Leveraging Collection-Wide Similarities for Unsupervised Document Structure Extraction

Gili Lior[1,2] **Yoav Goldberg**[1,3] **Gabriel Stanovsky**[1,2]
[1]Allen Institute for AI [2]The Hebrew University of Jerusalem [3]Bar-Ilan University
gili.lior@mail.huji.ac.il

Abstract

Document collections of various domains, e.g., legal, medical, or financial, often share some underlying collection-wide structure, which captures information that can aid both human users and structure-aware models. We propose to identify the typical structure of document within a collection, which requires to capture recurring topics across the collection, while abstracting over arbitrary header paraphrases, and ground each topic to respective document locations. These requirements pose several challenges: headers that mark recurring topics frequently differ in phrasing, certain section headers are unique to individual documents and do not reflect the typical structure, and the order of topics can vary between documents. Subsequently, we develop an unsupervised graph-based method which leverages both inter- and intra-document similarities, to extract the underlying collection-wide structure. Our evaluations on three diverse domains in both English and Hebrew indicate that our method extracts meaningful collection-wide structure, and we hope that future work will leverage our method for multi-document applications and structure-aware models.[1]

1 Introduction

Knowing the structure of a typical document within a collection can be useful in various use cases across different domains. For example, consider the legal domain, where lawyers seek to analyze corpora of legal proceedings, looking for trends and patterns over time. In a retrieval scenario, they may look for punishment trends over many verdict decisions (Wenger et al., 2021). While each verdict decision normally includes a dedicated punishment section, it is often hard to locate it, because it is not consistently marked – different verdicts may use different headers (“Punishment”, “Sentencing Decision”, or “Incurred Penalty”), and position the section in different document locations, requiring scholars to scan vast amounts of text. In an exploratory use case, lawyers may not have a well-defined apriori question, but instead interested in emerging collection-wide trends, for example, analyzing what judges take into account in verdict.

Furthermore, a collection-wide signal about document structure can also be leveraged by structure-aware models for multi-document downstream applications, e.g., infusing the document structure as part of the Transformer architecture, to improve multi-document applications (Liu and Lapata, 2019; Zhang et al., 2023).

To support both human users and structure-aware models, we propose to identify the typical document structure within the collection (§2). This requires to capture recurring topics across the collection, while abstracting over arbitrary header paraphrases, and ground each topic to respective document locations. For example, in Figure 1 we want to identify “Case Summary”, “Evidence Presented” and “Verdict” as the recurring and prominent topics of a typical document within the collection of legal verdict decisions, as opposed to “Software Development Standards Implications”, which is document specific. This also requires to identify that “Verdict” and “Judgement Decision” represent the same topic, i.e., abstract over header paraphrases. The colored bounding boxes in Figure 1 represent the grounding unto each document.

Automatically extracting the typical document structure is challenging. While topic boundaries are often loosely defined by explicit headers, it is hard to use them directly to understand collection-level properties, as headers indicating the same information often vary in phrasing, for example “Verdict”, “Judgement Outcome” and “Decision”. Moreover, some section headers are local to individual document, and do not participate in the global structure.

[1]Our code and data are available at https://github.com/SLAB-NLP/Doc-Structure-Parser

Findings of the Association for Computational Linguistics: ACL 2024, pages 9538–9550

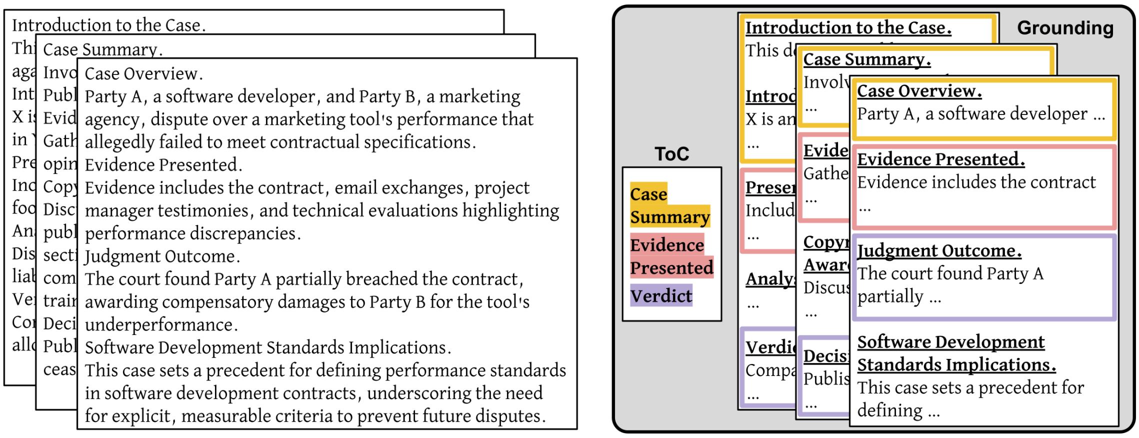

Figure 1: The input for our method (on the left) a document collection with some shared underlying structure. The output (on the right) is a list of the most prominent topics across the collection (ToC), grounded to the documents, which is represented by the colored bounding boxes.

For example, "Software Development Standards Implication" in Figure 1. Finally, while section order provides some signal, it is often inconsistent, and may vary across different documents. The challenge is then align section headers across the collection, while being flexible enough to discard sections that do not reflect a global shared structure.

Following, we devise an unsupervised method using a collection-wide signal to perform the structure extraction (§3). To achieve this, we represent the document collection using a complete undirected weighted graph, whose nodes represent topic boundary candidates, and the weight of edges between each pair of nodes represents their semantic similarity. Such formulation allows us to model relations both within document, as well as across documents. For example, set large edge weight between "Case Overview" and "Introduction to the Case", as they convey semantically similar topics. Finally, we find communities within the graph, where each community is a group of similar nodes that form a single component of the collection-wide structure, and filter the communities that best outline the typical document structure, which we call the collection-wide table of contents (ToC).

To illustrate the robustness of our method across various domains and languages, we curate three distinct datasets (§4). These include two English datasets from different domains (financial and legal), and a Hebrew dataset, composed of legal documents. This variety shows that our method is adaptive and effective under different linguistic and domain-specific contexts.

To evaluate our model, we propose three distinct metrics (§5). First is the "header intrusion" human evaluation, which is adapted from the popular "word intrusion" metric for clustering evaluation (Chang et al., 2009), to evaluate the quality of the representation of the collection. Second, an automatic evaluation for the document-level grounding, to evaluate the predicted grounding coverage. Last, a qualitative analysis over the predicted ToC entries, manually exploring how meaningful are the produced entries, comparing them to an existing ToC of the financial documents collection.

We find that our method extracts meaningful typical document structure, capable of retrieving both the overall collection structure, while also mapping it unto individual documents, and that it is robust to varying domains and languages, with no supervision and little domain-specific adaptation. Detailed error analysis shows that while it is capable of mapping most topics unto documents, it struggles with identifying the exact topic boundaries.

Our released code allows future work to leverage collection-wide structure to enrich both user-facing applications, and to integrate document structure within LLMs to cope with multiple documents.

Our main contributions are: (1) we formally define a novel task of identifying the typical document structure from a document collection, (2) we curate three datasets for the task from diverse domains and languages, and (3) we develop an unsupervised approach for the task, by leveraging collection-wide signal in a community detection algorithm.

2 Task Definition

Given a collection of documents which share an underlying latent structure, we aim to recover a

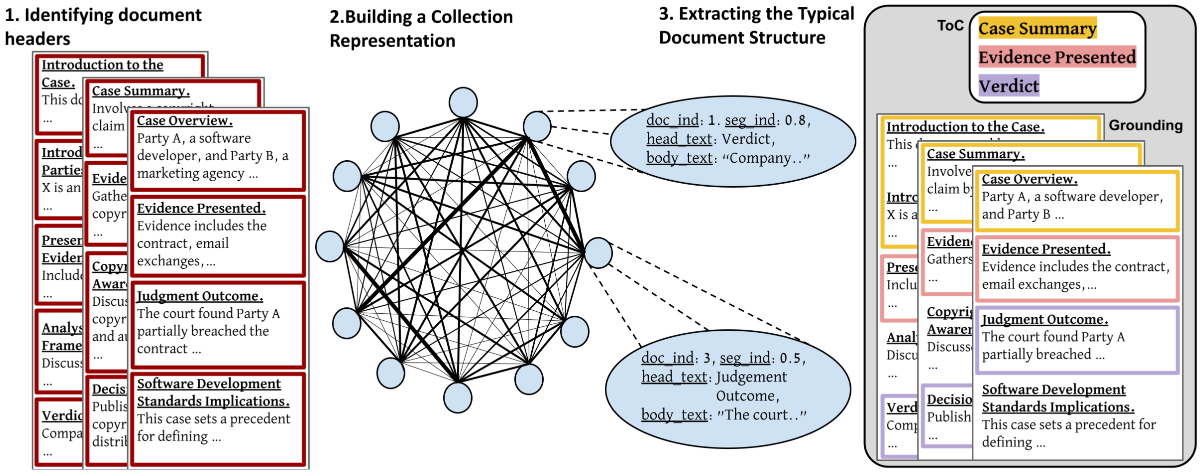

Figure 2: A high-level description of the three steps in our approach for extracting structure from a collection of documents. First, we identify topic boundaries in individual documents, based on lexical cues (§3.1). Second, we build a collection representation using a complete graph whose nodes are the topics identified in the previous step, and weighted edges represent semantic similarity between them (§3.2) Finally, we extract the typical document structure using unsupervised community detection, which we filter based on coverage maximization (§3.3).

structure which satisfies the three following requirements. First, we would like to list *the most dominant topics* which appear throughout the collection, and omit topics which appear only in few documents. This allows users to readily observe overall trends within the collection, without reading it in its entirety. Second, the representation should *abstract over header paraphrases*, allowing users to identify that certain sections discuss similar semantic topics, despite being phrased differently. Third, the structure should *ground topics unto the documents in the collection*, thus providing users with document-level structure, amenable for manual as well as automatic manipulation.

Formally, we denote a collection of documents $D = \{d_1, \ldots, d_n\}$, where a model is required to output a tuple (T, M). $T = \{t_1, \ldots, t_k\}$ is a set of k most prominent topics across D. In the scope of this work, each topic t_i should appear as a continuous text span in some document d_j. For example, in Figure 1, $T = \{$"Case Summary", "Evidence Presented", "Verdict"$\}$. $M : (T \times D) \mapsto (\mathbb{N}^+ \times \mathbb{N}^+)$ grounds topics to individual documents, mapping each topic to start and end indices. M is exemplified with colored bounding boxes in Figure 1.

3 Method

We describe our method for generating the ToC and grounding, which captures the typical document structure within a given document collection, and composed of three steps as outlined in Figure 2.

3.1 Identifying Document Headers

Following Erbs et al. (2013), in the scope of this work we assume document-level topics appear as explicit headers within documents. Later we show that this assumption holds in real-world domains.

Since header formatting may vary across domains, we consider their detection a corpus-specific task. We identify header candidates via rule-based heuristics, by leveraging a collection-wide signal that allows us to differentiate between certain patterns which are prominent throughout the collection (e.g., sentence length and capitalization patterns (Gutehrlé and Atanassova, 2022)), while filtering out other document elements which are sometimes styled as headers (e.g., page numbers, recurring signatures, copyright licensing, etc.).

3.2 Building a Collection Representation

Next, we face a collection-level challenge. After segmenting each document to topics, it is not clear how to represent both intra- and inter-document similarities across the entire collection. Furthermore, the similarity between topics hinges on a myriad of factors, including semantic similarities between the topic headers, their content, and their location within the document. For example, in Figure 2 we want to identify that "Verdict" in one document and "Judgement Desicion" in another are both related to a shared 'Verdict' topic, which is dominant in legal documents, and usually located towards the end of the verdict decision.

To model the different intra- and inter-document similarities, we represent a document collection using a complete undirected weighted graph $G = (V, E)$ as elaborated below.

Graph nodes. The nodes in the graph include all headers identified previously. Each node $v \in V(G)$ is associated with a 4-tuple: {*doc_ind, seg_ind, head_text, body_text*}. *doc_ind* $\in \mathbb{N}$ denotes the index of the document within the collection, *seg_ind* $\in [0, 1]$ denotes the *normalized* sequential position within the document, while *head_text* and *body_text* are the two continuous text spans of the header and following body text. For example, one of the nodes in Figure 2 represents the segment at the 0.5 normalized position from the 3rd document, its header is "Judgement Outcome" and its body text starts with "The court..".

Edge weights. As described above, G is a complete graph over all identified topic boundary candidates from documents within the collection. For each edge (v_1, v_2), we define $weight(v_1, v_2) \in \mathbb{R}$ as a weighted sum of three similarity metrics:

$$\begin{aligned} weight(v_1, v_2) = \lambda_{head} \cdot head\_sim(v_1, v_2) \\ + \lambda_{body} \cdot body\_sim(v_1, v_2) \\ + \lambda_{pos} \cdot pos\_sim(v_1, v_2) \end{aligned} \tag{1}$$

Where $head\_sim(v_1, v_2) \in \mathbb{R}$ represents the similarity between the headers corresponding to the two nodes, and $body\_sim(v_1, v_2) \in \mathbb{R}$ represents the similarity between their bodies. We compute both using cosine similarities over a language model embedding. The $pos\_sim(v_1, v_2) \in \mathbb{R}$ similarity metric computes the similarity of ordering within the document, which encodes the assumption that documents sometimes follow similar topic ordering. In particular, we define pos_sim as follows:

$$\begin{aligned} pos\_sim(v_1, v_2) = \\ (|seg\_ind(v_1) - seg\_ind(v_2)|)^{-1} \end{aligned} \tag{2}$$

Hence, pos_sim is larger the more the respective segments are in similar positions within their respective documents.

Finally, $\lambda_{head}, \lambda_{body}, \lambda_{pos}$ are hyperparameters, non-negative and sum up to 1, weighting the three metrics according to different corpus characteristics. For example, if the document collection follows a strict order across all documents, λ_{pos} could be large, as opposed to the case that the order is not strict but the headers themselves are similar, and then λ_{head} would be larger.

3.3 Extracting the Typical Document Structure

Finally, we want to find the the most representative collection-wide ToC and its corresponding grounding unto each document. At this step we want to find list of important topics which appear in many documents throughout the collection. In addition, we are interested in mapping the list of important topics unto each document in the collection.

As we outlined in Section 2, a good output would consist of coherent topics (e.g., all topics mapped to "Case Summary" would discuss the facts of the case), covering as many documents in the collections, and much of each individual document.

We achieve this by finding communities in the graph. Each community is a group of nodes heavily connected to each other by high-weight edges and lightly connected to nodes outside the community by low-weight edges. Ideally, the inter-community edges represent high similarity between nodes, and hence the community represents a coherent topic across different documents within the collection.

For community detection, we apply the Louvain algorithm (Blondel et al., 2008). We use this algorithm as it has a small number of hyper-parameters compared to other community detection algorithms, and it has shown reliable results over various NLP tasks (Lucy et al., 2023). The objective of the Louvain algorithm is to maximize modularity, which correlates with the ratio between edges inside the community to edges outside the community.

Finally, we generate the collection-wide ToC out of the best communities. We find a subset of k communities that maximize the coverage of the entire collection, which indicates that the chosen communities represent the topics that are relevant to many of the documents and covers the most of each document. For example, in Figure 1 we omit "Software Development Standards Implications" as it appears only in a single document.

Formally, let C represent the set of all communities output by Louvain, where each community $c \in C$ is a list of nodes. We want to find the subset $S \subset C$, containing k communities, that maximizes the coverage of each individual document d across the entire collection D:

$$\underset{S \subset C \ s.t. \ |S|=k}{\text{argmax}} \sum_{d \in D} \left[\frac{1}{|d|} \sum_{v \in d} \mathbb{1}_{[\exists c \in S \ s.t. \ v \in c]} \right] \tag{3}$$

To form the k ToC entries, we then find the centroid of each of the k best communities, and ground

Dataset Name	# docs	# words	Domain	Language	Strict structure	Preserve sections order	Headers similarity
Form-10K	500	18M	Financial	English	✓	✓	High
CUAD	389	3M	Legal (contracts)	English	✗	✗	Medium
Heb-Verdicts	277	3.4M	Legal (verdicts)	Hebrew	✗	✓	Medium

Table 1: Description of the three datasets we use for evaluation, spanning different domains and languages, with varying properties, as detailed in Section 4.

each representative centroid to the text spans of its corresponding community.

4 Data Curation

To evaluate our method, we curate three document collections. As presented in Table 1, these collections cover different domains and languages, with varying structure properties, showing the robustness of our method.

4.1 Form-10K

Form-10K is an annual financial report required by U.S. Securities and Exchange Commission, summarizing a company's annual financial performance. It includes information such as company history, organizational structure, executive compensation, equity, and audited financial statements.

The Form-10K files were collected from the SEC EDGAR platform.[2] We sampled 500 company CIK tickers from a list of public companies, and used these as the basis for EDGAR form retrieval. We then extracted the document's texts by cleaning the markup and filtering non-textual lines (e.g. tables) using regular expressions.

4.2 CUAD

Hendrycks et al. (2021) presented Contract Understanding Atticus Dataset (CUAD), consisting of 510 commercial legal contracts. This dataset was curated to evaluate the extraction of key clauses in contracts, providing labeled clauses that were manually extracted from the documents. Since we are only interested in predicting the structure of the documents, we ignore the provided labels.

To run our method over this dataset, we converted the provided pdf files to raw texts, resulting in raw texts of 389 legal contracts.[3]

In contrast with Form-10K, CUAD consists of different types of contracts, such as affiliate agreements, license agreements, marketing, or manufacturing. Hence, the contracts may diverge in their structure, content and order.

4.3 Hebrew Verdict Decisions

We use Habba et al. (2023)'s curated dataset of verdict decisions from the Israeli court, in Hebrew. The original dataset consists of 855 verdict decisions in sexual offense cases, which we then filtered manually, keeping only the documents with explicit headers, resulting in a collection of 277 Hebrew verdict decisions.

Since all verdict decisions in this dataset are of the same domain (sexual assault cases), we expect them to contain some overlapping topics and high-level structure similarities. But, since there is no pre-defined structure and each judge has their personal style, there are also expected differences and some parts may be added or omitted.

5 Results

We evaluate three different aspects of our method, including human evaluation of the communities we produce (§5.2), document-level evaluation of the ToC grounding (§5.3), and a qualitative analysis of the predicted ToC (§5.4).

5.1 Experimental Setup

Our architecture introduces several hyperparmeters designed to capture different datasets properties. Below we outline their configuration for each of our three collections. These are summarized in Table 1, and presented in detail in Table 4 in the Appendix.

First, our method embeds the document texts using a pre-trained language model. For the two English datasets we use a version of the MPNet language model (Song et al., 2020), as it was reported with the highest average performance on sentence embedding and semantic search tasks.[4,5] For our

[2] Using the secedgar python package `https://github.com/sec-edgar/sec-edgar`

[3] We used Adobe API package to convert pdf to text `https://developer.adobe.com/document-services/docs/overview/pdf-extract-api/`

[4] The MPNet version we use `https://huggingface.co/sentence-transformers/all-mpnet-base-v2`

[5] Performance of different pre-trained English LMs `https://www.sbert.net/docs/pretrained_models.html`

Hebrew Verdict Decisions dataset we use a version of AlephBERT (Seker et al., 2021).[6]

Second, we configure the weighting of the different similarity metrics in Equation 1, according to apriori knowledge of the domains. For example, we set a high header similarity weight (λ_{head}) for the Form-10K corpus, to reflect the expectation for very similar headers across documents. In contrast, the structure of the documents in CUAD is more flexible, so we set a more uniform weighting.

5.2 Graph Representation Evaluation

In this evaluation, participants are shown 10 headers, 9 of which are chosen headers from the same community at random, and the last header is chosen at random from outside the community. Participants are then tasked with identifying the intruder.

This task assesses if our predicted communities are well separated. We expect meaningful communities to demonstrate high internal similarity and low similarity with other communities. Therefore, if our communities are well-defined and meaningful, identifying intrusions should be straightforward, as they will noticeably differ from the rest of the community.

Our evaluation is an adapted version of the "word intrusion" task (Chang et al., 2009), widely used for human evaluation of topic clustering (Bhatia et al., 2017, 2018; Prouteau et al., 2022). We adapt this metric to account for the higher variability presents in headers as opposed to single-word topics by allowing annotators to choose an arbitrary number of intruders. The number of options they mark is reflected in a *confidence* metric:

$$conf = 1 - \frac{num\_marked - 1}{num\_options} \quad (4)$$

Intuitively, the more intruders that the annotator marks, the less they are certain of their annotation. At the extremes, $conf = 1$ when an annotator marks a single intruder, and $conf = \frac{1}{10}$ if they mark all options as intruders. This metric does not account for the accuracy of the annotation, which is calculated separately, if either any of the marked options is the actual intruder.

Crowdsourcing configuration. We run this evaluation through Amazon Mechanical Turk, with a pool of 12 participants, where each unique sample was annotated by exactly one participant, for a total

	Accuracy	Confidence
Form-10K	67.5	85.6 ± 22.4
CUAD	61.7	77.8 ± 21.3
Hebrew Verdict Decisions	65.2	89.2 ± 21.9
Randomly choosing 3 candidates	30.0	80.0

Table 2: Header intrusion crowdsource human evaluation. Confidence is determined based on how many options the participant marked as an intruder candidate according to Equation 4, presenting the average $\pm$ std confidence across all annotations. Accuracy is gained if either one of the marked candidates is the actual intruder. The last row is a random baseline of choosing three random intruder candidates, which induces 80% confidence, resulting in expected accuracy of 30%.

of 900 collected annotations (around 450 samples for each English dataset). We pay 0.1-0.2 USD for a single annotation, aiming for an hourly pay of 12 USD. To ensure quality annotations, we start with a qualification test where participants need to reach accuracy above 0.4 with confidence above 0.25 over a few examples of our method's output. In Figure 5 in the Appendix we provide an example of the annotation interface. As for the Hebrew Verdict Decisions, we collect 132 annotations via a group of 8 in-house Hebrew-speaking graduate students, who show similar performance to MTurk qualified workers.

5.2.1 Key Findings

In Table 2 we present the results for the *header intrusion* human evaluation, and highlight key findings below.

Our method predicts a meaningful collection representation on par with traditional topic modelling. In Table 2 we see that the accuracy for intrusion detection is more than double the chance accuracy for the observed ~80% confidence. Moreover, we find that our reported accuracy falls inside Chang et al. (2009)'s interquartile range (IQR) of word-intrusion results.

Our method performs better on document collections with a more strict nature of structure. As seen in Table 2, accuracy is highest for the Form-10K collection, which may attributed to its stricter structure, compared to our other two collections.

Our method sometimes conflates opposite headers from opposite topics. We manually analyze

[6]The AlephBERT version we use https://huggingface.co/imvladikon/sentence-transformers-alephbert

	Form-10K				Hebrew Verdict Decisions			
	Macro F1		Micro F1		Macro F1		Micro F1	
	partial	exact	partial	exact	partial	exact	partial	exact
Our method	93.7	63.7	92.6	43.9	78.1	63.7	82.8	64.2
Most frequent class	6.3	0	21.5	0	8.3	0	18.7	0
Random	16.1	1.1	25.8	1.1	21.6	6.4	16.6	3.2

Table 3: The accumulated ToC grounding scores for the Form-10K and Hebrew Verdict Decisions datasets, comparing our method to two different baselines, as elaborated in §5.3.

the Hebrew Verdict Decisions annotations, finding that some of the low confidence annotations (i.e., when annotators mark many options for the intruder) happen when our method conflates opposite topics in the same community. For example, "defense arguments" and "prosecution arguments" are wrongly clustered in the same community.[7] Considering opposite topics as close in the embedding space is a known property of language models (Vahtola et al., 2022).

5.3 Document Grounding Evaluation

In a second evaluation effort, we aim to quantify the quality of our method's automatic grounding from collection-wide ToC topics unto text spans in individual documents.

Annotating gold labels. To evaluate our method's grounding performance, we annotate some of our unsupervised data in two of our domains. The Form-10K dataset follows a strict structure, so we apply simple heuristics to automatically generate gold label grounding, resulting in a collection of 266 documents with grounding gold labels.[8] In contrast, for the Hebrew Verdict Decisions we use human annotations to generate its grounding gold labels, done by a domain expert (Hebrew speaking law student). Each manual annotation took approximately 45 minutes, with a total of 15 labeled documents. A screenshot of the annotation interface is provided in Figure 4 in the Appendix. We omitted CUAD from this particular evaluation due to budget constraints.

Metrics. We formulate exact and partial match scores, evaluating the overlap between the gold and the predicted grounding. Partial match credits *any* overlap between the text spans in predicted and gold, while exact match counts only *perfectly* aligned predictions. The formal mathematical definitions are presented in Appendix B.1.

Baselines. To the best of our knowledge, we present the first method which aims to identify ToC at the document-collection level, and hence we are not aware of applicable baselines. Instead, to gauge the difficulty of this task we include two naive approaches. First, a baseline predicting *most frequent class* in the gold annotations. For the Form-10K corpus this is "Exhibits, Financial Statement Schedules" (22% of headers) and for Hebrew Verdict Decisions this is "prosecution evidence" (20% of headers). The second baseline predicts each gold ToC header uniformly *at random*, averaged over 100 different random seeds.

5.3.1 Key Findings

The results for our two metrics and two document collections are presented in Table 3. Several observations can be drawn based on these results.

Our method captures the correct structure of a document. The high partial match scores indicate that our method is able to predict the correct document structure, as almost every gold topic grounding has some overlap with the predicted topic grounding (Micro F1 score of 92.6). As shown in Table 3, our method outperforms the two naive baselines in all F1 scores.

Exact match score is noisy. Even if only a single sentence is separating between a gold and a predicted topic grounding boundaries, the exact match score for the topic will be 0. We find this score very strict, and somewhat arbitrary. In addition, topics related to longer text spans, are more probable to reach a low exact match score. This is indicated by the first three rows in Table 5 in the Appendix, representing the three largest classes, showing lower exact match scores than all other classes.

[7] All examples from Hebrew Verdict Decisions are translated from Hebrew.

[8] Gold ToC for Form-10K is based on https://en.wikipedia.org/wiki/Form_10-K

Gold ToC

1. Business
   a. Risk Factors
   b. Unresolved Staff Comments
2. Properties
3. Legal Proceedings
4. Mine Safety Disclosures
5. Market
6. Consolidated Financial Data
7. Management's Discussion and Analysis of Financial Condition and Results of Operations
   a. Quantitative and Qualitative Disclosures about Market Risks
8. Financial Statements
9. Changes in and Disagreements With Accountants on Accounting and Financial Disclosure
   a. Controls and Procedures
   b. Other Information
10. Directors, Executive Officers and Corporate Governance
11. Executive Compensation
12. Security Ownership of Certain Beneficial Owners and Management and Related Stockholder Matters
13. Certain Relationships and Related Transactions, and Director Independence
14. Principal Accounting Fees and Services
15. Exhibits, Financial Statement Schedules Signatures

Predicted ToC	Map to gold #
Service Providers	-
Interest Income	-
Part Iv Exhibits Consolidated Financial Statement Schedules	15
Stock Incentive Plan	11
Global Crossing Ltd S Dan J Cohrs	-
Legal Proceedings	3
Security Ownership Of Certain Beneficial Owners And Management And Related Security Holder Matters A	12
Affirmative Covenants	-
Part Ii Market	5
Management's Discussion And Analysis Of Financial Condition And Results Of Operations	7
Certain Relationships And Related Transactions	13
A Quantitative And Qualitative Disclosures About Market Risk	7
Part Iii Directors And Executive Officers Of The Registrant	10
Selected Financial Data	6
Properties	2
Yes X	-
Principal Accountant Fees And Services	14
Changes In And Disagreements With Accountants On Accounting And Financial Disclosure	9
B Other Information	9
A Controls And Procedures	9
Refer To Note	-
Part I Business	1
Summary Of Significant Accounting Policies Continued	-
Submission Of Matters To A Vote Of Security Holders	4
A Risk Factors	1
Table Caption December	-
B Unresolved Staff Comments	1
Description Exhibit	-
Table	-
Total	-
Caption	-

Figure 3: Mapping between the gold and predicted ToC for the Form-10K dataset. The predicted ToC entries are ordered according to communities coverage ranking (§ 3.3). The entries in gray does not have mapping.

5.4 Qualitative ToC Analysis

Lastly, we suggest a collection-level qualitative evaluation, to explore the quality of the output ToC. Below we provide several key observations derived from our manual mapping between the predicted and gold Form-10K ToC entries, visualized in Figure 3.

We were able to map 14 out of the 15 gold ToC entries to the predicted ToC entries. The only gold entry that was not mapped is “8. financial statements”. After manually exploring the ToC grounding, we find that our method sometimes confuses this topic with “6. Consolidated Financial Data”.

The 14 gold ToC entries align with 19 predicted entries, in the following manner:

- 13 predicted ToC entries that were textually almost identical to the gold entries, with a straight-forward mapping.
- 1 mapping which followed a manual exploration of the dataset – we mapped “4. Mine Safety Disclouser” to “Submission Of Matters To A Vote Of Security Holders”, as we find that in many of the documents the latter is the actual header of the 4^{th} topic.
- 5 *subsection* headers (‘A Risk Factors’, ‘B Unresolved Staff Comments’, etc.), that we map to their gold topic headers, as our method predicts a flat structure while the Form-10K consists of an hierarchical structure.

Beyond the 19 predicted entries that align with gold entries, our method predicted additional 12 entries. Some topics indeed recur throughout the collection and accordingly receive high coverage scores, e.g., “Service Providers” or “Interest Income”, and may be considered too granular. Others, attributed to noise, e.g., “Total”, “Table” or “Yes X”, receive low coverage scores, and their emergence as topic candidates could stem from degraded performance at the header detection step.

6 Related Work

A seminal theory of a pragmatic meaning is rhetorical structure theory (RST Mann and Thompson, 1988), that inspired much follow-up work, mostly approaching document structure extraction as a supervised task (Arnold et al., 2019; Kang et al., 2022; Xia et al., 2022; Hua and Wang, 2022). The need for supervision narrows the application of such tools only to high-resource languages and small number of domains. Our method is unsupervised, does not require labeling, and is applicable to many languages and domains.

Another line of work suggest to approach structure extraction as an unsupervised task, to extend its applicability (Xing and Carenini, 2021; Born et al., 2022; Li et al., 2023), but they still approach this as a single-document task. In our work we take the entire collection as our input, leveraging signal regarding the underlying collection-wide structure.

Xing et al. (2022) discovered that injecting external (above-sentence) information can help with identifying discourse dependency structure, but they only look at individual documents. We extend this observation into modeling relations across an entire collection, to inject external information about the structure components of the collection.

Finally, theories like CST (Radev, 2000) and CAST (Altmami and Menai, 2020), or (Lu et al., 2018), suggest to look at the entire data collection to extract the main components that capture a typical structure layout of a document within the collection, but, their approach too relies on supervision and labeled data.

7 Conclusion

In this paper we suggested an unsupervised method to extract the typical document structure within a collection, leveraging similarities across documents that come from the same collection with some shared underlying latent structure.

We evaluated the extracted typical document structure through three different evaluation metrics, over three different datasets, showing our method's robustness to different domains and languages. The evaluations indicated that our method succeeds in identifying the different topics consisting the typical document structure, while showing room for improvement in its accuracy to detect the exact location of where a topic starts and where it ends.

Our method can be leveraged by users of multi-document applications, allowing them more focused browsing over a collection of documents, which can be useful for downstream tasks like retrieval and summarization. As for models, future work can utilize our method and inject the collection-wide structure to structure-aware models, to improve downstream applications like multi-document summarization or cross document information extraction.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments. Members of the SLAB group at the Hebrew university for their productive discussion. This research was supported by grants from Allen Institute for AI, the Israeli Ministry of Science and Technology (grant no. 2088), and the Council for Higher Education.

Limitations

Currently, our method relies on segmentation induced by the headers in the document, which limits our work only applicable on documents with explicit headers. Future work can extend our collection-wide approach to develop a multi-document unsupervised method to induce segmentation for document collections without explicit headers.

Furthermore, our method only predicts a flat list of the prominent topics, and does not express the hierarchical nature of structure that might be present. Future work can extend our method to predict also hierarchical structure.

Ethics Statement

All annotators were informed ahead that their annotations are curated for research purposes, and how the data is going to be used.

The Hebrew Verdict Decisions dataset contains sensitive information pertaining to sexual harassment offenses. The original dataset we utilized underwent a rigorous anonymization process to protect individual privacy. We do not publicly publish this dataset, and accessing it requires authorization from the dataset's original proprietors (Habba et al., 2023).

Recognizing the potential for distress by annotating the Hebrew Verdict Decisions, we issued a trigger warning for all annotation tasks and obtained explicit consent from annotators before their participation. This ensured they were fully informed of the risks associated with exposure to sensitive and potentially triggering content.

References

Nouf Ibrahim Altmami and Mohamed El Bachir Menai. 2020. Cast: A cross-article structure theory for multi-article summarization. *IEEE Access*, 8:100194–100211.

Sebastian Arnold, Rudolf Schneider, Philippe Cudré-Mauroux, Felix A. Gers, and Alexander Löser. 2019. SECTOR: A neural model for coherent topic segmentation and classification. *Transactions of the Association for Computational Linguistics*, 7:169–184.

Shraey Bhatia, Jey Han Lau, and Timothy Baldwin. 2017. An automatic approach for document-level topic model evaluation. *ArXiv*, abs/1706.05140.

Shraey Bhatia, Jey Han Lau, and Timothy Baldwin. 2018. Topic intrusion for automatic topic model evaluation. In *Conference on Empirical Methods in Natural Language Processing*.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Logan Born, M. Monroe, Kathryn Kelley, and Anoop Sarkar. 2022. Sequence models for document structure identification in an undeciphered script. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9111–9121, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jonathan D. Chang, Jordan L. Boyd-Graber, Sean Gerrish, Chong Wang, and David M. Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Neural Information Processing Systems*.

Nicolai Erbs, Iryna Gurevych, and Torsten Zesch. 2013. Hierarchy identification for automatically generating table-of-contents. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 252–260, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Nicolas Gutehrlé and Iana Atanassova. 2022. Processing the structure of documents: Logical layout analysis of historical newspapers in french. *Journal of Data Mining & Digital Humanities*, (Digital humanities in languages).

Eliya Habba, Renana Keydar, Dan Bareket, and Gabriel Stanovsky. 2023. The perfect victim: Computational analysis of judicial attitudes towards victims of sexual violence. *arXiv preprint arXiv:2305.05302*.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. CUAD: An expert-annotated NLP dataset for legal contract review. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Xinyu Hua and Lu Wang. 2022. Efficient argument structure extraction with transfer learning and active learning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 423–437, Dublin, Ireland. Association for Computational Linguistics.

Juyeon Kang, Abderrahim Ait Azzi, Sandra Bellato, Blanca Carbajo Coronado, Mahmoud El-Haj, Ismail El Maarouf, Mei Gan, Ana Gisbert, and Antonio Moreno Sandoval. 2022. The financial document structure extraction shared task (FinTOC 2022). In *Proceedings of the 4th Financial Narrative Processing Workshop @LREC2022*, pages 83–88, Marseille, France. European Language Resources Association.

Chuyuan Li, Patrick Huber, Wen Xiao, Maxime Amblard, Chloe Braud, and Giuseppe Carenini. 2023. Discourse structure extraction from pre-trained and fine-tuned language models in dialogues. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2562–2579, Dubrovnik, Croatia. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Hierarchical transformers for multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5070–5081, Florence, Italy. Association for Computational Linguistics.

Wei Lu, Yong Huang, Yi Bu, and Qikai Cheng. 2018. Functional structure identification of scientific documents in computer science. *Scientometrics*, 115:463 – 486.

Li Lucy, Jesse Dodge, David Bamman, and Katherine Keith. 2023. Words as gatekeepers: Measuring discipline-specific terms and meanings in scholarly publications. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6929–6947, Toronto, Canada. Association for Computational Linguistics.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Thibault Prouteau, Nicolas Dugué, Nathalie Camelin, and Sylvain Meignier. 2022. Are embedding spaces interpretable? results of an intrusion detection evaluation on a large french corpus. In *International Conference on Language Resources and Evaluation*.

Dragomir Radev. 2000. A common theory of information fusion from multiple text sources step one: Cross-document structure. In *1st SIGdial Workshop on Discourse and Dialogue*, pages 74–83, Hong Kong, China. Association for Computational Linguistics.

Amit Seker, Elron Bandel, Dan Bareket, Idan Brusilovsky, Refael Shaked Greenfeld, and Reut Tsarfaty. 2021. Alephbert: A hebrew large pre-trained language model to start-off your hebrew nlp application with. *arXiv preprint arXiv:2104.04052*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*, volume 33, pages 16857–16867. Curran Associates, Inc.

Teemu Vahtola, Mathias Creutz, and Jörg Tiedemann. 2022. It is not easy to detect paraphrases: Analysing semantic similarity with antonyms and negation using the new SemAntoNeg benchmark. In *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 249–262, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Mohr Wenger, Tom Kalir, Noga Berger, Carmit Klar Chalamish, Renana Keydar, and Gabriel Stanovsky. 2021. Automated extraction of sentencing decisions from court cases in the Hebrew language. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 36–45, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jinxiong Xia, Cao Liu, Jiansong Chen, Yuchen Li, Fan Yang, Xunliang Cai, Guanglu Wan, and Houfeng Wang. 2022. Dialogue topic segmentation via parallel extraction network with neighbor smoothing. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 2126–2131, New York, NY, USA. Association for Computing Machinery.

Linzi Xing and Giuseppe Carenini. 2021. Improving unsupervised dialogue topic segmentation with utterance-pair coherence scoring. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 167–177, Singapore and Online. Association for Computational Linguistics.

Linzi Xing, Patrick Huber, and Giuseppe Carenini. 2022. Improving topic segmentation by injecting discourse dependencies. In *Proceedings of the 3rd Workshop on Computational Approaches to Discourse*, pages 7–18, Gyeongju, Republic of Korea and Online. International Conference on Computational Linguistics.

Zixuan Zhang, Heba Elfardy, Markus Dreyer, Kevin Small, Heng Ji, and Mohit Bansal. 2023. Enhancing multi-document summarization with cross-document graph-based information extraction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1696–1707, Dubrovnik, Croatia. Association for Computational Linguistics.

A Parameters For Evaluation

As mentioned in Section 5.1, below we describe the different parameters used for running the evalution.

Dataset	Language Model	λ_{head}	λ_{body}	λ_{index}
10k	all-mpnet-base-v2	0.7	0	0.3
CUAD	all-mpnet-base-v2	0.5	0.3	0.2
Heb-Verdicts	sentence-transformers-alephbert	0.5	0.25	0.25

Table 4: The implementation parameters we used for running our method on different data collections.

Headers detection. In our implementation we detect headers for each dataset with a different set of heuristics. For example, for the Form-10k dataset, we search for sentences shorter than 10 tokens, with capitalization over > 50% of the words in the sentence. This produces high recall (most headers are caught through this heuristic), but low precision (many of the caught header candidates are not really headers). An example for a scenario where a sentence is wrongly marked as a header candidate is for missing sections, in which the only section content is 'None' or 'Omitted'. Furthermore, some of the header candidates are real headers, but not informative ones, for example 'Section A'. In such cases, we also drop these sentences from the set of header candidates using regex.

B Additional Evaluation Analysis

B.1 Exact and Partial Match Scores

Topic boundaries are a set of consecutive text spans that are labeled with the same ToC header. Following, we define an *exact match* of a topic grounding if the text spans forming the gold label topic grounding, match perfectly with the text spans that are predicted with that header, while for *partial match* it is enough if there is one text span within that topic boundaries that was predicted correctly.

These two matching scores are formalized as follows: let $s(w, d)$ be the range of text spans in document d, that are labeled as class w. So, we evaluate the intersection between $s_{gold}(w, d)$ and $s_{pred}(w, d)$. Full match precision is if $s_{pred}(w, d) \setminus s_{gold}(w, d) = \emptyset$, i.e., all text spans in d that were predicted w are labeled correctly, and full match recall is if $s_{gold}(w, d) \setminus s_{pred} = \emptyset$, i.e., all text spans in d that are labeled w are covered by the correct predictions. For partial match it is enough to have a non-empty intersection between $s_{gold}(w, d)$ and $s_{pred}(w, d)$.

Gold Header	Class Size		Precision		Recall		F1 Score	
	# segments	# sections	partial	exact	partial	exact	partial	exact
Exhibits, Financial Statement Schedules	3386	210	77.4	2.6	97.6	17.1	86.3	4.6
Management s Discussion and Analysis of Financial Condition and Results of Operations	2556	254	95.1	38.5	99.2	27.6	97.1	32.1
Business	1833	207	83.3	65.7	93.7	17.4	88.2	27.5
Certain Relationships and Related Transactions, and Director Independence	1509	257	96.9	75.2	97.3	82.9	97.1	78.8
Principal Accounting Fees and Services	1039	210	93.6	83.2	98.1	71.9	95.8	77.1
Changes in and Disagreements With Accountants on Accounting and Financial Disclosure	665	238	90.5	64.6	100	79.8	95.0	71.4
Executive Compensation	452	255	97.3	35.2	99.6	87.1	98.4	50.2
Market	400	247	96.3	73.6	94.3	71.3	95.3	72.4
Legal Proceedings	398	260	98.9	83.7	100	88.1	99.4	85.8
Properties	362	253	98.7	94.8	89.7	78.7	94.0	86.0
Security Ownership of Certain Beneficial Owners and Management and Related Stockholder Matters	335	247	95.0	73.1	100	88.3	97.4	80.0
Directors, Executive Officers and Corporate Governance	327	218	88.6	69.9	95.9	76.1	92.1	72.9
Selected Financial Data	320	241	97.9	82.6	97.9	87.1	97.9	84.8
Mine Safety Disclosures	292	220	93.5	89.0	65.9	55.5	77.3	68.3

Table 5: ToC grounding scores for the Form 10k dataset. Each row represents a single ToC entry, with its size within the dataset, and our method's precision, recall and F1 scores on each class. The partial/exact match scores differ in their minimal requirement to classify a prediction as correct. While partial score is gained if the prediction and label has at least one overlapping segment, exact match is gained only if the prediction and gold label fully agree. We further elaborate on partial/exact match scores in Section 5.3.

B.2 Document-Level Grounding For Form-10K

In Table 5 we provide our method's detailed results for each ToC entry's grounding, i.e., precision, recall and F1 exact and partial scores, over the Form-10K dataset.

B.3 Mapping Between Predicted and Gold Form-10K Conceptual ToC

In Figure 3 we provide the manually mapping between the predicted Form-10K ToC entries, to its gold label ToC.

C Human Annotations

C.1 Annotating Gold Labels For Grounding

Figure 4 is a screenshot of the annotation interface for grounding gold labels. We use this interface to annotate the Hebrew Verdict Decisions dataset, but for readability we provide here the interface over a document from CUAD.

C.2 Header Intrusion Task

Figure 5 is a screenshot of the annotation interface for the header intrusion evaluation, As described in Section 5.2. Annotators were chosen from English-speaking countries: USA, Canada, and UK. Mturk does not provide further demographic information about annotators.

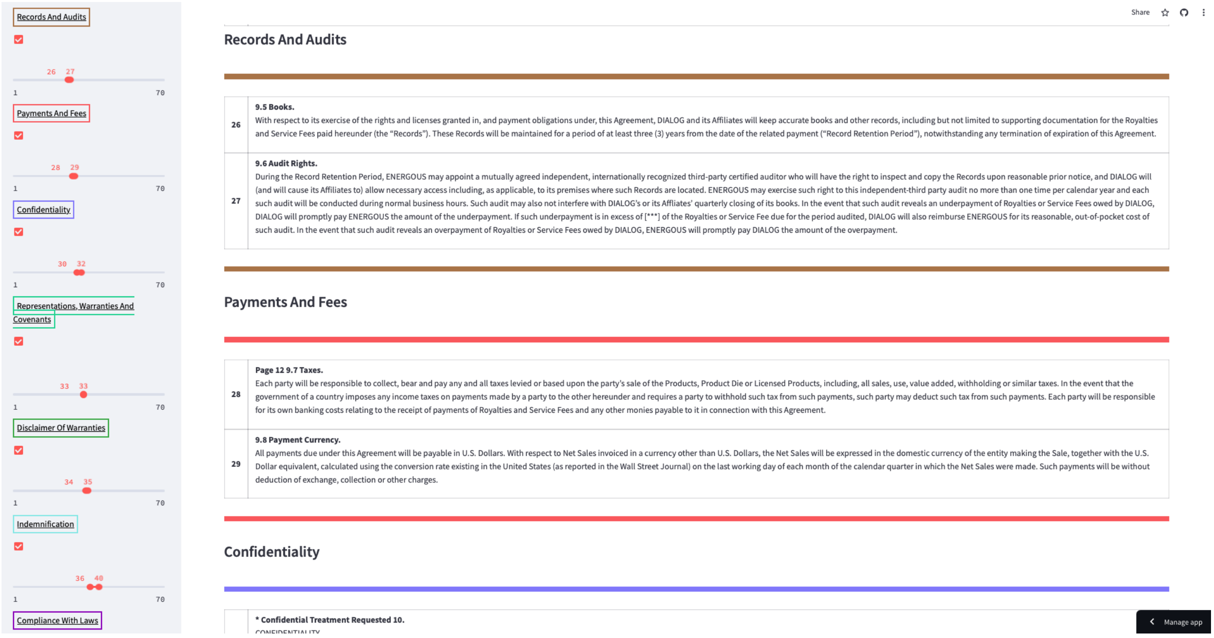

Figure 4: Example of the annotation interface for document-level grounding. On the left side is the ToC entries, and the bounding boxes on the around the text content marks the grounding boundaries. Below the ToC entries are editable bars, which than the annotator can edit according to its belief gold labeling. The presented document is partial, due to visbility limitation, but annotator can scroll and see the entire document.

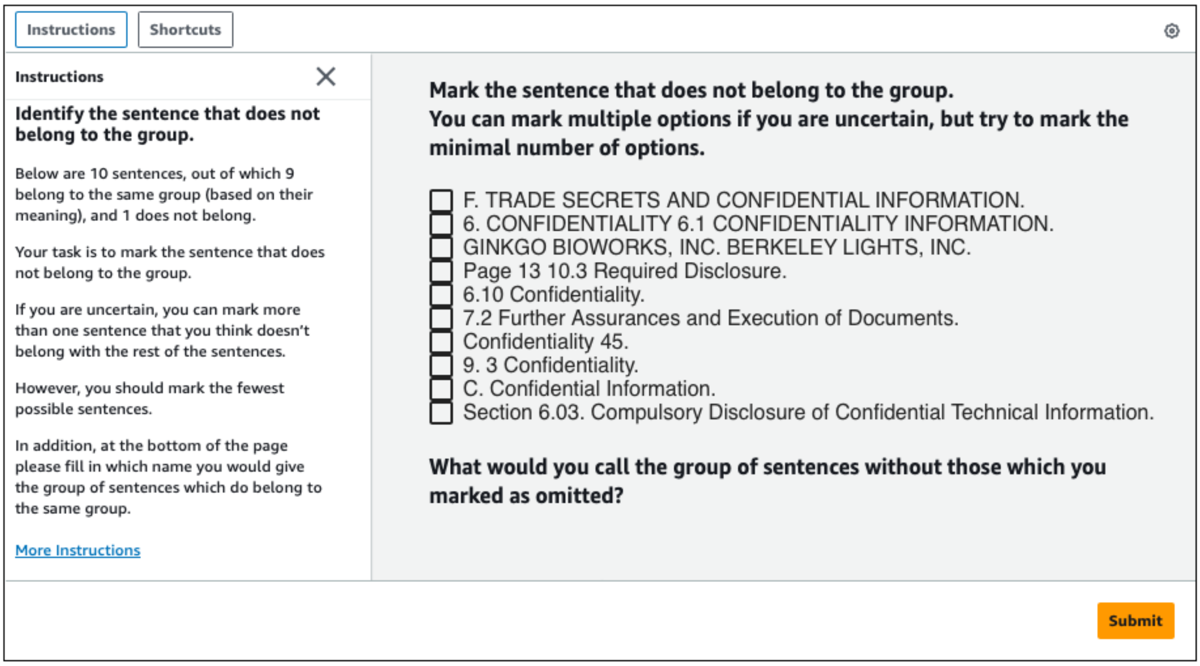

Figure 5: Screenshot of the human annotation interface for the header intrusion task. “7.2 Further Assurances...” and the rest are all headers describing a “confidentiality” section.