An Analysis of Collaborative Efforts in Scientific Communities - A Social Network Based Approach

Michael Bowen Mackenzie Sweeney

May 14, 2014

Contents

An aside to the reader

We hope that this documentation will serve as a dual purpose; not only as a summary of the experiments performed during the Spring 2014 semester, but also as a reference and guide going forward with our research.

Abstract

Social Network Analysis (SNA) is an increasingly popular methodology used to study emerging patterns as a result of interactions between individuals within a community. The wealth of information extracted from the analysis of such interactions is used for a multitude of applications; these commonly include describing behaviors of social entities, constructing predictive models, and inferring missing data. This project seeks to glean insight about the characteristics and members of communities present in the greater National Science Foundation (NSF) research community using SNA techniques. At this stage, the problem is

expanding and emerging as data exploration illuminates the potential information present in the NSF grant award data. The current approach is two-fold. First, the objective is to utilize traditional methods of network analysis, namely to draw conclusions via network structure and node attributes separately through visual analysis and topic modelling. The second is to utilize a SNA tool recently released by a data mining group at Stanford University known as CESNA, or Communities from Edge Structure and Node Attributes. A multi-perspective approach will provide a deeper insight when exploring the reaches of the NSF research community.

1 Introduction and Background

1.1 Social Network Analysis and Community Detection Techniques

Communities, in the context of data mining, are groups of interacting objects (data records). The relationships formed by these objects, and how they impact individuals or the community as a whole, are generally the focal point of Social Network Analysis. Traditionally these networks are analyzed via a careful study of the emergent network structure formed by these relationships, or the commonality of attributes across records. Both methodologies are used to detect underlying communities present in the data. Only recently have new approaches (CESNA) been proposed which seek to detect communities by combining these two methods.

1.1.1 Network Structure

The main benefit of using network structure to analyze social networks is that it provides a visual representation of the data from which conclusions about the data can be inferred. A large portion of the human brain is dedicated to visual processing and analysis, including deduction via spatial relations, so this technique plays to our natural strengths.
Network structures are visually depicted though two and three-dimensional graphs. General graph theory is applied for construction and analysis purposes. For SNA, nodes are often people, and edges represent a link between two people or groups of people. Graphs can be either directed or undirected. Directed graphs use arrows to show an explicit relation between two nodes which is used to indicate further information. Undirected graphs merely indicate that there exists an edge (link) between the two nodes. For the purposes of data mining algorithms used in SNA graphs and networks differ in a few key ways. Graphs are simple representations where primitive types (a single int, string, float, etc.) connect nodes of singleton primitive values. Networks expand upon this by making each edge and node an object. As stated previously, it is common in SNA for nodes to represent people, this is further expanded upon in networks by attaching attributes to this person which further describe the individual. For example, an individual node object could contain a document vector attribute which ontologically represents the individual. Edges also become objects, and thus also have attributes attached to them. The purpose of attaching attributes to edges is to perform a weighting analysis which indicates strength of relation between two nodes. These values are generally normalized between 0 and 1, with 0 being a weak relation, and 1 being a very strong relation.

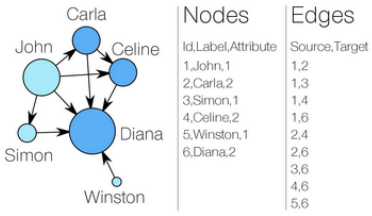

Figure 1: A directed graph.

The key metrics considered during network structure analysis are:

- Bridges
  - A node which provides a link between two communities for which otherwise there would be no connection. Can be removed to break communities apart farther.
- Centrality
  - Indicator of "importance" of an individual node, cluster of nodes, or the population of nodes. Common centrality measures include degree, closeness, betweenness, and eigenvector.
  - Example - Betweenness centrality describes a single nodes importance in terms of overall network centrality using shortest paths between nodes. Nodes with a high betweenness centrality are more well connected and are considered more important.
- Density
  - Proportion of direct ties to total possible ties in a network. Density $= \frac{direct\ ties}{all\ possible\ ties}$ Low values indicate a sparse graph, while high values indicate a very heavily connected graph.
- Distance
  - Number of edges separating two nodes. Used for "small world" analysis showing degrees out separation.
- Tie strength
  - How close the tie is between two nodes; generally indicated by edge weight.

1.1.2 Node Attributes

One of the most common ways of analyzing commonality between nodes, individuals in SNA, is through topic modeling. In topic models each node has a document vector which indicates the presence of abscence of a string of characters. A comparison is performed across the document vectors of each node in order to assess how much commonality two nodes, or a group of nodes, possess. Word clouds, where more frequent words are large and less frequent proportionately smaller, are often used to provide a quick visual representation of an individual node.

One popular topic model technique is Latent Dirichlet Allocation (LDA). LDA is a generative model, stochastic model generation based on statistical parameters, through which document analysis is performed. Each given topic in an LDA model has a probability of generating words which are deemed to be either related or unrelated to the topic at hand. These generated word models are compared to the constructed document vectors for each node in order to assign relational tie strengths between nodes. Communities are formed by analyzing which nodes are associated with which topics.

1.1.3 CESNA

As previously mentioned, detection of communities within a data set has been traditionally performed by utilizing one of two modalities, an analysis of node attributes or network structure. CESNA takes both of these modalities into account when performing analysis. Independently, both techniques have several drawbacks. Network structure clustering techniques, such as Single-Assignment Clustering, cannot detect overlapping communities. Topic models can detect overlapping communities, however they assume what's known as "soft" node-community relationships, which weakens the ability for a single node to belong to several different communities at the same time. Also, the temporal performance of previous community detection algorithms prohibits them from being effective for analysis of extremely large data sets such as the entire population of NSF grant data. CESNA has a linear run time in terms of number of edges, which

makes it ideal for data sets of this size.

CESNA claims to not only detect overlapping communities, and also not suffer from soft node community membership assumptions. Individuals can retain their strength to relative communities that they overlap with. CESNA's dual modality approach also is very robust in the presence of noise and outliers, as both modalities are used for weighing data points and constructing the overall statistical model.

1.2 Data

The NSF is structured into an organizational hierarchy which consists of three main braches: directorate, division, and program. Divisions are a subgroup of directorates, and programs are a subgroup of divisions. There are seven core directorates, however for this experiment a sample of a single directorate, Computer Information Science and Engineering (CISE), was selected from the population as is shown in Figure 2.

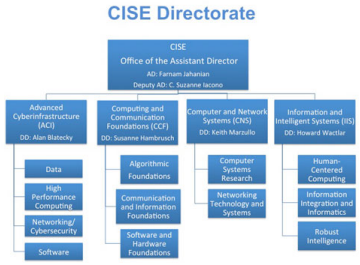

Figure 2: CISE, directorate 05, organizational chart. Directorate → Division → Program. This chart does not represent the full CISE hierarchy, only the top divisions and programs.

The data set utilized for this experiment was obtained through an NSF STEM initiative, Deep Insights Anytime, Anywhere (DIA2 - `http://dia2.org/`), which is a knowledge mining engine funded by the NSF. The website is currently in early development, however a PHP script has been made available to a few Universities so they can begin downloading and interacting with the hosted data. The data is comprised of all of the accepted grant proposals, and their associated attributes, from every directorate across the past twenty years (1995-2014). In its entirety, the full data population is approximately 1.5 terabytes of JSON files. The sample from the CISE directorate worked with in this experiment is about 57MB. Due to the nature of the sample, the main focus of analysis was on communities and ego networks at the division and program levels. This was by design, as it is our hope that the techniques used here will be applicable when performing a large scale investigation at the directorate level once several samples have been statistically measured. Our sample from directorate 05 consists of 22,633 awards, each with 8 attributes. Each JSON file contains the following fields:

- awardID :: integer string
- title :: string
- abstract :: string
- effectiveDate :: date string
- expirationDate :: date string
- PIcoPI :: [integer] or [integer string]
- PO :: [integer] or [integer string]

- fundingAgent :: [funding agent object]

The main fields of interest for this investigation are the awardID, title, abstract, PIcoPI, and fundingAgent. The PIcoPI field (PI - Principal Investigator, coPI - co-Principal Investigator) is a list of all the PI's who are associated with a given awardID. The fundingAgent field is an array of objects which indicates the directorate, division, and program the awardID is associated with.

1.3 Areas of Interest

There are both practical and theoretical applications for the problem we are seeking to address. The theoretical profits will come from a comparison of CESNA with the more traditional methods of community detection. If comparable results can be produced, the comparison could prove useful for future research. The practical results are perhaps greater in number.

1. The distinction of unique sub-communities within the greater NSF research community will illuminate exactly what fields of study are actually being funded.
2. An analysis of the funding spread across these communities overtime will clarify how NSF priorities have developed over time and may serve as an adequate foundation for predicting future funding trends; this information could be used by researchers to anticipate demand and by the NSF to identify underfunded initiatives.
3. The community membership information can be used as a foundation for a researcher recommender system which would simplify the process of finding collaborators and increase collaboration across NSF initiatives.
4. Identification of hubs and bridges in the community network will clearly highlight key players in the research world, which could be useful for a variety of purposes.
5. The use of generative models such as LDA and CESNA for network analysis might also yield the ability to infer (generate) missing data attributes, which could improve related analyses.

2 Experimental Design

We performed three distinct experiments, to varying degrees of completeness, and have identified a fourth as the next step moving forward.

1. Explore community structure using basic graph analysis methods.
2. Identify communities based on funding agents and perform a time-series analysis of funding trends over time; here we also seek to create a model for predicting future funding trends.
3. Use LDA topic modeling for community detection with the award abstracts as a document corpus.
4. Use CESNA to perform community detection, leveraging both network structure and attributes in conjunction.

The structural approach involves standard graph analysis techniques such as identification of communities through connected components discovery, detection of sub-communities via bridge removal, and measuring popularity/importance through centrality and degree comparisons. The two attribute-based approaches deal with the funding agents and the abstracts of the data. A time-series analysis of the funding data can be used to detect funding trends over time, predict future funding hot-spots, and identify communities as defined by the sources of funding received for research. The abstracts for the grants can be used as a document corpus for the purpose of community detection via LDA topic modeling.

While the first three stages are composed of traditional methods, the fourth seeks to utilize the cutting edge community detection algorithm CESNA. CESNA incorporates both network structure and network attributes together in order to detect communities in social networks. The authors of CESNA claim its methods of community detection are superior to traditional methods, such as those identified for stage one of this project. After performing both methods and accumulating results, we hope to produce results which can be compared to confirm or refute this claim. At this time, we are involved primarily in the first three phases of the project, though some preliminary progress has been made in preparing the data for a CESNA analysis.

In order to manipulate the data more naturally, a Python parsing tool was written to lazy load each file and return award objects as native Python data types. Python was also used for each of the three traditional analysis approaches.

2.1 Explore Community Structure: Basic Graph Theory Approaches

The first experiment was designed as a means to perform initial data exploration and basic community detection.

This experiment was performed using the igraph package for Python and Gephi, which is a popular graph visualization and analysis tool written in Java. In order to construct the graph, each PI was treated as a vertex and each shared award was treated as an edge. So for each award, each PI in the list was added as a vertex, then each combination of PIs was added as an edge between the vertices for those PIs to represent a collaboration between them. Since most attributes from the data were tied to awards, the attributes were stored in the edges of the graph. The only attributes associated with vertices were the PI IDs. Once the graph was constructed, it was written to a GraphML file, which is a standard graph file format that Gephi can read in. At this point, the data was fully represented in a graphical format both of our chosen tools could recognize, and we were able to move forward with the analysis.

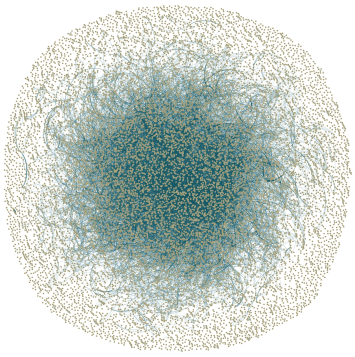

Figure 3: Network structure generated with igraph.

2.2 Funding Agent Time-series Analysis and Community Exploration

The second experiment was designed to identify communities based on funding agents and perform a time-series analysis of funding trends over time, both to understand the past and help predict future funding trends.

The funding agent data was more conducive to treatment as a table, so the Python pandas package was used

to parse the funding agents for all awards into records in a DataFrame, which is a data structure analogous to an SQL table. For each award, an entry was created for every funding agent who funded the award, since there can be more than one per award, and for every PI on the award. So for every PI, a record was created for every funding source; if an award had three funding agents and three PIs, nine records would be created. This resulted in a table with 65,386 records. Some initial exploration was performed using a table without dates to get a sense of the distribution of awards across directorates, divisions, and programs. A full time-series analysis has been left to a later time due to time constraints.

2.3 LDA Topic Modeling for Community Detection

Using LDA for community detection requires that each person in the network can be represented as a document and the network as a whole can be viewed as a corpus of documents. In order to facilitate this representation, the abstracts of the awards were treated as the corpus of documents and a representative document was created for each PI based off the awards they have worked on. The Python Natural Language Toolkit (NLTK) and the topic modeling package called gensim were used for this purpose.

For each PI, a list of the awards they have worked on was obtained, then a list of the abstracts for those awards was retrieved and combined to form the representative document. For each document, word tokenization was performed and then the following preprocessing methods were applied.

1. Puncuation removed
2. Words lowercased
3. Whitespace stripped

Then the list of words was filtered. Any word meeting one of the following criteria was removed.

1. Empty or only one character
2. Stop word
3. All digits
4. Starts with a digit

Finally, the Porter Stemming algorithm was applied to reduce each word to its most common stem. This was done in order to facilitate a bag-of-words (BoW) representation of the documents, in which each document is represented by a vector of its term frequencies. With all documents represented as a BoW, an LDA model was built from the corpus of abstracts with the number of topics set.

There was some deliberation about the number of topics to use for the model. Using the number of divisions would not result in granular enough topics, while using the number of programs would be far too granular. As a compromise, the average number of programs per division was used; this number came out to be exactly 96, so 96 topics were used. While this is by no means a perfect estimation of the number of communities in the network, it is a reasonable starting point.

3 Conclusions and Future Work

At this point, most of our experimental efforts have been devoted to data exploration and discovery. As a result, all three of our experiments together constitute a breadth-first approach to exploring our data from multiple perspectives in limited time. The hope is to gain more insight in this manner than from a depth-first approach to any single of the three methodologies. With this understanding, our primary achievements have been increased understanding of the data and the tools involved for analyzing it, and the creation

of a powerful API for parsing information from the raw data and asking questions about it. These results lay the foundation for meaningful analysis of the complete NSF dataset and therefore constitute essential components of our efforts moving forwards.

3.1 Explore Community Structure: Basic Graph Theory Approaches

The graph we parsed from the JSON files contained 14,979 vertices, each representing a unique PI, and 28,474 undirected edges, each representing a unique collaboration between PIs. An initial exploration of connected components showed 3,923 connected components. This number included 2,601 loners (PIs with no collaborations/vertices with no edges). Once the loners were filtered out, the number of connected components remaining was 1,322 as shown in Figure 5. 17.364% of the PIs were loners and of the 3,923 connected components, 66.301% of them were composed only of loners. So roughly 1/3 of those originally detected could be considered communities. The igraph package did not have any built in methods for bridge detection or removal, so we have left that analysis for a later time.

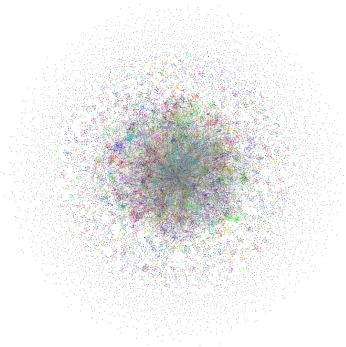

Figure 4: Clustering Analysis of the NSF CISE directorate via Markov Clustering.

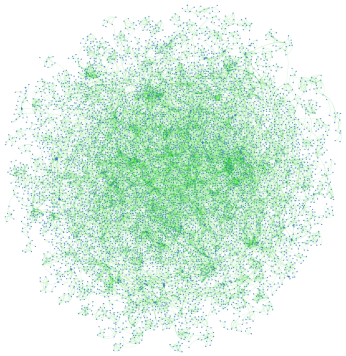

Figure 5: Largest Connected Component in the data set.

Some measures were also obtained from the graph as a whole to get a sense for the overall characteristics of the network. The modularity of the full graph is 0.918, with an average clustering coefficient of 0.742. The average path length is 8.345, and the average degree of all vertices in the graph is 3.191. These measures indicate the research network has a high clustering tendency and communities tend to be tightly knit.

3.2 Funding Agent Time-series Analysis and Community Exploration

The primary results of this analysis at this time include an API for querying directorates, divisions, and programs for a particular PI or group of PIs and some basic information about the number of funding entities. The API is built in Python pandas and includes a variety of querying capabilities specific to our data, as well as underlying query functionality with similar breadth and power to that of SQL. It allows for selection of funding entities based on a PI and filterable by single or multiple directorates, divisions, and programs. It also allows for selection based on shared funding entities across a list of PIs.

Some preliminary information was also obtained about the dataset. There were 12 directorates present in the data, with the most prominent being the CSE, ENG, EHR, MPS, BIO, GEO, O/D, and SBE. All remaining analyses were done with only CSE (directorate 05). Within the CSE directorate, there are eight divisions, with the most prominent by far being CNS, IIS, and CCF. Across these eight divisions, there are 768 programs. Of these, 150 are CNS programs, 304 and IIS programs, and 244 are CCF programs. Together these programs account for roughly 91% of all awards funded.

3.3 LDA Topic Modeling for Community Detection

The primary result of this experiment was an API which allows a user to input a PI ID and receive back the LDA topic distribution for that PI. This can be used to investigate community membership by relating topics back to communities. So for a list of PIs, if their top X most likely topics are the same, then they can be considered to part of the same community. This is of course a somewhat crude measurement, and more accurate and meaningful relationships are being explored at this time.

3.4 CESNA

At this point in time we have not yet utilized CESNA for analysis on our sample set. We have performed preliminary setup of the tool itself and analyzed several provided data sets. Unfortunately, CESNA is very new tool, and as such has very little documentation provided. We are in the process of processing our data into a format readable by CESNA, and evaluating the results from the samples in order to determine viable parameter settings for

4 Appendices

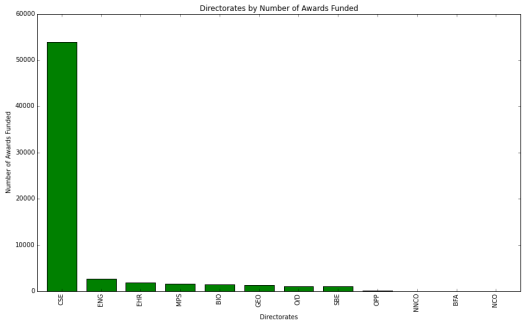

Figure 6: Directorates by Number of Awards Funded

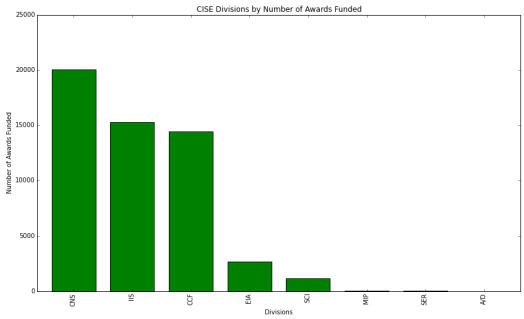

Figure 7: CISE Divisions by Number of Awards Funded

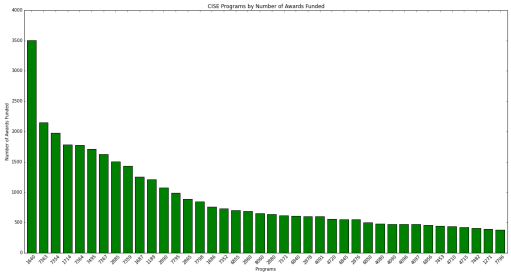

Figure 8: CISE Programs by Number of Awards Funded

References

[1] G. Cheliotis. *Social Network Analysis (SNA)*. Communications and New Media, National University of Singapore, ?

[2] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, 2010.

[3] S. Fortunato. Community detection in graphs. *Physics Reports 486, 75-174*, 2010.

[4] R. Hanneman and M. Riddle. *Introduction to Social Networks*. Riverside, CA: University of California, Riverside, 2005.

[5] Hoffman, Blei, and Bach. Online learning for latent dirichlet allocation. *NIPS*, 2010.

[6] J. Yang, J. McAuley, and J. Leskovec. Community detection in networks with node attributes. *IEEE International Conference On Data Mining (ICDM)*, 2013.