UTRECHT UNIVERSITY

Department of Information and Computing Science

Applied Data Science master thesis

Enhancing Information Retrieval through the Identification and Mapping of Dynamic Communities within Textual Data

First examiner
Dr. Ioana Karnstedt-Hulpus

Second examiner
Prof. dr. Lynda Hardman

Candidate
Bart J. Hiemstra

In cooperation with
Auditdienst Rijk

July 24, 2023

Abstract

In the digital age, the exponential increase in available data presents both an opportunity and a challenge, particularly in data-rich fields such as auditing. For example, the Auditdienst Rijk, the Dutch central government's audit service, grapples with substantial volumes of textual data.

To address this data challenge, we propose the use of a framework combining natural language processing, graph theory, and relevance measures. Instead of solely focusing on traditional document extraction, our approach is built around the identification and mapping of dynamic communities within a textual dataset.

This framework includes the extraction and analysis of semantic entities, tracing these entities over time, and employing a semantic search system. Viewing documents as interconnected elements within broader community contexts allows for a more efficient and precise retrieval, offering a significant enhancement over existing strategies.

Although our work is directly applied to the realm of auditing, the methodology within our framework can provide valuable insights to any field dealing with voluminous unstructured textual data. Consequently, our research contributes to fostering a culture of efficiency and precision in data handling, enhancing our capacity to navigate and leverage the data-rich digital landscape across various sectors.

Results show a moderate relevance for found communities and documents. While there are certain limitations, the framework's potential is acknowledged, and it provides a foundation for future work in leveraging community-based approaches in large-scale text data analysis.

Contents

Appendices

1. Introduction

1.1 Background and research question

With the age of digitization comes an exponential growth of available information [1]. The field of auditing is no exception to this trend, and for audit companies, the process of effectively extracting relevant information from large sets of data has become both an opportunity and a challenge [2]. One such organization dealing with big amounts of data is the Auditdienst Rijk (ADR), the internal audit service of the Dutch Government.

The ADR, being part of the Ministry of Finance, is responsible for a range of duties both financial and non-financial. Examples of tasks include ensuring the correctness of financial statements by checking accounting records and reports, assessing the security and functionality of IT systems, and reviewing the effectiveness of government policies [3]. With most types of audits comes a vast array of documents, making the task of information retrieval an important factor of the ADR's operations. As such, alongside the increase of digital information, the organization has developed an increasing interest in utilizing machine learning techniques to deal with the data.

One of those techniques is the extraction of communities from vast corpora of textual data. This method forms an important element of a larger operational strategy that seeks to understand 'Who did What, When' within a given dataset, ranging from external projects such as the discussion about gas extraction in Groningen, to internal communication (e.g. all e-mail conversations that occurred within the organisation). Previously, the organization utilized Named Entity Recognition (NER) techniques for extracting information from their large-scale textual data. NER focuses on identifying and classifying named entities, like individuals, organizations, or locations, within a text. However, while NER can give some insight into entities associated with certain topics, the insight it provides on its own can be insufficient when applied to lots of documents spread over time, as it can be difficult to keep track of which entity appears where.

Given the more advanced possibilities that NER as a foundation can provide, such as community detection and tracking, there is room to improve the process. These techniques not only deepen our understanding of the entities within a dataset, but also uncover their interactions, thereby leading to the formation of 'communities' of related entities. Moreover, tracking these communities over time may provide insight in the temporal evolution of the different relationships. Furthermore, providing a way to retrieve relevant documents based on the identified relevant communities may help with more efficiently extracting the right information.

Therefore, the research question for this study is: How can the identification and mapping of dynamic communities within a textual dataset improve the efficiency of information retrieval for organizations requiring extensive document analysis?

By investigating a potential solution to a challenge in auditing organizations like the ADR, this study aims to explore the possibility and effectivity of utilizing latent communities within textual data to enhance information retrieval. Furthermore, while the focus lies on applications performed by auditing companies, the proposed methodology should work with any kind of textual data containing entities.

1.2 Preliminary research

As discussed, the goal of this research is to explore the use of a framework that enhances the efficiency and effectiveness of information retrieval processes for organizations requiring extensive document analysis, specifically auditing companies, with a focus on analysis of latent entity relationships. To ensure that the solution aligns with the needs of these organizations, we first conducted a series of interviews with relevant stakeholders: data scientists working at the Auditdienst Rijk, hereafter referred to as participants. This approach allowed us to gain in-depth information about their existing practices, limitations, and possible improvements.

Our preliminary research was conducted using semi-structured interviews. This method balanced our need for specific information while also letting participants share their thoughts freely. This flexibility helped us dig into different parts of the participants' experiences. As a result, we gained insights into their current approach, the problems they face, and the ADR's view on possible improvements.

Current Practices and challenges

Participants revealed that entity tagging and topic modeling are central in their current approach. Topic modeling is used iteratively until a specific subtopic is reached. Entity tagging is utilized to identify and visualize entities within documents for the mentioned subtopics. The names of found entities are shown in a long list, along with their number of occurrences.

Participants' responses shed light on the main challenge associated with their current practices: limited insight. The existing method of visualizing entities as a lengthy list, although beneficial to some extent, lacks support for insight of changes over time. They expressed their awareness of the potential benefits of temporal analysis. Furthermore, the use of a query system was brought up, where groups of entities could come up not through the iterative use of topic-modeling, but by providing certain keywords. Ideally, documents belonging to these groups of entities are ordered on relevance.

Insights and research direction

The preliminary research conducted via semi-structured interviews has highlighted several aspects:

- **Central Role of Entity Tagging and Topic Modeling:** These methods form the essence of the existing information retrieval processes. It is therefore important that our framework further builds upon these existing capabilities.
- **Need for Temporal Analysis:** Experts identified a lack of temporal analysis capabilities as a significant shortcoming in their current processes. Thus, our framework must include temporal analysis as a fundamental component.
- **Shortcomings in Current Visualization Methods:** Presenting entities as a long list along with their occurrences may offer some insights but lacks comprehensive understanding. Visualization methods that are more organized, intuitive, and efficient would offer significant benefits.
- **Desire for document ranking:** Experts discussed the possibility for search queries to return a ranking of documents based on relevance, aiding them in prioritizing their document analysis process.

Based on these insights, we will focus our research on developing a framework to enhance the efficiency and effectiveness of information retrieval in document analysis. Our primary goal is to overcome the limitations of the current methods while introducing new functionalities that experts have identified as necessary.

1.3 Literature review

The concept of tracking communities and their evolution has been an area of interest in many fields, among which sociological and network studies. A community, being a subset of network nodes within which connections are dense, but between which connections are less dense [4], can be useful in distinguishing the different groups of nodes - or, in the case of social networks, entities - that are present in a network graph.

While static social network graphs and the communities therein can give valuable insight in finding latent groups of people, the connections and even the presence of people themselves can change over time. In a 2019 survey, Dakiche et al.[5] describe these dynamic networks as having 'interactions represented by a time-series of static networks, each network corresponding to interactions aggregated over a small time period, such as a day or an hour'. In the survey, several different methods for tracking communities within a temporal network are outlined. Once a method has been implemented, the identified set of communities can be analyzed to study their evolution and transformation over time.

A different approach to dealing with a large set of communities is community search (CS). Fang et al.[6] in a 2019 survey present an overview of CS and present it as a high-efficiency method that works well with large graphs, both static and temporal. They discuss the approach as "given a vertex q of a graph G, it aims to find a community, which contains q and satisfies the properties: (1) connectivity, i.e., vertices in the community are connected; and (2) cohesiveness, i.e., vertices in the community are intensively linked to each other w.r.t. a particular goodness metric." The reason for this approach being labeled 'high-efficiency' is that it only looks for communities based on the queried vertices, instead of performing community detection on all graphs in advance.

In a 2016 paper by Fang, Cheng, Luo, and Hu[7] a specific type of community search is proposed, where communities are found through groups of vertices that share their relevance to a certain query, more specifically a specified set of keywords. Again, the advantage here being that communities do not have to be detected for all data beforehand.

However, as participants pointed out during the preliminary research, the auditing organization would like to gain a comprehensive understanding of the temporal network's evolution and the underlying community dynamics. While the community search querying method by Fang et al. is faster and efficient, it only provides insights into communities centered around the queried vertices. This approach may miss out on discovering latent or evolving communities that are not directly connected to the query vertices at a given time point. On the other hand, the snapshot-based dynamic community detection method considers the entire network's structure at different time intervals, allowing for the detection of communities that might emerge, dissolve, or transform over time, even if they are not explicitly queried.

By adopting a hybrid approach that combines snapshot-based dynamic community detection with querying, the auditing organization can benefit from a balanced perspective. The dynamic community detection part provides a broader view of the network's temporal evolution, allowing us to explore trends and patterns in community formation and dissolution over time. Simultaneously, the community search with specified keywords offers targeted insights into communities related to specific entities of interest. This combination ensures a more comprehensive analysis, capturing both global and local changes within the temporal network, thus providing a richer understanding of the network's dynamics and community structure.

The proposed framework, therefore, aims to strike a balance between the broad scope and temporal tracking of snapshot-based community tracking, and the precise, query-based search of keyword-based CS. By doing so, it aims to enhance the practicality and efficiency of community detection in large networks, offering an approach that can cater to the specific needs of organizations seeking to extract relevant information from complex network data.

2. Methodology

Our proposed framework consists of three main stages: detecting and visualizing communities within temporal network graphs; finding the most relevant communities based on a given search query; and identifying the most interesting documents for those communities. Each of these stages consists of smaller components, and each requires careful consideration. The stages, the challenges they come with and the chosen approaches are discussed in the following sections. An overview of the framework can be found in Appendix A.

2.1 Detecting communities within temporal network graphs

As part of this research is aimed at finding a solution that works on any given corpus of documents, the proposed framework functions without making use of meta-data, i.e. all data should come from the text in the documents. Therefore as a first step we keep the foundation used by the ADR by making use use of named entity recognition. Recognized entity types can, among others, be names of people, organizations, locations, numeric expressions including dates and times. NER typically involves two main steps: tokenization, where the text is split into individual words or tokens, followed by entity recognition, where each token is analyzed and classified into predetermined categories if it matches the criteria. Using machine learning algorithms, these models are trained on large datasets to recognize a variety of entities with an often high precision[8].

Temporal network graphs

The entities found in each document in the corpus can be represented as vertices in a network graph. These graphs, denoted by G, store the state of a network by modeling the connections, or edges, between all vertices, such that $G = (V, E)$, with V representing the set of vertices and E denoting the set of edges [5].

A connection between two entities does not exist naturally in the available data. Therefore, we operationalize the concept of 'connection' as two entities co-occurring, i.e. appearing in the same document. Entities can appear together in different documents; more frequent co-occurrences indicate stronger relationships. Therefore, edges are assigned a weight based on the amount of co-occurrence, where a higher co-occurrence leads to a higher weight. The assigned connection weights helps determining the borders of communities when applying community detection. As the connections do not have a direction, the used graphs are undirected.

Community detection and matching

With the set of temporal network graphs G established, community detection can be performed to identify communities: subsets of vertices within which connections are dense, but between which connections are sparse [4]. These communities signify the groups of entities that have a higher co-occurrence and are more likely to be associated with each other.

2.2 Querying and community relevance

In our research methodology, after identifying the communities within the temporal network graphs, the next step involves enabling users to access relevant communities based on topic-keywords. To achieve this, we utilize a querying system, allowing the user to specify the topics that relevant communities may revolve around.

Semantic Search

As one community can encompass many different textual documents spanning different topics, retrieving relevant communities based on a small set of keywords can be a challenging task. While one possibility is to allow queries using exact keyword matching, the textual data in large corpora can often be more effectively understood and retrieved through the application of semantic search techniques, which focus on understanding the user's intent and the contextual meanings of terms to produce more accurate results[9]. This approach will also prevent spelling mistakes to exclude otherwise relevant communities.

To implement Semantic Search, the textual data in a corpus needs to be transformed into a form that could be interpreted and analyzed semantically by machines. This transformation is achieved through the use of word embeddings: high-dimensional vectors capturing the semantic properties of words. Words with similar meanings are positioned close together in this high-dimensional space[10].

Following our operationalization of ′connection′, we interpret shared document presence as indicative of a contextual relationship between two entities, represented as nodes in a network graph, G. Each document thus effectively acts as an edge, linking two nodes and encapsulating a shared semantic space between them.

Each identifiable community, C, within the network is associated with a specific sub-corpus. This sub-corpus comprises all documents that represent edges where the nodes belonging to the community co-occur. Through the aggregation of these documents, we are able to capture the specific context and semantic relationships that are unique to each community.

2.3 Within-community document relevance

Once the semantic search is complete, and the relevant communities are identified, the next stage in our research methodology involves assessing the relevance of the documents within these communities to the user′s query. This process, if well-executed, can potentially decrease the time users would otherwise spend examining numerous documents.

The determination of an effective method for document relevance extraction depends on several parameters. This includes the characteristics of the dataset, the complexity of the user′s query, and the desired level of detail in the results. Given that datasets used by auditing companies often comprise a multitude of diverse documents, the queries can be highly varied.

To achieve our goal of surfacing the most relevant documents, we′ll use a method that values the prominence of terms within each document and their prevalence across the entire corpus. This approach highlights significant terms within a document while maintaining its unique context. Users will receive relevant documents that carry distinctive semantic richness, enhancing their understanding.

Term Frequency-Inverse Document Frequency (TF-IDF) and Latent Semantic Indexing (LSI) are two viable options for document relevance assessment. TF-IDF, when adapted to this study, assigns weights to documents based on their frequencies within communities relative to their prevalence across all communities. This technique effectively emphasizes documents that are important within a community while downplaying common documents, making it well-suited for preserving unique context and promoting semantic richness. On the other hand, LSI, when adapted to this study, utilizes singular value decomposition (SVD) to unveil latent semantic relationships between documents and communities, facilitating the identification of documents with shared thematic content[11].

3. Experiment and evaluation

To evaluate our proposed framework, we set up an experiment and conduct a user study to measure its performance.

3.1 Data description and preparation

For our study, we made use of documents associated with the *Tweede Kamer* (i.e. the Dutch House of Representatives), which are publicly available through their API [12]. Document types vary from being transcripts of debates, letters to the Speaker of the House, and infographics (e.g. about Covid-19 policies). This dataset was suggested by the Auditdienst Rijk for being both rich in textual data, and being a publicly available source, ensuring that no complications with obtaining clearance or permissions could arise.

We created a Python script to request batches of documents from the API, after which the attached textual files were downloaded for each document in the batch. These files varied in their data format (e.g. .pdf, .doc), and so could not easily be parsed. To solve this issue, we used a Python port of Apache Tika, a unified-parser framework for content analysis and detection [13]. Using this tool, textual data could be extracted regardless of the file extension.

Thre resulting dataset contains an extensive set of columns with meta-data, including document id, date, title, organisation (e.g. 'Tweede Kamer' or 'Eerste Kamer'), content type of the attached document (.e.g. PDF), and API version. One aspect in the design of this framework is to ensure its flexibility and adaptability to a multitude of corpora types. Thus, corpus-specific meta-data is excluded from the setup. The remaining columns are as follows:

id (string): Unique identifier of the document;

date (datetime): Date of origin;

text (string): Textual contents of the document.

The resulting dataset consists of 539,419 rows, of which 58,836 contain no value for the column 'text'. After removing these, a dataset of 473,225 rows remains,

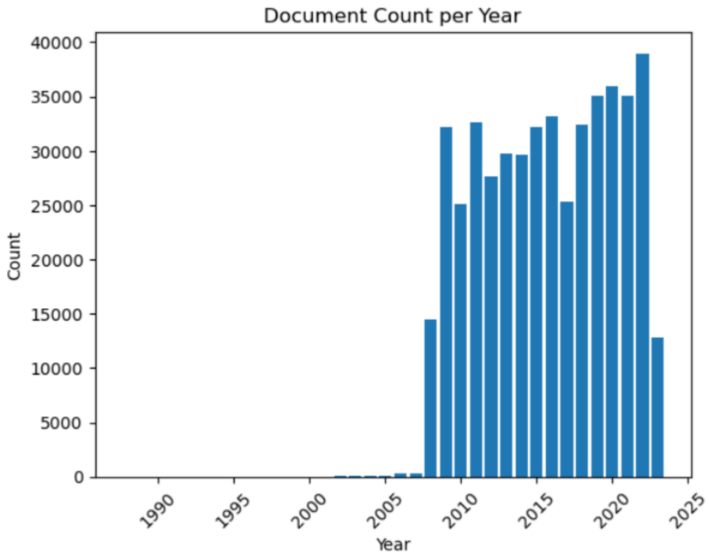


Figure 3.1: Distribution of documents relating to the Dutch House of Representatives, 1988 - 2022. The distribution exhibits a negative skew, with the amount of available data being significantly higher in more recent years.

spanning from the second half of 1988 to the first quarter of 2023. The document distribution is not uniform and exhibits a heavy negative skew, as can be seen in figure 3.1. As performing entity tagging on the full dataset would cost a significant amount of time, we took a random and uniformly distributed sample of 120,000 rows, spanning ten years, from 2012 through 2022.

3.2 Experimental setup

Finding entities

In the implementation of our study, we utilize the Named Entity Recognition capabilities of the spaCy library. This Python library employs rule-based tokenization of text, which provides us with a set of tokens. A machine-learning model trained on a large corpus then reviews each token to determine whether it is an entity and if so, its entity type [14]. The library has support for Dutch corpora, and while several better-performing transformer-based libraries are available, such as RobBert[15] and Flair-ner-dutch[16], spaCy was ultimately chosen due to its faster speed, while remaining sufficient performance for this study [8]. We used its largest model `nl_core_news_lg`.

Creating temporal network graphs

We use the Python library NetworkX to create snapshots of the aforementioned graphs with a bin of one month, so that we get a set of snapshots $G = (G_0, \ldots, G_t)$ where G_i represents a single snapshot of all entities in that graph, along with their edges and edge weight, for month i. The snapshot window of one month was chosen to strike a balance between too many fragmented communities that could result from a smaller window, and overly large, potentially less interpretable communities that could form with a larger window.

Community assignment

The survey on community evolution tracking by Dakiche et al.[5], discussed in the literature review section, distinguishes four different ways of tracking communities throughout a temporal network:

- **Independent Community Detection and Matching:** These are methods that first detect communities at each time step and then match them across different time-steps.
- **Dependent Community Detection:** These include methods that detect communities at time t based on the topology of the graph at t and on previously found community structures.
- **Simultaneous Community Detection on All Snapshots:** These are methods that first construct a single graph by adding edges between instances of nodes in different time-steps, and then run a classic community detection on this graph.
- **Dynamic Community Detection on Temporal Networks:** These include methods that do not detect communities from scratch at each time but instead update the ones previously found according to network modifications.

Due to the nature of our data, where entities often appear only once, or disappear only to reappear much later in the time series, the network graphs do not see any smooth transitions in between the time frames. Therefore, we opt to make use of independent community detection & matching, where community detection is first applied to all snapshots G, after which communities get matched based on their overlap in entities. One advantage is that this approach works with any chosen community detection method[5], making the framework as accessible as

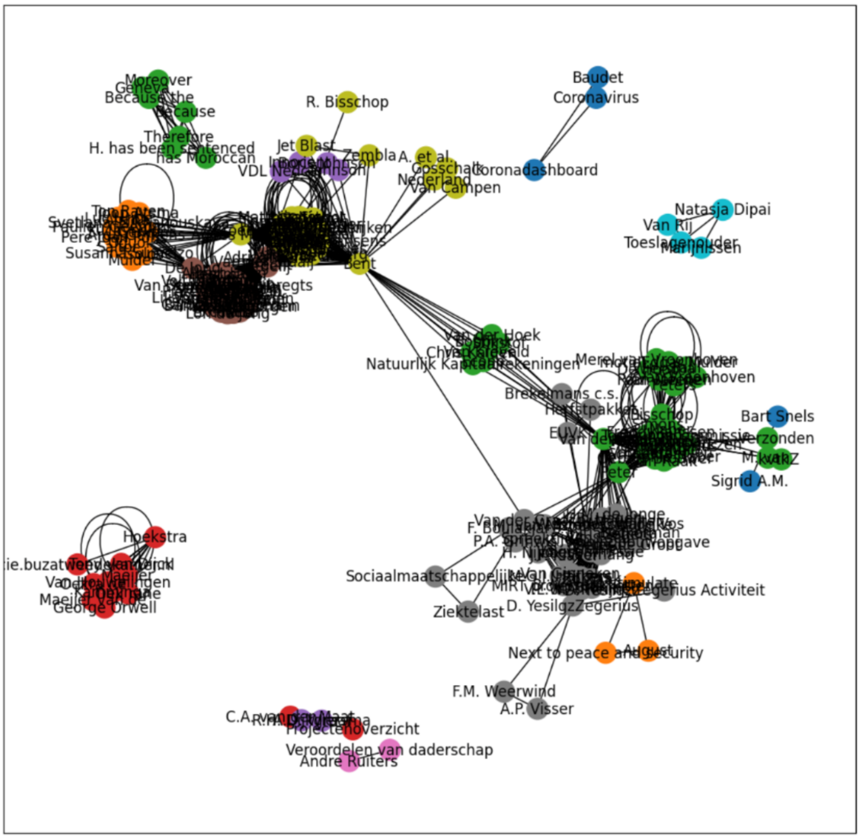


Figure 3.2: Network graph using co-occurence of entities in the dataset from the Tweede kamer as edges. Communities are assigned through the Louvain algorithm, and are visualized using different colors.

possible.

For community assignment, we make use of the Louvain method for community detection. This method by Blondel et al.[17] is known for its ability to handle large networks efficiently, utilizing its greedy optimization technique: it seeks to maximize the modularity of the network, essentially creating communities with stronger intra-connectivity and weaker inter-connectivity. The Louvain method works iteratively, first assigning a community to each node. Then, for each node, it evaluates the gain in modularity that would come from removing it from its community and placing it in a neighboring community. The result of applying the Louvain method is a set of communities for each snapshot G_i, denoted by $C = (C_{i0}, \ldots, C_{ik})$ where C_{ij} represents community j for snapshot i. The communities C_{ij} consist of entities and their connections at month i. An example of a network graph with Louvain community detection applied can be seen in figure 3.2.

We then perform community matching to deal with the communities that are present in consecutive snapshots, by determining their level of entity overlap. We keep a threshold of 0.5; when two communities in adjacent snapshots exhibit a considerable overlap of 50% or more, we consider them equivalent and assign them the same label. If more matching communities follow, e.g. $C_{1,5} = C_{2,10} = C_{3,50}$, all matching communities get named after the last matched community. This approach works on exact matches, so any inconsistencies in entity named may not count towards overlap. On our data, the method still manages to recognize around 20% of the communities as overlapping.

Text embedding and search

Among the techniques employed for generating word embeddings, Word2Vec[10] emerged as one of the first methods. It offered a robust approach to capture semantic relations between words by mapping them into a high-dimensional vector space. However, despite their success, Word2Vec and other similar methods have inherent limitations. One significant shortcoming is their inability to capture the context of words in a sentence effectively. These methods provide a single, static representation for each word, ignoring the fact that the meaning and significance of words can change based on their usage in different contexts.

The emergence of transformer models has effectively addressed this limitation. Equipped with an attention mechanism, these models weigh the significance of different words in a given context when generating word embeddings, thereby providing superior handling of context and sequence in textual data. The attention mechanism assigns different 'attention', or importance, to words in a context, reflecting their contribution to the overall meaning [18].

To illustrate, consider the term "infrastructure" in documents from the government. Depending on context, it could refer to either transport networks or digital systems. Traditional methods like Word2Vec would generate the same embedding for "infrastructure" regardless of its context. However, transformer models, leveraging their attention mechanism, create unique embeddings that accurately capture the distinct semantic value of "infrastructure" in each scenario. This context-aware representation significantly enhances the precision and relevance of information retrieval, making for more accurate search.

To capture the semantic representation of the documents belonging to each community, we then use the `all-MiniLM-L6-v2` transformer model. This is a type of sentence-BERT model, introduced by Reimers & Gurevych in 2019[19], that maps text to 384-dimensional vector space. The sentence transformer model was trained by comparing pairs and triplets of sentences and updating weights so that the resulting embeddings are semantically meaningful[20]. Then, when applying the model to new batches of text, the composition of these texts define their embeddings, and semantically-similar texts will produce similar embeddings.

In our Semantic Search approach, the key parameter is the value of 'k,' determining the number of top-k results to be returned for each user query. We set 'k' to five to strike a balance between providing sufficient and relevant information without overwhelming the user with an excessive number of results.

After retrieving the initial top-k results, we employed the Pagerank algorithm to discover five additional communities within the search results, the goal being to see if closely-positioned community nodes may be relevant or bring fresh context complementary to the query-related communities. Pagerank, developed by Larry Page and Sergey Brin[21], is an algorithm created for ranking webpages, viewing them as nodes in a network connected by edges representing links. It models a random walk that moves from node to node by following connections between those nodes. These communities represent clusters of documents that are highly interconnected and likely to be thematically cohesive, thereby providing the user with additional context and alternative perspectives on the query topic.

To be able to visualize the communities, their attached topics (determined through the ADR's prior methodology of topic assignment) and the flow of entiies between these communities, we opt to use a Sankey diagram[22]. The resulting visualization can be seen in figure 3.1.

Within-community document relevance

Once the set of relevant communities is identified, the next stage in our research methodology involves assessing the relevance of the documents within these communities to the user's query. This process, if well-executed, can potentially decrease the time users would otherwise spend examining numerous documents.

In the context of our research involving a dataset of heterogeneous documents within a community, and considering complex queries spanning across all communities, we have adopted the Term Frequency-Inverse Document Frequency (TF-IDF) method for Document Relevance Assessment. Our decision to employ TF-IDF stems from our prior experience with this technique in natural language process-

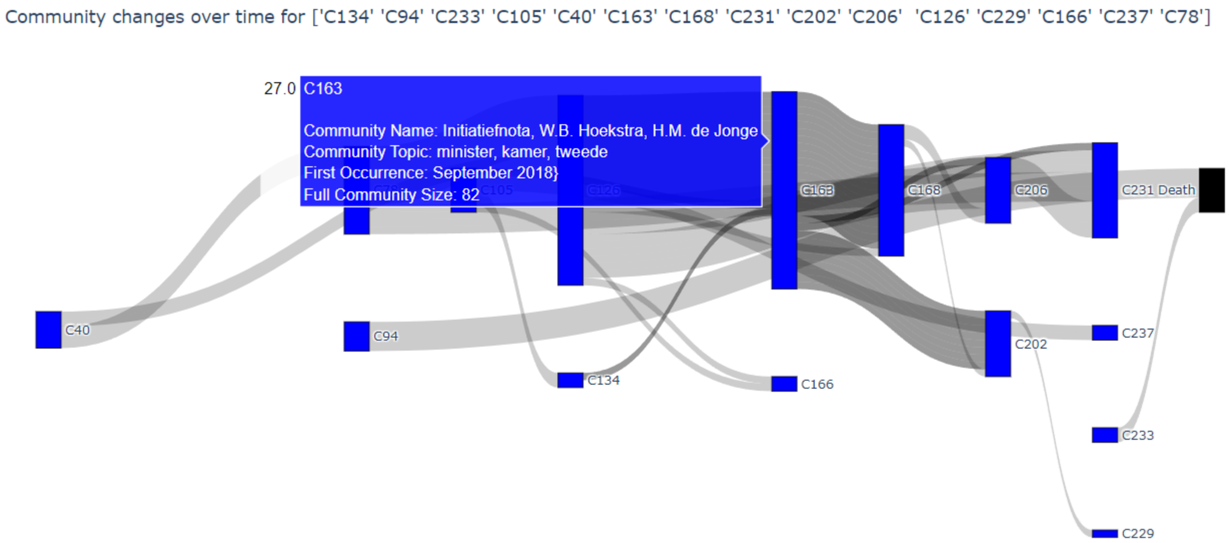


Figure 3.3: Example of a Sankey diagram for visualization of communities, their top members, associated topics, and entity flow (through the grey connections). Communities are ordered chronologically, with the earlier community appearing at the leftmost side.

ing. With TF-IDF, each word's value is determined by its frequency within a specific document (Term Frequency or TF) and is contrasted against its occurrence frequency in the entire corpus of documents (Inverse Document Frequency or IDF). This approach emphasizes the importance of words that are common across the corpus but rare within individual documents, enabling us to effectively highlight their significance within the context of the community's documents.

3.3 User study

Since the relevance of communities and their associated documents to certain search queries is a qualitative matter, we cannot use quantitative measures to evaluate the performance of the implementation of the proposed framework. Instead, we conducted a user study, where we asked experts working at the ADR to interpret and score the output of the model applied on specific search queries. The purpose of this study was is to assess the relevance and utility of the communities and associated documents identified by our model in response to specific search queries.

Each expert in the study (n=3) was given a predetermined set of search queries. Each query resulted in 10 identified communities, making a total of 20 communities for the study. Associated with each community was a collection of documents. The results for each search query were presented to the experts in an Excel file, facilitating easy navigation through the data. As different experts had different domains of expertise, different queries were used for some of them. To ensure some level of measurement reliability, two of the three experts were given the same set of queries.

The experts were requested to evaluate the model's results based on the following criteria:

- **Relevance of the community to the query:** This metric evaluated how closely an identified community related to the search query. Scoring was done on a scale of 1-10, where 1 indicated no relevance and 10 indicated high relevance.
- **Relevance of the documents to the community:** This metric assessed how well the associated documents corresponded to the identified community. Scoring was also done on a scale of 1-10, with 1 representing no relevance and 10 indicating high relevance.

Furthermore, participants where asked these qualitative questions to determine how useful the current visualization is:

- **Visualization:** How useful did you find the presented way of displaying communities and entity flow?
- **Other:** Is there any feature that you found missing

After the study, scores were collected and analyzed. An average score was calculated for each community's relevance to the query and the documents' relevance to the community. This gave us a clear insight into the effectiveness and accuracy of our proposed framework.

3.4 Results

The average ratings of all results were moderate to low. The document relevance scores, determined by embeddings, tend to be higher than the scores determined by PageRank. This discrepancy might be explained by the different mechanisms of these methodologies. The embeddings capture semantic information of the documents based on the text they contain, and thus can more accurately reflect their relevance to a search query. On the other hand, the PageRank approach operates on the assumption that the relevance of a document can be inferred from its importance within a network of documents, which is not always the case.

Average	Community relevance (embeddings)	Community relevance (pagerank)	Document relevance (embeddings)	Document relevance (pagerank)
Query 1 ('Fraude')	5.5	2.7	8.1	3.4
Query 2 ('Toeslagenaffaire')	2.1	1.7	5.2	1.7
Query 3 ('Algoritme')	2.8	2	3.2	3.4
Query 4 ('Tegemoetkoming vaste lasten')	2.6	4.8	3.2	4.8

Table 3.1: Relevance scores for different queries. Query 1 and Query 2 are averages over multiple participants (n=2) The full results can be found in Appendix B.

In contrast, the community relevance scores were generally lower. This suggests that while the proposed framework can, to some extant, identify relevant documents within a community, it may struggle to determine the relevance of entire communities to a given query.

It is also clear that certain queries perform better in terms of community relevance. For instance, the more specific the query, like 'Tegemoetkoming vaste lasten', the higher the community relevance score in both embeddings and pagerank methodologies. This could be due to the precise natsure of such queries, making it easier to form tight-knit communities with higher intra-connectivity. However, in the case of broader or more ambiguous queries, like 'Fraude', the scores are lower.

Participants found that the chosen technique of visualization is useful, but also hard to interpret from the start. Participants all asked how they should interpret the diagram, and hovering over certain communities to see details such as their most prominent member entities proved slightly uneasy to perform. A common suggestion is to replace the abstract current community names, such as 'C10492', with the community name, i.e. the members with the highest degree centrality.

Another interesting find was that a small number of documents were classified, with black-out names. This did not help with determining the relevance of the documents to the community, but it provided for interesting insights.

3.5 Limitations

The current research has a few limitations. Firstly, the user study was only conducted on three participants, of which one had his own set of search queries. Furthermore, due to the time-consuming nature of assessing relevance of all communities and documents associated with a query, the amount of different queries used was limited. This setup restricts the diversity of perspectives and may not provide a comprehensive understanding of the system's performance. Additionally, the limited sample size also increases the likelihood of bias and reduces the generalizability of our results.

Secondly, there are some potential scalability and performance limitations: The processing time of entity recognition was quite long for the subset of the corpus. Entity tagging, text embedding, will increase linearly with an increase in corpus size. Since both these time-consuming methods have to be executed only once, however, it's more about the ability to manage a large up-front computational load. After this initial investment of resources, the framework becomes quick to use for subsequent community identifications, searches, and document relevance assessments.

And, while our framework leverages the powerful capabilities of transformer models for creating context-aware text embeddings, we have only used a specific transformer model, `all-MiniLM-L6-v2`, which is a sentence-transformer made for use on sentences or paragraphs. While we used an increase max token count, this will very likely have limited the relevance search to apply only on the first pages of a community's documents.

Lastly, an inherent limiitation of our approach is the use of artifically-assigned communities. These communities come from connections based on entity co-occurrence, which do not necessarily reflect real-life connections. As documents are grouped on the artificial community that they are part of, a perfect method of relevance may never be reached.

4. Conclusion

4.1 Discussion

The implications of our research are promising for organizations that have to handle extensive document analysis as part of their core operations. The ability to dynamically identify and map communities within a textual dataset means these organizations can potentially better understand what hidden community structures lie there and what entities are of interest.

For example, governmental organizations, research institutes, or large corporations often handle massive volumes of reports, articles, and other document types. The ability to identify communities and track their evolution over time can help these organizations spot emerging trends, track the development of specific topics, and gain an overall better understanding of their information landscape.

However, the current approach sees lots of room for improvement. As seen in the results, communities were not particularly rated as having a high relevance to the queries. This might be due to a mixture of limitations, among which are some of the implemented techniques, such as the use of a sentence transformer for document-sized texts.

In response to the research question - "How can the identification and mapping of dynamic communities within a textual dataset improve the efficiency of information retrieval for organizations requiring extensive document analysis?" - we believe our approach provides a feasible framework to enhance information retrieval efficiency. However, the full potential of the approach can only be realized with further research and technical improvements. Thus, while the findings are encouraging, they signal the beginning rather than the endpoint, staying open for continued exploration and refinement.

Our study also raises some ethical considerations. One is the potential for undue exposure or undue significance placed on certain entities. The process of community detection and mapping based on entity overlap could inadvertently highlight entities that are merely incidentally mentioned or involved in the documents, potentially misconstruing their role or influence within the identified communities. Furthermore, our framework's potential application in a wide array of fields also brings up questions about the ethical implications in each specific context. For example, in the context of law enforcement or surveillance, the focus on entities might be considered undesirable.

4.2 Future work

Our research provides a foundation for community detection, document relevance assessment, and information retrieval in large-scale, heterogeneous document collections. The identified limitations and areas of potential improvement pave the way for promising avenues of future exploration and development.

Currently, our approach leverages traditional TF-IDF for document relevance assessment. However, TF-IDF doesn't adapt to the specific features of individual documents or communities. Future research could explore adaptable variants of TF-IDF that take into account document-specific and community-specific characteristics. This could involve customizing TF-IDF to better handle varying document lengths, different entity types, and other unique characteristics of each community's documents.

Furthermore, the current approach to community detection and matching doesn't fully account for more complex transitions like the merging of multiple communities into one or the splitting of a single community into multiple ones. We can view them, but we do not make use of them in the determination of any similarly relevant communtiies. Future work could explore methods that can handle these more complex community transitions, providing a more complete and accurate picture of community evolution, to serve as a complementary aid to the current way of finding relevant communities.

Lastly, the transformer model used in our research, `all-MiniLM-L6-v2`, has proven effective for generating text embeddings, it's primarily designed for sentence-level embeddings. Future work could explore transformer models specifically designed for full document embeddings. One such model is the Longformer model [23], which uses a 'sliding window' self-attention mechanism, allowing for the efficient processing of tens of thousands of tokens, such as in large documents.

Bibliography

[1] J. Gantz and D. Reinsel, “Extracting value from chaos,” 2011. [Online]. Available: https://www.yumpu.com/en/document/view/3703408/extracting-value-from-chaos-emc.

[2] G. Salijeni, A. Samsonova-Taddei, and S. Turley, “Big data and changes in audit technology: Contemplating a research agenda,” *Accounting and Business Research*, vol. 49, no. 1, pp. 95–119, Apr. 2018. DOI: 10.1080/00014788.2018.1459458. [Online]. Available: https://doi.org/10.1080/00014788.2018.1459458.

[3] “Evaluatie auditdienst rijk,” p. 6, Feb. 2023. [Online]. Available: https://open.overheid.nl/documenten/ronl-21d0e78010506621b4d67bb0f9967e227f29b6c2/pdf.

[4] M. Girvan and M. E. J. Newman, “Community structure in social and biological networks,” *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002. DOI: 10.1073/pnas.122653799. [Online]. Available: https://doi.org/10.1073/pnas.122653799.

[5] N. Dakiche, F. B.-S. Tayeb, Y. Slimani, and K. Benatchba, “Tracking community evolution in social networks: A survey,” *Information Processing &amp Management*, vol. 56, no. 3, pp. 1084–1102, May 2019. DOI: 10.1016/j.ipm.2018.03.005. [Online]. Available: https://doi.org/10.1016/j.ipm.2018.03.005.

[6] Y. Fang, X. Huang, L. Qin, *et al.*, *A survey of community search over big graphs*, 2019. DOI: 10.48550/ARXIV.1904.12539. [Online]. Available: https://arxiv.org/abs/1904.12539.

[7] Y. Fang, R. Cheng, S. Luo, and J. Hu, “Effective community search for large attributed graphs,” *Proceedings of the VLDB Endowment*, vol. 9, no. 12, pp. 1233–1244, Aug. 2016. DOI: 10.14778/2994509.2994538. [Online]. Available: https://doi.org/10.14778/2994509.2994538.

[8] S. Vychegzhanin and E. Kotelnikov, “Comparison of named entity recognition tools applied to news articles,” in *2019 Ivannikov Ispras Open Conference (ISPRAS)*, IEEE, Dec. 2019. DOI: 10.1109/ispras47671.2019.00017. [Online]. Available: https://doi.org/10.1109/ispras47671.2019.00017.

[9] H. Bast, B. Buchhold, and E. Haussmann, “Semantic search on text and knowledge bases,” *Foundations and Trends® in Information Retrieval*, vol. 10, no. 1, pp. 119–271, 2016. DOI: 10.1561/1500000032. [Online]. Available: https://doi.org/10.1561/1500000032.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of word representations in vector space*, 2013. DOI: 10.48550/ARXIV.1301.3781. [Online]. Available: https://arxiv.org/abs/1301.3781.

[11] B. Rosario, “Latent semantic indexing: An overview,” *Techn. rep. INFOSYS*, vol. 240, pp. 1–16, 2000.

[12] “Open data portaal,” [Online]. Available: https://opendata.tweedekamer.nl/over.

[13] C. A. Mattmann and J. L. Zitting, *Tika in Action*. Manning, Nov. 2011, p. 22, ISBN: 9781935182856.

[14] M. Honnibal, "Introducing spacy," Feb. 2015. [Online]. Available: `https://explosion.ai/blog/introducing-spacy`.

[15] P. Delobelle, T. Winters, and B. Berendt, *Robbert: A dutch roberta-based language model*, 2020. DOI: `10.48550/ARXIV.2001.06286`. [Online]. Available: `https://arxiv.org/abs/2001.06286`.

[16] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf, "FLAIR: An easy-to-use framework for state-of-the-art NLP," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 54–59. DOI: `10.18653/v1/N19-4010`. [Online]. Available: `https://aclanthology.org/N19-4010`.

[17] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," Jul. 2008. DOI: `10.48550/ARXIV.0803.0476`. [Online]. Available: `https://arxiv.org/abs/0803.0476`.

[18] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, 2017. DOI: `10.48550/ARXIV.1706.03762`. [Online]. Available: `https://arxiv.org/abs/1706.03762`.

[19] N. Reimers and I. Gurevych, *Sentence-bert: Sentence embeddings using siamese bert-networks*, 2019. DOI: `10.48550/ARXIV.1908.10084`. [Online]. Available: `https://arxiv.org/abs/1908.10084`.

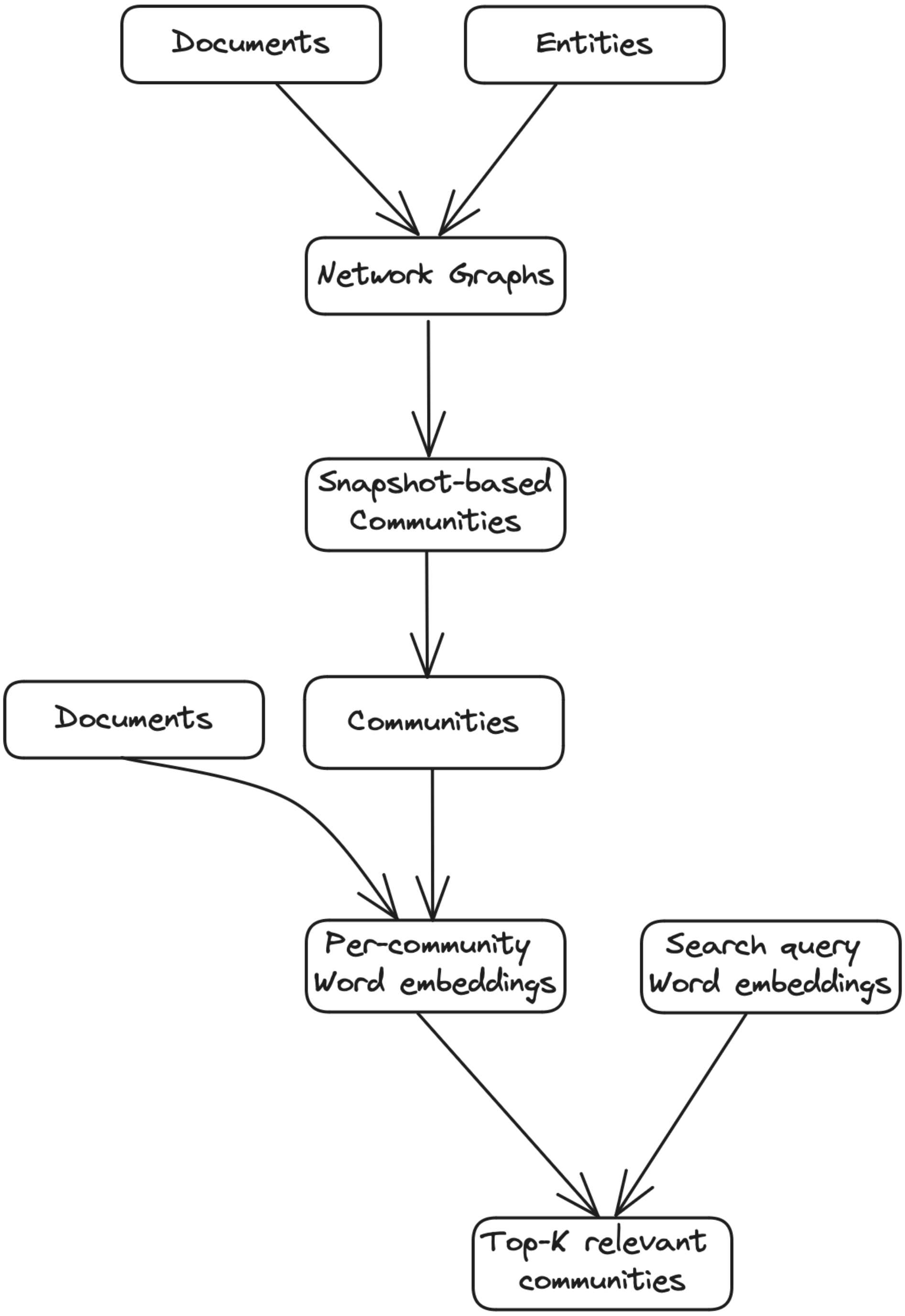
[20] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," 2015. DOI: `10.48550/ARXIV.1503.03832`. [Online]. Available: `https://arxiv.org/abs/1503.03832`.

[21] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 107–117, Apr. 1998. DOI: `10.1016/s0169-7552(98)00110-x`. [Online]. Available: `https://doi.org/10.1016/s0169-7552(98)00110-x`.

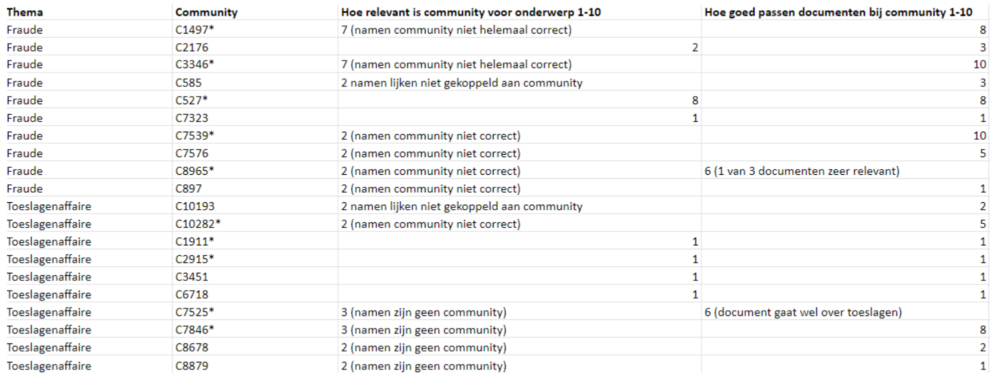
[22] F. Beck, M. Burch, S. Diehl, and D. Weiskopf, "A taxonomy and survey of dynamic graph visualization," *Computer Graphics Forum*, vol. 36, no. 1, pp. 133–159, Jan. 2016. DOI: `10.1111/cgf.12791`. [Online]. Available: `https://doi.org/10.1111/cgf.12791`.

[23] I. Beltagy, M. E. Peters, and A. Cohan, *Longformer: The long-document transformer*, 2020. DOI: `10.48550/ARXIV.2004.05150`. [Online]. Available: `https://arxiv.org/abs/2004.05150`.

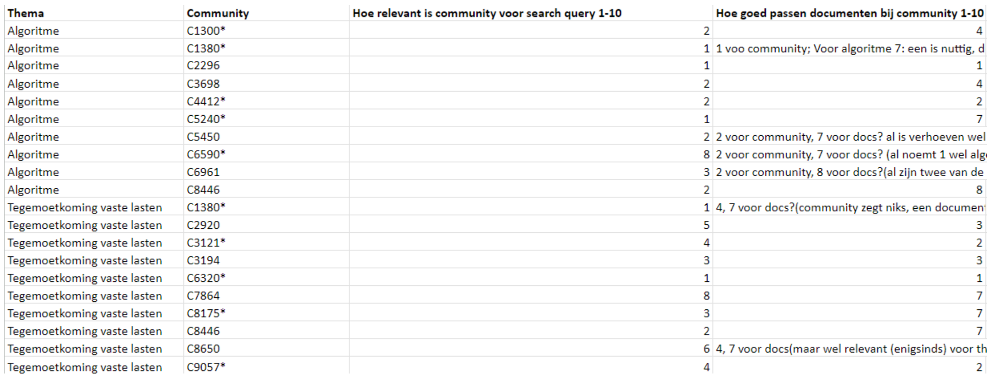
A. Framework visualization


Documents
Entities
Network Graphs
Snapshot-based Communities
Documents
Communities
Per-community Word embeddings
Search query Word embeddings
Top-K relevant communities

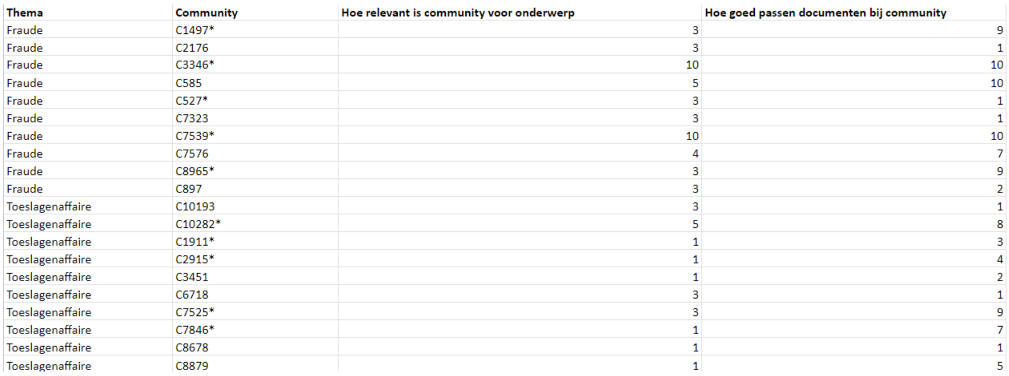
B. Full user study results



Thema	Community	Hoe relevant is community voor onderwerp 1-10	Hoe goed passen documenten bij community 1-10
Fraude	C1497*	7 (namen community niet helemaal correct)	8
Fraude	C2176	2	3
Fraude	C3346*	7 (namen community niet helemaal correct)	10
Fraude	C585	2 namen lijken niet gekoppeld aan community	3
Fraude	C527*	8	8
Fraude	C7323	1	1
Fraude	C7539*	2 (namen community niet correct)	10
Fraude	C7576	2 (namen community niet correct)	5
Fraude	C8965*	2 (namen community niet correct)	6 (1 van 3 documenten zeer relevant)
Fraude	C897	2 (namen community niet correct)	1
Toeslagenaffaire	C10193	2 namen lijken niet gekoppeld aan community	2
Toeslagenaffaire	C10282*	2 (namen community niet correct)	5
Toeslagenaffaire	C1911*	1	1
Toeslagenaffaire	C2915*	1	1
Toeslagenaffaire	C3451	1	1
Toeslagenaffaire	C6718	1	1
Toeslagenaffaire	C7525*	3 (namen zijn geen community)	6 (document gaat wel over toeslagen)
Toeslagenaffaire	C7846*	3 (namen zijn geen community)	8
Toeslagenaffaire	C8678	2 (namen zijn geen community)	2
Toeslagenaffaire	C8879	2 (namen zijn geen community)	1



Thema	Community	Hoe relevant is community voor search query 1-10	Hoe goed passen documenten bij community 1-10
Algoritme	C1300*	2	4
Algoritme	C1380*	1	1 voo community; Voor algoritme 7: een is nuttig, d
Algoritme	C2296	1	1
Algoritme	C3698	2	4
Algoritme	C4412*	2	2
Algoritme	C5240*	1	7
Algoritme	C5450	2	2 voor community, 7 voor docs? al is verhoeven wel
Algoritme	C6590*	8	2 voor community, 7 voor docs? (al noemt 1 wel alg
Algoritme	C6961	3	2 voor community, 8 voor docs?(al zijn twee van de
Algoritme	C8446	2	8
Tegemoetkoming vaste lasten	C1380*	1	4, 7 voor docs?(community zegt niks, een documen
Tegemoetkoming vaste lasten	C2920	5	3
Tegemoetkoming vaste lasten	C3121*	4	2
Tegemoetkoming vaste lasten	C3194	3	3
Tegemoetkoming vaste lasten	C6320*	1	1
Tegemoetkoming vaste lasten	C7864	8	7
Tegemoetkoming vaste lasten	C8175*	3	7
Tegemoetkoming vaste lasten	C8446	2	7
Tegemoetkoming vaste lasten	C8650	6	4, 7 voor docs(maar wel relevant (enigsinds) voor th
Tegemoetkoming vaste lasten	C9057*	4	2



Thema	Community	Hoe relevant is community voor onderwerp	Hoe goed passen documenten bij community
Fraude	C1497*	3	9
Fraude	C2176	3	1
Fraude	C3346*	10	10
Fraude	C585	5	10
Fraude	C527*	3	1
Fraude	C7323	3	1
Fraude	C7539*	10	10
Fraude	C7576	4	7
Fraude	C8965*	3	9
Fraude	C897	3	2
Toeslagenaffaire	C10193	3	1
Toeslagenaffaire	C10282*	5	8
Toeslagenaffaire	C1911*	1	3
Toeslagenaffaire	C2915*	1	4
Toeslagenaffaire	C3451	1	2
Toeslagenaffaire	C6718	3	1
Toeslagenaffaire	C7525*	3	9
Toeslagenaffaire	C7846*	1	7
Toeslagenaffaire	C8678	1	1
Toeslagenaffaire	C8879	1	5