DocInfer: Document-level Natural Language Inference using Optimal Evidence Selection

Puneet Mathur*, Riyaz Bhat[2], Gautam Kunapuli[3], Manish Shrivastava[4],
Dinesh Manocha[1], and Maneesh Singh[3]

[1]University of Maryland, College Park, MD, USA
[2]IBM, Bengaluru, India
[3]Motive, San Francisco, USA
[4]IIIT, Hyderabad, India

Abstract

We present **DocInfer** - a novel, end-to-end Document-level Natural Language Inference model that builds a hierarchical document graph enriched through inter-sentence relations (topical, entity-based, concept-based), performs paragraph pruning using the novel *SubGraph Pooling* layer, followed by optimal evidence selection based on REINFORCE algorithm to identify the most important context sentences for a given hypothesis. Our evidence selection mechanism allows it to transcend the input length limitation of modern BERT-like Transformer models while presenting the entire evidence together for inferential reasoning. We show this is an important property needed to reason on large documents where the evidence may be fragmented and located arbitrarily far from each other. Extensive experiments on popular corpora - DocNLI, ContractNLI, and ConTRoL datasets, and our new proposed dataset called CaseHoldNLI on the task of legal judicial reasoning, demonstrate significant performance gains of 8-12% over SOTA methods. Our ablation studies validate the impact of our model. Performance improvement of $\sim 3-6\%$ on annotation-scarce downstream tasks of fact verification, multiple-choice QA, and contract clause retrieval demonstrates the usefulness of `DocInfer` beyond primary NLI tasks.

1 Introduction

Natural Language Inference (NLI) is a fundamental textual reasoning task seeking to classify a presented hypothesis as *entailed by, contradictory to or neutral to* a premise (Dagan et al., 2010). Prior NLI datasets and studies have focused on sentence-level inference where both the premises and hypotheses are single sentences (SNLI (Bowman et al., 2015), MultiNLI(Williams et al., 2018), QNLI and WNLI(Wang et al., 2018)) Document-level NLI extends the reasoning of NLI beyond sentence granularity where the premises are in the document granularity, whereas the hypotheses can vary in length from single sentences to passages with hundreds of words(Yin et al., 2021).

Document level NLI is an important problem for many tasks including verification of factual correctness of document summaries, fact-checking assertions against articles, QA on long texts, legal compliance of contracts, etc. Even so, it challenges modern approaches due to the limited input bottleneck of modern Transformer models. Consider that the universally used BERT model (Devlin et al., 2018) can only encode 512 input sub-tokens due to its quadratic self-attention complexity. Consequently, evidence in the document premise relevant to the hypothesis can potentially be distributed in several textual spans located arbitrarily far away from each other in long documents, and may not be simultaneously available to draw inference.

Recent approaches, notably SpanNLI (Koreeda and Manning, 2021), HESM (Hanselowski et al., 2018)) and others, have shown that chunking the premise into multiple document spans, scoring them, and aggregating the scores helps mitigate the limited input length problem. Such approaches do not allow the inference module to reason over the complete evidence. In contrast to encoding the document as a set of sentences fed into a transformer for inferential reasoning, a recent line of work, e.g. EvidenceNet (Chen et al., 2022), GEAR (Zhou et al., 2019) and HGRGA (Lin and Fu, 2022)), encodes documents as graphs and uses graph reasoning to perform textual inference. Graphs allow encoding of various morphological and semantic relationships at various granularities. However, these approaches use graph-based processing subsequent to evidence selection.

We address the above challenge with a reasonable assumption that the portion of the premise (the ground truth evidence) necessary and sufficient for inference can fit entirely into the length

*Corresponding Author:puneetm@umd.edu

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 809 - 824
December 7-11, 2022

limit of language model for effective representation learning. Our proposed system achieves this by selecting sentences in the document that are contextually relevant for a given hypothesis through pruning irrelevant paragraphs and reinforce learning based optimal sentence selection. Our **main** contributions:

- **`DocInfer` – a novel DocNLI model** that simultaneously performs successive optimal evidence selection and textual inference on large documents. It utilizes a novel graph representation of the document encoding structural, topical, concept and entity-based relationships. It performs subgraph pooling and asynchronous graph updates to provide a pruned, hypothesis-relevant and richer sub-document graph representation and uses a reinforcement-learning based subset selection module to provide the contextually-relevant evidences for inference. Experimental results show that `DocInfer` outperforms the current SOTA on DocNLI, ContractNLI and ConTRoL datasets with a significant improvement of 8-12%.
- We propose **CaseHoldNLI - a new document-level NLI dataset** in the domain of legal judicial reasoning with over 270K document-hypotheses pair with maximum premise length of 3300 words. We observe similar performance gains on this dataset.
- **Application on downstream tasks:** We demonstrate the usefulness of the `DocInfer` evidence selection module on downstream tasks of fact verification, multiple choice QA and few shot clause retrieval from legal texts using no or small amounts of data for supervised fine-tuning. Results on FEVER-binary, MCTest and Contract Discovery dataset show significant improvement of $\sim$ 3-6% F1.

2 Related Work

Document-level NLI Datasets: Yin et al. (2021) introduced Doc-level NLI on news and Wikipedia articles. Liu et al. (2021) proposed the multi-paragraph ConTRoL dataset focused on complex contextual reasoning (logical, coreferential, temporal, and analytical reasoning). Several datasets comprising legal documents like case laws, statutes, and contracts have been proposed. COLIE-2020 (Rabelo et al., 2020) and Holzenberger et al. (2020) support identification of relevant paragraphs from cases that entail the decision of a new case. However, the combined input length of their premise-hypothesis pairs remains within 512 tokens with the premise lengths at paragraph-level, reasonably suited for input to BERT-like models. Koreeda and Manning (2021) released ContractNLI dataset for document-level NLI task on multiple page NDA contract documents along with ground truth evidence labeling for interpretability. We benchmark `DocInfer` on DocNLI, ContractNLI and ConTRoL datasets. CaseHOLD dataset (Zheng et al., 2021) is a multiple choice QA dataset for selecting relevant governing laws required to reason about the legal decision text. **Document-scale and Corpus-Scale Reasoning**: In order to handle document-scale premises in the doc-NLI corpora, approaches like SpanNLI (Koreeda and Manning, 2021), HESM (Hanselowski et al., 2018)) chunk the premise into multiple document spans for reasoning. A similar approach was followed by legal language models such as Legal-BERT (Chalkidis et al., 2020) and Custom Legal-BERT (Zheng et al., 2021) for legal reasoning tasks. More recently, language models (e.g., Longformer (Beltagy et al., 2020) with 4096 token input) have been proposed to overcome the limited input field bottleneck. Fact Extraction and Verification (FEVER) (Thorne et al., 2018) tasks require extracting evidence and claim entailment given an input claim and the Wikipedia corpus. Prior works in this domain address the length limitation for claim verification by relevant evidence identification and its chunking which are individually scored and probabilistically aggregated (Subramanian and Lee, 2020; Jiang et al., 2021). Hierarchical graph modeling may be used to handle the large scale of the premise (Liu et al., 2019b; Zhou et al., 2019; Zhong et al., 2020; Zhao et al., 2020; Chen et al., 2022; Lin and Fu, 2022; Si et al., 2021). **Context Selection for Document-level NLP**: Recent works have investigated selection of relevant context for document-level NLP tasks such as Neural Machine Translation (Kang et al., 2020), Event Detection (Ngo et al., 2020; Veyseh et al., 2021), Relation Extraction (Trong et al., 2022). Recently, some of the work on document-level NLP has looked at temporal relation extraction (Mathur et al., 2021) , temporal dependency parsing (Mathur et al., 2022b), and speech synthesis (Mathur et al., 2022a) using graphs and sequence learning. However, none of them have considered an end-to-end trainable approach for graph learning with to identify the relevant evidence extraction.

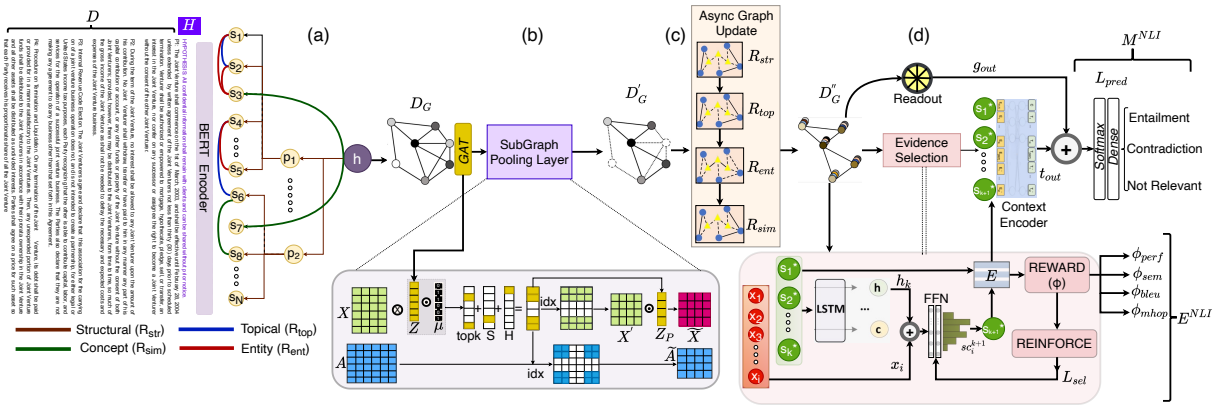


Figure 1: DocInfer Architecture: Document D and hypothesis H pass through BERT; Hierarchical graph using (R_{str}), (R_{top}), (R_{sim}) and (R_{ent}) relations. SubGraph Pooling extracts relevant paragraph nodes; Asynchronous graph updates learn relation-specific node embeddings. Evidence Selection optimized by REINFORCE rewards (ϕ_{perf}), (ϕ_{bleu}), (ϕ_{sel}), (ϕ_{mhop}).

3 DocInfer

Given a textual hypothesis H, the task of document-level NLI is to classify whether the hypothesis is *entailed by*, *contradicting to* or *not mentioned by (neutral to)* the document D. We present DocInfer, a neural architecture (Figure 1) that can select a set of evidence sentences E from document D to form a shortened document D^e which is then used for NLI prediction. Here, for the document level NLI task, we need to constrain D^e to fall within the length limit of BERT-like context encoder to enable it to consume the evidence entirely for improved representation learning for NLI.

Our model can been seen as a sequence of four phases: (a) Representation of document D in the presence of the Hypothesis H to form a hierarchical document graph with sentences and paragraphs as nodes and Structural, Topical, Entity-centric and Concept-similarity relations as edges. (b) Paragraph node pruning using the novel *Subgraph Pooling* layer to select highly relevant paragraphs. (c) Asynchronous graph update for improved node representations and finally. (d) Optimal evidence selection using REINFORCE from the graph for the task of document-level NLI.

Document Representation: Let premise document D be defined as a sequence of n sentences $s_1, s_2, \cdots, s_n$ such that $D = [s_1, s_2, \cdots, s_n]$. These sentences are naturally grouped into m consecutive paragraphs $P = [p_1, p_2, \cdots, p_m]$ such that each each sentence s_i belongs to only one paragraph p_j. We leverage pre-trained BERT language model to obtain the embedding of every sentence and paragraph nodes. The final representation for each sentence s_i and paragraph p_j is obtained by extracting the hidden vector of the CLS token as given by Emb(s_i) = BERT ($[[CLS]; H; [SEP]; s_i; [SEP]]$) and Emb($p_j$) = BERT ($[[CLS]; H; [SEP]; p_j; [SEP]]$), respectively. Here H denotes the hypothesis text which is also encoded as h =BERT($[[CLS]; H; [SEP]]$). $[CLS]$ and $[SEP]$ are symbols that indicate the beginning and ending of a text input, respectively.

Document Graph Construction: The document is then modeled as a hierarchical graph $D_G = (V, E)$ to capture the premise document structure. Here, $V = \{V_p, V_s, V_h\}$, where V_p, V_s, V_h are nodes corresponding to all the paragraphs, all the sentences and the hypothesis, respectively. The set of edges (E) of the Document Graph encodes four types of relations between the nodes mentioned below:

(1) Structural Relations (R_{str}): Hypothesis-Paragraph edges and Paragraph-Sentence Affiliation edges model the hierarchical structure of the document through a directed edge from the hypothesis node to each paragraph node and from a paragraph node to each constituent sentence, respectively. Further, Paragraph-Paragraph Adjacency and Sentence-Sentence Adjacency links preserve the sequential ordering for consecutive paragraph and sentence nodes through directed edges.

(2) Topical Relations (R_{top}): Sentence-Sentence Topical Consistency connections model the topical consistency between a pair of sentences by constructing sentence-level topical representations via latent Dirichlet allocation (Blei et al., 2003). Given a pair of sentences s_i and s_j, we extract latent topic distribution $lda_i, lda_j \in R^l$ for each sentence which are joined if the Helinger $H(lda_i, lda_j)$ distance between them is greater than 0.5.

(3) Entity-centric Relations (R_{ent}): Sentence-

Sentence Entity Overlap connections explicitly model the sentence-level interactions between entity spans by adding an undirected edge between two sentence nodes if they share one or more named entities. Further, Sentence-Sentence Entity Coreference connections join two sentences by an undirected edge if the sentences share mentions referring to the same real world entity.

(4) Concept-Similarity Relations (R_{sim}): Sentences conceptually similar to other sentences and the hypothesis are connected to each other to account for presence of related events and topics in two sentences. We propose Sentence-Sentence ConceptNet Similarity using ConceptNet Numberbatch (CN). Let $A_i^{cn} = [a_1, a_2, \cdots, a_l]$ be the ConceptNet Numberbatch embeddings for the words in sentence $s_i = [w_1, w_2, \cdots, w_M]$ respectively. Here, if a word w_q does not have its corresponding embedding in CN, we simply set its vector a_q to zero. Further, we introduce Hypothesis-Sentence Knowledge Similarity (using KnowBert embedding) connections that add weighted undirected edges between sentence-sentence and hypothesis-sentence node pairs, respectively. KnowBert representations are obtained by encoding text using the pre-trained KnowBert language model as $A_i^{kbrt} =$ KnowBERT($[s_i]$). The edge weights $\varepsilon_(i, j)$ between the input vector pairs (a_i, a_j) is cosine similarity between the knowledge-based semantic embeddings of the input texts.

$$\varepsilon_(i,j) = \begin{cases} cosine(A_i^{cn}, A_j^{cn}) & \text{if } a_i, a_j \in S \\ cosine(A_i^{kbrt}, A_j^{kbrt}) & \text{if } a_i == H, a_j \in S \end{cases}$$

Paragraph Pruning using *Subgraph Pooling*: Long documents are structured as a sequence of paragraphs such that each paragraph may be topically coherent to itself and neighboring paragraphs. As such, paragraphs unrelated to a given hypothesis may be ignored to reduce distractor cues. Graph pooling (Grattarola et al., 2021) is a popular method for graph coarsening. Unlike previous methods such as gPool (Gao and Ji, 2019) and SAGPool (Lee et al., 2019) that pool entire graph, we propose attention-based *Subgraph Pooling* layer which can select top rank nodes from a predefined subset of nodes in the graph. *Subgraph Pooling* layer can selectively drop irrelevant paragraph nodes while retaining the remaining paragraph nodes, their corresponding sentence nodes and the hypothesis node in the graph.

Suppose there are N nodes in document graph D_G with node embedding of size C with adjacency matrix $A \in \Re^{NxN}$ and feature matrix $X \in \Re^{NxC}$ .We apply GAT (Veličković et al., 2017) over D_G to obtain self-attention scores Z for all nodes. The pooling ratio η is a hyperparameter that determines the number of paragraph nodes to keep based on the value of Z. We want to select the top-rank nodes only from the set of paragraph nodes. Hence, we use a hard mask $\mu = \{1|x_i \in P \forall X; 0\}$ that is 1 for all paragraph nodes P, otherwise zero. We perform an element-wise multiplication ($\odot$) between attention scores and mask values to get a soft mask $Z_P = Z \odot \mu$. Top-rank operation ranks returns the indices of top η paragraphs based on Z_P. Node indices corresponding to the set of selected top-η paragraphs added to the set of sentence nodes minus those belonging to the pruned paragraphs ($\text{idx}_{S-S_{P-P_\eta}}$) and hypothesis ($\text{idx}_H$) are selected as follows: idx = top-rank(Z_P, η)+$\text{idx}_{S-S_{P-P_\eta}}$ + idx_H. The combined index tensor (idx) contains the indices of all the nodes selected in the final graph $D_G^{'}$. $X^{'}(\text{idx}, :)$ and $\widetilde{A} = A(\text{idx}, \text{idx})$ perform the row and/or column extraction to form the adjacency matrix and the feature matrix of $D_G^{'}$. The attention scores for selected nodes Z_{idx} act as gating weights for node features after filtering which controls the information flow and makes the whole procedure trainable by back-propagation as given by: $\widetilde{X} = X^{'} \odot (Z_{idx})$.

Asynchronous Graph Update: Graph Neural Networks (GNN) are useful for multi-hop reasoning on hierarchical graphs comprising of different levels of granularity (questions, paragraphs, sentences, entities)Fang et al. (2019); Zhang et al. (2020); Chen et al. (2021). However, GNN's perform message passing synchronously at each step of the graph update, ignoring the fact that different relationship (edge) types may have different priorities. In order to overcome this challenge, we propose to use Asynchronous Graph Update (Li et al., 2021) to perform sequential graph updates corresponding to all relationship types in $R \in \{R_{str}, R_{top}, R_{ent}, R_{sim}\}$ to enhance the effectiveness of multi-hop reasoning.

Optimal Evidence Selection (E^{NLI}): To select the set of most relevant evidence sentences E, we hypothesize that a sentence s_i from document D is important for NLI prediction if including the corresponding sentence as part of evidence set can improve the performance of NLI label prediction model (M^{NLI}). We design an iterative process

for sentence selection such that at step $k+1$ in the process ($k \geq 0$), a sentence s_i^{k+1} is chosen which has not been selected previously in evidence set $E_k = \{s_1*, \cdots s_k*\}$ at step k. We employ a Long Short Term Memory Network (LSTM) over previously selected k sentences to select a relevant sentence at time step $k+1$. At step 0, the initial hidden state h_0 for LSTM is set to zero. At step $k+1$, we use the hidden state h_k of LSTM from prior step to assign a score sc_i^{k+1} for each sentence node $s_i \in S - E_k$. The sentence with highest selection score is considered for selection at this step as given by $sc_i^{k+1} = sigmoid(FFN([x_i : h_k]))$ and $s_{k+1}* = argmax_{s_i \in S-E_k}(sc_i^{k+1})$, where FFN is a two-layer feed-forward network. In particular, if selecting $s_{k+1}*$ causes the number of words in the selected sentences so far to exceed the context encoder length limit (eg., 512 tokens for BERT), the selection process stops and $s_{k+1}*$ is not included in the evidence set E (i.e., $E = \{s_1*, \cdots, s_k*\}$ in this case). Otherwise, the selection process continues to the next step and $s_{k+1}*$ will be chosen and included in E (i.e.,$E = \{s_1*, \cdots, s_{k+1}*\}$). The hidden state of LSTM is also updated for the current step, i.e., $h_{k+1} = LSTM(h_k, x_{k+1}*)$, to prepare for the continuation of sentence selection.

Evidence Selection Reward Function: In order to train the evidence selection module, we employ the REINFORCE algorithm (Williams, 1992) and incorporate the following information signals in the reward function of REINFORCE to better supervise the training process. In order to train the evidence selection module, we employ the REINFORCE algorithm (Williams, 1992). We incorporate the following information signals in the reward function of REINFORCE to better supervise the training process:

(1) Task Reward ϕ_{perf}: We compute this reward based on the NLI task prediction performance. In order to measure the impact of the selected context, we use a T-5 model (Raffel et al., 2019a) pretrained on MNLI corpus (Williams et al., 2017) to predict the NLI label for the given hypothesis + context pair. $\phi_{perf}(E)$ is set to 1 if the final prediction is correct; and 0 otherwise.

(2) Semantic Reward ϕ_{sem}: We propose that the evidence sentences should be semantically similar to the hypothesis. Our motivation is that similar context sentences (e.g., discussing the same events or entities) provide more relevant information for the NLI prediction. We include the semantic similarity between the selected evidence sentences in E and the hypothesis as measured by the cosine similarity (i.e.,$\bigodot$) between their sentence embeddings computed using SimCSE[1](Gao et al., 2021).

(3) Evidence Reward ϕ_{bleu}: We seek to promote evidence sentences having a high overlap with the target ground truth evidence. In many cases, the target evidence length may be way less than 512 token limit. Hence, our motivation is to reward the lexical overlap while penalizing verbosity arising at evidence selection stage. We calculate the BLEU score between the selected evidence E and ground truth evidence E_{gt}: $\phi_{bleu} = BLEU(E, E_{gt})$ This reward can only be applied for cases where ground truth evidence annotation is present.

(4) Multihop Reward ϕ_{mhop}: The motivation for this reward is that a sentence should be preferred to be included in E by the selection process if there are common entities mentions with the hypothesis. Moreover, connected sentences by the virtue of common entity mentions are more likely to refer to the same events. Hence, we leverage the subgraph similarity of the learned node embeddings of the selected evidence and their first degree node connections through entity-centric relations with the hypothesis node in $G_D^{''}$. We perform max-pooling operation over the concatenated node embeddings of the corresponding evidence sentences and their first degree node connections joined by R_{ent}: $\hat{E} = maxpool(v_1 \bigoplus v_2, \cdots, v_k | s_i \in E, i \in \{1, \cdots, k\})$, where $\bigoplus$ means embedding concatenation. Finally, we compute the dot-product between $\hat{E}$ and node embedding of the hypothesis node h as $\phi_{mhop} = \hat{E}.h$.

3.1 Training DocInfer

NLI Prediction Loss: We combine the final representations corresponding to the learnt graph structure (g_{out}) and selected evidence text (t_{out}). We aggregate the embeddings corresponding to the selected sentence nodes in $D_G^{''}$ and the hypothesis node using a summation-based graph-level readout function (Xu et al., 2018) as $g_{out} = \rho(\sum_{v \in D_G^{''}} W_g V_i^T)$. The words in the evidence sentences are joined in order of their appearance in document D and input to the context encoder $t_{out} = Encoder([CLS]s_1; s_2, \cdots, s_k)$. g_{out} and t_{out} are concatenated and passed through two dense fully-connected layers: $z = ReLU(Dense(t_{out} \bigoplus g_{out})$. This is

[1] https://github.com/princeton-nlp/SimCSE

followed by a Softmax layer to predict entailment/contradiction/neutral by utilizing the negative log-likelihood loss: $L_{pred} = -P(y|z)$.

Evidence Selection Loss: The overall reward function to train our evidence selection module is $\phi(E) = \phi_{perf} + \phi_{sem} + \phi_{bleu} + \phi_{mhop}$. Using REINFORCE, we seek to minimize the negative expected reward $\phi(E)$ over the possible choices of E as $L_{sent} = -\mathbb{E}_{E\sim P(E|D,H)}[\phi(E)]$, and $L_{sent} = -\mathbb{E}_{E\sim P(E|D,H)}[\phi(E)]\nabla \log(P(E|H,D))$. Finally, the probability of the selected sequence E is computed via $P(E|H,D) = \prod_{k=0,\cdots,K-1} P(s_{k+1} * |H, D, s_{i\leq k}*)$, which is obtained via softmax over selection scores for sentences in S at selection step $k+1$.

Joint NLI Prediction and Evidence Selection: During training, the NLI prediction model M^{NLI} and the evidence selection module E^{NLI} are trained alternatively. At each update step, E^{NLI} first selects optimal evidence sentences E that form a shortened document D^e. M^{NLI} uses E to predict the NLI label. The parameters of M^{NLI} are updated using the gradient of NLI prediction loss L_{pred}, keeping parameters of evidence extraction module constant. Next, the parameters of the evidence selection module are updated using the gradient of L_{sent}, keeping parameters of M^{NLI} constant. This process repeats until convergence. At test time, evidence sentences are first selected and then consumed by the prediction model to perform NLI prediction.

4 Experiments

4.1 Datasets for Document-level NLI

We use the following three datasets to benchmark document-level NLI approaches. **(1) DocNLI** (Yin et al., 2021): A large-scale document-level NLI dataset obtained by reformatting mainstream NLP tasks such as question answering and document summarization. **(2) ContractNLI** (Koreeda and Manning, 2021): NLI dataset of 607 contract documents annotated with ground truth evidence sentences. **(3) ConTRoL (Liu et al., 2021)**: A passage-level NLI dataset of exam questions that requires logical, analytical, temporal, coreferential reasoning, and information integration over multiple premise sentences. **(4) CaseHoldNLI**, the fourth and novel NLI dataset introduced in this paper, in the legal judicial reasoning domain for identifying the governing legal rule (also called "Holding") applied to a particular set of facts. It is

	System	DocNLI Dev F1	DocNLI Test F1
Baselines	Majority	19.7	19.9
	BERT (Hypothesis-only)	21.9	22.0
	BERT$_{base}$ (Devlin et al., 2018)	63.1	60.1
	BERT$_{large}$ (Devlin et al., 2018)	63.5	61.1
	RoBERTa$_{base}$ (Liu et al., 2019a)	61.0	59.5
	RoBERTa$_{large}$ (Liu et al., 2019a)	63.1	61.3
	T5 (Raffel et al., 2019b)	62.9	61.1
	Longformer (Beltagy et al., 2020)	46.1	44.4
	GEAR (Zhou et al., 2019)	67.8	63.3
	KGAT (Liu et al., 2019b)	68.5	64.8
	HESM (Subramanian and Lee, 2020)	68.9	65.0
	DREAM (Zhong et al., 2020)	69.7	65.9
	TARSA (Si et al., 2021)	70.4	66.4
	EvidenceNet (Chen et al., 2022)	72.6	68.5
Ours	`DocInfer (w/ RoBerta)`	**75.5**	**72.3**

Table 1: Results comparing performance of `DocInfer` with baselines on DocNLI dataset. **Bold** denotes the best performing model. LightCyan and Yellow show best performing baseline and Transformer model.

	System	ContractNLI Acc (%)	F1 (C)	F1 (E)	mAP	PR@80
Baselines	Majority	67.4	8.3	42.8	-	-
	BERT$_{large}$ (Devlin et al., 2018)	77.5	25.7	76.4	0.822	0.763
	T5 (Raffel et al., 2019b)	73.2	21.2	69.1	0.786	0.575
	Longformer (Beltagy et al., 2020)	71.2	19.2	70.4	0.755	0.648
	BigBird (Zaheer et al., 2020)	71.5	18.8	70.9	0.776	0.630
	GEAR (Zhou et al., 2019)	78.4	26.9	78.3	0.909	0.774
	KGAT (Liu et al., 2019b)	78.9	27.8	79.2	0.914	0.773
	HESM (Subramanian and Lee, 2020)	28.2	79.5	79.9	0.916	0.789
	DREAM (Zhong et al., 2020)	79.8	29.3	80.4	0.919	0.786
	TARSA (Si et al., 2021)	80.4	29.4	80.5	0.916	0.783
	SpanNLI-Bert$_{large}$ (Koreeda and Manning, 2021)	87.5	35.7	83.4	0.922	0.793
Ours	`DocInfer (w/ Bert)`	**91.8**	**38.2**	**89.1**	**0.956**	**0.832**

Table 2: Results comparing performance of `DocInfer` with baselines on ContractNLI dataset. **Bold** denotes the best performing model. LightCyan and Yellow show best performing baseline and Transformer model.

sourced from the CaseHOLD dataset (Zheng et al., 2021) comprising over 53,000+ multiple choice questions. Each multiple choice question comprises of a snippet from a judicial decision along with 5 semantically similar potential holdings, of which only one is correct. We obtain the NLI-version by combining the question and the positive (negative) answer candidate as a positive (negative) hypothesis. To evaluate the dataset quality, we asked an expert to select the NLI using only the hypothesis for 10% of the test data sampled at random. The poor performance of this human baseline ($\sim 0.24F1$) validates that the dataset doesn't suffer from hypothesis bias. CaseHoldNLI dataset is comparable to challenging document-level NLI datasets with average premise length at document-scale and exceeds the maximum input length limit of BERT models. We report train/dev/test splits of each dataset.

4.2 Experiments on Downstream Tasks

(1) Fact Verification: The NLI-version of FEVER (Thorne et al., 2018) task, released by Nie et al. (2019), considers each claim as a hypothesis while the premises consist of ground truth textual evi-

	System	ConTRoL				
		Acc (%)	F1 (E)	F1 (N)	F1 (C)	F1 (O)
Baselines	Majority	40.6	57.7	0.0	0.0	19.2
	BERT$_{base}$ (Devlin et al., 2018)	47.4	42.4	50.2	46.0	46.2
	BERT$_{large}$ (Devlin et al., 2018)	50.6	45.9	53.1	49.3	49.4
	RoBERTa$_{base}$ (Liu et al., 2019a)	45.9	45.3	45.9	45.6	45.6
	BART (Lewis et al., 2020)	56.3	49.1	59.5	53.8	54.0
	Longformer (Beltagy et al., 2020)	49.8	45.6	46.8	46.2	46.2
	BigBird (Zaheer et al., 2020)	49.3	46.0	45.1	46.0	46.1
	BART-NLI (Liu et al., 2021)	57.2	49.0	60.4	54.2	54.5
	BART-NLI-FT (Liu et al., 2021)	57.5	49.3	60.6	54.6	55.0
	KGAT (Liu et al., 2019b)	59.1	50.6	61.8	55.7	56.6
	HESM (Subramanian and Lee, 2020)	59.3	50.9	62.3	56.1	56.6
	DREAM (Zhong et al., 2020)	59.8	51.1	62.0	56.1	56.3
	HGRGA (Lin and Fu, 2022)	60.6	52.9	62.4	58.7	58.0
	EvidenceNet (Chen et al., 2022)	61.8	56.4	64.2	64.3	61.6
Ours	DocInfer (w/ BART)	**66.7**	**60.6**	**67.1**	**69.6**	**67.4**

Table 3: Results comparing performance of DocInfer with baselines ConTRoL dataset. **Bold** denotes the best performing model. LightCyan and Yellow show best performing baseline and Transformer model.

	System	CaseHoldNLI		
		P	R	F1
Baselines	Majority	0.0	1.0	0.0
	BERT$_{base}$ (Devlin et al., 2018)	42.2	46.3	44.2
	RoBERTa$_{base}$ (Liu et al., 2019a)	42.2	46.3	44.2
	T5 (Raffel et al., 2019b)	41.5	43.5	42.5
	Legal-BERT (Zheng et al., 2021)	46.5	47.9	47.1
	Longformer (Beltagy et al., 2020)	40.1	43.3	41.6
	GEAR (Zhou et al., 2019)	42.9	46.8	44.8
	HESM (Subramanian and Lee, 2020)	44.0	48.0	45.9
	HGRGA (Lin and Fu, 2022)	45.4	49.4	47.3
	EvidenceNet (Chen et al., 2022)	47.3	50.5	48.8
Ours	DocInfer (w/ Legal-Bert)	**51.3**	**53.1**	**52.2**

Table 4: Results comparing performance of DocInfer with baselines on CaseHoldNLI dataset. **Bold** denotes the best performing model. LightCyan and Yellow show best performing baseline and Transformer model.

dence and other randomly sampled related text.

(2) Multi-choice Question Answering: The NLI-version MCTest (Richardson et al., 2013) combines the question and the positive (negative) answer candidate as a positive (negative) hypothesis. Presence of limited labeled data makes them both good benchmarks to investigate the performance of document-level NLI models on annotation-scarce tasks. We evaluate DocInfer trained on DocNLI dataset and report F1 scores for both tasks. We follow the "FEVER-binary" and "MCTest-NLI" settings proposed in Yin et al. (2021).

(3) Contract Clause Retrieval(Łukasz Borchmann et al., 2020): is a task to identify spans in a target document representing clauses analogous (i.e. semantically and functionally equivalent) to the provided seed clauses from source documents. We reformulate this as an NLI task where the seed clauses are concatenated to form the hypothesis, and the target document is the premise. We test the evidence selection capabilities of DocInfer trained on ContractNLI dataset for identifying relevant sentence-level spans in the premise for the clause retrieval task. The dataset has 1300 examples each for validation and test to tune and test the paragraph selection hyperparameter η. We followed the evaluation framework specified in Łukasz Borchmann et al. (2020) of few (1-5) shot setting and report Soft F1 score.

5 Results

Table 1-4 compares the performance of DocInfer against other baselines on DocNLI, ContractNLI, ConTRoL, and CaseHoldNLI datasets. Similar to (Yin et al., 2021), we truncate the hypothesis-premise pair sequence to appropriate maximum input length for input to Transformer models. BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019a), DeBERTa (He et al., 2020), BART (Lewis et al., 2020) show superior performance for DocNLI, ContractNLI, and ConTRoL datasets, respectively. Legal-BERT (Chalkidis et al., 2020) outperforms other Transformer language models on CaseHoldNLI dataset due to its high domain-specificity of legal language. However, they are challenged by their input length restriction of 512 tokens for contextually reasoning over long premise lengths. Consistent with observations of (Yin et al., 2021), large input Transformer models such as Longformer (Beltagy et al., 2020) and BigBird (Zaheer et al., 2020) that can handle up to 4096 tokens underperform traditional BERT-like models on all four datasets. We attribute this to the presence of distractors in long documents and the inability of these models to reason in a multihop fashion. BART-NLI which is pretrained on sentence-level NLI (Liu et al., 2021) improves over naive Transformers but still struggles due to limited captured context.

We also re-purpose several strong baseline methods from the Fact Extraction and Verification (FEVER 1.0) task. by reformulating the document retrieval and claim verification steps to paragraph retrieval and textual entailment, respectively. GEAR, KGAT, and HGRGA model the document as a dense fully-connected graph, leading to distractor interactions confounding the reasoning process. They are also devoid of linguistic information about entities, topics or commonsense knowledge. HESM uses document chunking which hinders contextual reasoning for far-away chunks. DREAM and TARSA use semantic role labeling and topic modeling, respectively, to identify phrase interaction but lack entity-level information required to resolve coreferences across document. EvidenceNet and SpanNLI emerge as strong baseline models for our work. DocInfer outperforms SpanNLI and EvidenceNet due to its ability to iteratively

select important evidence sentences in the premise and simultaneously utilize multihop interactions between related evidences. **Impact of Input Length**: DocInfer achieves SOTA performance on all four datasets and maintains steady improvements over corresponding baseline models with increasing in input lengths. **Choice of context encoder in NLI prediction**: One of the merits of the our approach is that it is extensible and can utilize any domain-specific transformer language models for context encoding to further augment performance. We evaluate the choice of context encoder for different datasets. DocInfer gives SOTA performance using RoBERTa for DocNLI, BERT for ContractNLI, BART for ConTRol, and Legal-BERT for CaseHoldNLI, in the prediction model.

Ablation Study of DocInfer: Table 5 shows ablations for the document graph relations, module components and reward functions. We observe that concept relation is critical in all data settings due to the need for external knowledge-based semantic representation for connecting related concepts across sentences. Removing any of the relations does not degrade the performance below EvidenceNet (Chen et al., 2022) or SpanNLI baselines. This is important for adapting our method to new domains where existing linguistic parsers maybe noisy or non-existent. Cells in Table 5 highlighted in red shows the ablation of individual components such that removing paragraph pruning mechanism severely deteriorates model performance as the model has to evaluate an exponentially larger number of candidate evidences during evidence selection stage. In absence of optimal evidence selection, we treat evidence extraction as a binary classification task over each sentence node along with NLI label given by the "readout" function similar to KGAT (Liu et al., 2019b). The severe performance drop of DocInfer model in absence of evidence selection component highlights its importance for document NLI task. Asynchronous graph update adds incremental value to DocInfer owing to its relation-specific message passing. Evidence Selection and Paragraph Pruning components are most critical for SOTA performance of DocInfer. Greedy selection instead of REINFORCE significantly decreases performance. Concept relations are most beneficial for DocInfer, followed by topical and entity relations. Evidence, semantic, multihop and task rewards most help ContractNLI, ConTRoL, DocNLI, and CaseHoldNLI.

Impact of reward function: Table 5 shows that removing any reward component (i.e., task, semantic, evidence, multihop) significantly hurts the overall performance, thus clearly demonstrating their individual importance. To assess the necessity of the multi-step selection using REINFORCE, we eliminate multistep selection strategy and perform one-shot sentence selection where the top k sentences with highest selection scores from the first step are selected. We call this setting as greedy evidence selection and show that the elimination of multistep selection drops performance, suggesting that selecting sentences incrementally conditioning on previously selected sentences is advantageous.

Performance of DocInfer on downstream tasks: Table 6 shows the evaluation of DocInfer along with RoBERTa-large and EvidenceNet (Chen et al., 2022) baselines and RoBERTa model from Yin et al. (2021) on FEVER-binary and MCTest tasks. We train all models on DocNLI dataset to benefit from cross-task transfer and for minimizing domain shift. We then inference all models in two settings: (i) without task specific fine-tuning, and (ii) with fine-tuning on the end task. DocInfer model consistently outperforms baselines across both tasks in case of without fine-tuning (FEVER-binary: +0.8 F1, MCTest v160: +1 F1, MCTest v500: +0.6 F1) and with fine-tuning (FEVER-binary: +0.9 F1, MCTest v160: +0.5 F1, MCTest v500: +0.2 F1). We observe that both the tasks require the models to capture topic coherence, knowledge-based semantics, and entities interactions as removing graph relations severely degrades the performance.

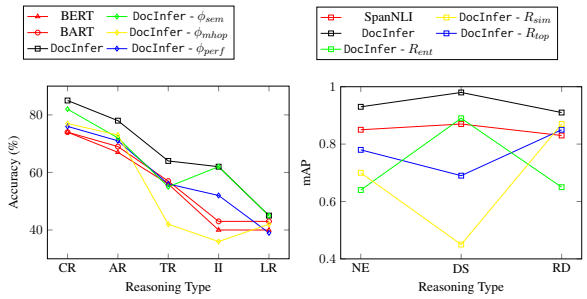
Evidence selection for clause retrieval focuses on selecting evidence spans in the target document (premise) given the entailment relation with seed clauses (hypothesis). The task is unsupervised in nature (has no training set). We test the evidence selection module (E^{NLI}) of the DocInfer model and its ablated variants (without paragraph pruning and reward functions), all pre-trained on ContractNLI dataset. Table 7 shows that DocInfer model with BERT as the context encoder outperforms strong baselines by approximately 5%. Removing paragraph pruning significantly degrades the performance, highlighting the need to prune distractor paragraphs for retrieving relevant information. Presence of each reward function to maintain the performance of DocInfer indicates the linguistic importance of each reward. Formulating the task as NLI helps contextualize the seed clauses with

	System	DocNLI		ContractNLI					ConTRoL					CaseHoldNLI		
		Dev F1	Test F1	Acc (%)	F1 (C)	F1 (E)	mAP	PR@80	Acc (%)	F1 (E)	F1 (N)	F1 (C)	F1 (O)	P	R	F1
	Context Encoder	RoBERTa		BERT					BART					Legal-BERT		
Ours	**DocInfer**	75.5	72.3	91.8	38.2	89.1	0.956	0.832	66.7	63.6	67.1	69.6	67.4	51.3	53.1	52.2
Ablation	DocInfer w\Concept Relations	72.6	70.7	90.7	35.4	87.2	0.928	0.810	62.3	62.6	66.6	66.1	64.2	50.6	52.9	51.7
	DocInfer w\Topical Relations	72.2	70.4	90.6	33.6	86.6	0.925	0.819	60.5	59.4	66.7	63.2	63.1	49.5	51.5	50.5
	DocInfer w\Entity Relations	72.5	70.2	90.2	31.1	85.7	0.921	0.812	59.8	58.8	66.0	62.8	62.5	49.0	51.4	50.2
	DocInfer w\o Asynchronous Graph Update	73.2	71.5	89.5	36.3	84.3	0.923	0.813	57.6	61.0	59.7	65.1	64.8	48.8	50.5	49.6
	Greedy Evidence Selection	67.5	64.9	88.3	36.0	84.1	0.876	0.780	56.5	56.3	56.1	55.2	55.8	46.8	46.2	46.7
	DocInfer w\o Paragraph Pruning	65.6	64.5	85.4	35.9	83.9	0.855	0.742	51.8	47.0	45.5	48.0	46.8	44.8	46.2	45.5
	DocInfer w\o Evidence Selection	65.0	63.6	83.7	35.5	83.5	0.825	0.715	51.6	46.7	45.0	47.8	46.5	44.4	45.9	45.1
	DocInfer w\Task Reward	70.5	68.5	90.1	33.9	86.7	0.907	0.769	61.9	61.0	64.0	65.2	63.4	47.4	47.4	47.4
	DocInfer w\Evidence Reward	-	-	90.8	36.4	87.5	0.916	0.805	-	-	-	-	-	-	-	-
	DocInfer w\Semantic Reward	71.4	69.0	90.6	34.2	84.4	0.912	0.778	63.3	62.5	63.9	64.7	63.7	46.6	46.8	46.7
	DocInfer w\Multihop Reward	71.6	69.5	90.5	35.5	85.6	0.910	0.785	61.6	62.0	63.2	64.0	63.0	46.1	46.9	46.5

Table 5: Results comparing ablative components of DocInfer model and analysis of using a single reward/relation at a time. Darker green represents better F1 performance, darker red shows negative impact. Evidence reward is applicable only for ContractNLI which has ground truth evidence annotations.

System	Fine-tune	FEVER binary	MCTest v160	MCTest v500
RoBERTa	✗	88.4	90.0	85.8
EvidenceNet	✗	88.7	90.6	86.0
DocInfer†	✗	**89.2***	**91.0***	**86.4***
DocInfer† w/o R	✗	86.3	87.5	82.5
RoBERTa	✓	89.4	91.0	91.0
EvidenceNet	✓	89.9	90.8	90.6
DocInfer†	✓	**90.5***	**91.5***	**91.2***
DocInfer† w/o R	✓	86.6	88.5	87.5

Table 6: Performance comparison of DocInfer† with RoBERTa (large) and EvidenceNet on FEVER-binary and MCTest-NLI. † means using RoBERTa as context encoder.



(a) ConTRoL Dataset (b) ContractNLI Dataset

Figure 2: Error analysis across reasoning types (accuracy%) and challenging phenomenon (mAP) on the test set of ConTRoL and ContractNLI datasets.

premise as opposed to earlier techniques of isolated vectorization and naive aggregation by (Łukasz Borchmann et al., 2020) .

Qualitative Analysis: Figure 2 shows qualitative analysis across different reasoning types on the test set of ConTRoL dataset. The results provide evidence that the multihop and semantic similarity rewards are important for coreference reasoning (CR) due to reasoning over multiple mentions and noun phrases. Multihop reward also helps improve Information aggregation (II) which requires combining information from multiple paragraphs. Task reward benefits logical reasoning as it focuses on logical inference of human language. DocInfer is unable to handle temporal and analytical reasoning cases. We further analyze the evidence extraction

System	Soft F1
Tf-IDF	0.39
GloVe (300D, EDGAR)	0.41
Sentence-BERT	0.32
USE	0.38
BERT	0.35
RoBERTa	0.31
GPT-1	0.49
GPT-2 (large)	0.51
DocInfer‡(pretrain ContractNLI)	**0.53***
DocInfer‡ w\o Paragraph Pruning	0.42
DocInfer‡ w\o Task Reward (ϕ_{perf})	0.48
DocInfer‡ w\o Semantic Reward (ϕ_{sem})	0.45
DocInfer‡ w\o Evidence Reward (ϕ_{bleu})	0.45
DocInfer‡ w\o Multihop Reward (ϕ_{mhop})	0.44
Human	0.84

Table 7: Performance comparison of DocInfer and its configurations pretrained on ContractNLI and tested for clause retrieval without fine-tuning on Contract Discovery dataset (Łukasz Borchmann et al., 2020). ‡: BERT as context encoder.

mAP on ContractNLI dataset across diverse challenging phenomena. Entity relations are critical for resolving reference to definitions (RD) as they are anchored together through common mentions. Concept similarity links play an important role in resolving information spread out between discontinuous spans based on commonsense reasoning. DocInfer handles evidence identification for all studied phenomena better than SpanNLI.

6 Conclusion and Future Work

We introduce DocInfer, a document-level NLI model that uses enriched hierarchical document graph through inter-sentence relations, performs paragraph pruning using *SubGraph Pooling* layer, and optimally selects evidence sentences using REINFORCE algorithm to outperform SOTA methods on four doc-NLI datasets, including our propose CaseHoldNLI on legal judicial reasoning.DocInfer is useful for downstream fact verification, multi-choice QA and legal clause retrieval tasks. For future work, we intend to integrate temporal knowledge and analytical reasoning into our model to improve the performance.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

David M. Blei, A. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Chonghao Chen, Fei Cai, Xuejun Hu, Wanyu Chen, and Honghui Chen. 2021. Hhgn: A hierarchical reasoning-based heterogeneous graph neural network for fact verification. *Information Processing & Management*, 58(5):102659.

Zhen-Heng Chen, Siu Cheung Hui, Fuzhen Zhuang, Lejian Liao, Fei Li, Meihuizi Jia, and Jiaqi Li. 2022. Evidencenet: Evidence fusion network for fact verification. *Proceedings of the ACM Web Conference 2022*.

Ido Dagan, Bill Dolan, Bernardo Magnini, and Dan Roth. 2010. Recognizing textual entailment: Rational, evaluation and approaches–erratum. *Natural Language Engineering*, 16(1):105–105.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2019. Hierarchical graph network for multi-hop question answering. *arXiv preprint arXiv:1911.03631*.

Hongyang Gao and Shuiwang Ji. 2019. Graph u-nets. In *international conference on machine learning*, pages 2083–2092. PMLR.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Daniele Grattarola, Daniele Zambon, Filippo Maria Bianchi, and Cesare Alippi. 2021. Understanding pooling in graph neural networks. *ArXiv*, abs/2110.05292.

Andreas Hanselowski, H. Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018. Ukp-athene: Multi-sentence textual entailment for claim verification. *ArXiv*, abs/1809.01479.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Nils Holzenberger, Andrew Blair-Stanek, and Benjamin Van Durme. 2020. A dataset for statutory reasoning in tax law entailment and question answering. *arXiv preprint arXiv:2005.05257*.

Kelvin Jiang, Ronak Pradeep, Jimmy J. Lin, and David R. Cheriton. 2021. Exploring listwise evidence reasoning with t5 for fact verification. In *ACL*.

Xiaomian Kang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2020. Dynamic context selection for document-level neural machine translation via reinforcement learning. *ArXiv*, abs/2010.04314.

Yuta Koreeda and Christopher Manning. 2021. ContractNLI: A dataset for document-level natural language inference for contracts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1907–1919, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Junhyun Lee, Inyeop Lee, and Jaewoo Kang. 2019. Self-attention graph pooling. In *International conference on machine learning*, pages 3734–3743. PMLR.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*.

Ronghan Li, Lifang Wang, Shengli Wang, and Zejun Jiang. 2021. Asynchronous multi-grained graph network for interpretable multi-hop reading comprehension. In *IJCAI*, pages 3857–3863.

Hongbin Lin and Xianghua Fu. 2022. Heterogeneous-graph reasoning and fine-grained aggregation for fact checking. *Proceedings of the Fifth Fact Extraction and VERification Workshop (FEVER)*.

Hanmeng Liu, Leyang Cui, Jian Liu, and Yue Zhang. 2021. Natural language inference in context-investigating contextual reasoning over long texts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13388–13396.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019a. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2019b. Fine-grained fact verification with kernel graph attention network. *arXiv preprint arXiv:1910.09796*.

Puneet Mathur, Franck Dernoncourt, Quan Hung Tran, Jiuxiang Gu, Ani Nenkova, Vlad Morariu, Rajiv Jain, and Dinesh Manocha. 2022a. Doclayouttts: Dataset and baselines for layout-informed document-level neural speech synthesis. *Proc. Interspeech 2022*, pages 451–455.

Puneet Mathur, Rajiv Jain, Franck Dernoncourt, Vlad Morariu, Quan Hung Tran, and Dinesh Manocha. 2021. Timers: document-level temporal relation extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 524–533.

Puneet Mathur, Vlad Morariu, Verena Kaynig-Fittkau, Jiuxiang Gu, Franck Dernoncourt, Quan Hung Tran, Ani Nenkova, Dinesh Manocha, and Rajiv Jain. 2022b. Doctime: A document-level temporal dependency graph parser. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 993–1009.

Nghia Trung Ngo, Tuan Ngo Nguyen, and Thien Huu Nguyen. 2020. Learning to select important context words for event detection. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 756–768. Springer.

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6859–6866.

Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2020. Coliee 2020: methods for legal document retrieval and entailment. In *JSAI International Symposium on Artificial Intelligence*, pages 196–210. Springer.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019a. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019b. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 193–203.

Jiasheng Si, Deyu Zhou, Tongzhe Li, Xingyu Shi, and Yulan He. 2021. Topic-aware evidence reasoning and stance-aware aggregation for fact verification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

Shyam Subramanian and Kyumin Lee. 2020. Hierarchical evidence set modeling for automated fact extraction and verification. In *EMNLP*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.

Hieu Man Duc Trong, Nghia Ngo Trung, Linh Van Ngo, and Thien Huu Nguyen. 2022. Selecting optimal context sentences for event-event relation extraction.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Amir Pouran Ben Veyseh, Minh Van Nguyen, Nghia Ngo Trung, Bonan Min, and Thien Huu Nguyen. 2021. Modeling document-level context for event detection via important context selection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5403–5413.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*.

Wenpeng Yin, Dragomir Radev, and Caiming Xiong. 2021. Docnli: A large-scale dataset for document-level natural language inference. *arXiv preprint arXiv:2106.09449*.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: Transformers for longer sequences. *ArXiv*, abs/2007.14062.

Min Zhang, Feng Li, Yang Wang, Zequn Zhang, Yanhai Zhou, and Xiaoyu Li. 2020. Coarse and fine granularity graph reasoning for interpretable multi-hop question answering. *IEEE Access*, 8:56755–56765.

Chen Zhao, Chenyan Xiong, Corby Rosset, Xia Song, Paul N. Bennett, and Saurabh Tiwary. 2020. Transformer-xh: Multi-evidence reasoning with extra hop attention. In *ICLR*.

Lucia Zheng, Neel Guha, Brandon R Anderson, Peter Henderson, and Daniel E Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, pages 159–168.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Reasoning over semantic-level graph for fact checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. Gear: Graph-based evidence aggregating and reasoning for fact verification. *arXiv preprint arXiv:1908.01843*.

Łukasz Borchmann, Dawid Wisniewski, Andrzej Gretkowski, Izabela Kosmala, Dawid Jurkiewicz, Lukasz Szalkiewicz, Gabriela Pałka, Karol Kaczmarek, Agnieszka Kaliska, and Filip Grali'nski. 2020. Contract discovery: Dataset and a few-shot semantic retrieval challenge with competitive baselines. In *FINDINGS*.

Appendix

A Limitations

Through careful analysis of error cases, we found that there are two main types of prediction errors from the proposed model. First, the model is unable to reason over temporal and causal aspects. For example, in the hypothesis "Repayment terms will be finalized before disbursement but prior to loan approval" while the evidence states "Repayment terms are subject to loan approval and monthly disbursement of interest amount". `DocInfer` does not recognize the fact that there is a temporal order between events "repayment", "approval", and "disbursement". Tackling this type of error requires temporal relation prediction between different events. The second type of errors is mainly due to contradictory/missing information in the retrieved evidence required for analytical inference. For example, the model predicts that the hypothesis "Insurance prices are all time high" contradicts with evidences "Insurance prices increase with increase in pollution" and "Protest marches for restoring pollution control board were censored". The model prunes a relevant piece of evidence - "Pollution control board tabled policy for curbing air contamination in residential areas" in an otherwise irrelevant paragraph which causes loss of logical flow. **Potential Risks:** Our models are exploratory and academic in nature and should not be used for real-world legal/contractual/healthcare purposes without extensive investigations into its shortcomings/randomness/biases. **Unhandled Cases**: The current work is limited to English language and would need suitable tools in other languages to process semantic similarity, concept knowledge and topic models. Moreover, our method has been tested on limited domains of Wikipedia text, narrative stories, exam-style questions, case laws, and contracts. Applying it to life-critical scenarios such as healthcare, public safety will need further investigations.

B Ethics Statement

We utilize three publicly available datasets - DocNLI, ContractNLI and ConTRoL for evaluating document-level NLI. We also curated dataset for doc-level NLI on legal judicial case documents. We source these contract documents from a publicly available resource - CaseHOLD dataset (Zheng et al., 2021). We repurpose the dataset for our task and provide new annotations. CaseHoldNLI dataset does not violate any privacy as these documents are already in public domain as part of Harvard Case Law Corpus[2]. There is no human bias involved in such documents as they are already annotated expertly and provided openly after anonymizing any identifiable information. These documents do not restrict reuse for academic purposes and any personal information was already redacted before their original release. All documents and our experiments are restricted to English language. FEVER-binary, MCTest, and Contract Discovery datasets are also publicly available for research purposes. There was no sensitive data involved in the studies.

C Impact of Input Length

`DocInfer` achieves SOTA performance on all four datasets and maintains steady improvements over corresponding baseline models with increasing in input lengths for performance vs premise length comparison for ConTROL, DocNLI, ContractNLI, andCaseHoldNLI datasets.

D Experiments on Downstream Tasks

D.1 Fact Verification

The NLI-version of FEVER (Thorne et al., 2018) task, released by Nie et al. (2019), considers each claim as a hypothesis while the premises consist of ground truth textual evidence and some other randomly sampled related text. Yin et al. (2021) combined the "refute" and "not-enough-info" labels into a single class of "not entail", organized the dataset into train/dev/test split of 203,152/8,209/10,000 and renamed it as "FEVER-binary".

D.2 Multi-choice Question Answering

MCTest (Richardson et al., 2013) is a multi-choice QA benchmark in the domain of fictional story with one correct and three incorrect answers. The NLI-version MCTest combines the question and the positive (negative) answer candidate as a positive (negative) hypothesis. We test two versions of MCTest – MCTest-160 (70 train, 30 dev, 60 test) and MCTest-500 (300 train, 50 dev, 150 test). Presence of limited labeled data makes it a good benchmark to investigate the performance of document-level NLI models on annotation-scarce tasks. We evaluate `DocInfer` trained on DocNLI dataset and report F1 scores for both tasks.

[2]https://casestudies.law.harvard.edu/

D.3 Contract Clause Retrieval

Clause Discovery (Łukasz Borchmann et al., 2020) is a task to identify spans in a target document representing clauses analogous (i.e. semantically and functionally equivalent) to the provided seed clauses from source documents. We reformulate this as an NLI task where the seed clauses are concatenated to form the hypothesis, and the target document is the premise. We test the evidence selection capabilities of `DocInfer` for identifying relevant sentence-level spans in the premise. To this end, we test `DocInfer` trained on ContractNLI dataset without supervision for the clause retrieval task. The dataset has 1300 test examples and we tuned the paragraph selection hyperparameter η on the validation set of 1300 examples. We followed the evaluation framework specified in Łukasz Borchmann et al. (2020) of few (1-5) shot setting and report Soft F1 score.

E Data Statistics

We present the dataset statistics in Table 8.

A1: Limitations:

Through careful analysis of error cases, we found two main types of prediction errors from the proposed model: (1) unable to reason over temporal and causal relations; (2)contradictory/missing information in the retrieved evidence required for analytical inference.

A2: Potential Risks:

Our models are exploratory and academic in nature ans should not be used for real-world legal/contractual/healthcare purposes without extensive investigations into its shortcomings/randomness/biases.

B1: Citation to creators of artifacts:

We use four datasets: (i) DocNLI (Yin et al., 2021), (ii) ContractNLI (Koreeda and Manning, 2021), (iii) ConTRoL (Liu et al., 2021), (iv) CaseHold (Zheng et al., 2021). We repurpose the fourth dataset for doc-level NLI task All datasets and documents are publicly available.

Further, we use three datasets for downstream applications: (i) FIVER-binary (Nie et al., 2019), (ii) MCTest (Richardson et al., 2013), (iii) Contract Discovery (Łukasz Borchmann et al., 2020).

DocNLI Dataset: https://github.com/salesforce/DocNLI

ContractNLI Dataset: https://stanfordnlp.github.io/contract-nli/

ConTRoL Dataset: https://github.com/csitfun/ConTRoL-dataset

CaseHold Dataset: https://github.com/reglab/casehold

Contract Dicovery Dataset: https://github.com/applicaai/contract-discovery

B2: License and terms for use of data artifacts:

All the datasets are available to use for research purposes.

B3: Intended use of data artifacts:

The intended use of NLI datasets is to improve NLP reasoning and semantic understanding of text and languages. Use cases in legal and contract domains can make increase accessibility amongst non-experts and lead to AI for social good.

B4: Steps taken to protect / anonymize names, identities of individual people or offensive content:

We do not use any identifiable user data for any experiments. All persons mentioned in the dataset are anonymous or have their information publicly available.

B5: Coverage of domains, languages, linguistic phenomena, demographic groups represented in data:

Our work uses NLI dataset from Wikipedia, story narrations, exam questions, contracts and case laws in English language. Adaptation to other languages may need appropriate processing.

B6: Data statistics (train/test/dev splits):

The data statistics are given in Table 8.

E.1 Training Setup

Hyperparameters: We tune the hyperparameters for the proposed model using a grid search. All the hyperparameters are selected based on the F1 scores on the development set to find the best configurations for different datasets. In our model we use the BERT-base to encode document embeddings; LSTM and 2 layers for feed forward neural networks. The trade-off parameters α, β and γ are set to 0.5, 0.1, 0.05, respectively. We use Adam optimizer and the batch size of 16 is employed during training. We vary paragraph selection η between 1 to 10.

Dataset	Original Task	Domain	Premise Length	Hypothesis Length	Evidence	Train				Validation				Test			
						E	C	N	T	E	C	N	T	E	C	N	T
DocNLI (Yin et al., 2021)	NLI	Wiki, News	Doc ($\sim 318, < 26K$)	Para ($\sim 51, < 252$)	✗	466K	475K	-	942K	28K	205K	-	234K	33K	233K	-	267K
ContractNLI (Koreeda and Manning, 2021)	NLI	NDA Contracts	Doc ($\sim 2254, < 11.5K$)	Sent ($\sim 13, < 23$)	✓	3530	841	2820	7191	519	95	423	1037	968	220	903	2091
ConTRoL (Liu et al., 2021)	NLI	Exam	Doc ($\sim 441, < 1604$)	Multi-Sent ($\sim 12, < 93$)	✗	2480	2293	1946	6719	286	276	237	799	327	242	236	805
CaseHoldNLI (Zheng et al., 2021)	MCQ	Legal Case Laws	Doc ($\sim 143, < 3328$)	Multi-Sent ($\sim 28, < 156$)	✗	43K	174K	-	217K	5.4K	21.6K	-	27K	5.4K	21.6K	-	27K

Table 8: Data statistics for DocNLI, ContractNLI, ConTRoL, and CaseHoldNLI datasets. The average ($\sim$) and maximum ($<$) lengths of the premise and hypothesis were measured by NLTK tokenization. We describe the train/validation/test distributions across labels C: Contradiction, E: Entailment, N: Neutral, T: Total. ($-$) indicates non-existent, LightCyan is our contribution.

	Dataset			
Hyperparameters	DocNLI	ContractNLI	ConTRoL	CaseHoldNLI
Dropout Ratio	0.4	0.3	0.3	0.5
Optimizer	Adam	Adam	Adam	Adam
GAT Layers	2	2	2	2
Input Dimension (g_{out})	(n,256)	(n,256)	(n,256)	(n,256)
Input Dimension (Dense)	(n,1024)	(n,1024)	(n,1024)	(n,1024)
Hidden Dimension	200	200	200	200
Epochs	20	20	20	20
Batch Size	16	16	16	16
Activation Function of Linear layers	ReLU	ReLU	ReLU	ReLU
Output Classes	2	3	3	2

Table 9: **Hyperparameter Details:** Training hyperparameters of `DcInfer` for DocNLI, ContractNLI, ConTRol and CaseHoldNLI datasets.

We summarize the range of our model's hyper parameters such as: size of hidden layers in LSTM $\{100, 200, 300\}$, size of hidden layers in FFN $\{100, 200, 300, 400\}$, BERT embedding size, dropout $\delta \in \{0.2, 0.3, 0.4, 0.5.0.6\}$, learning rate $\lambda \in \{1e-5, 2e-5, 3e-5, 4e-5, 5e-5\}$, weight decay $\omega \in \{1e-6, 1e-5, 1e-4, 1e-3\}$, batch size $b \in \{16, 32, 64\}$ and epochs (≤ 100).

Contextual Encoder: We used BERT-base-uncased for generating token embedding of size 1x 768. As BERT-base Transformer provides a stronger baseline as compared to RoBERTa, we utilized BERT Transformer for Contextual Encoder in `DocInfer` architecture. We use the default dropout rate (0.1) on BERT's self attention layers but do not use additional dropout at the top linear layer The output from the Contextual Encoder is a 1-D vector of size 768.

Loss Function and Inference: `DocInfer` is trained end to end using Cross Entropy loss for context encoder and REINFORCE loss for evidence selection with Adam optimizer. Across all four datasets, we found the best results correspond with the use of Adam optimiser set with default values $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e-8$, weight-decay of $5e-4$ and an initial learning rate of 0.001. We evaluate the performance of NLI prediction with the following corresponding metrics for each dataset:

- **DocNLI**: devF1, test F1
- **ContractNLI**: NLI label using Acc (%), F1(entails), F1(contradicts), Evidence extraction using mAP,PR@80 precision and recall score.
- **ConTRoL**: Acc(%), F1(E), F1(C), F1(N), F1(Overall)
- **CaseHold**: P, R, F1
- **MCTest**: F1
- **FEVER-binary**: F1
- **Contract Discovery**: Soft F1

Parameter Size: Our model's total parameters is asymptotically close to the context encoder it uses in the backbone. Our model does not add a lot more parameters than the existing context encoder.

C4: Implementation Software and Packages

We implemented our solution in Python 3.6 using PyTorch framework. We used the following libraries and modules:

- Huggingface's implementation for BERT/RoBERTa/BART/Legal-BERT/T5/Longformer/BigBird transformers.
- PyTorch Geometric[3] for graph learning methods.
- LDA-Pypi[4] package for topical relations.

[3] https://pytorch-geometric.readthedocs.io/en/latest/
[4] https://pypi.org/project/lda/

- Standford CoreNLP[5], Spacy[6], and
- AllenNLP Library[7] for coreference and Named Entity Recognition in entity relations.
- ConceptNet Numberbatch[8] and KnowBert[9] for concept relations.
- SimCSE embeddings[10] for semantic reward.
- Bleu-Pypi[11] library for evidence reward.

[5]https://stanfordnlp.github.io/CoreNLP/
[6]https://spacy.io/usage
[7]https://allenai.org/allennlp/software/allennlp-library
[8]https://github.com/commonsense/conceptnet-numberbatch
[9]https://github.com/allenai/kb
[10]https://github.com/princeton-nlp/SimCSE
[11]https://pypi.org/project/bleu/