University of Rhode Island

Department of Mathematics

Initial Applications of Data Science to the Study of Genizah Documents

Author:
Daniel Gladstone

Adviser:
Dr. Nancy Eaton

June 2014

Contents

Abstract

The Jewish people of medieval Fustat amassed an unprecedented collection of documents depicting the vast social and economic structures of their society. This aggregate dataset is known as the Genizah Collection and has subsequently been distributed among academic institutions around the world. This paper will argue that this collection is early example of what is known today as Big Data. This classification allows for the contemporary advents of data science to provide a fresh perspective on millennia old information. Specifically the paper will use the tools of graph theory and computer driven algorithms to examine networks contained in the Genizah Documents. This paper demonstrates the viability of using computer generated graph drawings to further our understanding The Genizah Documents. The results presented are likely the first visualizations of this millennia old network.

I. Introduction

I. Intended Audience

This paper is intended for readers who are interested in the application of data science to the study of history. Readers with a background in history or data science may find utility here. Accordingly, the paper has been divided into subsections; each section may be read independently and may be further developed in the future.

II. The Genizah Documents

In this paper the term Genizah Documents will refer to an assorted archive of approximately 450,000 documents that were originally collected by the Jewish communi-

ties in the city of Fustat (present day Cairo). Such collections are actually a byproduct of a deep and commonly held reverence for the godâĂŹs name. When a document that contains GodâĂŹs name or a scriptural passage is no longer needed, it is not treated with the same disregard as ordinary writings; such a document is set aside to await natural disintegration inside a genizah. In Fustat, the arid climate combined with the cityâĂŹs economic centrality has culminated in an unprecedented historic archive of documents. These documents have provided tremendous insight into medieval culture and economics.

These documents have provided tremendous insight into medieval culture and economics. It is important to understand that these documents were not kept with a regard for future historians, but rather are a consequence of religious practices. This makes the history contained inside them markedly different than conventional historic archives. When the genizah documents were written they served mundane and utilitarian purposes for common people; they are mostly legal documents, marriage contracts, correspondences, and miscellaneous writings. This disregard for future examination provides an honest account of everyday life and is alike to present day Big Data methodology. The logistics of such a sizable collection forces historians to examine a sample of documents that is relatively small when compared to the entire collection. For example the study of three thousand genizah documents would cover just 1% of the collection. Accordingly modern historians have not been able come to an agreement on a cohesive depiction.

II.1 Big Data Characteristic: Size

Before a data scientist can open her toolkit she must first diagnose the relative size of the available data set; she must be able to access all the available information for it to considered Big Data. Already the data scientistâĂŹs approach is markedly different then earlier statisticians or historians. Her predecessors had been concerned with optimizing their sample size and area of focus (Viktor20). Such notions are evident in the existing historic research conducted with the genizah documents; historians have only been able to infer causality using a sampling of documents. With the very recent advent of digitized document translations, the Genizah collection is now becoming a data set suitable for the data scientist. The digitized portion of the collection is approaching the size requirements to be considered Big Data. It is this authorâĂŹs understanding that this study marks the first application of data science to the Genizah documents.

II.2 Big Data Characteristic: Content

The big data characteristics are further exemplified by the scope and content of the documents. These documents were written for a variety of applications and thus encompass the entire scope of medieval life. The document types range from merchant correspondences, marriage contracts, deeds, biblical commentaries, and a variety of miscellaneous writings. Such diversity allows for the tools of data science to use the size of the data set to discover new commonalities. The processing of all the data, even if it seems unrelated, is a tenant of big data.

II.3 Big Data Characteristic: Reuse

As data has become commoditized it has also become a reusable or non-rivalrous good (SOURCE Vikctor 100). Data scientists have come to recognize that data often has many uses beyond its original intended application. This realization has accrued in harmony with the plummeting cost of data storage. It has become common practice to store up massive amounts of seemingly unimportant information with hopes of some vague future application. The resulting petabytes of digital data is akin to the medieval vaults that housed these Genizah Documents.

III. Graphing

IV. General Theory

A graph $G = (V, E)$ can be use to model a given social network. G is constructed from set vertices with each vertex V representing an individual (vertex and node are used to mean the same this paper). An edge E, between two vertices represents a document. The edge begins at the author of document and ends at the person mentioned in it. These edges form directed graphs. G can also be constructed by generating a graph using languages and keywords as vertices. That is a document written in language (V1) forms an edge with a keyword (V2) when that keyword is used in the document. These are just two of countless methods for determining edges and node definitions. The goal of any method is to visualize how individual actions create networks of ideas, emotions, and expressions.

V. Applications in Social Networks

Social networks have existed for as long as information has been exchanged between individuals. The fundamental concept behind the use of graphs to model social networks is that seemingly autonomous individuals are embedded in a web of connected interactions (SOURCE Handbook pg.808). Analysis of social networks by means of interdisciplinary study has a long history in applied and theoretical setting. During the past century, the rapid development of communication technology has created new and massive networks. Such global networks remained intangible when records of communication were restricted to less measurable methods (letter writing). With the advent of computer graphing programs any such network can be modeled by a graph G = V, E wherein individuals are represented by the set of vertices V and their communications by the set of edges E. The digitized Genizah Documents provide an opportunity to use these modern techniques on millennia old data. This simple property has enormous implications, because it quantifies human behavior so that it can be studied with the tools of mathematics. The research conducted in this paper has utilized that property in combination with computer automation to generate graphs. These graphs of social networks are formally known as Sociograms (SOURCE Handbook). Though communication is the dominant property for edge drawing it is not the only one.

Graph drawing can be used to model networks beyond simple correspondences. In this study the use of language and keywords proved to be an effective method for developing an understanding of the Genizah Documents. Selecting what data will become a vertices and what an edges is the essential choice that is made in the process of graph drawing. This is especially true when using messy or incomplete data this choice can mean the difference between trivial and meaningful visualizations. Those who are interested in the application of graph drawing to historical information need to be particularly considerate of the ways in which innovative graphs can vastly increase their value.

VI. Graph Drawing Algorithms

In the computer automated drawing methods used in this paper there exist two stages in which algorithms determine the layout of a graph. To understand the difference between these two algorithms one must recognize the difference between constructing and drawing a graph. When a graph is constructed nodes are generated and edges are determined. In this study those nodes and edges were generated from the data contained in the Genizah Documents. Once the graph is constructed we can then use that information to draw a visualization of that graph. The flow chart in figure 1 demonstrates the ordering of this process. In this study the Neworkx library in the python programing environment was used to generate the graphs and the Gephi graphing program was used to draw those graphs.

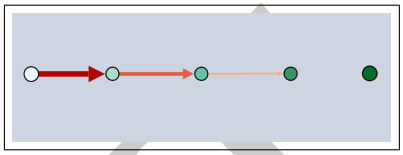
It is at the initial step of generating a graph

when algorithms can be applied. Such an algorithm will assign attributes to the nodes or edges. Generally these attributes have numerical values. One of the most basic examples of these attributes is number of edges that a node is connecting to (a feature of all connected graphs). When a graph is generated an algorithm assigns the attribute to each connected node. Attributes can be assigned nodes, edges, groups of edges, and groups of nodes. Attributes provide a method for highlighting valuable information.

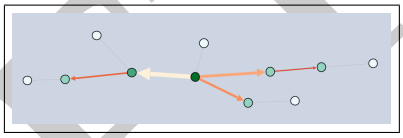
Once the graph is generated and attributes are assigned the process of graph drawing can begin. During this process another algorithm will examine the attributes of the nodes edges and determine the optimal layout for the graph. Such algorithms use vector principles in conjunction with unique energy model to set the layout of the graph (Forceatlas2, a graph layout algorithm for handy network visualization). These nodes are governed by rules in the same that real objects are controlled with the laws of physics; it is angelus to the dispersant of objects by attributes like weight, spring tension, and heat in the real world. An essential feature in this virtual graphing environment is that the plane can also be assigned attributes (Handbook). The fields in which these nodes and edges exist are often given attractive or repulsive forces. This provides further structure for the nodes and edges.

VII. Path Length

Graph theory is often concerned with shortest path between two vertices; the path which requires passing the least number of vertices. In this paper the weight of the edges is not considered in the shortest path, though it is common for edge weight to equal distance in navigation problems; weighted edges are used when a program such as Google Maps determines the fast route home. As figure 1 shows, this study used the library NetworkX in the Python programming environment to find shortest paths. This library uses Dijkstra's algorithm to find the shortest path. This algorithm was originally published in 1959 and is frequently used as an efficient method for finding shortest paths (Dijkstra, E. W. (1959). "A note on two problems in connexion with graphs". Numerische Mathematik 1: 269âĂŞ271.).



(a) *Small Directed Connected Graph*



(b) *Large Directed Connected Graph*

Figure 1: *Visualization of Path Length Values*

In data such as the Genizah Documents, communication often spans multiple generations. It seems important to have a method for highlighting the series of correspondences that span the most generations. That is to say it is important to have a method for demonstrating where knowledge had been passed down. To understand this concept we must think of shortest path as being the same as a purely efficient path. Accordingly we are looking for the longest purely efficient path. An algorithm for finding such paths was developed and applied to this data set.

VIII. Bipartite Graphs

A bipartite graph is a two-mode graph in which there exist two independent groups of vertices. Such a graph has group of vertices U and V connected by a set of edges E such that $G = (U, V, E)$. Though these graph types are not common in most Sociograms, there does exist data sets wherein nontrivial social information can be determined from a graph that is bipartite.

II. Methods

The methodology and procedure used in this study follows the transformations outlined in figure 1. The chart is designed to exhibit how the format of the documents has been transformed as they have sifted through the system. The procedures conducted in this study begin at the server storage point and end the point of visualization.

I. Choice in Digital Library

As seen in figure1 the visualizations in this study are a product of only one of the three major online databases. The process of moving data from the Internet into a table structure that can be graphed has a very high upfront cost; a process known as web scraping. Due to the unknown nature of this data it did not make sense to use all the data Internet sources before the viability of the visualization could be determined. Before the results were graphed there was no guarantee that such graphs could be obtained; it was assumed, but not proven that such a network existed. Accordingly the library with the most accessible data was chosen; The Taylor-Schechter Cairo Genizah Collection at Cambridge University Library. As Table 1 demonstrates this library represents the smallest portion of available online data. Furthermore, the data available accounts for only a portion of all Genizah documents.

Cambridge University Library	1200
Bodleian Libraries	4000
Friedberg Manuscript Society	448288

The primary reason for choosing the Cambridge collection over the other available library was the structure of its web content. This library gives each document its own webpage where the important features of the document are clearly displayed. As explained the SECTION different website features can have a huge effect on the ease of data collection.

II. Data Collection

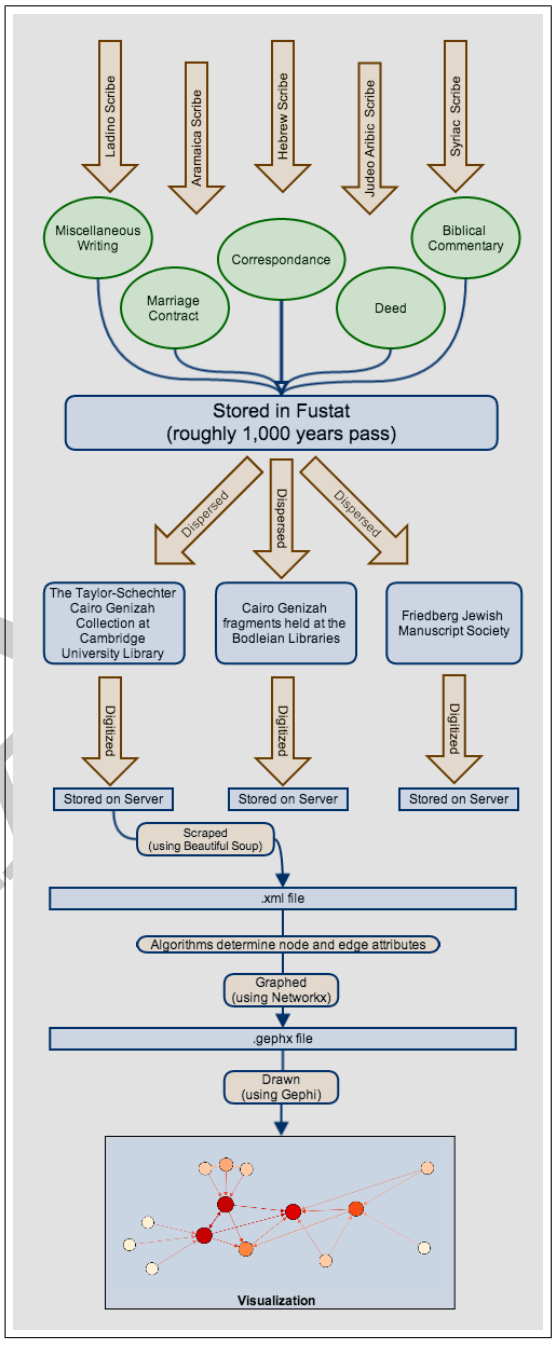


Figure 2: *Transformation of Data*

The process wherein a computer program imports data from multiple Internet sources is commonly referenced to as web scraping.

Most websites have a unique html structure and thus there is no universal method for scraping Genizah documents. The differences in the layout websites pose new programming challenges. Of all the Internet Genizah collections, the Cambridge Library seemed to have the most straightforward and structured layout. In particular the authors, mentions, keywords, and languages of each document were already delineated.
The Beautiful Soup library in the Python programming environment was chosen to scrape this information. This library includes many essential tools for interpreting html tags. The process of collecting this information was done in two steps. The first step was collecting all the roughly 4,500 web addresses that corresponded to the individual documents. This was achieved by manipulation of the websiteâĂŹs search feature. Once all the addresses were stored locally another program visited each page and deciphered the pertinent information. In order to not overload the Cambridge Servers a 2 second pause time was introduced resulting in a roughly 6 hour total run time. Table 2 shows the various types of information that was taken for the web pages.

III. Repeating Names

The significant recurrence of names such as Israel, Isaac, and Abraham point to a potential source of error in these graphs; two or more people with the same name. Establishing the identity of people who share a name is beyond the current scope of this project. Figure 2 shows that there exists two stages where these identities can be established; during initial translation/digitization or through computer aided analysis. The latter method could be done using Python in conjunction with input from historians in this field of study.

IV. Complete Communication Network Graph

The Complete Communication Network Graph in figure 4 is the product of four algorithms run in two different programs. The first algorithm was designed just for this study and use to highlight the value of shortest paths. This proceeding algorithm was developed to best utilize the tools of Python. Many ideas about how to value path length cannot efficiently by executed by Python.
The algorithm for weighting shortest path lengths can be best understood when looking the graph in figure 2. Every time the algorithm recognized edge a to b was part of a connected path on 4 occasions. Each time that incidence was recognized it adding weight to that edge. In figure 2a and figure 2b the attribute of weight is visualized in the form of color and thickness. The visualization of those attributes was neglected in Figure 4; repetition of paths made the graph more difficult to understand. The attribute is accounted for in the layout algorithm and does make the network easier to understand.

V. Bipartite Graph with Correlation between Language and Keywords

This can be found in figure 5. The graph was drawn with each edge representing a document. On the left side of the graph are the languages each documents was written in. On the right side are the all keywords that occurred in more then 5 documents. The culminating graph demonstrates how some keywords were restricted to one language while others were not.

Algorithm for Weighting Shortest Path

- Test each and every node of graph.
- For each node find unique shortest paths. This is start node.
- For every shortest, find the *last node* in the path.
- Find the path between *last node* and *start node*.
- If there exists a node in the path such that is has an edge with *start node*.
  - Then add 1/(node weight).

Title	Name	Volume	Description	Acquisition	Destination	Languages	Authors	Mentions	rty to the De	Key Words	Notes	URL
[u'MS. Heb. c	[u'Arabic lett	[u'MS. Heb. c	[u'Syr\u200e	[u'Bought thi	[u'20a', u'20b	[u'Main langu	["Musa b. Ab	[u"Musa b. A	[]	[u"Musa b. A	[]	http://gen
[u'MS. Heb. c	[u'Business le	[u'MS. Heb. c	[u'Oriental cl	[u'Bought thi	[u'34a', u'34b	[u'Main langu	["Musa b. Ab	[u"Musa b. A	[]	[u'Letters. Ar	[u'from \u05	http://gen
[u'MS. Heb. c	[u'Letter, in A	[u'MS. Heb. c	[u'Oriental cl	[u'Bought thi	[u'6a', u'6b']	[u'The main l	["Joseph b. A	[u"Joseph b.	[]	[u"Joseph b.	[u'to \u05d0	http://gen
[u'MS. Heb. c	[u'Business le	[u'MS. Heb. c	[u'Oriental cl	[u'Bought thi	[u'40a', u'40b	[u'Main langu	["Abu Ibrahir	[u"Abu Ibrah	[]	[u"Abu Ibrah	[u'from \u05	http://gen
[u'MS. Heb. c	[u'Letter, in A	[u'MS. Heb. c	[u'Oriental cl	[u'Bought thi	[u'87a', u'87b	[u'Main langu	["Abu 'l-ridha	[u"Abu 'l-ridh	[]	[u"Abu 'l-ridh	[u'from \u05	http://gen

Figure 3: *Sample of Collected Data*

III. Results

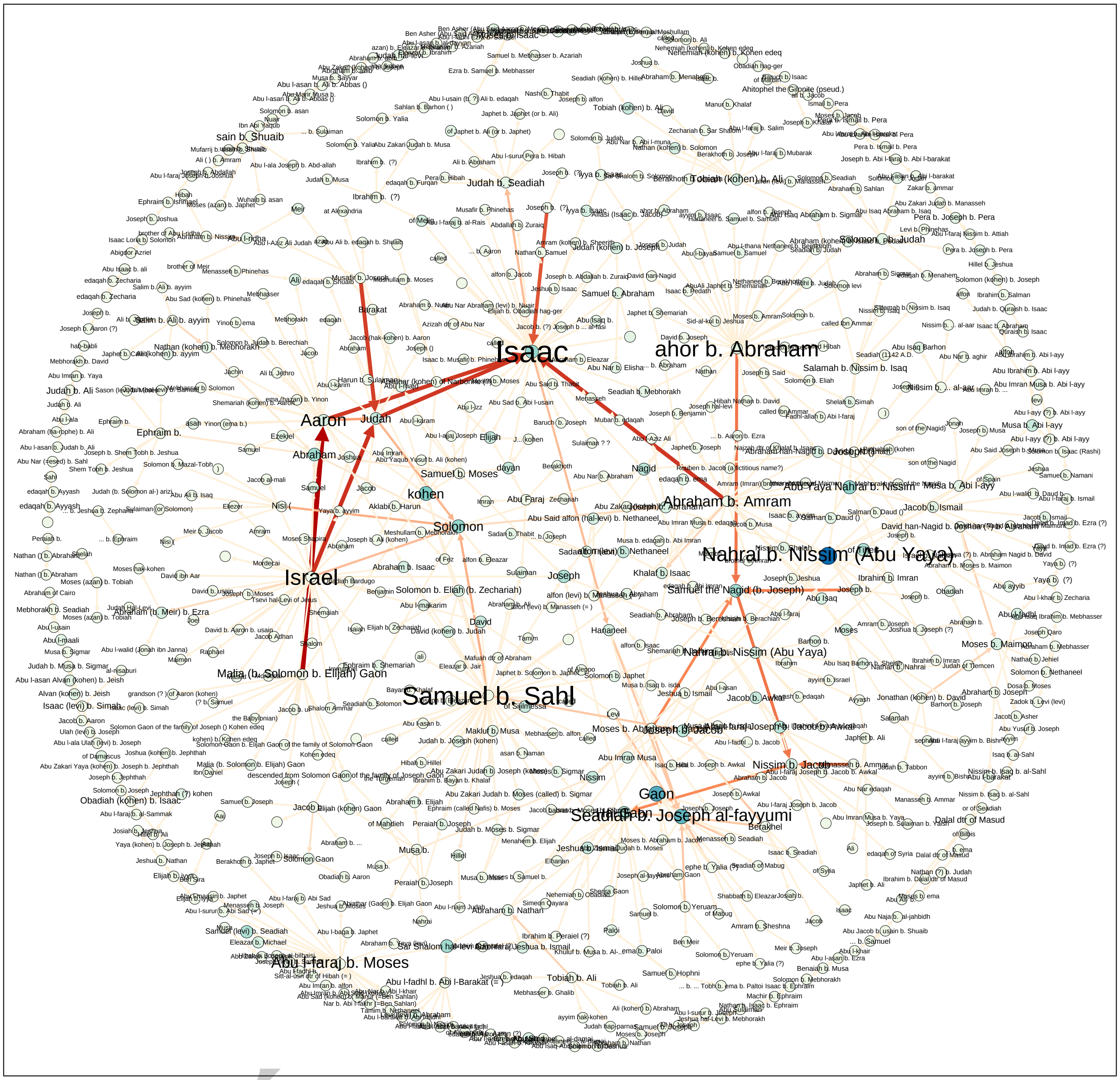


Figure 4: *Full Network Graph with Fruchterman Reingold Layout*

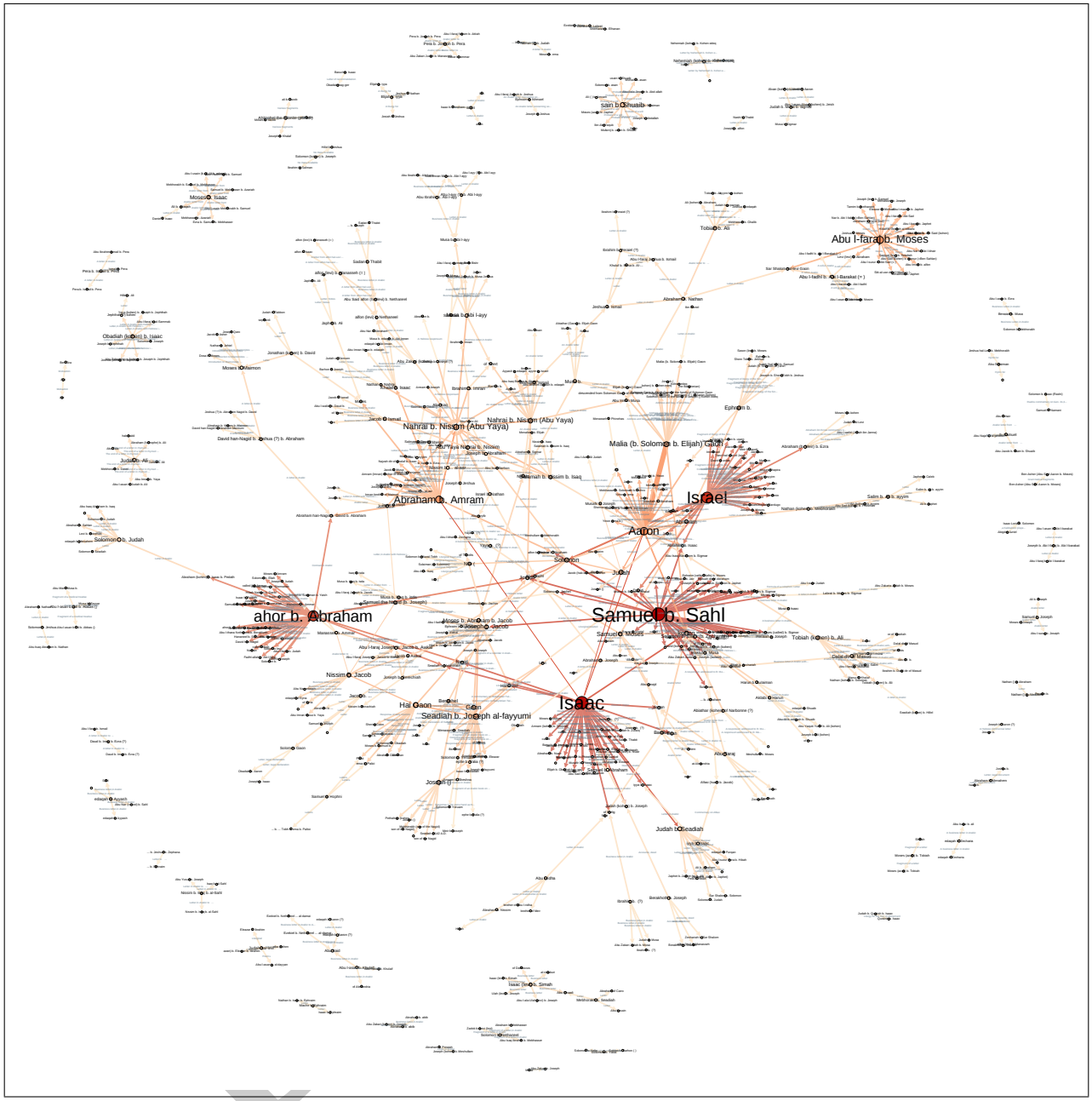


Figure 5: *Full Network graph with Tugan Hu Layout and Edge Labels*

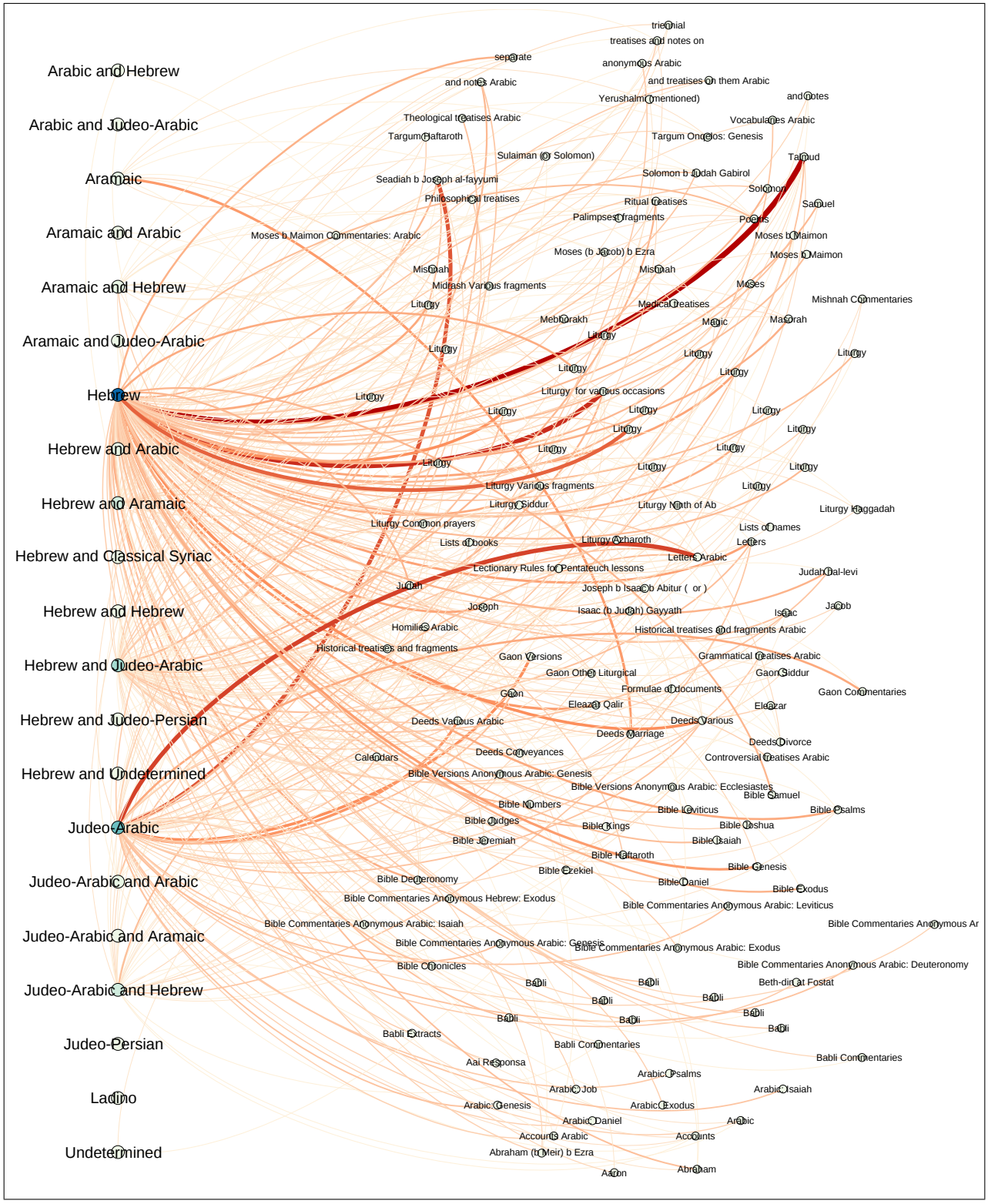


Figure 6: *Bipartite Graph with Correlation between Language and Keywords*

IV. Analysis

I. Regarding Full Network Graph

II. Regarding Bipartite Graph with Correlation between Language and Keywords

References