Journal of Data Science 21 (3), 470–489
July 2023
DOI: 10.6339/22-JDS1038
Statistical Data Science

Scalable Community Extraction of Text Networks for Automated Grouping in Medical Databases

Tomilayo Komolafe[1], Allan Fong[2], and Srijan Sengupta[3,*]

[1]*Qlik, 211 S Gulph Rd, King of Prussia, PA 19406, USA*
[2]*MedStar Health Research Institute, Hyattsville, Maryland, USA*
[3]*Department of Statistics, North Carolina State University, Raleigh, NC, 27695, USA*

Abstract

Networks are ubiquitous in today's world. Community structure is a well-known feature of many empirical networks, and a lot of statistical methods have been developed for community detection. In this paper, we consider the problem of community extraction in text networks, which is greatly relevant in medical errors and patient safety databases. We adapt a well-known community extraction method to develop a scalable algorithm for extracting groups of similar documents in large text databases. The application of our method on a real-world patient safety report system demonstrates that the groups generated from community extraction are much more accurate than manual tagging by frontline workers.

Keywords *community structure; community extraction; natural language processing; patient safety*

1 Introduction

Many complex systems in today's world consist, at an abstract level, of *agents* who *interact* with one another. This general agent-interaction framework arises in a range of disciplines, such as biological sciences (Lynall et al., 2010), physical sciences (Huberman and Adamic, 1999; Pagani and Aiello, 2013), social media (Guo et al., 2020), and epidemiology (Leitch et al., 2019), to name a few. By denoting *agents* as *nodes* and their *interactions* as *edges*, any such system can be represented as a network. Such network data provide a versatile framework for analyzing a broad spectrum of complex systems.

Community structure is a well-known feature of many empirical networks. Nodes in a network are often found to belong to groups or communities that exhibit similar behavior. The identification of this network structure, called community detection, is an important problem in network analysis. Community detection has important scientific implications; these communities often turn out to be groups of agents which share common properties and/or play similar roles within the network. For example, in Jonsson et al. (2006), the communities in a protein interaction network turned out to be functional groups (proteins having the same or similar function) - this conclusion has important implications for cancer research. Fortunato (2010) provides a multidisciplinary exposition on community detection in networks. Fittingly, several useful tools for community detection have been developed and studied in the statistics literature. These include spectral methods (Rohe et al., 2011; Jin, 2015; Sengupta and Chen, 2015), modularity

*Corresponding author Email: ssengup2@ncsu.edu.

Received November 4, 2021; Accepted February 13, 2022

based methods (Newman and Girvan, 2004; Bickel and Chen, 2009; Sengupta and Chen, 2018), likelihood based methods (Amini et al., 2013), to name a few. Most of these methods are known to have theoretical guarantees for accuracy of community detection.

In this paper, we study text networks, where vertices represent documents and edges represent similarity between document pairs. Similarity between text documents can be measured in a number of ways based on representational learning (Mikolov et al., 2013, 2017; Hofmann, 1999; Landauer et al., 1998; Papadimitriou et al., 2000; Dumais, 2004). We provide more details on document representation in Section 3. Text networks provide a useful framework for representing large databases of documents, and statistical network analysis techniques can be applied for the analysis of such databases. However, there has not been much work on community detection of text networks, with some very recent exceptions such as Yan and Wang X (2021) and Dong et al. (2020).

In patient safety research, analysts study free-text narratives written by the front-line staff describing medical events that impact the safety of patients. Given the large size of patient safety reporting databases, it is important to combine similar reports into groups that can be analyzed together. However, this cannot be accomplished by standard clustering techniques. This is because clustering methods enforce each document to belong to some group, which is neither desirable nor practical for patient safety reports. There could be many documents which are unique, essentially forming a "miscellaneous" category whose members are not similar to each other. Rather, we want to identify groups of documents that are highly similar to each other, and separate these homogeneous groups from the "miscellaneous" group.

The starting point of this paper is in identifying that the above task is analogous to community extraction in networks, where communities of nodes are extracted from the network, while allowing for arbitrary structure in the remaining nodes (Zhao et al., 2011). In particular, the community extraction framework allows for certain nodes to not belong to any community while simultaneously facilitating the analysis of community structure in the network. Building on this, we develop a method for community extraction in text networks by integrating representational learning with a well-known community extraction method proposed by Zhao et al. (2011). Most real-world text databases are large, which can lead to high computational expense when applying community extraction. We propose a novel divide and conquer strategy to address this issue. We demonstrate our method by applying it to a large patient safety event database, where it generates much better groups than manual tagging, as measured by document similarity.

The rest of the paper is structured as follows. In Section 2, we describe the scientific application area of medical errors and patient safety events which motivated this work, and we also introduce the patient safety error database on which our method is applied. In Section 3, we describe the proposed methodology. In Section 4, we report the results of our analysis, and we conclude the paper with a short discussion in Section 5.

2 Medical Errors and Patient Safety Event Reports

The Institute of Medicine (IOM), an authority at the intersection of medicine and society, released a report titled "To Err is Human: Building a Safer Health System" in November 1999 (Kohn et al., 2000). Its goal was to break the cycle of inaction regarding medical errors by advocating a comprehensive approach to improving patient safety. Based on two studies conducted in 1984 and 1992, the IOM concluded that between 44,000 and 98,000 patients die every year in United States (U.S.) hospitals due to medical errors. Costs alone from medical errors were approximated to be $37.6 billion per year. About $17 billion were associated with preventable errors (Kohn et al., 2000). Given the intense level of public and scientific reaction to the report, various stakeholders responded swiftly to take action. In February 2000, President Clinton announced a national action plan to reduce preventable medical errors by fifty percent within five years (The White House, 2020). Congress mandated the monitoring of progress in preventing patient harm. In July 2004, a Healthgrades Quality Study asserted that IOM had in fact vastly underestimated the number of deaths due to medical errors, citing 195,000 deaths per year (Harrington, 2005).

Two decades later, medical errors continue to be a leading cause of death in the United States (Makary and Daniel, 2016). The Institute of Medicine and several state legislatures have recommended the use of patient safety event reporting systems (PSRS) to better understand and improve safety hazards (Aspden et al. (2004); Rosenthal and Booth (2005)). Numerous healthcare providers have adopted these systems which provide a framework for healthcare provider staff, including frontline clinicians, nurses, and technicians to report patient safety events, ranging from 'near misses', where no patient harm occurs, to serious safety events that result in patient harm (Clarke, 2006). However the potential of these reports to systematically identify hazards and reduce harm has been lacking, in part because of the limited techniques used to analyze these data. If the reported data can be analyzed effectively, reporting systems have the potential to dramatically improve the safety and quality of care by exposing possible weaknesses in the care process (Pronovost et al., 2008).

Patient safety event (PSE) reports are free-text narratives written by the front-line staff. These narratives describe incidents whereby a healthcare service delivery did not go as expected. During these instances, the front-line staff witnessing the incident can document his/her perspective of the events that occurred. Therefore, aggregating similar PSEs has the potential to give insights into trends of the different types of medical errors healthcare organizations encounter. There is a significant amount of variation between documents because these narratives do not have to follow any specified format. For example, documents describing similar events can vary drastically in their word usage, vocabulary, document length, and prevalence of grammatical errors. Therefore, the notion of similarity has to be based on semantic representation rather than simple features defined on the documents.

Reporters are often required to tag reports with a general and specific event type categories (i.e. 'Wrong drug', 'Drug confusion') using the taxonomy integrated into their reporting system (Chang et al., 2005; Griffey et al., 2019; Dovey et al., 2002). However, the default categories in reporting systems are challenging to interpret and differentiate especially for complex events, for example a drug confusion event can lead to the wrong drug prescribed to a patient. In addition, front-line staff who select the event categories may have limited understanding of the category nuances because these categories are often not defined for the reporter (Gong et al., 2015; Johnson, 2003). These factors contribute to a large spurious tagging of categories resulting in the need for manual review (Puthumana et al., 2021). This process requires an extensive amount of effort by the patient safety analyst and takes away from the analyst's time to identify patterns and develop solutions (Puthumana et al., 2021).

Therefore, relying on tagging with default categories in a reporting system is not reliable. The community extraction method proposed in this paper is a data driven approach to identify categories using the report narratives rather than complex and often confusing default categories. Grouping documents with similar words is a useful starting point for patient safety analysts to make sense and extract insights from the data. The shared keywords and phrases provide safety analysts a data driven approach to focus their efforts. The keywords themselves are not enough

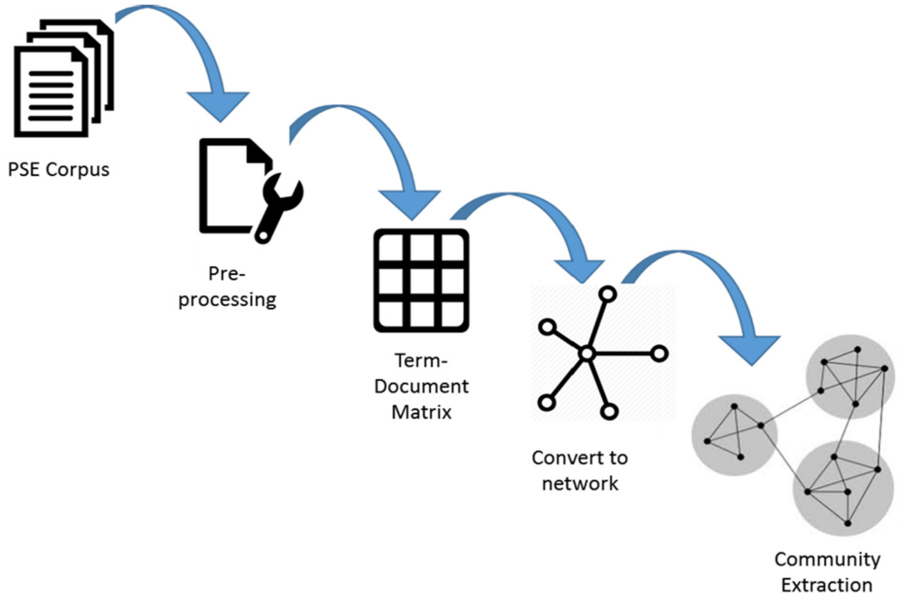

Figure 1: Framework for community extraction of PSE corpus.

to identify problems and hazards in the healthcare system but it gives the analysts a way to filter and prioritize how reports are reviewed. In this work, we consider a PSE database from MedStar Health consisting of 2,072 documents. Our goal is to develop a community extraction algorithm to find homogeneous groups of documents.

3 Methodology

In this section, we describe the process of community extraction to find homogeneous document groups in a text database. Figure 1 provides a schematic representation of the different steps involved. The subsequent subsections provide details on each step. Note that while this work is motivated by patient safety event reports, this general methodology can be applied on any text database. The first two steps (pre-processing and term-document matrix construction) are well-known strategies from natural language processing, while the last step (community extraction) is a well-known method from the statistical network analysis literature. We integrate these well-known approaches in our work.

3.1 Text Pre-Processing

A pre-processing step is critical to the performance of any natural language processing (NLP) model to reduce errors (Vijayarani et al., 2015). Pre-processing of text can be compared to exploratory data analysis in traditional statistical analysis.

In our work, we first create a manually curated dictionary of commonly misspelled words in our corpus and replace them with their proper spelling. To do this, we extracted terms that appeared in 2 or more documents and correct any misspelled terms. As PSEs typically contain information such as the date an event occurred, the time it occurred, or dosage of a particular medication, any permutation of a date, dosage, or time is replaced with the words “date”, “dose”, and “time” respectively. This is because the exact time an event occurred or the exact dosage of a medication is irrelevant for our analysis. However, we should not remove the word because then the sentence will lose its syntactic coherence. Therefore, we simply replace specific times by the general concept word *time*. In addition, special characters are removed, except for periods and all other numbers are removed from the text.

For example, this sentence: "*On Dec. 13 at 5PM resident was prescribed 2mc/mg of oxycotine*" is converted to "*On date at time resident was prescribed dose of oxycotine*"

Furthermore, to ensure that words with similar morphology are presented as the same, we carried out stemming of the words, which is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form—generally a written word form. The goal of stemming is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form, and this is a common pre-processing step in text analytics (Vijayarani et al., 2015).

3.2 Construction of Term Document Matrix

The next step is to represent the text database as a numeric matrix. This is a common approach in natural language processing, where the entire corpus is converted to a term-document matrix where rows represent terms and columns represent documents. The weighting of the terms in our term document matrix is critical to any future analysis. We use the common methodology Term Document - Inverse Document Frequency methodology referred to as "tf-idf" in the literature (Aizawa, 2003; Ramos et al., 2003). Here, *term frequency* is an adjusted version of the number of times a term appears in the document. Let t be a term and d be a document in the corpus. Then, term frequency is defined as

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}, \tag{1}$$

where $f_{t,d}$ is the raw count of a term in a document, i.e., the number of times that term t occurs in document d. Note that the denominator is the total frequency of all terms in the document, i.e., the total length of the document, which scales the raw count and allows for comparison between documents of differing length. The inverse document frequency is a measure of how much information the word provides, i.e., how common or rare the term t is across all documents in the corpus. Let D denote the set of all documents in the corpus and let $N = |D|$ be the total number of documents. Then, inverse document frequency is defined as

$$idf(t,D) = \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right), \tag{2}$$

where the denominator is the number of documents which contain the term t. Finally, the tf-idf score is calculated as

$$tf\text{-}idf(t,d) = tf(t,d) \times idf(t,D). \tag{3}$$

The tf-idf term-document matrix is constructed as follows. First, we consider the set of all unique terms that appear in the corpus. Then, for each term t and each document d, we compute the tf-idf score and populate the entries of the matrix. For a toy illustration, consider the following short patient safety event reports.

- "The patient schedule did not match the script."
- "Script and schedule mismatch. Script stated vasculab and schedule xray"
- "Xray monitor will not transmit images"

The resulting term-document matrix is displayed in Table 1.

Table 1: Toy illustration of TF-IDF matrix.

Terms	Doc1	Doc2	Doc3
did	0.264	0.000	0.000
image	0.000	0.000	0.264
match	0.264	0.000	0.000
mismatch	0.000	0.176	0.000
monitor	0.000	0.000	0.264
not	0.097	0.000	0.097
patient	0.097	0.065	0.000
schedule	0.097	0.130	0.000
script	0.097	0.130	0.000
state	0.000	0.176	0.000
schedule	0.000	0.000	0.000
transit	0.000	0.000	0.264
vasculab	0.000	0.176	0.000
will	0.000	0.000	0.264
xray	0.000	0.065	0.097

3.3 Text Network Construction via Latent Semantic Analysis

Once we have a weighted term document matrix, we apply the well-known technique of Latent Semantic Analysis (LSA) for dimension reduction (Turney, 2001; Dumais, 2004). LSA has the ability to handle obstacles prevalent in natural language processing and analysis such as presence of synonyms and polysemy. In what follows, we provide only a brief description of LSA. For a more detailed description of the approach, see Landauer et al. (1998).

For a term-document matrix X of m terms and n documents with rank r, its singular value decomposition (SVD) can be written as

$$X = T\Sigma D^T, \tag{4}$$

where X is the m x n term-document matrix, T is a m x m matrix whose columns are the orthogonal eigenvectors of XX^T where we denote X^T as the transpose of the matrix X. The matrix D is a n x n matrix whose columns are the orthogonal eigenvectors of X^TX and Σ is a m x n diagonal matrix whose diagonals are $\sqrt{\lambda_i}$ where λ corresponds to the eigenvalues of XX^T and $1 \leqslant i \leqslant r$ and 0 everywhere else. The eigenvalues of XX^T are the same as the eigenvalues of X^TX. The values $\sqrt{\lambda_i}$ are called the singular values of X.

The implementation of LSA used in this work is a low rank approximation of the SVD. For this, we find a positive integer, $k \leqslant r$ such that it closely approximates the term document matrix. The value k is selected such that it minimizes the error between the original matrix X and its low rank approximation X_k. This is achieved through the following steps; Since $\lambda_i \geqslant \lambda_{i+1}$, setting $\lambda_{i+1} = 0$ if it is close to zero will not significantly affect the original matrix X. We therefore find a k where $1 \leqslant k \leqslant r$ such that it minimizes the difference in the Frobenius norm between X and X_k. If $k = r$, then the difference in the Frobenius norm is 0 but if $k \ll r$, we have a low rank approximation of our matrix that is also easy to manipulate. By keeping only the k columns or entries for each of our matrices, we obtain X_k and furthermore a low rank approximation of both terms and documents. Therefore, we have

$$X_k = T_k \Sigma_k D_k^T \tag{5}$$

Where we only keep the k columns of matrix T so T_k is a m x k matrix, D^T so D_k^T is a k x n matrix and Σ_k is a diagonal k X k matrix. Then, the rows of the matrix D_k are the LSA-based vector representations of the documents in the corpus. In our application, we used the R package *lsa* to implement this method (Günther et al., 2015), where the appropriate dimension k is chosen by a thresholding technique on the singular values (Wild et al., 2005).

Finally, we generate a network of documents by creating a similarity matrix from the matrix D_k. For this purpose, we have used Pearson correlation between LSA embeddings as the measure of similarity between documents. We define the similarity between two documents d_i and d_j as the correlation between the corresponding rows of D_k, resulting in a n x n correlation matrix. The correlation matrix serves as our adjacency matrix for the next step of community extraction. We note that this is a weighted adjacency matrix.

3.4 Community Extraction for Text Networks

Most community detection methods aim to partition a network into communities with the goal of maximizing the number of edges within communities and minimizing edges between communities. This framework assumes that all nodes belong to some community. However, there could be scenarios where some nodes do not belong to any particular community and forcing these nodes into a community will distort the community detection results. For example, let's assume we have a network of high school students where links between students signifies that these students participate in similar extra-curricular activities. Applying some of the traditional community detection algorithms to this network will result in unsatisfactory results. This is because some students naturally do not participate in any extra-curricular activity and therefore do not belong to a community. However, these community detection algorithms will force these nodes to one of the formed communities.

The text networks from PSE databases also have this property. We expect that the majority of PSE reports will fall into groups, but there could be some "miscellaneous" documents that do not belong to any group. Community detection methods that partition all nodes into communities are going to enforce such "miscellaneous" reports into groups, which is unwarranted. Therefore, we use the community extraction method, proposed by Zhao et al. (2011), which can handle these types of networks.

We describe a network graph G as composed of vertices V and edges E, and $G = (V, E)$. The total number of vertices in a network graph G gives us the network size N. That is, $N = |V|$. Also the number of edges in a network graph is M, where $M = |E|$. We consider only non-overlapping communities in this paper, therefore once community extraction is applied to a network G, the partition results in two distinct sets, V_1 and V_2 where $V_1 \cap V_2 = \emptyset$ and $V_1 \cup V_2 = V$. A network can also be represented as an N x N adjacency matrix referred to as $\mathbf{A}$, where its elements are $\mathbf{A}_{ij}$ and $i, j = 1, 2, \ldots, N$, $\mathbf{A}_{ij} = (-1, 1)$ making it a weighted network. For text networks, the adjacency matrix $\mathbf{A}$ is equal to the correlation matrix of $D_k T$ from the preceding subsection. Communities are extracted one at a time with the criterion of extracting a set of nodes with the sum of its weights largest within that set and smallest between the set and its complement (Zhao et al., 2011). We will call this set of extracted nodes S, and its complement, S^c. The objective function we are therefore maximizing in each iteration step is given by

$$\tilde{W}(S) = |S||S^c| \left[\frac{O(S)}{|S|^2} - \frac{B(S)}{|S||S^c|} \right], \tag{6}$$

where $O(S) = \sum_{i,j \in S} A_{ij}$ and $B(S) = \sum_{i \in S, j \in S^c} A_{ij}$. The term $O(S)$ is twice the weight of the edges within S and $B(S)$ represents the weights from the set S to the rest of the remaining network. In large sparse networks, particularly as in our application, a small community S could result in a large $\tilde{W}(S)$ value, the term $|S||S^c|$ serves to ensure that sufficiently sized communities are extracted at each step as very large communities or very small communities will be penalized. This is because the term, $|S||S^c|$ is maximized at $|S| = \frac{N}{2}$.

To maximize the objective function, we implement the *tabu* search maximization technique which is a local optimization technique based on label switching (Beasley, 1998; Glover and Laguna, 1998). In this optimization technique, a string of binary values representing nodes in either community S or S^c is passed to the *tabu* search function (Zhao et al., 2011; Beasley, 1998; Glover and Laguna, 1998). The function tracks which nodes have been switched, ensuring that they are not switched again until a certain number of iterations have passed, making these nodes, "*tabu*". To guard against being trapped at a local maxima, the algorithm is run with random label assignments each time.

In our implementation, the community extraction algorithm is repeated till only a small subset of nodes, 30 nodes or less, are left in the network and this was sufficient for our application. Zhao et al. (2011) proposed a stopping criteria only for a network that can be represented by the block model. Future works will investigate a more appropriate stopping criteria.

3.4.1 Scalability via Divide and Conquer Approach

In practice, we observed the community extraction method is computationally infeasible for large networks. Running one iteration of the *tabu* search algorithm on the entire corpus of 2,072 documents takes over 120 hours. One alternative is to use the divide and conquer strategy. Suppose that the full network has n nodes, and we randomly partition the set of nodes into P roughly equal sized subsets. We apply the community extraction method of Zhao et al. (2011) on each subnetwork of approximately n/P nodes. Consider the p^{th} such subnetwork, where $1 \leqslant p \leqslant P$ and suppose that K_p communities are extracted from this network. We applied the divide and conquer strategy to the PSE corpus with $P = 10$ and $P = 5$, which brought down the runtime to 22 hours and 44 hours respectively.

Partitioning the entire network will also result in some communities being arbitrarily split up. Therefore, we also developed a methodology for combining similar communities from different partitions. Next, we would like to check whether a pair of communities from different partitions should be aggregated into a single community. To this end, define $V_{a,p}$ to be the set of nodes assigned to an extracted community, where p refers to the partition number, and $1 \leqslant a \leqslant K_p$ is the community number in that partition. Consider two such sets of indices, namely, V_{a_1,p_1} and V_{a_2,p_2}, where $p_1 \neq p_2$. We compute the within-community edge densities for the two sets, and compare that with the across-community edge density between the two sets. If the across-community edge density is sufficiently high when compared to the within-community edge densities, we aggregate the two communities. For illustration, suppose we want to check whether community 1 in the first partition should be combined with community 4 in the second partition. Here, we have $a_1 = 1$, $a_2 = 4$, $p_1 = 1$, $p_2 = 2$. We compute the within-community edge densities, denoted by $D_{a,p}$, and the across-community edge density, denoted by $D_{(a_1,p_1),(a_2,p_2)}$, as,

$$D_{1,1} = \frac{1}{|S_{1,1}|^2} \sum_{i,j \in S_{1,1}} A_{ij}, \;\; D_{4,2} = \frac{1}{|S_{4,2}|^2} \sum_{i,j \in S_{4,2}} A_{ij}, \;\; D_{(1,1),(4,2)} = \frac{1}{|S_{1,1}| * |S_{4,2}|} \sum_{i \in S_{1,1}, j \in S_{4,2}} A_{ij}. \quad (7)$$

We aggregate the communities if $D_{(1,1)(4,2)} > \alpha \max(D_{1,1}, D_{4,2})$. The constant α is a heuristic choice which represent the trade-off between false positives and false negatives — if α is set too high then legitimate pairs might not get selected for aggregation, and on the other hand, if α is set too low then too many community pairs might get selected for aggregation. In our work, we used $\alpha = 0.85$ which a heuristic choice to balance these two factors.

4 Empirical Results

In this section, we report the results from applying the methodology proposed in Section 3 on the MedStar PSE corpus of 2,072 documents.

4.1 Benchmark Results from Manual Tagging

First, we establish a reference method for benchmarking. These PSE reports are manually tagged by the front-line staff with options available from a drop-down menu. Tags include both a general event description, and there are 20 options to select from in our report, and 187 specific event descriptions which are sub-categories of any one of the general event descriptions. If the tags are descriptive enough, then we would expect the diagonals of the correlation matrices, representing average correlation within a group, to be high, and conversely, the off diagonals to be low. This would suggest that front-line staff are tagging similar documents with similar tags. However, if the correlation matrices do not follow this pattern, then it suggests that the tags available to the front-line staff are not descriptive enough for each report type.

The benchmark results from manual tagging are displayed in Figure 2 as a heatmap. Clearly, manual tagging fails to obtain high correlation within groups and low correlation between groups.

Besides the visual illustrations, we can also look at statistics of the correlation matrices obtained by manual tagging. Here, document groups are created as per manual tags, and we compute within-group and between-group correlations using the LSA embedding described in Section 3.3. Specifically, are there communities or tags whereby the documents within the community are more related to another set of documents in another community or tag. We do this by looking at the percentage of off diagonal cells that have a value equal to or greater than the value of the cell in the diagonal for a given column in the correlation matrix. Some examples of manually tagged categories that are more similar to other categories than within themselves are below.

- “Medication”: more related with “Fluid-Outdated” and “Unusable Medication”
- “Equipment”: more related with “Medical Device-Sterilization” and “Cleanliness Issue”
- “Diagnostic Imaging-Test - Wrong Side (L vs. R)”: more related with “Blood Bank-Patient Testing (Blood Bank Use Only)”, “Diagnostic Imaging-Image - Misidentified”, and “Diagnostic Imaging-Test - Test Delayed”

4.2 Results from Community Extraction

Next, we applied our methodology described in Section 3 to obtain automated tags via community extraction. Recall that implementing the method on the full network of 2,072 documents

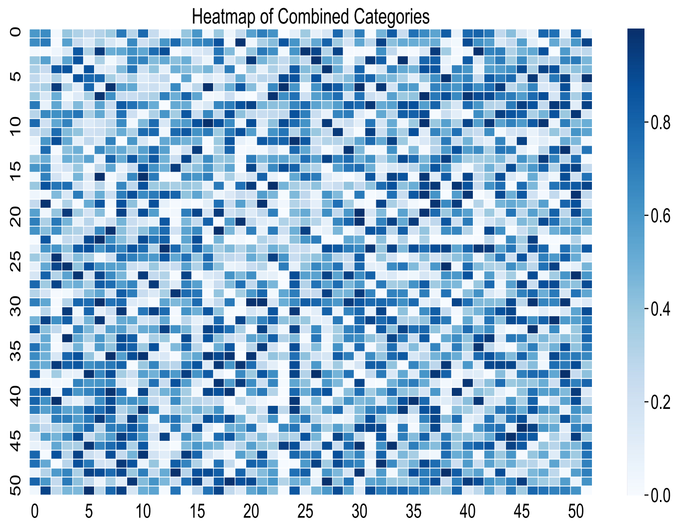

Figure 2: Heatmap of groups generated by manual tagging from 52 X 52 combined event types, i.e., a General event tag paired with its Specific event.

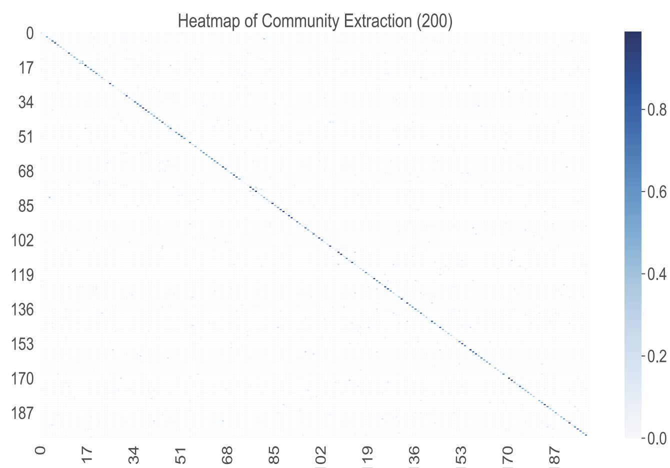

Figure 3: Heatmap of communities generated from correlation matrix of documents that fall into the respective communities after community extraction is applied. Partitions of 200 documents with threshold 0.2 and 113 communities.

is computationally very intensive, and therefore we applied the divide and conquer approach described at the end of Section 3. We used subnetworks of 200 and used a correlation cut-off of 0.15 to stop the network from getting too dense. The results are plotted in Figure 3. Note that the community extraction method does not require pre-specification of the number of communities, rather, the number of communities is an output of the method. From the correlation heatmap, it is clear that the documents have very high within-group correlation (values close to 1) and very low between-group correlation (values close to 0), which indicates that the grouping is effective. This is a substantial improvement over manual tagging (Figure 2). Note that the results from community extraction are better than manual tagging across the range of tuning parameters, i.e., subgraph size and correlation threshold.

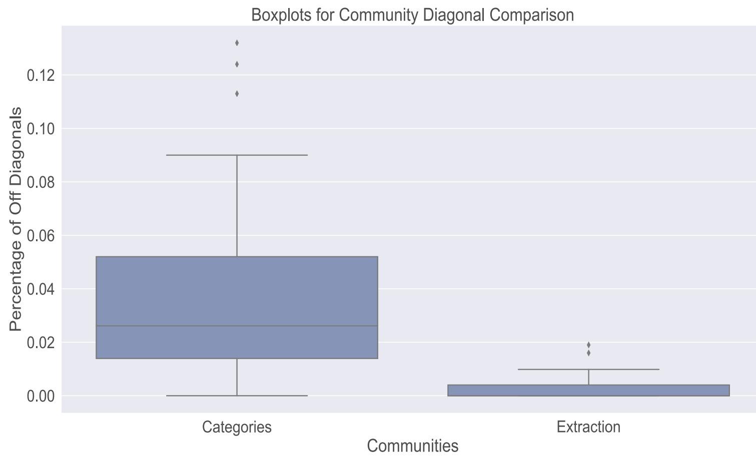

Figure 4: Comparisons of communities between the predefined PSE categories/tag against community extraction by looking at the distribution of percentage of communities that are more similar, higher correlation score, than documents within that community.

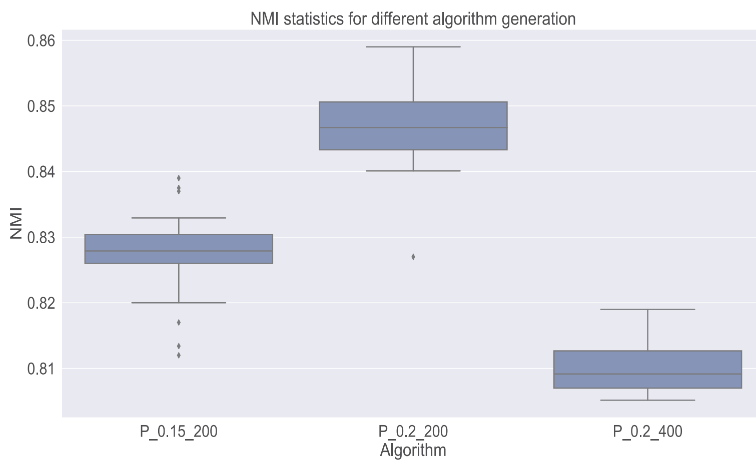

Figure 5: Normal Mutual Information (NMI) statistics for comparing the relatedness of communities extracted for the different permutations of partition size and correlation matrix threshold.

Next, recall that we observed "heterophilic" behavior with manual tagging, where documents in some groups have higher between-group correlation than within-group correlation. To compare manual tagging vs community extraction with respect to this property, we looked at each group, and computed what fraction of other groups have higher between-group correlation than within-group correlation. The boxplots are shown in Figure 4, where we compare manual tagging to a representative community extraction. We observe that the groups from community extraction have very little "heterophilic" behavior compared to manual tagging.

Finally, recall that our divide and conquer strategy involves random partitioning of the large text network into a number of smaller subnetworks. A natural question is "How stable are the groupings generated due to random partitioning? To answer this question, we implemented several random iterations of the divide and conquer strategy, and computed the Normalized Mutual Information (NMI) for document groups arising in different iterations. A high value of

NMI indicates high stability of document grouping across random iterations. The results are plotted in Figure 5 for several tuning parameter values. We observe that the NMI values are quite high indicating stability of clustering.

5 Discussion

This analysis demonstrates the advantage of a network driven approach to extract communities in patient safety event free-text. The results clearly show categories with less overlapping categories compared to the categories manually selected by the front-line staff. It is likely that the analysis can identify communities of reports or themes in the reports that are not dependent on the structured categories. For example, communication and hand-off are often prevalent themes in patient safety reports that are not typically captured in structured fields. Structured fields are predefined and often difficult or time consuming to change and update. A network driven approach that leverages the free-text is more flexible and can identify more timely hazards with changing environments and care processes. Having a flexible approach is particularly important as new workflows are being introduced (e.g., COVID-19 protocols, telehealth).

A network driven approach to identify communities and themes in free-text can help reduce the burden of reporters from choosing through complex taxonomies which are both time consuming and can result in errors. In addition, these results highlight the potential to identify communities of related reports that might be missed from analyzing just the structured categories. Such categorization flexibility could greatly help safety analysts and safety leaders better identify meaningful signals and insights from all the data.

Next, we discuss some limitations of this work. This analysis was performed on data from one healthcare system. As a result, the comparison of extracted communities with the structured categories are specific to the structured categories implemented at the healthcare system. It is possible that other healthcare systems use different categorization taxonomies highlighting the need to understand the generalizability of this approach across taxonomies and healthcare systems. In addition, the present method does not consider temporal effects on communities. Expanding this approach to include temporally stable communities or emerging communities would be important especially as changes to policy, workflow, safety hazards can often occur.

Appendix

Table 2: List of General Event Tags from PSE.

	General Event Types
1	Airway Management
2	Blood Bank
3	Diagnosis/Treatment
4	Diagnostic Imaging
5	Equipment/Medical Device
6	Facilities
7	Fall

(continued on next page)

Table 2: Continued.

	General Event Types
8	Healthcare IT
9	Infection Prevention
10	Lab/Specimen
11	Lines/Tubes/Drain
12	Maternal/Childbirth
13	Medication/Fluid
14	Miscellaneous
15	Patient ID/Documentation/Consent
16	Professional Conduct
17	Restraints/Seclusion Injury
18	Safety/Security
19	Skin/Tissue
20	Surgery/Procedure

Table 3: List of Specific Event Types from PSE.

	Specific Event Types
1	Abandonment
2	Abrasion
3	Abuse/Assault (Physical)
4	Abuse/Assault (Verbal)
5	Administration Technique Incorrect
6	Adverse Drug Reaction
7	Adverse Reaction (Non Med)
8	Air Quality/Odor/Smoke/Fumes
9	Airway Mgmt Equipment Issue
10	Airway Obstructed
11	Apgar Score < 5 at 5 min
12	Armband Issue
13	Bed Malfunction
14	Birth Trauma
15	Blister
16	Break in Sterile Technique
17	Broken Item
18	Bruise
19	Burn
20	Cardiac and/or Respiratory Arrest Requiring ACLS Intervention
21	Cardiac or Circulatory Event
22	Cardiopulmonary Arrest Outside of ICU Setting
23	Circulation Impeded

(continued on next page)

Table 3: Continued.

	Specific Event Types
24	Collection Issue
25	Combination or Interaction of Device Defect and Use Error
26	Communication
27	Complications of Anesthesia
28	Complications of Surgery/Procedure
29	Consent Issue
30	Contamination
31	Contrast/Radiopharmaceutical - Allergic Reaction
32	Contrast/Radiopharmaceutical - Event
33	Contrast/Radiopharmaceutical - Extravisation
34	Count Issue
35	Date of Birth Issue
36	Delay/Difficulty With Resuscitation
37	Delivery Without Provider
38	Diagnosis - Delayed
39	Diagnosis - Missed
40	Diagnosis Issue
41	Diaper Dermatitis
42	Dietary Issue
43	Disconnected
44	Discontinued
45	Discontinued Incorrectly
46	Dislodgement
47	Disorderly Person
48	Disrupted Utility (Electric/Water/HVAC/Med Gas)
49	Documentation Error
50	Documentation Issue
51	Dose/Concentration Incorrect
52	Drug Incorrect
53	Drug Interaction/Incompatibility
54	Drug Preparation/Labeling Issue
55	Drug With Known Allergy
56	Duplicate Therapy
57	Elevator Malfunction
58	Elopement
59	Equipment - Faulty
60	Equipment - Not Available
61	Equipment - Wrong/Inappropriate
62	Equipment (Blood Bank Use Only)
63	Equipment/Device Function
64	Exposure - Prolonged Fluro Time

(continued on next page)

Table 3: Continued.

	Specific Event Types
65	Extubation - Unplanned
66	Extubation Issue - Self
67	Failure to Assess Patient
68	Failure to Follow Order
69	Failure to Respond to Request for Service
70	Fetal pH <7.05 Cord Blood Gas
71	Foreign Object Retained Post Procedure
72	Friction/Shear
73	From Bed
74	From Bed - Over Rails
75	From Chair
76	From Exam Stool
77	From Exam/Operating Table
78	From Stretcher
79	From Therapy Equipment
80	From Toilet/Commode
81	From Wheelchair
82	Hand Hygiene Compliance Issue
83	Hardware Failure or Problem
84	Illegible Order
85	Image - Misidentified
86	Implant Issue
87	Inadequate Supplies
88	Inappropriate Admission
89	Inappropriate Discharge
90	Inconsiderate/Rude/Hostile/Inappropriate Behaviors
91	Infiltration Event
92	Infiltration/Extravasation
93	Intimidation/Verbal Abuse
94	Intubation - Unplanned
95	Isolation - Failure to Follow Protocol
96	Labeling Issue
97	Laceration
98	Lack of Responsiveness
99	Left Against Medical Advice
100	Left Without Being Seen
101	Line Not Changed
102	Lost Specimen
103	Medication Administered Not Ordered
104	Monitoring Issue
105	MRI Safety Issue

(continued on next page)

Table 3: Continued.

	Specific Event Types
106	Narcotic Count Incorrect
107	Network Failure or Problem
108	Non Head Injury - Restraint Related
109	Noncompliant/Uncooperative/Obstructive Behaviors
110	Not Activating the Chain of Command
111	Occlusion
112	Omission
113	Ordering Issue
114	Other (please specify)
115	Outdated/Unusable Medication
116	Patient Exposure - Blood/Body Fluid
117	Patient Testing (Blood Bank Use Only)
118	Personal/Associate Property Lost/Theft
119	Phlebitis
120	Post-Partum Hemorrhage
121	Preparation Incorrect
122	Prescriptions Not Given at Discharge
123	Pressure Ulcer
124	Procedure Issue
125	Process Issue
126	Product Administration (Clinical Services)
127	Product Receipt/Handling (Blood Bank Use Only)
128	Product Test Request (Clinical Services)
129	Property Damage/Vandalism
130	Pump Programming Issue
131	Radiation Onclogy Issues
132	Referral Issue
133	Reporting Issue
134	Requisition Incorrect
135	Respiratory Mgmt - Inappropriate
136	Restraint Improperly Applied
137	Restraints Applied - Not Ordered
138	Restraints Ordered - Not Applied
139	Results - Delay in Critical Results Communication
140	Results - Posted to Wrong Patient
141	Risky/Reckless/Dangerous Behaviors
142	Route Incorrect
143	Sample
144	Self Injury
145	Shoulder Dystocia
146	Site Infection

(continued on next page)

Table 3: Continued.

	Specific Event Types
147	Skin Tear
148	Slip/Trip/Fall
149	Smoking
150	Specimen Acceptability Issue
151	Specimen Processing Issue
152	Sterilization/Cleanliness Issue
153	Storage Incorrect
154	Suicide/Suicide AttemptSuspicious Package
155	Test - Incorrectly Performed
156	Test - Ordered, Not Performed
157	Test - Test Delayed
158	Test - Wrong Side (L vs. R)
159	Testing Issue
160	Time/Date Incorrect/Delayed
161	Tissue
162	Treatment - Delayed
163	Treatment - Inappropriate
164	Treatment - Incorrectly Performed
165	Treatment - No Order for
166	Unable to Access
167	Unauthorized Access/Trespassing
168	Unauthorized Drugs
169	Unauthorized Weapons on Premises
170	Unexpected Return to the OR
171	Unexpected Software Design Issue
172	Unexpected Transfer to ICU/NICU
173	Unknown/Found on Floor
174	Use Error
175	Visitor Policy Issue
176	Water Leak/Flood
177	Weapons on Premises
178	While Ambulating
179	While Held by Staff
180	While Running/Playing
181	While Standing
182	While Transferring
183	Workplace Violence
184	Wound
185	Wrong Body Part (Site/Side/Level)
186	Wrong Insertion Location
187	Wrong Patient

Acknowledgement

We thank the Edtitor, the Associate Editor and two anonymous reviewers for their constructive suggestions, which were really helpful towards the improvement of the manuscript.

Supplementary Material

Supplementary material online include R code for implementing the proposed method.

Funding

We acknowledge the support from the NIH R01 grant 1R01LM013309 from the National Library of Medicine.

References

Aizawa A (2003). An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1): 45–65.

Amini AA, Chen A, Bickel PJ, Levina E (2013). Pseudo-likelihood methods for community detection in large sparse networks. *Ann. Statist.*, 41(4): 2097–2122.

Aspden P, Corrigan JM, Wolcott J, Erickson SM, et al. (2004). Patient safety reporting systems and applications. In: *Patient Safety: Achieving a New Standard for Care*. National Academies Press (US).

Beasley JE (1998). Heuristic algorithms for the unconstrained binary quadratic programming problem, *Technical report, Citeseer*.

Bickel PJ, Chen A (2009). A nonparametric view of network models and Newman–Girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106: 21068–21073.

Chang A, Schyve PM, Croteau RJ, O'Leary DS, Loeb JM (2005). The JCAHO patient safety event taxonomy: a standardized terminology and classification schema for near misses and adverse events. *International Journal for Quality in Health Care*, 17(2): 95–105.

Clarke JR (2006). How a system for reporting medical errors can and cannot improve patient safety. *The American Surgeon*, 72(11): 1088–1091.

Dong R, Yang J, Chen Y (2020). Overlapping community detection in weighted temporal text networks. *IEEE Access*, 8: 58118–58129.

Dovey S, Meyers D, Phillips R, Green L, Fryer G, Galliher J, et al. (2002). A preliminary taxonomy of medical errors in family practice. *BMJ Quality & Safety*, 11(3): 233–238.

Dumais ST (2004). Latent semantic analysis. *Annual review of information science and technology*, 38(1): 188–230.

Fortunato S (2010). Community detection in graphs. *Physics Reports*, 486(3): 75–174.

Glover F, Laguna M (1998). Tabu search. In: *Handbook of Combinatorial Optimization*, 2093–2229. Springer.

Gong Y, Song HY, Wu X, Hua L (2015). Identifying barriers and benefits of patient safety event reporting toward user-centered design. *Safety in Health*, 1(1): 1–9.

Griffey RT, Schneider RM, Todorov AA, Yaeger L, Sharp BR, Vrablik MC, et al. (2019). Critical review, development, and testing of a taxonomy for adverse events and near misses in the emergency department. *Academic Emergency Medicine*, 26(6): 670–679.

Günther F, Dudschig C, Kaup B (2015). Lsafun: An r package for computations based on latent semantic analysis. *Behavior Research Methods*, 47(4): 930–944.

Guo Z, Cho JH, Chen R, Sengupta S, Hong M, Mitra T (2020). Online social deception and its countermeasures: A survey. *IEEE Access*, 9: 1770–1806.

Harrington MM (2005). Revisiting medical error: Five years after the iom report, have reporting systems made a measurable difference. *Health Matrix*, 15: 329.

Hofmann T (1999). Probabilistic latent semantic indexing. In: *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 50–57.

Huberman BA, Adamic LA (1999). Internet: growth dynamics of the World-Wide Web. *Nature*, 401: 131.

Jin J (2015). Fast community detection by SCORE. *The Annals of Statistics*, 43(1): 57–89.

Johnson C (2003). How will we get the data and what will we do with it then? issues in the reporting of adverse healthcare events. *BMJ Quality & Safety*, 12(suppl): ii64–ii67.

Jonsson PF, Cavanna T, Zicha D, Bates PA (2006). Cluster analysis of networks generated through homology: automatic identification of important protein communities involved in cancer metastasis. *BMC Bioinformatics*, 7(1): 2.

Kohn LT, Corrigan J, Donaldson MS (2000). *To err is Human: Building a Safer Health System.* National Academy Press.

Landauer TK, Foltz PW, Laham D (1998). An introduction to latent semantic analysis. *Discourse processes*, 25(2–3): 259–284.

Leitch J, Alexander KA, Sengupta S (2019). Toward epidemic thresholds on temporal networks: a review and open questions. *Applied Network Science*, 4(1): 105.

Lynall ME, Bassett DS, Kerwin R, McKenna PJ, Kitzbichler M, Muller U, et al. (2010). Functional connectivity and brain networks in schizophrenia. *Journal of Neuroscience*, 30(28): 9477–9487.

Makary MA Daniel M (2016). Medical error—the third leading cause of death in the US. *Bmj*, 353.

Mikolov T, Chen K, Corrado G, Dean J (2013). Efficient estimation of word representations in vector space. arXiv preprint: https://arxiv.org/abs/1301.3781.

Mikolov T, Grave E, Bojanowski P, Puhrsch C, Joulin A (2017). Advances in pre-training distributed word representations. arXiv preprint: https://arxiv.org/abs/1712.09405

Newman MEJ, Girvan M (2004). Finding and evaluating community structure in networks. *Physical Review E*, 69(2): 026113.

Pagani GA, Aiello M (2013). The power grid as a complex network: A survey. *Physica A: Statistical Mechanics and its Applications*, 392(11): 2688–2700.

Papadimitriou CH, Raghavan P, Tamaki H, Vempala S (2000). Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, 61(2): 217–235.

Pronovost PJ, Morlock LL, Sexton JB, Miller MR, Holzmueller CG, Thompson DA, et al. (2008). *Improving the value of patient safety reporting systems Advances in Patient Safety: New Directions and Alternative Approaches (Vol. 1: Assessment).*

Puthumana JS, Fong A, Blumenthal J, Ratwani RM (2021). Making patient safety event data actionable: understanding patient safety analyst needs. *Journal of Patient Safety*, 17(6): e509–e514.

Ramos J, et al. (2003). Using tf-idf to determine word relevance in document queries. In: *Proceedings of the First Instructional Conference on Machine Learning*, volume 242, 29–48. Citeseer.

Rohe K, Chatterjee S, Yu B (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, 39(4): 1878–1915.

Rosenthal J, Booth M (2005). *Maximizing the use of State Adverse Event Data to Improve Patient Safety*. National Academy for State Health Policy, Portland, ME.

Sengupta S, Chen Y (2015). Spectral clustering in heterogeneous networks. *Statistica Sinica*, 25: 1081–1106.

Sengupta S, Chen Y (2018). A block model for node popularity in networks with community structure. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(2): 365–386.

The White House (2020). *Clinton-gore administration announces new actions to improve patient safety and assure health care quality.* https://clintonwhitehouse4.archives.gov/textonly/WH/New/html/20000222_1.html.

Turney PD (2001). Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In: *European Conference on Machine Learning*, 491–502. Springer.

Vijayarani S, Ilamathi MJ, Nithya M, et al. (2015). Preprocessing techniques for text mining-an overview. *International Journal of Computer Science & Communication Networks*, 5(1): 7–16.

Wild F, Stahl C, Stermsek G, Neumann G (2005). Parameters driving effectiveness of automated essay scoring with lsa.

Yan S, Jia Y Wang X, (2021). Overlapping community detection in temporal text networks.

Zhao Y, Levina E, Zhu J (2011). Community extraction for social networks. *Proceedings of the National Academy of Sciences*, 108(18): 7321–7326.