



Exploring Ancient Networks

Exploração de Redes Antigas

<https://doi.org/10.21814/h2d.3508>

Niek Veldhuis, UC Berkeley, USA

Como citar

Veldhuis, N. (2021). Exploração de Redes Antigas. *H2D|Revista De Humanidades Digitais*, 3(1). <https://doi.org/10.21814/h2d.3508>

ISSN: 2184-562X



Exploring Ancient Networks

Exploração de Redes Antigas

<https://doi.org/10.21814/h2d.3508>

Niek Veldhuis, UC Berkeley, United States of America

Abstract

A small archive of texts from ancient Iraq is used to demonstrate an approach to network analysis in which traditional close reading and computational text analysis go hand-in-hand. The computational methods produce tables and graphs that link back to online editions of the primary material, enabling the user to check the results.

Keywords

Network analysis; Natural Language Processing; Sumerian

Resumo

Um pequeno arquivo de textos provenientes do Iraque antigo é utilizado para demonstrar uma abordagem de análise em rede, em que a leitura atenta tradicional e a análise de texto informatizada andam de mãos dadas. Os métodos computacionais produzem tabelas e gráficos que remetem para as edições online das fontes primárias, permitindo ao utilizador verificar os resultados.

Palavras-chave

Análise de rede; Processamento de linguagem natural; Sumério

1. Introduction

In this article, I will discuss the possibilities of using interactive visualizations for exploring social networks, based on ancient Sumerian texts from the period between ca. 2.100 – 2.000 BCE (the so-called Ur III period). Building such networks allows students and researchers to quickly get an idea of the important actors in the archive (or text group) that is being studied and to move from studying a single text in detail, to a bird's-eye view of the entire corpus¹.

Humanities research and teaching has traditionally put a lot of weight on carefully analyzing (or close reading) original texts and objects. Computational analysis (or distant reading, in Franco Moretti’s words) has the potential of displacing that emphasis and replacing it with conclusions that derive from analyses of large corpora that provide insights in broader patterns of an ancient culture. There are many reasons, however, to doubt such a scenario. First, ancient data sets are rarely ‘large’ in the current sense of that adjective. Even if we muster all 100,000 Ur III documents (possibly the largest corpus in the ancient world), this still is no match for the magnitude of data used to produce GLOVE or <https://fasttext.cc/FastText> and similar models. Second, digital representations of ancient texts are likely to contain errors. Errors are, of course, everywhere and in all data sets, but in smaller data sets, such errors may gain more weight and the impossibility to track unlikely outcomes back to the source data is undesirable. Most importantly, however, computational approaches should not displace the current research practice, but rather support and extend it by providing tools and concepts that can be understood from within traditional Humanities research.

Rather than distant reading, what I will argue for is an approach that enables going back and forth between bird’s-eye views in a visualization and access to the actual data (warts and all included) on which the visualization is based. Avoiding neural networks (which largely scramble the connection between data and result), I will use old-fashioned rule-based techniques to select nodes, assign roles, and create (directed) edges. The processes of assigning roles and creating edges are themselves accompanied by interactive displays that allow the user to check the validity of the results.

Figure 1 is an example of a visualization of the Treasure Archive from Puzriš-Dagan (Paoletti, 2012). The names of the most important actors (those with the highest degree) are labeled. The coloring of the node represents its eigenvector centrality.

The article will discuss how this visualization was built and how further analyses and visualizations may be employed to explore the network.

2. Building the Graph: Nodes, Roles, and Edges

The graph is built by first acquiring the text corpus, filtering for proper nouns (the nodes), assigning a role to each node, and defining (directed) edges between the nodes. The data is freely available through the Open Richly Annotated Cuneiform Corpus (ORACC). The script that does the work is available as a Jupyter notebook in Chapter 4.1 of the Computational Assyriology (Compass) project².

Before going into more details of this process, we will first briefly discuss the background of the Treasure Archive.

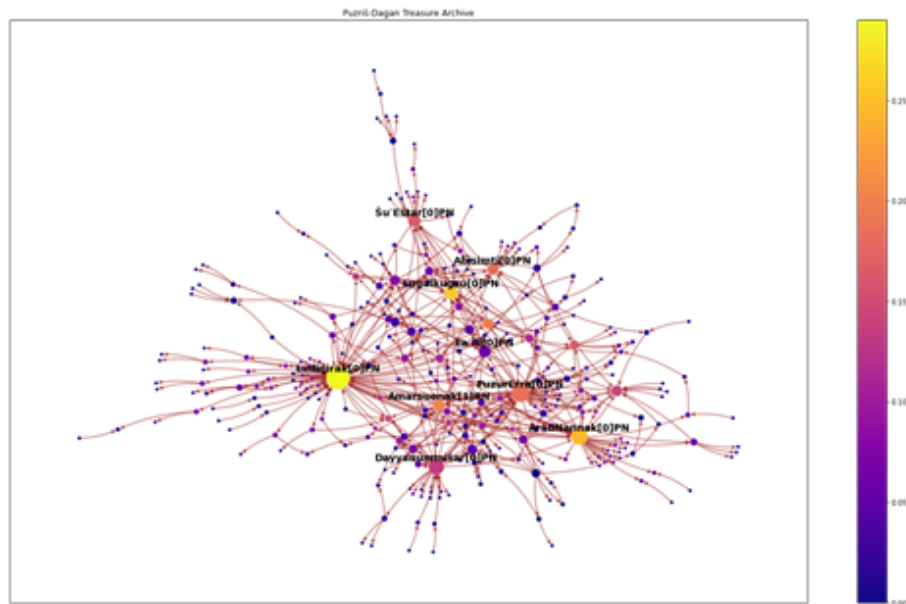


Figure 1: Treasure Archive Network

2.1. The Treasure Archive

The Treasure Archive is a relatively small group (< 300 documents) of administrative texts that date to the 21st century BCE, dealing with valuable objects (made of metals and valuable stones), leather products (shoes, boots, etc.) and weapons. The archive was studied in detail by Paoletti (2012), a publication to which we will come back for interpreting the visualizations.

The Ur III period (ca. 2100-2000 BCE) was only the second time in Babylonian history (present-day southern Iraq) when all the former city states were united under a single king. According to native historiography, this was the third time that kingship over Sumer and Akkad resided in the city of Ur – hence the modern moniker for the period: “Ur III.” The period is known for its extraordinary number of texts (over 100,000), primarily administrative documents that, in large measure, come from a time span of only fifty years. One of the important find spots is the site of Drehem, ancient Puzriš-Dagan, which housed the royal administration of domestic animals. Some 15,000 administrative texts have been assigned to Puzriš-Dagan, distinguished by their use of a specific calendar, by prosopographical connections, and by other aspects of text, spelling, writing conventions, subject matter, etc. The great majority of these texts (more than 99%) were looted from the site and reached museums and private collections all over the world through the antiquities market. All these texts are written in Sumerian in cuneiform writing on clay tablets; most of them have a date that may include day, month, and year (but not all date elements are present in

each document). Many tablets are small and may have no more than 6 to 10 lines; bigger, multi-column tablets, may represent monthly or yearly summaries. Damage to clay tablets is very common (in particular for larger ones) and presents challenges for their interpretation.

The Treasure Archive is a small subset of the finds from Puzriš-Dagan. Thanks to Paoletti's very thorough study of all aspects of this archive, the results of our efforts may easily be compared to the results of more traditional methodologies.

A somewhat random example of a text from this archive (AUCT 1, 502) is the following:

1. 10 ma-na kug-babbar -> 10 mina of silver (5kg)
 2. niḫ-sa10-ma kug-sig17 10-ta-še3 -> purchase price for gold at a 10:1 ratio
 3. i3-lal3-lum -> Ilalum
 4. mu-kux(DU) -> delivered.
 5. puzur4-er3-ra -> Puzur-Erra
 6. šu ba-ti -> received.
 7. šag4 puzur4-iš-^dda-gan -> In Puzriš-Dagan.
 8. iti še-sag11-kud -> Month 12.
 9. mu en ^dNanna ba-huḫ -> Year: the en priest of Nanna was installed.
- left edge: 10 ma-na -> 10 mina



Figure 2: Obverse of AUCT 1 502

The text documents that Ilalum brought in 10 mina (or about 5 kg) of silver for buying 1 mina of gold and that this silver was received by Puzur-Erra. In the network graph this should result in two nodes (Ilalum and Puzur-Erra) with a directed edge Ilalum -> Puzur-Erra. Note that the year name (line 9) contains

the name of the god Nanna, but this god is not part of the transaction. Nanna, therefore, should not become part of the network.

2.2. Acquiring Data

The documents that belong to the Treasure Archive are all available in three parallel databases: The Database of Neo-Sumerian Texts (<http://bdtns.filol.csic.es/>), the Cuneiform Digital Library Initiative (<http://cdli.ucla.edu>), and the electronic Pennsylvania Sumerian Dictionary (ePSD2) in the Open Richly Annotated Cuneiform Corpus (<http://oracc.org>) project. Each of these repositories provides open access to the data. For our purposes we will use the ePSD2 data, first, because in ePSD2 the texts are lemmatized and second, because the data can be acquired in JSON format, which is easier to process than the raw text data provided by BDTNS and CDLI. The ORACC JSON is quite complex in structure (see <http://oracc.museum.upenn.edu/doc/opendata/json/Oracchttp://oracc.museum.upenn.edu/doc/opendata/json/JSONData>). Since the JSON structure is identical for all ORACC data (which covers all of cuneiform), the complexity of the JSON is a minor issue. A standard parser, developed by the present writer, can be used (and, where necessary, adapted) to represent the text in the desired units (signs, words, lines, sentences, or texts). The parser transforms the data into a proper Pandas data frame, selecting the relevant data elements from the JSON.

Each word in the ORACC JSON data representation has a part-of-speech (POS) field. The POS field currently does not follow any particular international standard but is customized for the various cuneiform languages represented in ORACC. Proper Nouns have their own POS abbreviations, such as PN (Personal Name), RN (Royal Name), DN (Deity Name), etc. This data representation makes it straightforward to filter for the human beings and gods who participated in the transactions documented in the Treasure Archive. Year names are marked explicitly in the JSON and may thus be excluded from consideration. The lemmatization, finally, makes it possible to deal with the key words that indicate roles (such as recipient or intermediary) in documents of this time.

2.3. Name Role Activity Document

Each name instance in the documents is considered an NRAD instance: a Name in a Role in an Activity in a Document. The NRAD model was developed by Patrick Schmitz and Laurie Pearce for the Berkeley Prosopography Services. The NRAD model draws attention to different aspects of a name instance. A name instance appears in a Document, which has metadata, such as a document ID number, a date and/or a provenance. An Activity (such as Selling, Receiving, or Issuing) implies a particular set of roles that are necessary or possible. The Activity may have its own set of attributes, for instance the goods that are being received, or issued, or the date of the transaction. Finally, the attributes of the Name may include spelling and normalization (if different spellings of the same

name are known) as well as attributes directly expressed in the text (profession, or familial relationships).

The NRAD model was developed as an abstract way of representing a name instance and collecting the information needed for disambiguation. In the Berkeley Prosopography Services model, the attributes of Name, Role, Activity, and Document together provide the data for a probabilistic model of disambiguation in which one may assert that a person called Ur-Enlil (Name) who receives dead sheep (Role/Activity) in 2035 BCE (Document) is likely the same person called Ur-Enlil who receives dead lambs in 2033 BCE.

In this article I will not go into the complex issues of disambiguation. The set of documents that we will use as an example comes from a small office that was active during a restricted period of time, and the issue of namesakes is of minor importance here. The NRAD model also helps, however, in developing a model for creating edges for a social network. We may distinguish between different types of Documents that represent different Activities. Within each Activity we expect a limited number of Roles, and edges are drawn between those Roles according to specified patterns, for instance intermediary -> recipient (in the Delivery activity) but not recipient -> intermediary.

2.4. Name, Activity, and Role

The Treasure Archive has three types of documents that are marked by key words towards the end of the text: Income, Expenditure, and Transfer. The essential difference between Expenditure and Transfer is that in the first the goods leave the Puzriš-Dagan organization, whereas in the latter the goods are transferred from one official (or office) to another. The Roles in a document are explicitly indicated by key words that appear either before or after the name. In the example above we saw mu-kux(DU) (marking the deliverer) and šu ba-ti (a two-word key phrase that marks the recipient). There are a good number of such key words, marking not only deliverer and recipient, but also intermediaries of various kinds, producers (of valuable goods), senders, and offerers. A Name may be followed immediately by a qualifier: a profession, such as “general,” or a familial relationship (PN1 son of PN2). In such cases the key word will follow the qualifiers. Finally, there are participants who are not marked explicitly for a particular role. In Income texts those are deliverers, whereas in Expenditure texts those are recipients.

Putting all these rules together, the script will scan the documents to assign appropriate roles. This process results in a Pandas data frame in which all name instances are listed, with the assigned role and attributes (the qualifiers) and with a link that takes the user directly to the ePSD2 edition of the text, with the name instance highlighted. Widgets allow a user to see more or fewer rows, or to filter for a particular text or for a particular role. This is important, because the set of rules that is applied is complex and prone to errors. Different archives within the Ur III corpus will need slightly different rules, and the interactive

presentation of the outcome of the process allows a user to find errors, adjust the code and reevaluate the result.

	id	word	form	pos	lemma	role	attribute
8	P512773.4.4	(d)en-ili	DN	Enili(1)DN	recipient		Ninili(1)DN
9	P512773.4.5	(d)nin-ili-la	DN	Ninili(1)DN	recipient		
12	P512773.6.2	wa-ta-ru-um	PN	Watarum(0)PN	conveyer		sagga(official)N
15	P512773.8.2	puzur-er-ra-ta	PN	PuzurErra(0)PN	source		
36	P103347.5.1	li-lab-lum	PN	Ilalum(0)PN	deliverer		
38	P103347.8.1	puzur-er-ra	PN	PuzurErra(0)PN	recipient		
58	P102169.4.2	geme-(d)en-ili-la	PN	GemeEnili(0)PN	reason		
60	P102169.5.2	a-mur-dingir-ta	PN	Amuril(0)PN	source		
61	P102169.6.1	lugal-ku-zu	PN	LugalKuz(0)PN	recipient		
86	P103367.5.2	lu-dingir-ra-ta	PN	Ludigir(0)PN	source		
110	P135067.4.1	al-la-da-ru-um	PN	AllaSarum(0)PN	recipient	emeda(nursemaid)N Simatistar(0)PN dumumunus(daughter)N lugal(king)N	
112	P135067.4.3	me-(d)istar	PN	Simatistar(0)PN	relation		dumumunus(daughter)N lugal(king)N
114	P135067.4.5	lugal	N	lugal(king)N	relation		
117	P135067.6.2	a-bi-ili	PN	Apil(0)PN	conveyer	ragaba(rider)N Simatistar(0)PN dumumunus(daughter)N lugal(king)N	
119	P135067.6.4	me-(d)istar	PN	Simatistar(0)PN	relation		dumumunus(daughter)N lugal(king)N
121	P135067.6.6	lugal	N	lugal(king)N	relation		
123	P135067.8.2	a-hu-ni-ta	PN	Ahun(0)PN	source		
149	P110441.7.2	nu-ur-(d)utu	PN	NurSamas(0)PN	conveyer		
151	P110441.8.2	ta-hi-is-a-tal-ta	PN	Tahisata(0)PN	source		

Figure 3: Roles

2.5. Edges

Edges and directionality are assigned by looking forward in the text. In an Expenditure text, when the script runs into a Deliverer, it will look for a Recipient or an Intermediary – whichever comes first. A set of rules, comparable to the rules that assign roles, is thus responsible for creating edges. As before, the edges are displayed in a Pandas data frame with links to the editions of the texts in which the edges appear, allowing the user to check the validity of the outcome and, where necessary, to adjust the rules.

2.6. Building the Graph

Finally, the graph is built in the NetworkX package. An initial visualization of the graph is used, once again, for checking validity. A set of widgets is employed to create an interactive visualization that allows the user to select a single text and see the results for that one document, or to choose a single node and see its Ego network. The example shows the directed Ego network of Abumbašti. The button “Open Edition in ORACC” creates a link to the seven documents that attest this Ego network.

Once the user is satisfied that the nodes and edges in the network accurately represent the corpus, the graph of the Treasure Archive has been created. It is important to keep in mind that the graph is a mathematical object that consists of nodes and edges and that does not coincide with any of its visualizations. For Humanities scholars, a common step at this point in the process is to import the

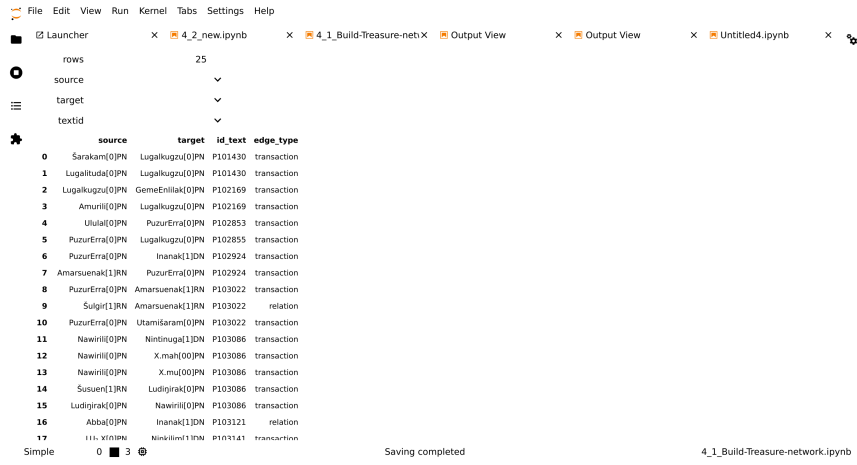


Figure 4: Edges

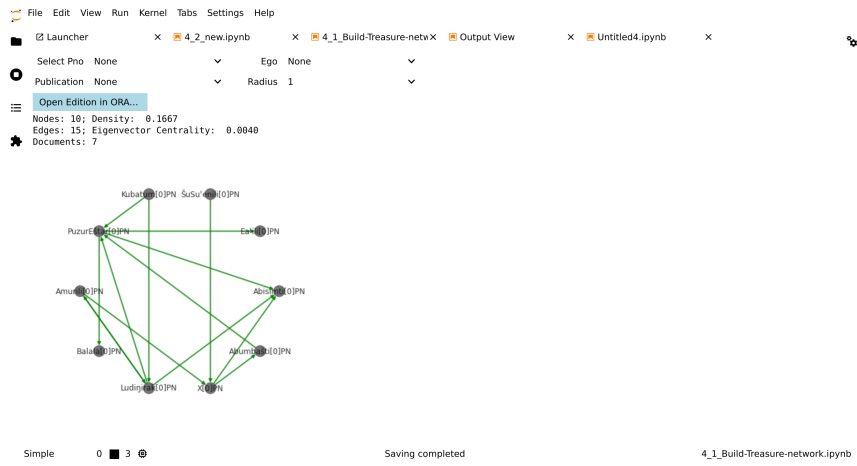


Figure 5: Ego

data in Gephi, a powerful open-source visualization program for exploring and manipulating networks. Gephi has been around for a long time, has a large user base, and creates attractive and flexible visualizations by means of a graphical user interface. Although Gephi has many strong sides and has facilities for computing numerous attributes of the nodes, the edges, and the graph as a whole, the graphical user interface makes it impossible to create a reproducible workflow. In the next section, therefore, we will discuss using Python packages (in particular NetworkX and hvPlot) for exploring and visualizing the graph.

3. Exploring the Graph

3.1. A Static Visualization

Once the graph is built, it can be explored in a variety of ways. Figure 1 represents a visualization of the graph that labels the nodes with the highest degree using a spring layout³. Node color indicates eigenvector centrality, ranging from yellow (high) to blue (low). The labeled nodes include AmarSuen, the king, Abisimti, the queen, and AradNannak, the prime minister. The most central figure in the graph is Ludinjirak, who was in charge of the goldsmiths and was succeeded by PuzurErra. The other names included in the visualization may also be recognized as performing important functions in the organization. Ea'ili and Šu'Eštar were responsible for the administration of luxury shoes and boots in consecutive periods. Dayyanummišar was in charge of weapons, including luxury weapons for display purposes. Lugalkugzu fulfilled an important coordinating position between the various offices within the treasury and between the (much larger) administration of domestic animals and the treasury. The information about the identity of the actors is not available in the graph itself but may be retrieved from Paoletti (2012). The persons with the highest degree (those labeled in the graph) include important political figures (the royal family and the prime minister), figures in leading positions in the institution, and people who mediate between different branches.

3.2. Interactive Visualization

One drawback of the visualization in Figure 1 is the absence of labels for the lower-ranked nodes. Printing all the labels is certainly possible but leads to an illegible graph. We may improve the visualization by adding interactivity and displaying additional information about the nodes in tooltips. A Python package that was written for such interactive purposes is Bokeh. We can transform Figure 1 into the interactive visualization in Figure 6⁴.

The Bokeh plot can be saved as an image, but also as a free-standing HTML file, in which the interactivity is preserved. Hovering over a node with the mouse turns that node and all its edges green and displays some basic information about the node in tooltips (name, degree, and eigenvector centrality). Clicking on the node will highlight the node and its edges, but it will turn all other graph

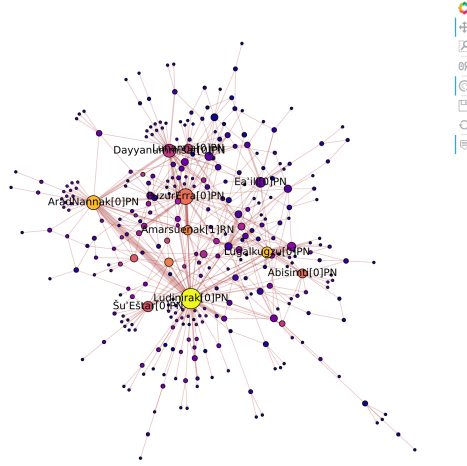


Figure 6: Interactive

elements gray. The tools in the toolbar may be used to zoom in, to save (as .png) or to reset.

We can now move around in the visualization and discover the names of the smaller nodes. Doing so, we will also find gods (marked with DN) who are the recipients of valuable objects.

3.3. Cliques and Clique Communities

The graph allows for a variety of further inquiries, based on the mathematics of graph theory. Cliques are sets of nodes that are all connected to each other. That is, if we have four nodes, A, B, C, and D, all possible connections exist. A related concept is the k-clique community. A k-clique community consists of adjacent cliques of at least k members, where adjacent cliques share at least k-1 nodes. As we will see, k-clique communities tend to overlap. That is, there are powerful actors who belong to multiple such communities and connect them to each other. If we compute the k-clique communities (k=4) for the present graph, we get the following names:

Clique 1: Amarsuenak[1]RN, AradNannak[0]PN, Ayaḅu[0]PN, Dada[0]PN, Dayyanummišar[0]PN, Enkik[1]DN, Lisin[0]PN, Ludinirak[0]PN, Lunanna[0]PN, PuzurErra[0]PN, Šulgir[1]RN

Clique 2: AradNannak[0]PN, Ludinirak[0]PN, Tahišatal[0]PN, Šušulgir[0]PN

Clique 3: Ludinirak[0]PN, Lugalkugzu[0]PN, PuzurErra[0]PN, Utamišaram[0]PN

Clique 4: Ludinirak[0]PN, Lugalkugzu[0]PN, Ribagada[0]PN, Tahišatal[0]PN

We see several of the names that also appeared as high-ranking nodes in Figures 1 and 6, but we also find several new names, including a god (Enkik) and a second king (Šulgir). Quite a few names appear in more than one k-clique community. The visualization below shows the nodes that belong to at least one k-clique community, plus the nodes with a degree of at least 8. The nodes are colored according to the k-clique community to which they belong – black represents nodes that do not belong to any k-clique community, and red nodes (not accidentally the larger ones) belong to multiple such communities. Hovering over a node will highlight the edges of that node in the color of the k-clique community. Clicking on Ludinirak will show that he participates in all four k-clique communities and is also connected to Abisimti, a black node.

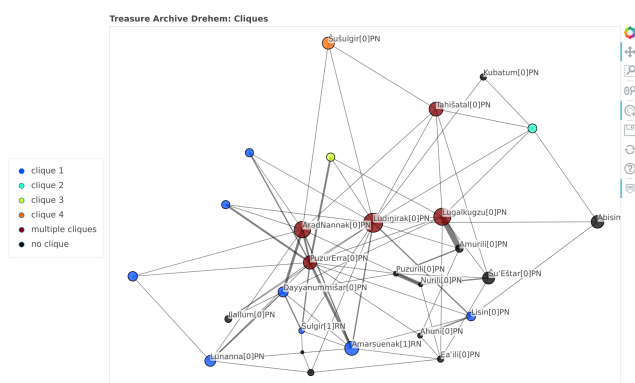


Figure 7: K-clique-com

This last visualization may invite further investigation of the people who do or do not show up among the k-clique communities.

4. Further Thoughts

The scripts that produce the graph and the visualizations were developed for this particular set of 300 documents – a tiny drop in the ocean of ancient texts. These scripts, in other words, cannot be used as out-of-the-box tools for visualizing and analyzing any data set. Most specific is the part of the script that selects nodes, assigns roles, and creates edges – that is, the actual creation of the graph. ORACC data will have part-of-speech annotations that will make it relatively easy to filter out personal names, royal names, gods, and/or place names. The assignment of roles, however, very much depends on the activities recorded in the archive at hand and how these activities were formulated. Even within the Ur III period, the set of key words developed for this text group is not necessarily

valid for any other text group.

I do hope that the steps formulated and exemplified in the scripts will help to do similar things for other text groups. Most administrative traditions will use key words to mark the most important actors in transactions. If it turns out that such an approach is not feasible, one may always collect edges by hand. For a corpus of several hundred documents, that is labor-intensive but doable (and may produce fewer errors than the approach discussed here). Manual entry of edges may be somewhat more prone to inconsistencies, and the various utilities for checking nodes, roles, and edges as proposed here may still fulfill important functions. Once the graph has been constructed the possibilities are almost endless – the examples shown here are, indeed, just examples. I hope they will inspire other researchers to use interactive visualizations for similar (or very different!) analyses.

One may ask: did our scripts produce new knowledge? Since we did not start with a research question, there is no hypothesis that was proven or refuted at the end of this story. The approach was, from the start, an exploratory one and so the question should be: does this approach, or one like this, help in exploring an ancient dataset?

For the interpretation of the graph we have relied heavily on the more traditional approach in Paoletti (2012). In fact, it would be difficult to make much sense of all of this without such work. The network analysis, therefore, is not going to replace traditional close reading of texts. I propose that graphs and interactive visualizations may play a role first, in working with students who are new to the material. It will show rather quickly, who the central people are and how they are (or are not) connected to each other. Second, for researchers and specialists the visualizations may add another layer to the understanding of the dynamics of the social network attested in the documents at hand. A five-hundred-page book gives insights that cannot be reproduced computationally. On the other hand, the bird’s-eye view provided by the various visualizations shows, for instance, the (relative) importance of the royal family in the day-to-day business of the treasury office. Figure 7 not only includes two kings among the most important nodes (AmarSuenak and Šulgir) but also two queens (Abisimti and Kubatum). Since this is a royal archive that deals with royal property, it is not surprising to see the king showing up, but the prominence of the king himself (and his wife) in such operations is unexpected. The most central figures, those who participate in more than one k-clique community, however, are the high administrators and the prime minister, AradNannak.

The visualizations shown here do not take into account the dates of the documents, and therefore we see people in the same graph that were not active at the same time. Dates, however, are part of the metadata in the NRAD model, attached to the Document. In a future incarnation we may well utilize such dates for restricting the documents to a range of dates. Similarly, one of the edge attributes in the graph discussed here contains the document IDs (the document or documents in which these edges occur), which can easily be turned into the

URLs of their online editions. It is not far-fetched, therefore, to imagine an interactive graph where clicking on an edge opens the editions of these documents, further enabling the exploration. The possibilities for explorative research seem endless.

Notes

1) This article is based on my Compass (Computational Assyriology) project (<http://github.com/niekveldhuis/compass>), Chapter 4. This chapter profited from the Sumerian Networks project (<http://github.com/niekveldhuis/sumnet>). Sumerian Networks is a Data Science Discovery project that was coordinated from 2017-2021 by Dr. Adam Anderson. The undergraduate UC Berkeley students who participated in the project for shorter or longer periods of time are Yashila Bordag, Colman Bouton, Jennie Chen, Tiffany Chien, Dalton Do, Zekai Fan, Kimberly Kao, Jason Kha, Anya Kulikov, Rachel Lim, Dominic Liu, Harini Rajan, Max Sullivan, Anjali Unnithan, and Lucie Yoonsun Choi. In addition, Aleksi Sahala, visiting graduate student from the University of Helsinki, participated for one semester.

2) The Compass project is still under construction. The code for data acquisition, analysis, and visualization is functional, but at the time of writing not all code is sufficiently documented.

3) The code for this visualization was written by Colman Bouton. The code for Figure 5 reuses his functions for sizing and placing the nodes.

4) The code for Figure 5 was written in HvPlot, a package that uses HoloViews and Bokeh for plotting data from a variety of Python libraries, including NetworkX. A good introduction to the combination of NetworkX and Bokeh is Chapter 6 of *Introduction to Cultural Analytics & Python* by Melanie Wash (2021).

Submitted 2021-07-06 | Published 2021-10-31

References

Paoletti, P. (2012). *Der König und sein Kreis. Das Staatliche Schatzarchiv der III. Dynastie von Ur. Biblioteca del Próximo Oriente Antiguo 10*. Consejo Superior de Investigaciones Científicas.