected, mode = "strong")
cat("Number of strongly connected components:", strong_components$no, "\n")
```

```
## Number of strongly connected components: 1569
```

```
cat("Size of largest strong component:", max(strong_components$csize), "\n")
```

```
## Size of largest strong component: 6
```

```
cat("Proportion of nodes in largest strong component:",
    round(max(strong_components$csize) / vcount(graph_connected) * 100, 2), "%\n\n")
```

```
## Proportion of nodes in largest strong component: 0.38 %
```

```
# Component size distribution
cat("Weak component sizes (top 10):\n")
```

```
## Weak component sizes (top 10):
```

```
weak_sizes <- sort(weak_components$csize, decreasing = TRUE)
print(head(weak_sizes, 10))
```

```
##  [1] 1433   27    9    5    5    4    3    3    3    3
```

```
cat("\nStrong component sizes (top 10):\n")
```

```
##
## Strong component sizes (top 10):
```

```
strong_sizes <- sort(strong_components$csize, decreasing = TRUE)
print(head(strong_sizes, 10))
```

```
##  [1] 6 2 2 2 2 2 2 2 2 1
```

```r
# Create comprehensive degree distribution analysis

# Prepare data for plotting
degree_data <- data.frame(
  node_id = V(graph_connected)$name,
  total_degree = all_degrees,
  in_degree = in_degrees,
  out_degree = out_degrees
)

# Add node attributes for additional analysis
degree_data$institution <- V(graph_connected)$institution
degree_data$subtopic <- V(graph_connected)$subtopic
degree_data$year <- V(graph_connected)$year
degree_data$citations <- V(graph_connected)$citations

cat("=== DEGREE DISTRIBUTION SUMMARY ===\n")
```

```
## === DEGREE DISTRIBUTION SUMMARY ===
```

```r
cat("Total degree quartiles:\n")
```

```
## Total degree quartiles:
```

```r
print(quantile(all_degrees))
```

```
##   0%  25%  50%  75% 100%
##    0    1    3    5  110
```

```r
cat("\nIn-degree quartiles:\n")
```

```
##
## In-degree quartiles:
```

```r
print(quantile(in_degrees))
```

```
##   0%  25%  50%  75% 100%
##    0    0    1    2  104
```

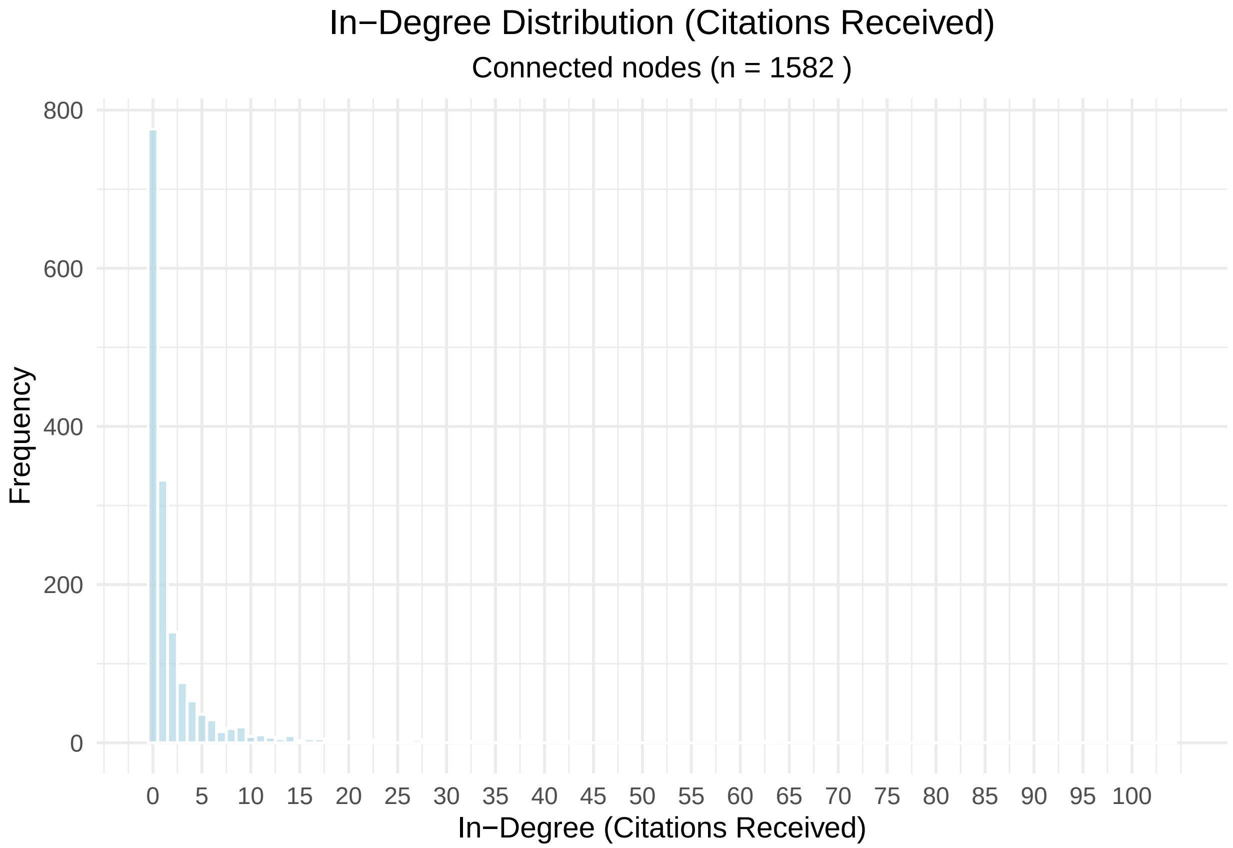
```r
cat("\nOut-degree quartiles:\n")
```

```
##
## Out-degree quartiles:
```

```r
print(quantile(out_degrees))
```

```
##   0%  25%  50%  75% 100%
##    0    1    2    3   29
```

```r
# In-degree distribution histogram
ggplot(degree_data, aes(x = in_degree)) +
  geom_histogram(binwidth = 1, fill = "lightblue", alpha = 0.7, color = "white") +
  labs(title = "In-Degree Distribution (Citations Received)",
       subtitle = paste("Connected nodes (n =", vcount(graph_connected), ")"),
       x = "In-Degree (Citations Received)",
       y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0, max(in_degrees), by = 5))
```



```r
# Print summary statistics
cat("\nIN-DEGREE DISTRIBUTION STATISTICS:\n")
```

```
##
## IN-DEGREE DISTRIBUTION STATISTICS:
```

```r
cat("Mean:", round(mean(in_degrees), 2), "\n")
```

```
## Mean: 2.36
```

```r
cat("Median:", median(in_degrees), "\n")
```

```
## Median: 1
```

```r
cat("Mode:", names(sort(table(in_degrees), decreasing = TRUE))[1], "\n")
```

```
## Mode: 0
```

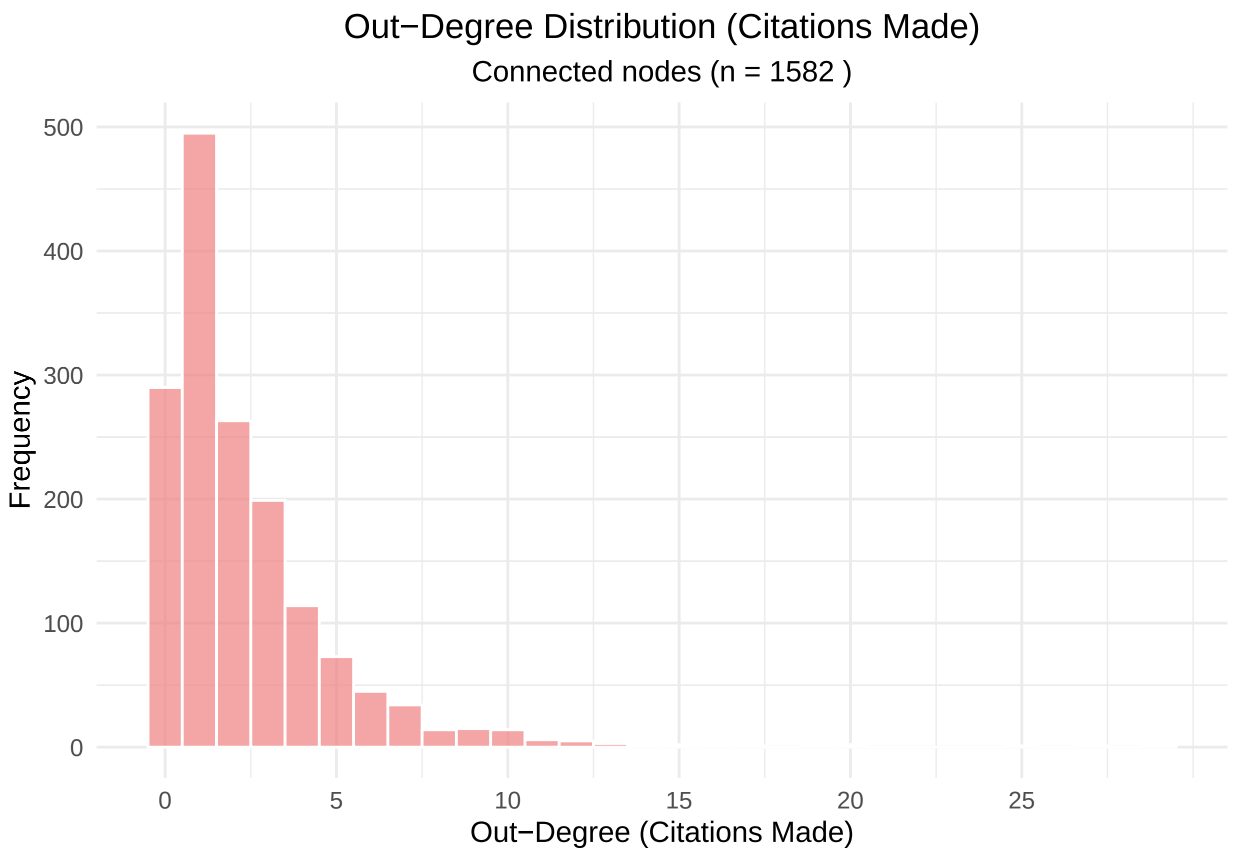
```r
cat("Nodes with 0 in-degree:", sum(in_degrees == 0), "\n")
```

```
## Nodes with 0 in-degree: 776
```

```r
cat("Nodes with >10 in-degree:", sum(in_degrees > 10), "\n")
```

```
## Nodes with >10 in-degree: 80
```

```r
# Out-degree distribution histogram
ggplot(degree_data, aes(x = out_degree)) +
  geom_histogram(binwidth = 1, fill = "lightcoral", alpha = 0.7, color = "white") +
  labs(title = "Out-Degree Distribution (Citations Made)",
       subtitle = paste("Connected nodes (n =", vcount(graph_connected), ")"),
       x = "Out-Degree (Citations Made)",
       y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0, max(out_degrees), by = 5))
```



```r
# Print summary statistics
cat("\nOUT-DEGREE DISTRIBUTION STATISTICS:\n")
```

```
##
## OUT-DEGREE DISTRIBUTION STATISTICS:
```

```r
cat("Mean:", round(mean(out_degrees), 2), "\n")
```

```
## Mean: 2.36
```

```r
cat("Median:", median(out_degrees), "\n")
```

```
## Median: 2
```

```r
cat("Mode:", names(sort(table(out_degrees), decreasing = TRUE))[1], "\n")
```

```
## Mode: 1
```

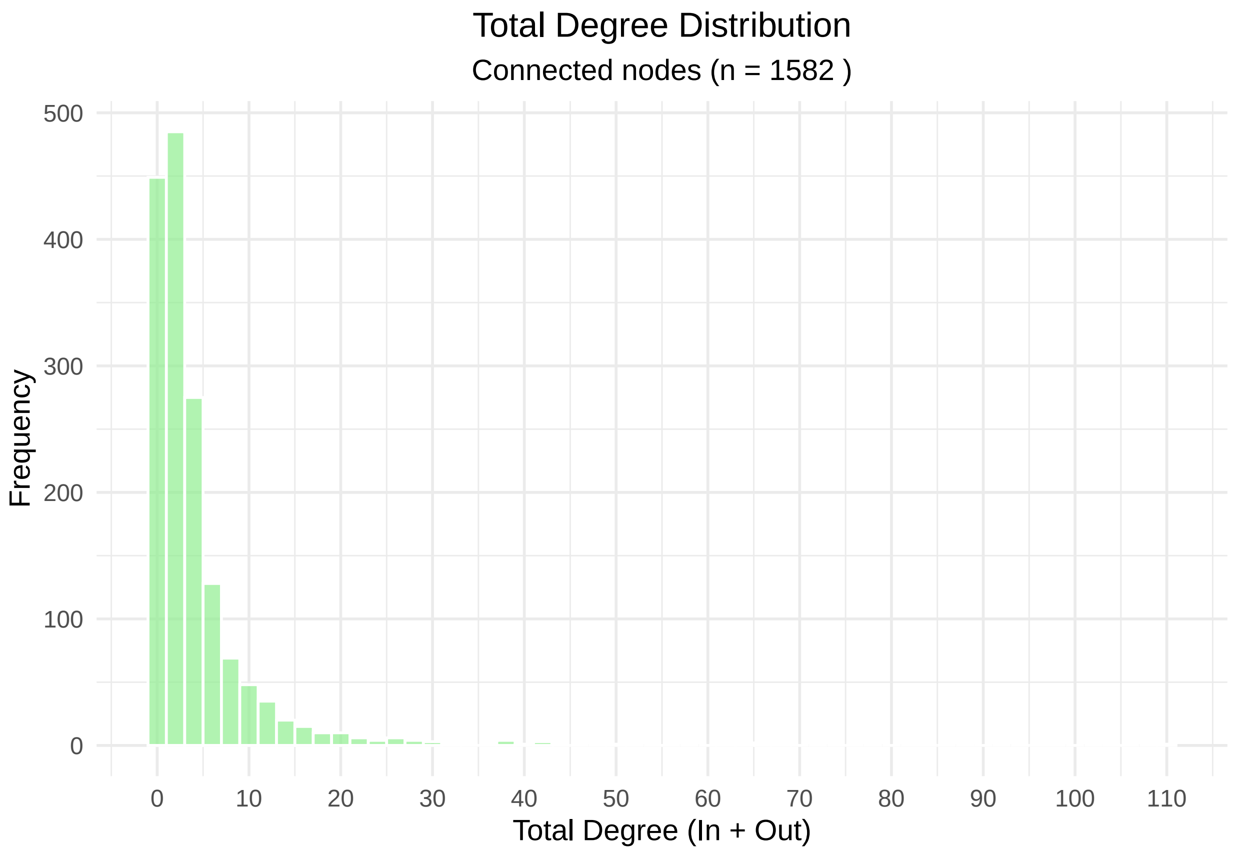
```r
cat("Nodes with 0 out-degree:", sum(out_degrees == 0), "\n")
```

```
## Nodes with 0 out-degree: 290
```

```r
cat("Nodes with >10 out-degree:", sum(out_degrees > 10), "\n")
```

```
## Nodes with >10 out-degree: 26
```

```r
# Total degree distribution histogram
ggplot(degree_data, aes(x = total_degree)) +
  geom_histogram(binwidth = 2, fill = "lightgreen", alpha = 0.7, color = "white") +
  labs(title = "Total Degree Distribution",
       subtitle = paste("Connected nodes (n =", vcount(graph_connected), ")"),
       x = "Total Degree (In + Out)",
       y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = seq(0, max(all_degrees), by = 10))
```



```r
# Print summary statistics
cat("\nTOTAL DEGREE DISTRIBUTION STATISTICS:\n")
```

```
##
## TOTAL DEGREE DISTRIBUTION STATISTICS:
```

```r
cat("Mean:", round(mean(all_degrees), 2), "\n")
```

```
## Mean: 4.72
```

```r
cat("Median:", median(all_degrees), "\n")
```

```
## Median: 3
```

```r
cat("Mode:", names(sort(table(all_degrees), decreasing = TRUE))[1], "\n")
```

```
## Mode: 1
```

```r
cat("Nodes with degree 1:", sum(all_degrees == 1), "\n")
```

```
## Nodes with degree 1: 434
```

```r
cat("Nodes with degree >20:", sum(all_degrees > 20), "\n")
```

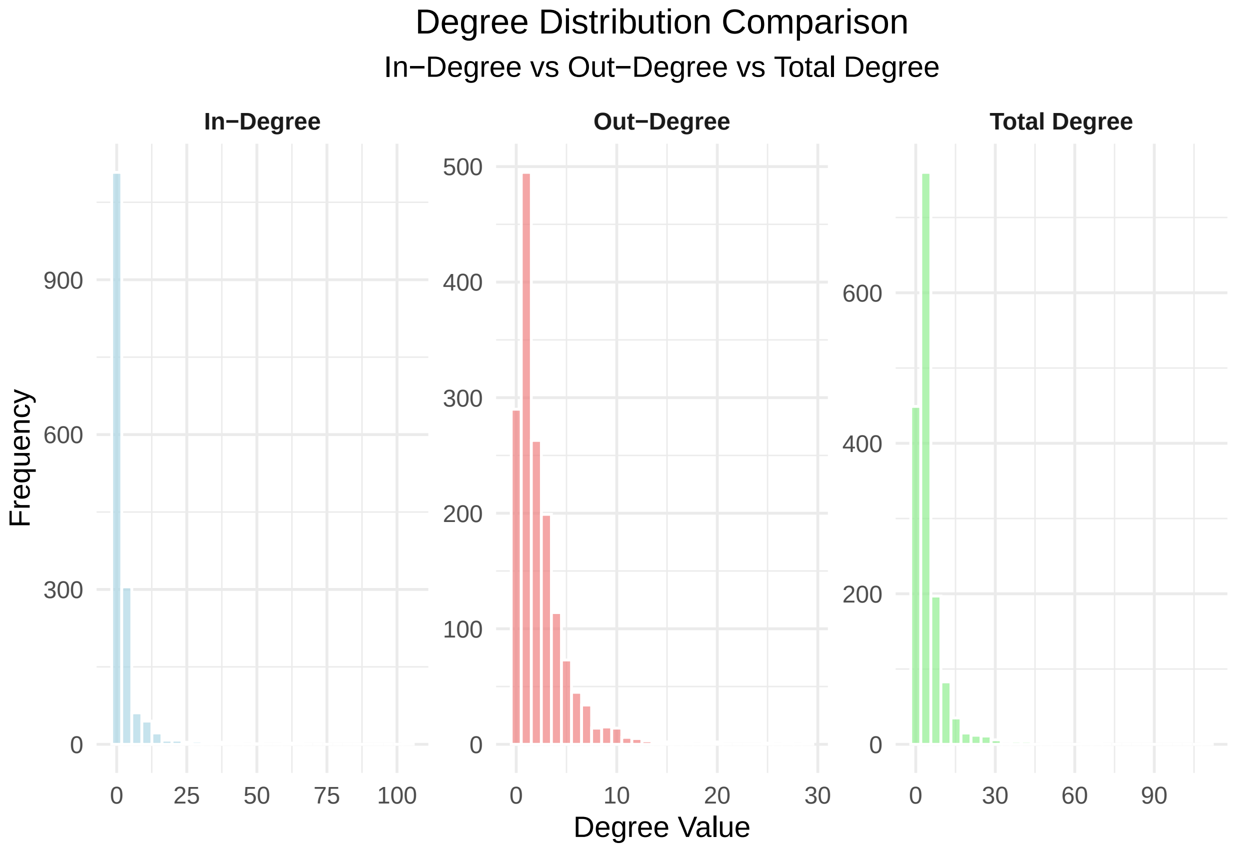
```
## Nodes with degree >20: 43
```

```r
# Create combined degree distribution plot
degree_long <- reshape2::melt(degree_data[, c("in_degree", "out_degree", "total_degree")],
                              variable.name = "degree_type",
                              value.name = "degree_value")
```

```
## No id variables; using all as measure variables
```

```r
# Rename for better labels
degree_long$degree_type <- factor(degree_long$degree_type,
                                  levels = c("in_degree", "out_degree", "total_degree"),
                                  labels = c("In-Degree", "Out-Degree", "Total Degree"))

# Create faceted histogram
ggplot(degree_long, aes(x = degree_value, fill = degree_type)) +
  geom_histogram(alpha = 0.7, color = "white", bins = 30) +
  facet_wrap(~degree_type, scales = "free") +
  scale_fill_manual(values = c("In-Degree" = "lightblue",
                               "Out-Degree" = "lightcoral",
                               "Total Degree" = "lightgreen")) +
  labs(title = "Degree Distribution Comparison",
       subtitle = "In-Degree vs Out-Degree vs Total Degree",
       x = "Degree Value",
       y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        legend.position = "none",
        strip.text = element_text(face = "bold"))
```



```r
# Analyze the largest component in detail
cat("=== LARGEST COMPONENT DETAILED ANALYSIS ===\n")
```

```
## === LARGEST COMPONENT DETAILED ANALYSIS ===
```

```r
# Extract the largest weakly connected component
largest_comp_nodes <- which(weak_components$membership == which.max(weak_components$csize))
largest_component <- induced_subgraph(graph_connected, largest_comp_nodes)

cat("Largest component statistics:\n")
```

```
## Largest component statistics:
```

```r
cat("Nodes:", vcount(largest_component), "\n")
```

```
## Nodes: 1433
```

```r
cat("Edges:", ecount(largest_component), "\n")
```

```
## Edges: 3634
```

```r
cat("Density:", round(edge_density(largest_component), 6), "\n")
```

```
## Density: 0.001771
```

```r
cat("Average in degree:", round(mean(degree(largest_component, mode = "in")), 2), "\n")
```

```
## Average in degree: 2.54
```

```r
cat("Average out degree:", round(mean(degree(largest_component, mode = "out")), 2), "\n")
```

```
## Average out degree: 2.54
```

```r
cat("Average degree:", round(mean(degree(largest_component, mode = "all")), 2), "\n")
```

```
## Average degree: 5.07
```

```r
cat("Diameter:", diameter(largest_component, directed = FALSE), "\n")
```

```
## Diameter: 13
```

```r
cat("Average path length:", round(mean_distance(largest_component, directed = FALSE), 2), "\n\n")
```

```
## Average path length: 4.5
```

```r
# Check if there are smaller components worth analyzing
if(length(unique(weak_components$csize)) > 1) {
  second_largest_size <- sort(weak_components$csize, decreasing = TRUE)[2]
  cat("Second largest component size:", second_largest_size, "\n")
  cat("Ratio (largest/second largest):", round(max(weak_components$csize) /
  second_largest_size, 2), "\n")
}
```

```
## Second largest component size: 27
## Ratio (largest/second largest): 53.07
```