Imprint

Université du Luxembourg 2023

Luxembourg Centre for Contemporary and Digital History (C²DH)
Université du Luxembourg
Belval Campus
Maison des Sciences Humaines
11, Porte des Sciences
L-4366 Esch-sur-Alzette

Editors
Asst. Prof. Dr. Marten Düring (Luxembourg Centre for Contemporary and Digital History | C²DH)
Apl. Prof. Dr. Robert Gramsch-Stehfest (Friedrich-Schiller-Universität Jena)
PD Dr. Christian Rollinger (Universität Trier)
Dr. Martin Stark (ILS – Institut für Landes- und Stadtentwicklungsforschung, Dortmund)
Clemens Beck, M. A. (Friedrich-Schiller-Universität Jena)

ISSN 2535-8863

Contact
Principal Contact
JHNR-editors@historicalnetworkresearch.org
Support Contact
Dr. Marten Düring (Université du Luxembourg)
JHNR-support@historicalnetworkresearch.org

Cover Design and Typesetting
text plus form, Dresden, Germany
Cover image
Martin Grandjean
Copyediting
Andy Redwood, Barcelona, Spain

Published online at
https://doi.org/10.25517/jhnr.v8i1

The Journal of HISTORICAL NETWORK RESEARCH

RYAN MUTHER/DAVID A. SMITH/
SARAH SAVANT

From Networks to Named Entities and Back Again

Exploring Classical Arabic *Isnād* Networks

Journal of Historical Network Research 8 (2023) 1-20

Keywords name disambiguation, network analysis, natural language processing, hadith

Abstract This paper explores new methods for disambiguating the identity of individuals in classical Arabic citations (*isnāds*) using a network-based approach. After training a model to extract name mentions from classical Arabic, we embed these mentions in vector space using fine-tuned BERT representations and use community detection to infer clusters of coreferent mentions. The best-performing clustering approach reduces error on the CoNLL metric by 30%. Then, as a case study, we examine the problem of determining the number of direct transmitters to Ibn ʿAsākir (d. 1176) in a set of *isnād*s taken from the 12th century historical text *Taʾrīkh Madīnat Dimashq* (*TMD, History of Damascus*), using our method to replicate human judgement.

C²DH
LUXEMBOURG CENTRE FOR CONTEMPORARY AND DIGITAL HISTORY

uni.lu
UNIVERSITY OF LUXEMBOURG

1. Introduction*

Many questions in literary history concern how certain authors worked, including what sources they used and how those sources were accessed. Often, evidence for the answers to such questions can be found through citations. In the classical Arabic written tradition, citations frequently took the form of *isnād*s (chains of transmission). Rather than giving a single source, *isnād*s give a more complete provenance by listing a sequence of transmitters that records who received information from whom, tracing back to a sufficiently reputable source, such as the Prophet, one of his companions or, as was often the case for later periods, well-respected scholars. An example *isnād* is shown below, with accompanying translation.

حدثنا أبو داود قال: حدثنا هشام عن قتادة عن الحسن عن سمرة أن النبي صلى الله عليه وسلم

> Abū Dāwūd transmitted to us, saying, Hishām transmitted to us, from Qatādah, from al-Ḥasan, from Samurah the Prophet (May the peace and blessing of God be upon him).[1]

In the above *isnād* there are five transmitters, each linked by a "transmissive term" such as *transmitted to* or *from*.

Taken collectively, a group of *isnād*s can be thought of as a natural language representation of the social network of textual production, describing links between pairs of individuals involved in the transmission of information. Using this collection of transmission data, we can understand the roles that different individuals played in the composition of a text, or the particular part of the text described by the *isnād*s. For instance, a collection of *isnād*s from a single text that dates back to a common individual would allow one to explore the processes through which the individual's words were transmitted to the author of the text in question. Despite the potential of this sort of analysis, this data is often difficult to work with directly, due to uncertainty concerning the breaks between names within *isnād*s and the variety of names used to refer to a particular transmitter. Two identical names may in reality refer to different individuals, while conversely, two names that have different surface forms might refer to the same individual. This requires names to be located and linked to individuals before the network of individuals can be constructed, let alone meaningfully analyzed. Since the same individual often occurs in different positions in different *isnād*s (e.g. sometimes

* **Acknowledgements:** This research is part of a project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant agreement No. 772989)
Corresponding author: Ryan Muther, muther.r@northeastern.edu
1 Al-Ṭayālisī, d. 204AH/819CE

they may be in the second position, while in others they are third), simpler inference models are ineffective, as we can't simply rely on observations like "if name X occurs in position Y, it is individual Z." Helpfully, the structure of *isnād*s provides substantial evidence that can be used to resolve both of these issues. Both the structure of Arabic names and the use of common transmissive terms can be used to locate names within the text of an *isnād* (hereafter referred to as mentions), while the presence of other transmitters in an *isnād* provides a useful signal for entity disambiguation. Leveraging this latter kind of information is essentially using the network of individuals as evidence for inferring identities. One person may go by multiple names, but the fact that the different forms of their name co-occur with the same set (or similar sets) of names in *isnād*s is evidence that the two names refer to the same individual. In this paper, we will present a method for inferring clusters of mentions that refer to the same individual given a collection of ambiguous name mentions in *isnād*s.

Our approach can be summarized as follows: given a collection of *isnād*s, we first run a named entity recognition (NER) model to locate mentions in each *isnād*. We then embed each mention in a high-dimensional vector space – a representation of the mention in its context – using a hybrid English-Arabic contextual word embedding model based on BERT[2, 3] and tuned to predict names in *isnād*s. These embeddings have the useful property that names that occur in similar contexts have similar embeddings, so we can use a measure of similarity between the embeddings of mentions to construct a network of embeddings in which close embeddings are linked by edges, with their similarity as the edge weight. Finally, we experiment with different algorithms to detect communities in the embedding network. These algorithms produce clusters that can be interpreted as an assignment of a shared identity to sets of names, although without explicitly giving a name to each cluster.[4]

This paper is organized as follows: Section 2 discusses related work in computer science on entity linking and a historiography of the book we annotate for our experiments, Ibn ʿAsākir's *Taʾrīkh Madīnat Dimashq,* i.e., *History of the City of Damascus,* hereafter *TMD,* from the 12th century CE. We also discuss the relevance of our work to historians working on Arabic books. Section 3 describes the data and the annotation process used to collect gold standard mention-entity links. Section 4 describes the process used to convert the *isnād* name data into networks for community detection, as well as some results for the general case of

2 Devlin et al, 2018.
3 Lan et al, 2020.
4 Data and code used for this article, including isnads with the disambiguated names and surface forms for each isnad, the embedding networks presented in our results, and the clusters we find in our data, as well as all code relevant to creating and evaluating these models, can be found at: https://github.com/mutherr/isnadNameDisambiguation.

attempting to disambiguate all names in the dataset at once. Section 5 presents a case study on the usefulness of these methods for answering an open question in the study of *TMD*. Section 6 discusses the broader ramifications of this work and potential avenues for future research.

2. Related Work

We decompose the problem of matching name mentions to individuals into two tasks: named entity recognition (NER) to locate the mentions within an *isnād,* and entity linking to map the mentions of the individuals they represent. Named entity recognition is a fairly straightforward and well studied sequence tagging problem,[5] and is comparatively easy to do in this setting. Entity linking presents some complications, however. In most settings, entity linking is done by using a broad-coverage resource like Wikipedia[6] as an authority list with descriptions of entities, using the similarity between a mention in context and an entity's description to assign mentions to entities, treating the problem as one of classification. In this instance, however, most of the entities in our data are not in Wikipedia or other commonly-used authority lists, and the context of the mentions we are working with consists almost wholly of other names, so the similarity between the mention's context and a description of an individual might not be a useful predictor of the identity associated with a particular mention – even if the individual in question is in the authority list. Furthermore, relying on authority lists becomes increasingly difficult when the same short form of a name is used for multiple individuals whose full names differ in the same text. For example, if there are two different individuals that the author refers to simply as "Muḥammad," the name gives less information about who the individual is. This is often the case in other texts in the OpenITI corpus,[7] such as those by the prolific 9th century historian Muḥammad b. Jarīr al-Ṭabarī (d. 923). This does not, it should be noted, occur in the data we use for our experiments, where each name is associated with exactly one individual.

To overcome the limitations of traditional entity linking, we recast the entity linking problem as a clustering problem similar to coreference resolution,[8] in which we try to find clusters of mentions that refer to the same individual, which allows us to work on this problem without an authority list. In the future, more annotated resources, such as classical Arabic biographical dictionaries, could

5 Lample et al, 2016.
6 See Durrett and Klein, 2014 and Muller and Durrett, 2018.
7 The OpenITI corpus is a machine-readable collection of medieval Arabic texts that we use as the source for the version of *TMD* we are working with. For more information, see https://zenodo.org/record/6808108.
8 Lee et al, 2017.

be useful as authority lists in this domain. Due to the complexity of this style of presenting evidence, most prior work on *isnād*s has focused on collections of hadith, which have a predictable structure amenable to processing with regular expressions and other lexical features.[9] In this paper, however, we focus on extracting scholarly networks from less formally structured historical texts.

We evaluate our approach by applying it to Ibn ʿAsākir's 12th-century CE (6th-century AH) book on the history of Damascus *Taʾrīkh Madīnat Dimashq* (*TMD*). Ibn ʿAsākir (d. 517AH) was a prolific Islamic scholar and historian orginally from Damascus, who wrote extensively on the history of Syria in *TMD.* The work consists of a first volume, treating the history of the city, including its ancient roots and seventh-century conquest, and a second volume covering the topography of the city. The remainder of the book comprises biographies of the elites who lived or passed through Damascus prior to Ibn ʿAsākir's time.

Historians have sought to identify Ibn ʿAsākir's sources and methods of working, relying on the *TMD* itself, as well as other early historical works, including a book called *Muʿjam al-shuyūkh* (Catalogue of Teachers) in which Ibn ʿAsākir lists his teachers. Recently, Jens Scheiner built on earlier scholarship in a description of Ibn ʿAsākir's "virtual library."[10] He created a list of 100 works that Ibn ʿAsākir consulted, including that of the 9th-century CE (3rd-century AH) scholar Ibn Saʿd. Ibn Saʿd is a particularly important source for Ibn ʿAsakir, so we limit our analysis to *isnād*s dating back to Ibn Saʿd. For 58 of the works, Scheiner provided one or two chains of transmission which he said documented recensions of a text, and for one work, four chains.[11] Scheiner maintained that Ibn ʿAsākir's use of these works, among others, illustrates "Ibn ʿAsākir's love for books."[12]

We believe that Scheiner and other scholars have understated the range and scale of Ibn ʿAsākir's sources, in large measure because of the great size of the book and the number and variety of its *isnād*s. There are over 75,000 *isnād*s in the *TMD,* containing many thousands of different surface forms for names.[13] Even simple searches for authors' names within the *TMD* show that Scheiner's estimate for the number of transmission chains in which authors' names feature is far too low. Scheiner's description of a library accounts for neither the great variety of surface forms, nor the greater number of citation networks in which they

9 Altammami, Atwell, and Alsalka, 2019.
10 Scheiner, 2017. Nūr Sayf, 1979; Conrad, 1991,1994, 1988 and 1990; al-ʿAmrawī and Shīrī, eds., 1995–2001; Judd, 2001; and Daʿjānī, 2004.
11 Scheiner lists three works by Ibn Saʿd in Ibn ʿAsākir's "library." Two have one citation chain each, while the third work has none.
12 Savant, 2022.
13 Savant and Seydi (forthcoming).

sit. It also assumes that complete works lie behind the quotes that Ibn ʿAsākir gives, but this is not supported by the language of Ibn ʿAsākir.

Tackling Ibn ʿAsākir's method is a large task, but a starting point is to understand the *isnād*s and the citation networks they represent. The case of Ibn Saʿd citations points to a model of transmission where Ibn ʿAsākir is relying on multiple direct informants, and citing them in many different ways. By understanding the situation with Ibn Saʿd, we can build an understanding of citation in the book as a whole. Furthermore, the *TMD* may be exceptionally large, but in format, content, and style it represents a dominant form of historical writing. If more efficient methods can be ascertained, it would be greatly beneficial for historians of Arabic literary history in general.

3. Data and Annotation

For our experiments, we worked with a dataset of 2,379 *isnād*s taken from the *TMD,* all of which go back to their sources through Muḥammed Ibn Saʿd. The *isnād*s were extracted from a machine-readable version of the *TMD* based on the 80-volume, 1995–2001 Dār al-Fikr edition edited by ʿUmar al-ʿAmrawī and ʿAlī Shīrī, but excludes volumes 71–80. Volumes 71–74 represent a *mustadrak,* or amendment by the editors (including additional biographical entries). Volumes 75–80 represent indices. A more recent edition published in 2020 exists, but we do not yet have a machine readable version of it, so this is our only option for a complete version of the text. While some of the *isnād*s go back further than Ibn Saʿd and give the sources for his information, our annotations only go back to him, regardless of the actual endpoint of the *isnād.* In total, the annotated sections of these *isnād*s contain 14,454 mentions, the technical term for individual instances of names occurring in a text, of which 13,072, around ninety percent, have been disambiguated by assigning them a known identity. The remaining ten percent were too ambiguous for the annotator to readily assign to a particular individual. The disambiguation is only meant to cover the most frequent transmitters, so leaving the less common transmitters ambiguous was considered acceptable for annotation purposes. The final set of disambiguated mentions contains 44 individuals, with anywhere from one to twenty-six different surface forms referring to the same individual. A histogram of the number of different surface forms assigned to each individual is shown in Figure 1.

Just under half of the individuals have a single surface form, while most of the remaining twenty-four individuals have less than ten different surface forms, and a small handful have more than ten distinct surface forms. It should be noted that, for this particular dataset, the mapping from mention surface forms to individuals is many to one, rather than many to many, as each disambiguated surface form refers to exactly one individual. This is a side-effect of the data annotation process, and would not necessarily hold true for a set of *isnād*s collected from

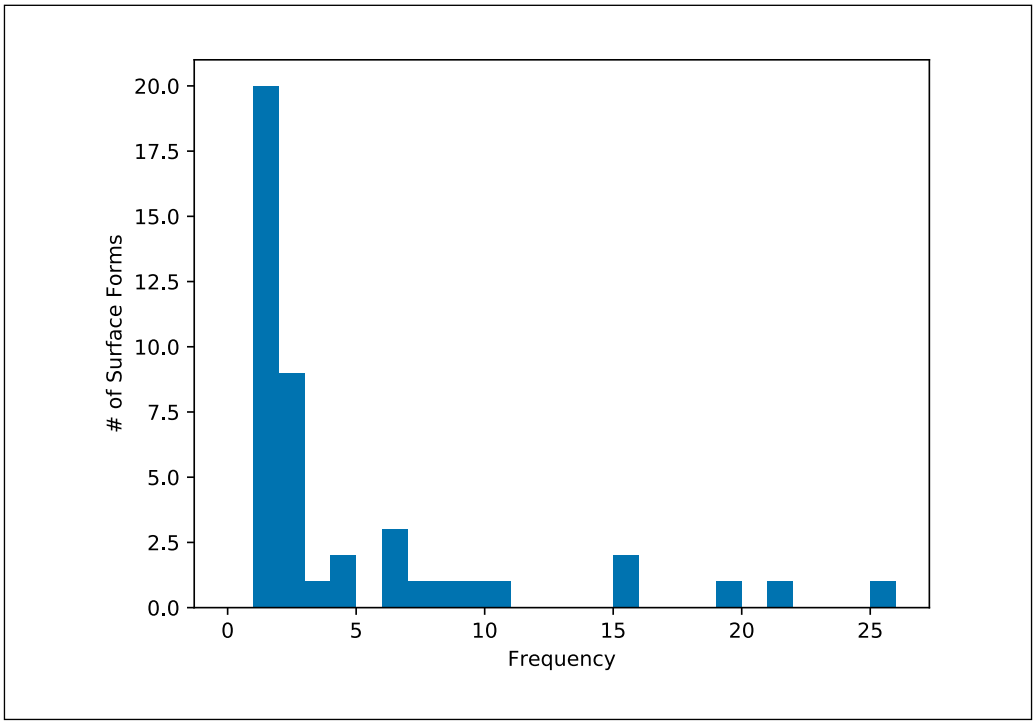

Fig. 1 Histogram of surface form counts for individuals in the *isnād* dataset

multiple texts, or even for a wider set of *isnād*s taken from the *Taʾrīkh Madīnat Dimashq* alone. With a corpus collected from multiple authors, the chance that two identical names refer to different people increases. Even just having a broader-coverage set of *isnād*s from this one work makes this more likely. If, for example, Ibn ʿAsākir uses two different sources at different points in the text whose citations involve identically-named but distinct individuals, it might be clear to the reader from context which of those people the shared names refer to, saving the writer the effort of constantly giving completely unique names to every transmitter. As an example, the narrator Abu Bakr Muḥammad b. Shujāʿ has six different surface forms, which can be seen in Figure 2.

When we look at the top narrators upon whom Ibn ʿAsākir relies – his direct informants – we can identify persons on whom Ibn ʿAsākir relies heavily across the *TMD,* and not just for the Ibn Saʿd material. This would be impossible without

أبو بكر اللفتواني, أبو بكر محمد بن شجاع, أبو بكر محمد بن شجاع اللفتواني, محمد بن شجاع, أبو بكر بن شجاع, أبو بكر محمد بن أبي نصر بن محمد اللفتواني

Fig. 2 Example surface forms of Abū Bakr Muḥammad b. Shujāʿ

the work on named entities. The most common network involves persons across five generations going back to Ibn Saʿd. Ibn ʿAsākir's most often cited direct informants within it include men by the names of Abū Bakr Muḥammad b. ʿAbd al-Bāqī (d. 1140), Abū Ghālib (d. 1133), and Abū Bakr Muḥammad b. Shujāʿ (d. 1138). Another major transmitter within the network is Ibn Ḥayyawayh (d. 983–84), upon whom Abū Bakr Muḥammad b. ʿAbd al-Bāqī and Abū Ghālib rely via different intermediaries. There is also Abū Bakr b. Abī Dunyā (d. 894), a famous tutor to caliphs and author in his own right; he passes on quotes directly from Ibn Saʿd. These transmissions then pass through three generations of persons to Abū Bakr Muḥammad b. Shujāʿ, and then to Ibn ʿAsākir.

4. Mention Network Creation

As discussed above, the process of creating networks out of mentions to distinguish between mentions of different individuals is done in two steps: named entity recognition and entity embedding. We will now discuss these two steps in more detail.

4.1 Named Entity Recognition

The first step in the process is that of finding all of the mentions in the collection of *isnād*s. This is usually done as a token-level tagging task where each token in a document is assigned one tag in {B, I, O}, where B indicates that a token begins a mention, I indicates that the token continues a mention, and O is used for tokens that are not part of mentions. As an example, the *isnād* from earlier would be tagged as:

O	O	O	O	O	B	O	B	0	B	O	O	I	B	O
عليه	الله	صلى	النبي	أن	سمرة،	عن	قتادة	عن	هشام	حدثنا	قال:	داود	أبو	حدثنا

Arabic readers will note that the term *nabī,* Prophet, is tagged as O. This is because the report is about the Prophet, not transmitted by him.

Sequence tagging problems like this are well-studied in NLP, and state of the art solutions involve finetuning pre-trained deep learning models, like BERT or its variants, for the task in question using an annotated dataset. For the named entity recognition step, we use a GigaBERT based English and Arabic model originally published by Lan et al[14] as our base model for fine tuning.

14 Lan et al., 2020.

Training Set	Precision	Recall	F1
Classical	.99	.99	.99
Modern	.95	.97	.96

Tab. 1 A comparison of NER models evaluated on classical Arabic finetuned on Classical and Modern Standard Arabic, respectively

One problem we run into with this approach is that our text is all in classical Arabic, for which very little annotated training data for NER exists, aside from the annotated names that were made in the process of creating the disambiguated data. We could finetune the model directly on that classical Arabic data, but the comparatively small size and limited coverage of the dataset could limit the generalizability of the resulting NER model. As an alternative, we could instead finetune the model on the Modern Standard Arabic NER corpus ANERCorp[15], then use that model to tag mentions in the *isnād* data. To give an understanding of the tradeoff involved in this choice of training data, we compare two NER models, one finetuned on classical Arabic and the other on modern Arabic, in Table 1. The model finetuned on classical Arabic is evaluated using ten-fold cross validation[16] on the annotated *isnād* dataset, so we present the average across the ten folds. For the modern data, the training set is completely distinct from the evaluation set, so cross validation isn't necessary.

As Table 1 shows, the model trained directly on classical Arabic only slightly outperforms the model trained on modern Arabic. Given this, and that the classical model is less likely to generalize well, we opt to use the model trained on modern data for the named entity recognition step.

4.2 Mention Embedding

With the named entity recognition step accomplished, we can now move on to creating mention representations which can be converted into a graph. To create the mention representations, we turn to word embeddings produced by the same base GigaBERT model as we previously used for named entity recognition. Instead of finetuning the model for the NER task, we use the contextual word embeddings produced by the model as a starting point for creating mention representations. Contextual embeddings differ from standard word embeddings in

15 Obeid et al, 2020.

16 To get a better estimate of the NER model's performance, we split the complete dataset into ten separate pieces, called folds, then trained ten separate models, each of which was tested on one of the folds and trained on the remaining nine, then reported the average of the ten model's scores.

that rather than assigning a single embedding to every instance of a word in a corpus, the model gives slightly different embeddings depending on the context in which the word appears. Using the model, each token in a document is given an embedding, in this case with 768 dimensions due to the choice of model architecture, which can be thought of as representing the meaning of the token in context. For multi-word mentions, the embeddings for all the tokens in a mention are averaged to create a representation of the whole mention.

As with the named entity recognition step, where we could finetune the model to improve performance on classical Arabic, we can also finetune[17] the embeddings used to create the mention representations. To do this, we train the base model using a masked language model (MLM) objective. Unlike standard MLM training, which masks tokens in the input at random, we focus the training process on the mentions by only masking out whole mentions in each training example, then train the model to predict a masked word based only on its context. To construct the training data for finetuning the embeddings, each *isnād* d_i with n_i mentions is converted to n_i distinct training examples, each with one of the mentions replaced with [MASK] tokens. Using this dataset, we train the model for three epochs[18] to refine the embeddings of the mentions so that the model better understands how to embed and predict names. Consequently, the resulting embeddings more accurately convey the nuanced linguistic understanding required to disambiguate individuals from their contexts within *isnād*s. To evaluate this model intrinsically, we tested the name prediction accuracy of the tuned model in a tenfold cross validation setup and found an average accuracy of .81. As we will show below, the choice of whether to finetune the embeddings has a significant effect on the downstream community detection performance.

4.3 Network Construction

Having embedded the mentions as points in a shared space, we can now use those points as the basis for constructing a network. Ideally, we want to construct a network so that each mention is connected by an edge to other mentions of the same individual, regardless of changes in the surface form used to refer to the individ-

17 The process of finetuning the embeddings themselves directly, rather than adapting the model to a particular task as with NER, is more properly called pretraining in the NLP literature, but here we employ pretraining in a slightly unconventional way and so refer to it as finetuning, despite the slight inconsistency with standard terminology.

18 In each epoch, the model is told to predict names in isnads by filling in the blank where a name is missing in each training instance in the entire training set. When the model predicts an incorrect name, the error in the model's prediction is used to update the parameters of the model, including the word embeddings, so it is more likely to be correct in the future. Over the course of several epochs, this improves the performance of the model at predicting names and refines the embeddings to more accurately convey the contextual meanings of the words they represent.

ual. To accomplish this, one could employ one of several possible heuristics to construct the network by selecting pairs of mentions to link. The most straightforward would be to simply link each mention to the k-nearest mentions. While this is simple to understand, the disadvantage is that it creates a network with a somewhat uninformative structure, with each mention connected to exactly k other mentions regardless of how similar those mentions might be. To alleviate this and create a potentially more informative network, one can take advantage of the relative sparsity of different surface forms by defining a radius r around each mention, using the distance to the furthest identical surface form mention as r, then connecting the mention to every other whose distance in embedding space is at most some multiple m of r. This, however, leaves singleton surface forms neighborless. To resolve this, we use the average of the radii of low-frequency surface forms, those with counts between 2 and 5, as an estimated radius for the neighborhood around singleton mentions. For the results below, m = 1 unless otherwise specified. Using this heuristic, each surface form's embeddings collectively form a clique inside the larger network, with common surface forms being more densely connected than rarer ones. To add weights to the edges between mentions, we use the cosine similarity between the two mentions as the weight of that edge.

5. Community Detection

To evaluate the quality of these graphs, we create graphs using a variety of heuristics and apply two different community detection algorithms – the label propagation (LP) algorithm described by Raghavan,[19] and the Leiden algorithm described by Traang et al.[20] – to produce clusters of mentions, the results of which can be seen in Table 2. As a baseline for comparison, we also show the scores for naive clustering algorithms that ignore the network and place all identical surface forms in the same cluster (Naive), or in a single cluster (Single).

5.1 Evaluation Metrics

For evaluation metrics, we use several common metrics used to evaluate coreference systems. B Cubed[21] computes average precision and recall at the mention level. Constrained Entity-Aligned F-measure (CEAF)[22] computes an alignment between model clusters and gold standard clusters using a bipartite graph matching algorithm, then uses that mapping to compute precision and recall. The authors proposed two variants of that metric, one based on mentions, the other based on

19 Ragavan, Albert, and Kumara, 2007.
20 Traang, Waltman, and van Eck, 2014.
21 Bagga and Balwin, 1998.
22 Luo, 2005.

		B Cubed1			$CEAF_m$			$CEAF_e$			BLANC1			
Link Method	**Algorithm**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**CoNLL**
–	Naive	**1.0**	.674	.805	.807	.784	.795	.209	.854	.336	**.988**	.834	.900	.706
–	Single	.098	**.944**	.178	.184	.178	.181	.310	.007	.014	.049	.475	.089	.392
kNN, k = 100	Leiden	.951	.627	.756	.744	.722	.733	.598	.353	.444	.961	.739	.819	.727
kNN, k = 100	LP	.951	.354	.515	.478	.464	.470	.250	.313	.278	.939	.594	.663	.591
Surface Form	Leiden	.866	.912	**.888**	.887	.861	.874	.484	.516	.499	.948	**.926**	**.936**	**.790**
Surface Form	LP	.880	.857	.868	**.910**	**.884**	**.897**	**.456**	**.612**	**.523**	.921	.904	912	**.790**

Tab. 2 A comparison of different methods of constructing graphs from mentions

gold standard individuals (which Luo calls entities) which use different similarity functions to determine the best mapping. These are referred to as $CEAF_m$ and $CEAF_e$, respectively. Finally, BLANC[23] computes two pairwise F-scores, one for coreference decisions and another for non-coreference decisions, which are averaged to produce a final score. Since each of these metrics has drawbacks, standard practice is to report the CoNLL 2012 score, which is the average of the F1 scores for CEAFe, B Cubed, and a third metric (Message Understanding Coreference[24] (MUC)). Since MUC is designed for working with the much smaller clusters one finds in coreference datasets, its scores for all of our models are around .99, making the metric uninformative and limiting the impact of that metric on the final ConLL scores across models, so we omit it from our evaluation.

5.2 Results

We will now report our community detection results, using the models and metrics mentioned above. We will begin by presenting results for the tuned mention representations, then comparing those to the untuned representations.

From the Table 2, we can see that the more nuanced network construction method using the surface form heuristic tends to give better results, regardless

23 Pradhan et al, 2014.
24 Vilain et al, 1995.

		B Cubed1			$CEAF_m$			$CEAF_e$			BLANC1			
Link Method	**Algorithm**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**CoNLL**
kNN, k = 100	Leiden	.892	.712	**.792**	**.794**	**.771**	**.782**	**.601**	.287	**.388**	**.958**	**.812**	**.875**	**.722**
kNN, k = 100	LP	**.925**	.354	.513	.459	.445	.452	.260	**.290**	.274	.931	.590	.657	.588
Surface Form	Leiden	.393	.745	.515	.531	.515	.524	.543	.111	.184	.654	.747	.666	.560
Surface Form	LP	.099	**.942**	.179	.184	.179	.182	.248	.023	.041	.506	.475	.091	.401

Tab. 3 A comparison of different methods of constructing graphs from mentions using untuned mention representations

of the algorithm chosen. The flaws in the more uniform k-nearest-neighbor network, which generally performs worse than the surface form heuristic based network, can be in part overcome by using a community detection algorithm that takes the similarity between mentions into account, as evidenced by the large performance gap between label propagation, which only uses the network topology, and the more advanced Leiden algorithm, which takes edge weights into account when finding communities.

We can also try the same methods of creating the network using the untuned mention embeddings to see the effect of finetuning on the clustering performance. Results for untuned embeddings can be seen in Table 3.

Interestingly, the performance gain of finetuning is less pronounced for the kNN models, perhaps because the network itself, especially for higher values of k, is less sensitive to changes in the embeddings. The surface form-based models, by contrast, benefit much more from the improved embeddings as the embeddings of different surface forms move to more distinct regions of the embedding space, making the surface form distance heuristic more effective at creating a network of distinct mention clusters corresponding to individuals, especially when, as noted, each surface form corresponds to one individual.

To get a better understanding of the relative performance of different community detection methods, rather than examining the differences in clusters between the methods at the mention level, it is worth looking at the distribution of cluster sizes produced by each method. For reference, Figure 2 shows the cluster size distribution of the gold standard data.

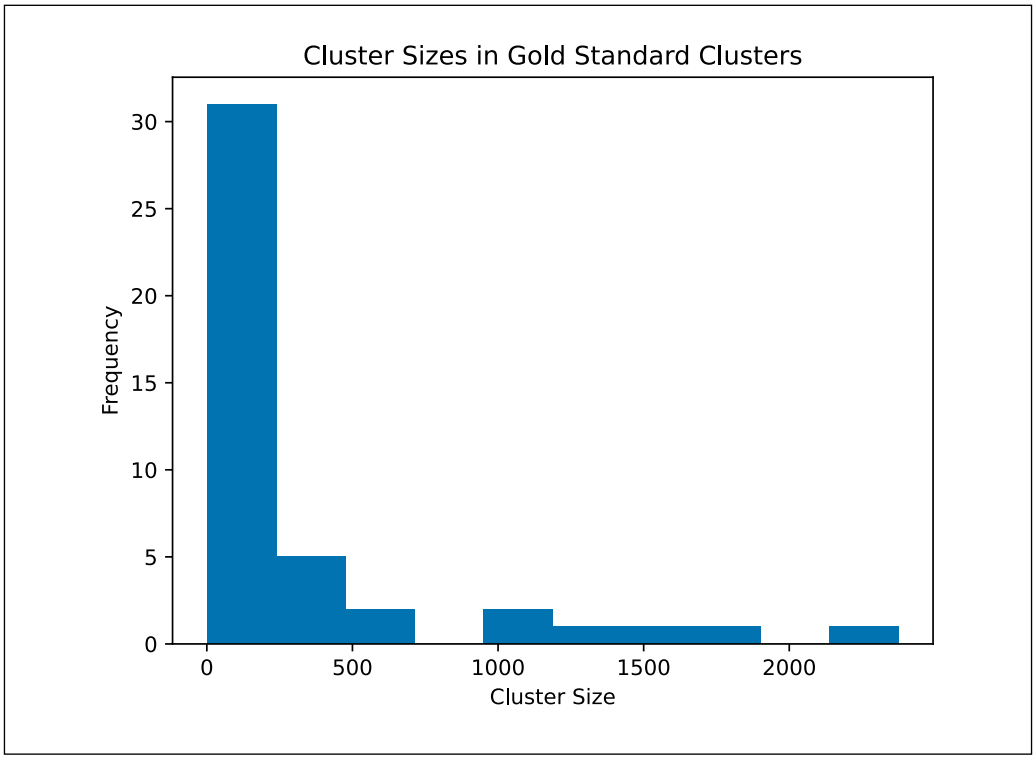

Fig. 3 The distribution of cluster sizes in the gold standard data.

From this, we can see that the data consists of a large number of small clusters from individuals with 500 or fewer mentions, and a small number of significantly larger clusters from a handful of individuals with 1000 or more mentions. Better clustering models will create distributions that look more like the above. Figure 4 shows the cluster size distributions for each clustering method on the tuned mention networks.

By comparing the graphs in Figure 4, we see that community detection on the kNN networks tends to omit the long tail of larger clusters, likely splitting one individual's mentions across multiple of the smaller clusters. By contrast, the surface form networks tend to produce a longer tail of a few larger clusters, with the majority of clusters being small, as in the gold standard data, further demonstrating the suitability of the surface form networks for this task. Both the surface form clustering results still have a larger number of small clusters than the gold standard data, however, so it may be that some surface forms of the same inidividual that are used in very distinct contexts remain difficult for the clustering algorithm to properly place in the same cluster, even after finetuning.

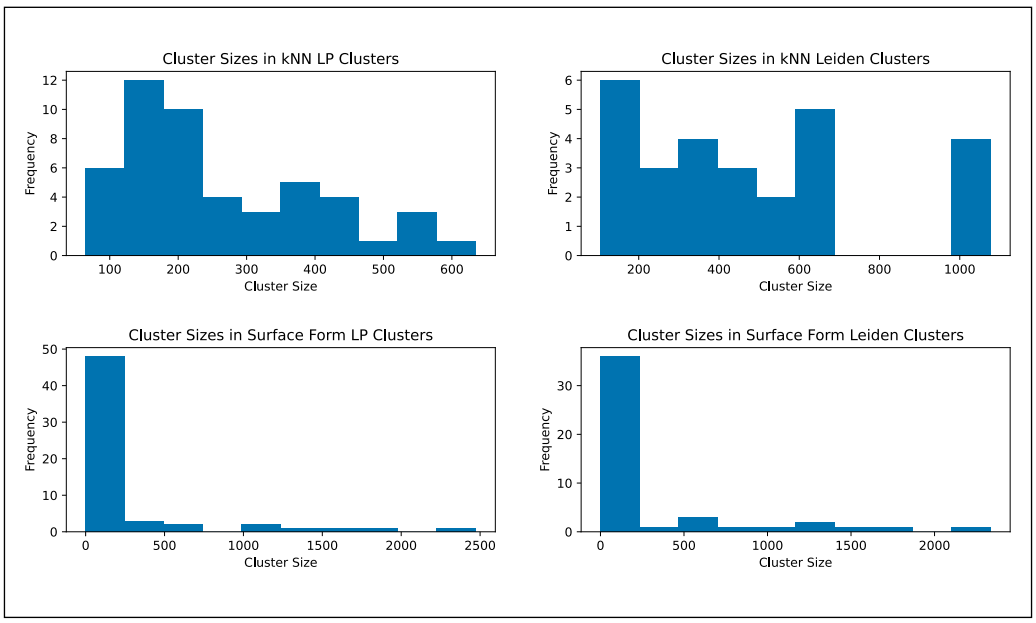

Fig. 4 The distribution of cluster sizes found by the community detection models on tuned mention graphs.

6. Case Study

As a case study on the usefulness of applying these methods, we attempt to quantify how one author, 12th century CE/6th century AH scholar Ibn ʿAsākir, received information from the work of Muḥammed Ibn Saʿd. In particular, we will concern ourselves with the first transmitter in each *isnād* (i.e., the most recent relative to Ibn ʿAsākir) and focus on determining the number of immediate transmitters to Ibn ʿAsākir himself, who conveyed information from Ibn Saʿd, in essence finding the fanout of the graph of transmission from Ibn ʿAsākir to Ibn Saʿd. The test set, which is made using a random sample of twenty percent of the *isnād*s, contains 25 individuals with a total of 63 surface forms. To find the optimal values for the hyperparameters (k and m) and evaluate how well we can solve the problem of determining how many different individuals transmit directly to Ibn ʿAsākir, we take all the mentions that are the first mention in their respective *isnād* and split that dataset randomly into two halves, one for development and one for testing. The development set is used to find the optimal values for k and m for a given community detection algorithm by finding the hyperparameter value which gives the best performance in terms of CoNLL score, while the test set is used to evaluate how well the optimized models perform at inferring the correct number of transmitters. Results for the test set can be seen in Table 4.

From the results in Table 4, it is clear that while neither method finds the correct number of transmitters, the surface form heuristic is significantly more effec-

Model	B Cubed F1	$CEAF_m$ F1	$CEAF_e$ F1	BLANC F1	CoNLL Score	# of Transmitters
Naive Clustering	.719	.698	.524	.786	.741	63
kNN, k = 135, Leiden	.814	.760	.188	.846	.661	4
kNN, k = 135, LP	.732	.641	.141	.739	.618	3
Surface Form m = 1.15, Leiden	.829	.812	.556	.875	.791	32
Surface Form m = 1.05, LP	.757	.744	.593	.796	.779	40

Tab. 4 Clustering evaluation metric scores and transmitter count results for first-name-only clustering of *isnād* test set

tive than constructing the mention graph using k nearest neighbors, regardless of the community detection algorithm chosen. The surface form distance heuristic gives better results using the Leiden algorithm to detect communities, both in terms of CoNLL score and the number of transmitters detected, regardless of the choice of clustering algorithm. These results could be used as a baseline for further refinement by human analysts rather than having to disambiguate the names from scratch by hand, as was done to create the training and evaluation data.

One other thing that should be noted about these results is the low dynamic range of the metrics. Many of the metrics still give fairly high scores to models that give worse estimates of the number of transmitters. The results for the kNN networks show this most clearly, where – by virtue of the metrics alone – the models appear to perform only slightly worse than the same methods on the surface form networks, but are in fact vastly worse at estimating the number of transmitters. As such, when using metrics originally intended for another task, it is important to consider how similar the two tasks are in terms of what is being evaluated. These metrics were largely designed for single-document coreference evaluation on small datasets, where there tend to be a small number of small clusters. As such, when evaluating on much larger datasets like this *isnād* data, some of these metrics may give misleading results. As an example, If you took an otherwise correct large cluster and merged it with an otherwise correct small one so that two individuals were conflated, B Cubed recall would be unaffected, and B Cubed precision for the larger cluster's mentions would only slightly decrease, while the precision for the smaller cluster would be more heavily impacted. However, when one computed the average F-score across all mentions, the average would tend towards the F-score of the larger cluster's mentions, as they repre-

sented a larger fraction of the dataset, artificially inflating the B Cubed score. Similar issues exist for the other evaluation metrics used. That is not to say that these metrics are useless, but that they need to be used with caution, as this task isn't what these metrics were originally designed for. The feasibility of the task itself is more important than the utility of the metrics.

7. Discussion

This work shows many of the tradeoffs involved in trying to work with networks inferred from textual data sources. As shown above, the process of converting raw text into clustered mentions contains several steps, each of which can have a significant effect on the outcome of the final results. Even before considering how to go about constructing the network, everything from the choice of NER model to the source of embeddings used to represent the mentions should be considered, as we see with the importance of finetuning the embeddings before constructing the network. Equally important are the decisions regarding how the network should be constructed once the mentions are embedded, bringing with them their own set of issues in hyperparameter selection and choice of clustering algorithm. The question of how well this finetuning process improves the mention representations of unseen names is still open. While the embeddings of unseen names would not be directly affected by the additional pretraining, the overlap in names between those found in Ibn ʿAsākir and other texts would indirectly influence the embeddings of other names, meaning that the finetuning process may improve performance even on unseen texts. In cases where there is no overlap between what the model saw in finetuning and a new text, there is likely to be litle benefit. It may also be that much of the performance gain from finetuning is due to the distiribution of names across individuals in the dataset, where no surface form is used to refer to more than one individual. Future work on other texts where this is not the case would be needed to investigate this. This research, however, is only the beginning of addressing a much larger set of potential future topics, ranging from how well these methods can be applied to collections with multiple texts where the authorial style can vary, both across individual works by the same author and between authors, to a whole host of related problems in analyzing *isnād*s from a network perspective. These include inferring missing individuals, dealing with still-ambiguous names, or inferring more complete networks from collections of *isnād*s and analyzing the roles played by particular individuals in the dissemination of information. The still-ambiguous names, in particular, represent a salient application of these methods, as the methods could provide useful aids to human annotators interested in this form of data by giving estimates of how likely two mentions are to refer to the same individual, potentially helping disambiguate previously difficult or impossible instances. The most interesting immediate avenue for future work is likely to extend these models to work with multiple texts, especially those that have drastically different background distributions of names between individuals. The problem becomes much

more complicated once names do not always refer to one individual unambiguously, as is often the case with other authors like Tabari, and the process of connecting entities across texts is non-trivial, making the problem computationally interesting as well as historically relevant.

8. References

Abū Dāwūd Sulaymān b. Dāwūd al-Ṭayālisī (d. 204AH/819CE), *Musnad Abī Dāwūd al-Ṭayālisī,* Muḥammad b. ʿAbd al-Muḥasin al-Turkī ed. (Cairo: Dār al-Hijr, 1999), vol. 2, 223–4.

Altammami, Shatha, Eric Atwell, and Ammar Alsalka. "Text Segmentation Using N-Grams to Annotate Hadith Corpus." In *Proceedings of the 3rd Workshop on Arabic Corpus Linguistics,* 31–39. Cardiff, United Kingdom: Association for Computational Linguistics, 2019. https://www.aclweb.org/anthology/W19-5605.

Bagga, A. and B. Baldwin. "Entity-Based Cross-Document Coreferencing Using the Vector Space Model." In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics,* Volume 1 1998: 79–85. Association for Computational Linguistics.

Conrad, Gerhard. *Abū'l-Husain al-Rāzī (–347/958) und seine Schriften: Untersuchungen zur frühen Damaszener Geschichtsschreibung* (Stuttgart: Franz Steiner, 1991).

idem. *Die quḍāt Dimašq und der Maḏhab al-Auzāʿī: Materialien zur syrischen Rechtsgeschichte* (Beirut: Orient-Institut der DMG, 1994).

idem. "Zur Bedeutung des Taʾrīḫ madīnat Dimašq als historische Quelle," in *Zeitschrift der Deutschen Morgenländischen Gesellschaft,* Supplement VIII: XXIV, ed. Werner Diem and Abdoldjavad Falaturi (Stuttgart: Deutscher Orientalistentag, 1988, 1990), 271–282.

Daʿjānī, Ṭalāl b. Saʿūd Daʿjānī. *Mawārid Ibn ʿAsākir fī taʾrīkh Dimashq* ([Medina]: al-Mamlaka al-ʿArabīya al-Saʿūdīya, Wizārat al-Taʿlīm al-ʿĀlī, al-Jāmiʿa al-Islāmīya biʾl-Madīna al-Munawwara, ʿImādat al-Baḥth al-ʿIlmī, 2004).

Devlin, Jacob, et al (2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." *CoRR* 2018 abs/1810.04805.

Durrett, Greg, and Dan Klein. "A Joint Model for Entity Analysis: Coreference, Typing, and Linking." *Transactions of the Association for Computational Linguistics 2* (December 2014): 477–90. https://doi.org/10.1162/tacl_a_00197.

Ibn ʿAsākir. *Taʾrīkh Madīnat Dimashq.* ʿUmar al-ʿAmrawī and ʿAlī Shīrī eds. 80 vols. Beirut: Dār al-Fikr, 1995–2001. Vols. 71–74 represent an amendment by the editors (including additional biographical entries); vols. 75–80 represent indices. The OpenITI file that we used for this chapter, 0571IbnCasakir.TarikhDimashq.JK000916-ara1 is based on this edition but excludes vols. 71–80. https://github.com/OpenITI/0575AH/blob/master/data/0571IbnCasakir/0571IbnCasakir.TarikhDimashq/0571IbnCasakir.TarikhDimashq.JK000916-ara1.mARkdown.

idem. *Muʿjam Shuyūkh.* Wafāʾ Taqī al-Dīn ed. (Damascus: Dār al-Bashāʾir, 1421/2000).

Judd, Steven. "Ibn ʿAsākir's Sources for the Late Umayyad Period," in Lindsay, ed., *Ibn ʿAsākir and Early Islamic History,* 78–99.

Judd, Steven and Jens Scheiner, eds. *New Perspectives on Ibn ʿAsākir* (Leiden: Brill, 2017).

Lample, Guillaume, et al. "Neural Architectures for Named Entity Recognition." *CoRR* 2016 abs/1603.01360.

Lan, W., et al. "An Empirical Study of Pre-trained Transformers for Arabic Information Extraction." In *Proceedings of The 2020 Conference on Empirical Methods on Natural Language Processing (EMNLP).* 2020

Lee, K., et al. "End-to-end Neural Coreference Resolution." In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* 2017: 188–197 Association for Computational Linguistics.

Lindsay, James E., ed. *Ibn ʿAsākir and Early Islamic History* (Princeton, NJ: Darwin Press, 2001).

Luo, X. "On Coreference Resolution Performance Metrics." In Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language *Processing 2005:* 25–32. Association for Computational Linguistics.

Mourad, Suleiman A. "Appendix A. Publication History of *TMD,*" in Lindsay, ed., *Ibn ʿAsākir and Early Islamic History,* 127–133.

Mueller, David, and Greg Durrett. "Effective Use of Context in Noisy Entity Linking." In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,* 2018: 1024–29. Brussels, Belgium: Association for Computational Linguistics. https://doi.org/10.18653/v1/D18-1126.

Nūr Sayf, Aḥmad M. "Maṣādir taʾrīkh Ibn ʿAsākir min kutub al-ḥadīth waʾl-rijāl," in ed. Wizārat al-Taʿlīm al-ʿĀlī, *al-Kalimāt waʾl-buḥūth waʾl-qaṣāʾid al-mulaqāt fīʾl-iḥtifāl bi-muʾarrikh Dimashq al-kabīr Ibn ʿAsākir* (Damascus: Wizārat al-Taʿlīm al-ʿĀlī, 1979), 475–504.

Obeid, Ossama, et al. "CAMeL Tools: An Open Source Python Toolkit, for Arabic Natural Language Processing." In *Proceedings of the Conference on Language Resources and Evaluation* (LREC 2020), 2020, Marseille.

Pradhan, S., et al. Scoring Coreference Partitions of Predicted Mentions: A Reference Implementation. *Proceedings of the conference. Association for Computational Linguistics. Meeting, 2014,* 2014: 30–35. https://doi.org/10.3115/v1/P14-2006

Raghavan, U., R. Albert, and S. Kumara. "Near linear time algorithm to detect community structures in large-scale networks." *Phys. Rev. E* 76, 2007, https://doi.org/10.1103/PhysRevE.76.036106.

Savant, Sarah Bowen. "People versus Books," In Bruce Fudge, Kambiz Ghanea-Bassiri, Christian Lange, and Savant, eds. *Non Sola Scriptura: Essays on the Qur'an and Islam in Honour of William A. Graham* (London: Routledge, 2022), 281-302. https://doi.org/10.4324/9781003252221-18.

Savant, Sarah Bowen and Masoumeh Seydi. "Dispatches from Ibn ʿAsākir." Forthcoming.

Scheiner, Jens. "Ibn ʿAsākir's Virtual Library as Reflected in his Taʾrīkh madīnat Dimashq," in Judd and Scheiner, eds. *New Perspectives on Ibn ʿAsākir,* 156–257.

Traag, V. A., L. Waltman, and N. J. van Eck. "From Louvain to Leiden: guaranteeing well-connected communities." *Sci Rep* 9, 5233 2019. https://doi.org/10.1038/s41598-019-41695-z

Vilain, Marc, et al. "A Model-Theoretic Coreference Scoring Scheme." In *Proceedings of the Sixth Message Understanding Conference (MUC-6).* 1995. https://aclanthology.org/M95-1005