discourseGT: An R package to analyze discourse networks in educational contexts

Qi Cui, Joshua P. Le, Albert Chai, Andrew S. Lee, Jitarth Sheth, Kevin Banh, Priya Pahal, Katherine Ly, and Stanley M. Lo

1 University of California, San Diego 2 University of California, Los Angeles ¶ Corresponding author * These authors contributed equally.

DOI: 10.21105/joss.05143

Summary

Student discussions in the classroom are important components of the learning process. Research methods for analyzing discourse in these educational contexts are predominantly qualitative. discourseGT is an R package that adapts graph theory to analyze discourse networks of students collaborating in small groups. This software package takes data on the sequential student talk-turns in a classroom discussion and produces statistics and generates plots based on graph theory and additional parameters. Overall, discourseGT provides new features that can yield insight on the dynamics of student discussions relevant to education researchers.

Statement of Need

Many contemporary applications and software packages are optimized for large-scale networks. For example, igraph (Csardi & Nepusz, 2006), network (Butts, 2021), and sna (Butts, 2020) were developed to analyze social media networks (Jones et al., 2017), epidemiological networks (Christakis & Fowler, 2011), and political networks (Hobbs et al., 2016), respectively. In contrast, discourse networks in educational contexts are substantially smaller, typically with only 3-8 students (Wagner & González-Howard, 2018). Consequently, certain parameters that are relevant for these larger networks are not necessarily applicable, and analysis of discourse networks demands additional parameters beyond what is available in graph theory (Chai et al., 2019; Lou et al., 2001).

Usage
discourseGT comes equipped with example data. We use this dataset to demonstrate the utility of discourseGT in examining discourse networks.

Preparing an igraph Object

An igraph object is the core input to many of the modular analytical functions offered in discourseGT. Prior to generating an igraph object, a weighted edge list needs to be generated from imported raw data, structured as two columns containing sequential nodes or individual students who start or continue a discussion episode (Chai et al. (2019)). This is addressed by the tabulate_edges function. By default, the weight of an edge is defined as the number...
of times an edge has occurred between two nodes. Weights can be redefined based on other available criteria, but this must be done manually.

```r
install.packages("discourseGT")
library(discourseGT)
data(sampleData1)
# Calculate the weighted edge list
tabEdge <- tabulate_edges(sampleData1, silentNodes = 0)
# Check the weighted edge list
head(tabEdge$master)
## source target weight
## 1 1 1 8
## 2 2 1 25
## 3 3 1 49
## 4 4 1 75
## 5 1 2 28
## 6 3 2 11
```

Generation of an igraph object is handled by the `prepareGraphs` function, which requires the following information:

- The variable that stores the weighted edge list
- The title of the project. Default: NULL
- Is the graph weighted? Default: TRUE

```r
prepNet <- prepareGraphs(
  tabEdge,
  project_title = "Sample Data 1",
  weightedGraph = TRUE
)
```

The graph settings specified by `prepareGraphs` will influence the analytical output of downstream functions.

**Running Graph Analysis**

discourseGT offers graph theory-based analytics via two separate functions: `coreNetAnalysis()` and `subgroupsNetAnalysis()`.

`coreNetAnalysis()` will perform core graph theory operations, such as the counting number of nodes and edges and calculating edge weights, average graph degree, centrality, and other graph theory parameters (Chai et al., 2019).

```r
coreNet <- coreNetAnalysis(prepNet)
```

`subgroupsNetAnalysis()` utilizes the Girvan-Newman algorithm to detect subgroups within the overall network (Girvan & Newman, 2002), such that:

```r
subNet <- subgroupsNetAnalysis(
  prepNet, raw_input = sampleData1,
  normalized = TRUE
)
```

**Generating Summaries**

While it is possible to display the generated igraph object, core network statistics, and subgroup statistics as separate outputs, it can be helpful to view them as an overall summary of a network’s graph theory analytics. Furthermore, combining all of these outputs into a single variable is a necessary step in exporting them as a single text file. The `summaryNet()` function will combine

the outputs from prepareGraphs(), coreNetAnalysis(), and subgroupsNetAnalysis() as such:

```r
summaryData <- summaryNet(
  netIntconfigData = prepNet,
  coreNetAnalysisData = coreNet,
  subgroupsNetAnalysisData = subNet,
  display = TRUE
)
```

### Basic Visualization

`discourseGT` offers several methods to visualize networks. For a basic network graph, `basicPlot()` is used, which offers parameters that modify the plotting algorithm, edge curvature, arrow size, and edge weight scaling.

The default plotting algorithm of `basicPlot()` is Fruchterman-Reingold, denoted by \(0\) (Fruchterman & Reingold, 1991). This is typically the best option to use because it attempts to minimize edge intersections in the final plot, improving readability. Other projections include Kamada-Kawai (Kamada & Kawai, 1989) and Reingold-Tilford (Reingold & Tilford, 1981), denoted by \(1\) and \(2\), respectively.

```r
basicPlot(prepNet, graph_selection_input = 0, curvedEdgeLines = TRUE,
          arrowSizeMultiplier = 2, scaledEdgeLines = TRUE,
          scaledMin = 1, scaledMax = 10)
```

### Running Non-Graph Theory Analysis

`discourseGT` does not require an `igraph` object to produce a non-graph theory (NGT) analysis. Rather, the `plotNGTData()` function utilizes the two-column raw data to generate its output. Additionally, it requires the duration of the conversation (in minutes) and the number of silent nodes (i.e., students who did not speak at all) in the discourse network.

```r
plotNGTData(data = sampleData1, convoMinutes = 90,
             silentNodes = 0)
```

### Conclusions

We developed an R package `discourseGT` that considers student discourse as a network and quantitatively examines the dynamics of small-group discussions. This paper offers a step-by-step case example that contextualizes the workflow with educational data. Details about usage and more elaborate examples are hosted online at the Comprehensive R Archive Network (https://cran.rstudio.com/web/packages/discourseGT/index.html), and the current version of the graphical user interface (GUI) is 1.1.0 (https://sites.google.com/ucsd.edu/dgt/home).

### Acknowledgements

We thank B. N. Nguyen and S. Lam for their contributions to the early testing of the package with example datasets. We thank L. Hobbie for his input on several package functions and usability. We are grateful for the support and work that CRAN Volunteers put into examining and publishing R packages on the Comprehensive R Archive Network.

This material is based upon work supported by the National Science Foundation under grant number DUE-2022975. The authors declare that there are no conflicts of interest.

References


