Length-Aware Multi-Kernel Transformer for Long Document Classification

Anonymous ACL submission

Abstract

Lengthy documents pose a unique challenge to neural language models due to substantial memory consumption. While existing state-of-the-art (SOTA) models segment long texts into equal-length snippets (e.g., 128 tokens per snippet) or deploy sparse attention networks, these methods have new challenges of context fragmentation and generalizability due to sentence boundaries and varying text lengths. For example, our empirical analysis has shown that SOTA models consistently overfit one set of lengthy documents (e.g., 2000 tokens) while performing worse on texts with other lengths (e.g., 1000 or 4000). In this study, we propose a **L**ength-**A**ware **M**ulti-**K**ernel **T**ransformer (*LAMKIT*) to address the new challenges for the long document classification. LAMKIT encodes lengthy documents by diverse transformer-based kernels for bridging context boundaries and vectorizes text length by the kernels to promote model robustness over varying document lengths. Experiments on four standard benchmarks from health and law domains show LAMKIT outperforms SOTA models up to an absolute 10.9% improvement. We conduct extensive ablation analyses to examine model robustness and effectiveness over varying document lengths.

1 Introduction

Lengthy documents widely exist in many fields, while the input limit (512 tokens) of transformer models prevents developing powerful pre-trained language models on those long documents, such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019). For example, a recent study shows that clinical documents have grown over 60% longer in a decade (Rule et al., 2021). Truncation is a common strategy to handle long documents and fit the input limit of BERT-based classifiers, however, the method may lose many critical contexts beyond the first 512 tokens and hurdle model effectiveness. One solution for lengthy documents is *long document modeling*.

Among existing transformer-based models, long document modeling has two major directions, hierarchical transformer and sparse attention (Dong et al., 2023; Qin et al., 2023). The hierarchical approach (Wu et al., 2021; Chalkidis et al., 2022; Dai et al., 2022; Li et al., 2023a; Chalkidis et al., 2023) splits document into small text chunks (e.g., 128 tokens) so that long document models can take shorter input per step. As the self-attention in transformer-style models causes quadratic complexity $O(n^2)$, the sparse attention aims to lower the complexity to linear and reduce context fragmentation caused by the segments (Beltagy et al., 2020; Zaheer et al., 2020; Guo et al., 2022; Zhang et al., 2023). For example, sparse attention in Longformer (Beltagy et al., 2020) lifts up the input limit from 512 tokens to 4096 tokens. Popular evaluation benchmarks also switch from social media data (e.g., IMDb and Amazon reviews (Wu et al., 2021)) to more complex data in health and legal domains (Qin et al., 2023; Chalkidis et al., 2022). For example, the median document length of IMDb is only 225 tokens (Li et al., 2023a), which is much smaller than the lengths in Table 1. Indeed, document lengths vary across datasets, and model performance can vary across length-varied corpora (Li et al., 2023a). However, very few studies have examined if long document models can handle varying-length texts, ranging from short to extremely long. A common question is: *will a long document model be capable to maintain robust performance across varying-length data?* Our analysis on SOTA baselines in Figure 1 says "No."

To understand the length effects and encounter the long document challenges, we conduct extensive analysis and propose **L**ength-**A**ware **M**ulti-**K**ernel **T**ransformer (*LAMKIT*) for robust long document classification. LAMKIT diversifies learning processes by a multi-kernel encoding (MK)

Dataset	Length-Quantile			L-mean	Size	|Label|	Splits		
	25%	50%	75%				Train	Valid	Test
Diabetes	408	608	945	720	1,265	10	885	190	190
MIMIC	1,432	2,022	2,741	2,200	11,368	50	8,066	1,753	1,729
ECtHR	668	1,328	2,627	2,139	11,000	11	9,000	1,000	1,000
SCOTUS	3,723	7,673	12,275	9,840	7,800	14	5,000	1,400	1,400

Table 1: Statistics of average token count per document (L-mean), data size (Size), and unique labels (|Label|).

so that the model can capture contexts from different perspectives. The MK contains multiple neural encoders with diverse kernel sizes and can relieve context fragmentation caused by a unique segment encoder on short text chunks. LAMKIT promotes model robustness over varying-length documents by a length-aware vectorization (LaV) module. The LaV encodes length information in a hierarchical way, position embedding on segment and length vectors on document level. We compare LAMKIT with 8 domain-specific models on four datasets (MIMIC-III (Johnson et al., 2016), SCOTUS (Chalkidis et al., 2022), ECtHR-A (Chalkidis et al., 2019), Diabetes (Stubbs et al., 2019)) from health and legal domains evaluated by F1 and AUC metrics. Additionally, we also conduct a case study on the performance of ChatGPT in these tasks. Classification results demonstrate that our LAMKIT approach's outperforms competitive baselines by an absolute improvement of up to 10.9%. We conduct further experiments on the length-varying effects and ablation analysis to examine the effectiveness of our individual modules.

2 Data

We have retrieved four publicly available data, Diabetes (Stubbs et al., 2019), MIMIC-III (Johnson et al., 2016), ECtHR-A (Chalkidis et al., 2019), and SCOTUS (Chalkidis et al., 2022), which are popular benchmarks for the long document classification. We obtained *Diabetes* (Stubbs et al., 2019) from the 2018 National NLP Clinical Challenges (n2c2) shared task with a collection of longitudinal patient records and 13 selection criteria annotations. We exclude 3 annotations due to less than 0.5 inter-rater agreements and discard documents with fewer than 40 tokens. *MIMIC-III* (Medical Information Mart for Intensive Care) (Johnson et al., 2016) is a relational database that contains patients admitted to the Intensive Care Unit (ICU) at the Beth Israel Deaconess Medical Center from 2001 to 2012. We follow previous work (Mullenbach et al., 2018; Vu et al., 2021) to select discharge summaries and use the top 50 frequent labels of International Classification of Disease codes (9th Edition, ICD-9), which are types of procedures and diagnoses during patient stay in the ICU. *ECtHR-A* collects facts and articles from law case descriptions from the European Court of Human Rights' public database (Chalkidis et al., 2019). Each case is mapped to the articles it was found to have violated in the ECHR, while in ECTHR-B (Chalkidis et al., 2021), cases are mapped to a set of allegedly violated articles. We follow the study (Chalkidis et al., 2022) to process and obtain 11 labels. *SCOTUS* is a data collection of US Supreme Court (the highest US federal court) opinions and the US Supreme Court Database (SCDB) (Spaeth et al., 2020) with cases from 1946 to 2020. SCOTUS has 14 issue areas, such as Criminal Procedure, Civil Rights, and Economic Activity. We summarize data statistics and splits in Table 1.

Table 1 shows each data has a varying length range, a critical yet under-explored question is: does the varying length effect model performance or will models be generalizable across all lengths? For example, the document length in Table 1 is either less than a few hundred or over ten thousand tokens surpassing input limitations of regular transformer-style models (e.g., BERT), and there are significant length variations across the data. While studies (Dong et al., 2023) have achieved improving performance overall to encode more contexts beyond the 512 token limit, there is very few work examining the effects of varying document lengths over model robustness. To answer the question, we conduct an exploratory analysis of existing state-of-the-art (SOTA) models and evaluate their performance.

Our exploratory analysis follows existing studies (Mullenbach et al., 2018; Dai et al., 2022; Chalkidis et al., 2022; Qin et al., 2023) to split data, includes three state-of-the-art transformer classifiers (BigBird, Longformer, and Hierarchi-

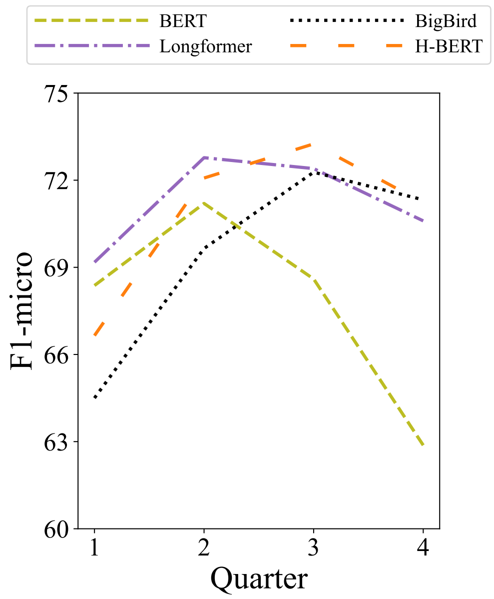

Figure 1: SOTA baseline performance across the quarter splits.

cal BERT (H-BERT)) for long document and a BERT classifier, and evaluates models performance by F1-micro (F1-μ) score. We refer to the details of experimental settings and SOTA baselines under the Experiments section. For each quarter, we maintain similar data sizes and run the classifier multiple times to take average performance scores. Finally, we visualize the relation between model performance and document lengths in Figure 1.

Figure 1 shows that model performance varies across document lengths, posing a unique challenge to build robust models on varying lengthy data. For example, while the SOTA classifiers achieve better scores on mid-lengthy texts, the performance drops significantly in either short (e.g., 400 tokens) or super long (e.g., 10K tokens) documents. The consistent observations can suggest that: 1) varying length can be a critical factor to make models perform better; 2) length-based splits are important to understand the capacity of classifiers on long documents. The findings inspire us to propose the **L**ength-**A**ware **M**ulti-**K**ernel **T**ransformer (*LAMKIT*) to encounter the length factor.

2.1 Ethic and Privacy Concern

We access four datasets in accordance with data agreements and underwent relevant training. To prioritize user privacy, we employ stringent data usage measures and conduct our experiments exclusively on anonymized data. For ethical and privacy reasons, we refrain from releasing any clinical data linked to patient identities. However, we commit to sharing our code, accompanied by comprehensive guidelines to reproduce our findings. All data used in this research is publicly accessible and has been stripped of identifying information. Our investigation is centered on computational techniques, and we do not gather data directly from individuals. Our institution's review board has confirmed that this research does not mandate an IRB approval.

3 Length-Aware Multi-Kernel Transformer

This section presents our Length-Aware Multi-Kernel Transformer (*LAMKIT*) for robust long document classification in Figure 2. LAMKIT consists of three major modules, 1) multi-kernel encoding, 2) length-aware vectorization, and 3) hierarchical integration, aiming to solve context fragmentation and augment model robustness on lengthy documents. We deploy different encoding kernels to diversify text segments with various contexts. Incorporating length as vectors can adapt classifiers across varying-length documents. Finally, we elaborate on how to learn robust document representations via a hierarchical integration.

3.1 Multi-kernel Encoding

Multi-kernel Encoding (MK) aims to diversify context to segment and encode documents from multiple perspectives. The mechanism is to solve the fundamental challenge of existing long document modeling (Beltagy et al., 2020; Wu et al., 2021; Dai et al., 2022; Dong et al., 2023) — splitting and vectorizing each document by a fixed size and a unified document encoder, which has been analyzed in our previous data section. Our MK mechanism gets inspirations from Convolutional Neural Network (Kim, 2014) that encodes each document into various sizes of text segments and deploys one document encoder per segment size to obtain various feature representations. By learning diverse document features with varying-size text chunks, we can enrich representations of lengthy documents with various sizes.

Specifically, we empirically choose three kernel sizes ($m \in \{128, 256, 512\}$) and three neural encoders to vectorize text chunks with a size of m. Following the CNN, we tried the other sizes (e.g., 300) and a stride ranging between $(2/3 * m, m)$, but we did not get significant improvements. In

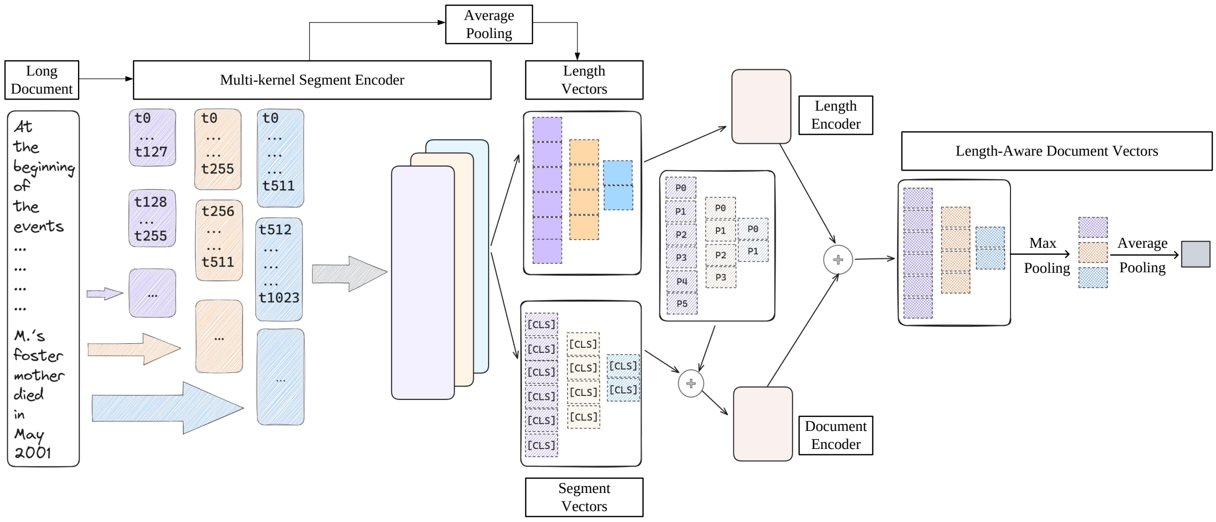

Figure 2: LAMKIT diagram overview. Our approach consists of three main components: multi-kernel encoding, length-aware vectorization, and hierarchical integration. We denote one color of segments and vectors per kernel. The arrows indicate model workflows, $\bigoplus$ is a sum operation.

the later section, our ablation analysis shows that the major performance drops come from the number of kernels. We infer the performance of kernel and stride sizes as encoding contexts with different kernels is more critical to augment classifiers on lengthy documents. For each chunk size of text, we deploy a pre-trained RoBERTa model (Liu et al., 2019) so that our MK has three varied RoBERTa encoders. While our MK mechanism allows other BERT variants, we choose the RoBERTa to keep consistent with existing SOTA approaches (Chalkidis et al., 2022; Li et al., 2023c; Dong et al., 2023) for fair comparisons. We take the embedding of the "[CLS]" token from each text chunk to represent its segment vector and feed to the following operation, combining with the segment position embedding of length-aware vectorization.

3.2 Length-aware Vectorization

We propose the Length-aware Vectorization (*LaV*) to incorporate lengthy contexts and augment model generalizability, as our Figure 1 presents that the model performance varies across document lengths. LaV achieves the grand goal by two levels: text chunk and document. On the text chunk level, we encode length information by the segment position embedding, and on the document level, we vectorize text length with MK outputs.

Segment Position Embedding vectorizes positions of text chunks into a learnable embedding by a Transformer encoder in Equation 1, where $|d|$ refers to the embedding size, i is the column index of a vector scalar, and pos is the index of the text chunk. For example, if we segment a 1024-token document into 15 chunks (with a stride) by the 128 kernel encoder, the total will be the 15 and the second chunk's index (pos) will be 2. Similarly, we can obtain segment position embeddings for other multi-kernel encoders and equip the segment vectors from the MK step with the length information, segment position. Finally, we sum the segment position embeddings up with the segment vectors and feed them to the document encoder.

$$PE_{(pos,i)} = \begin{cases} \sin\left(\frac{pos}{10000^{2i/|d|}}\right), & \text{if } i \text{ is even} \\ \cos\left(\frac{pos}{10000^{2i/|d|}}\right), & \text{if } i \text{ is odd} \end{cases} \quad (1)$$

Note that, our position embedding **differs** from previous studies. For example, majority of long document classifiers (Wu et al., 2021; Li et al., 2023b; Zhang et al., 2023) deploy position embeddings for tokens rather than the segment. There is one close study (Dai et al., 2022) that utilizes segment position embedding in classification models. In contrast, our position embedding diversifies segment positions from multiple kernels, aiming to incorporate text lengths and augment model generalizability over varying text lengths.

Length Vectors encode document length information into feature vectors. Instead of directly encoding a length scalar into a vector, we obtain the length vectors by applying averaging pooling over each MK encoder's outputs and vectorizing the chunk sizes per document by the position embedding. The length vectors not only encode document lengths by chunk sizes but also implicitly incorporate lengthy contexts from the MK encoders. Finally, we feed the length vectors into the length encoder to obtain learnable length-aware vectors, which will be integrated with the document encoder's outputs.

3.3 Hierarchical Integration

We obtain length-aware document representations through the hierarchical integration process from segment and length vectors. The integration process starts with a document encoder to encode segment vectors and a length encoder to encode length vectors. Both modules are Transformer (Vaswani et al., 2017) encoders but serve different purposes — while both encoders take length-related vectors, the document encoder focuses on learning diversified contexts from the MK encoders and the length encoder focuses on incorporating varying length features. We then combine the two encoders' outputs by a sum operation and feed the integration to a hierarchical pooling process to obtain length-aware document vectors.

Hierarchical pooling operations has two major processes in order, max pooling and average pooling. The max pooling aims to squeeze length-aware multidimensional representations of text chunks from the length and document encoders. We concatenate the pooling outputs and feed them to the average pooling operation. The average pooling aggregates the length-aware segment features into the length-aware document vectors. Finally we feed the document vectors to linear layer for classification. Our tasks cover both binary and multi-label classifications. We deploy a sigmoid function for binary prediction and a softmax function for the multi-label task.

4 Experiments

We follow the previous studies (Mullenbach et al., 2018; Stubbs et al., 2019; Chalkidis et al., 2022) on lengthy document to preprocess data and split data into training, validation, and test, as in Table 1. We follow SOTA baselines to set up our evaluation experiments. Our results include F1 and AUC metrics, covering both micro (μ) and macro (m) variations.

Our evaluation presents performance comparisons and ablation analysis to understand the length effects and the models better. More details of the hyperparameter settings for the baselines and LAMKIT are in the Appendix A, which allows for experiment replications.

4.1 Baselines

To demonstrate the effectiveness of LAMKIT, we compare it against both hierarchical transformer and sparse attention transformer SOTA baselines for long-document modeling, as well as with regular BERT.

Our experiments utilize baseline hyperparameters that achieved their best results in the previous studies. For example, we take publicly released models or source codes to train long document classifiers. As our data come from health and legal domains, we choose the pre-trained models on the domain data. For example, we report performance of Clinical-Longformer (Li et al., 2023c) on health data instead of vanilla Longformer (Beltagy et al., 2020).

BERT includes classifiers built on domain-specific pre-trained BERT models. Specifically, we include two types of pre-trained BERT model, *Legal-BERT* (Chalkidis et al., 2020) for the legal data and *RoBERTa-PM-M3* (Lewis et al., 2020) for the clinical data, which achieved the best performance on broad text classification tasks in legal and clinical domains. Due to the input limit, the BERT baselines truncate and only take 512 tokens per entry. We experiment two types of truncation, first and last 512 tokens of each data entry, and name the two types as $BERT_{First}$ and $BERT_{Last}$.

Hierarchical BERT (*H-BERT*) splits long document into equal-length segments, hierarchically integrate segment features into document vectors, and yield predictions on the document vectors (Dai et al., 2022; Qin et al., 2023; Dong et al., 2023). We follow the existing SOTA studies that achieved the best results using the H-BERT in health (Dai et al., 2022) and legal (Chalkidis et al., 2022) domains. The H-BERT models are close to our hierarchical architecture, while the H-BERT models do not incorporate our proposed multi-kernel mechanism (MK) and length vectors. If LAMKIT achieves better performance, the improvements over the H-BERT can prove the effectiveness of adapting varying-length texts.

Model	Diabetes				MIMIC				ECtHR				SCOTUS			
	F1-μ	F1-m	AUC-μ	AUC-m	F1-μ	F1-m	AUC-μ	AUC-m	F1-μ	F1-m	AUC-μ	AUC-m	F1-μ	F1-m	AUC-μ	AUC-m
BERT_{First}	72.0	43.2	86.9	72.4	56.8	47.0	87.1	84.0	64.2	52.6	91.6	88.6	73.9	61.6	95.9	90.0
BERT_{Last}	68.7	39.1	87.2	72.2	51.3	41.5	84.8	81.4	66.1	59.1	93.7	91.3	66.9	53.1	93.6	87.2
Longformer	71.5	41.2	88.4	71.6	67.2	58.2	92.5	89.8	71.4	59.0	95.4	93.3	74.3	62.9	95.6	89.9
BigBird	71.9	42.5	**88.5**	**76.4**	65.3	56.8	92.3	89.7	70.2	61.8	93.8	91.8	72.3	60.6	94.3	89.7
H-BERT	70.4	46.0	83.2	69.7	66.9	60.6	92.6	90.2	70.4	57.7	95.7	93.9	76.6	**68.0**	95.5	**95.0**
LAMKIT	**73.4**	**49.9**	88.4	74.5	**69.5**	**63.7**	**93.3**	**91.2**	**73.0**	**65.0**	**96.0**	**94.7**	**78.5**	67.8	**97.1**	94.9
$\overline{\Delta}$	2.5	6.9	1.6	2.0	8.0	10.9	3.4	4.2	4.5	7.0	2.0	2.9	5.7	6.6	2.1	4.5

Table 2: Overall performance in percentages of F1 and AUC metrics, both micro (μ) and macro (m). We **bolden** the best performance and underline the second best value. $\overline{\Delta}$ denotes the absolute improvement of LAMKIT over the baselines average.

Longformer (Beltagy et al., 2020; Guo et al., 2022; Saggau et al., 2023) solves the 512-length limit by replacing self-attention with a local (sliding window) attention and unidirectional global attention and thus can process sequences up to 4096 tokens. We deploy domain-specific Longformer to keep consistent experimental settings. Specifically, we utilize *Clinical-Longformer* (Li et al., 2023c) and *Legal-Longformer* (Chalkidis et al., 2023) to build our document classifiers for the health and legal data, respectively.

BigBird deploys a block sparse attention to relieve the length limit that reduces the Transformer quadratic dependency to linear (Zaheer et al., 2020). BigBird utilizes a fusion of local, global, and random attention, extending the maximum processable sequence length to 4096 tokens. We utilize its domain-specific variants, Clinical-BigBird (Li et al., 2023c) and Legal-Bigbird (Dassi and Kwate, 2021) to conduct experiments.

5 Result Analysis

This section reports the performance of SOTA baselines and LAMKIT in terms of F1 and AUC metrics, both micro (μ) and macro (m) modes. Besides the overall performance, we examine varying-length effects and conduct ablation analysis on our individual modules (e.g., MK and LaV). The results show that LAMKIT not only surpasses the baselines by a large margin on long documents from both health and legal domains but also shows more stable performance on documents of varying lengths.

5.1 Overall Performance

We present the results of long document classification benchmarks in Table 2 that our LAMKIT significantly outperforms the other SOTA baselines. For example, compared to the baselines' average performance, LAMKIT shows an improvement of 5.2% in F1-micro and 7.9% in F1-macro. Long document models do not perform better than regular BERT models on shorter texts. For example, $BERT_{first}$ outperforms most of the SOTA baselines on Diabetes, of which 50% clinical notes are less than 608 tokens. In contrast, we can observe our LAMKIT is robust on both shorter and longer text documents, highlighting the unique contribution and effectiveness of our approach.

Document characteristics of health and legal data can impact baselines performance. For example, we find that H-BERT performs better on the SCOTUS compared to models with sparse attention networks (e.g., Longformer and BigBird), while its performance on other datasets is comparable. We infer this as the SCOTUS dataset has clear segment boundaries that H-BERT can utilize the boundaries as segments, however, other data is compressed and dense, which can cause context fragmentation (Beltagy et al., 2020) and weaken effectiveness of H-BERT. *However*, our LAMKIT demonstrates superior performance on the issue, and we think the MK and length-aware vectors play critical roles, which is shown in our ablation analysis.

5.2 Performance on Varying-length Splits

To assess the model's robustness and generalizability across documents of varying lengths, we follow the approach described in the Data Section, dividing each dataset into quarters based on the lengths of the documents, ensuring similar data sizes in each quarter.

Table 3 presents F1-micro scores across four quarters of each dataset that LAMKIT outperforms baselines on most quarters across the datasets. Surprisingly, SOTA baselines tend to favor and overfit one quarter data with a specific length, which does not exceed their input limit (e.g., 4096 for Longformer). In contrast, our LAMKIT shows more generalizable performance across varying-length documents. The stable performance of our LAMKIT

Model	Diabetes				MIMIC				ECtHR				SCOTUS			
	Q-1	Q-2	Q-3	Q-4	Q-1	Q-2	Q-3	Q-4	Q-1	Q-2	Q-3	Q-4	Q-1	Q-2	Q-3	Q-4
BERT_{First}	65.7	**74.1**	73.4	74.2	57.9	63.0	57.5	52.9	74.9	73.4	62.6	54.4	**75.0**	74.3	80.9	70.0
BERT_{Last}	63.4	66.9	71.6	71.8	51.6	57.8	50.3	48.4	72.6	73.0	62.5	61.6	68.8	64.4	69.4	66.0
Longformer	64.6	72.7	72.2	75.8	63.8	71.0	68.1	66.4	79.0	74.0	72.4	65.7	69.3	73.4	76.9	74.5
BigBird	61.0	72.1	71.7	**79.9**	62.9	70.2	66.3	62.6	68.8	65.9	73.9	**70.7**	65.3	70.4	77.2	72.1
H-BERT	61.2	67.6	74.2	77.8	62.1	69.6	66.8	66.5	79.1	**75.3**	69.1	64.1	64.2	75.8	82.9	76.5
LAMKIT	**66.0**	71.2	**77.0**	78.1	**66.4**	**72.6**	**70.4**	**68.0**	**79.7**	74.6	**74.3**	67.5	72.2	**76.4**	**83.0**	**78.5**
$\overline{\Delta}$	2.8	0.5	4.4	2.2	6.7	6.3	8.6	8.6	4.8	2.3	6.2	4.2	3.7	4.7	5.5	6.7

Table 3: F1-micro scores across four quarters following our Figure 1. We **bolden** the best performance and underline the second best value. $\overline{\Delta}$ refers to the absolute improvement of LAMKIT over the average of baselines.

highlights the effectiveness of our multi-kernel and length vectors in adapting classifiers on varying lengths and promoting classification robustness on the health and legal domains.

5.3 Ablation Study

We conduct an ablation analysis to assess the effectiveness of individual LAMKIT modules focusing on the multi-kernel mechanism (MK) and length-aware vectorization (LaV). *w/o MK* replaces multi-kernel encoders with a single kernel encoder (RoBERTa) and shrinks segment vectors accordingly. *w/o LaV* removes length-related vectors and encoders from LAMKIT. And, *w/o MK and LaV* removes both MK mechanism and length-related encoding.

We can observe that removing one of the modules or removing all modules can significantly reduce model performance. Replacing the MK mechanism can result in a 1.3% and 1.8% drop in F1-micro and F1-macro on average, respectively. The performance drop indicates multi-kernel encoding mechanism can relieve context fragmentation to promote model performance by diversifying document representations. Removing LaV leads to 1.4% and 2.5% drops in F1-micro and F1-macro on average, respectively. The performance drop shows that the length information can be critical to building robust classifiers on the health and legal data.

We can observe the most significant performance drop in LAMKIT after removing both MK and LaV modules, with F1-micro and F1-macro scores decreasing by 3.0% and 3.5%, and AUC-micro and AUC-macro scores by 1.5% and 1.8%, respectively, demonstrating the effectiveness of these methods

6 Case Study on ChatGPT

To examine the ability of large language models on the long document classification task. Due to privacy concerns and data usage agreement, we do not test ChatGPT (OpenAI, 2022) on MIMIC and Diabetes. We utilize GPT-3.5-Turbo via *ChatCompletion API*[1] in a zero-shot strategy with multiple templated instructions summarized by (Lou et al., 2023; Chalkidis, 2023), and report the best performing template results. The results in Table 5 suggest that large language models do not exceed the performance of task-specific models in long-text classification. For the prompt template, we refer more details in the Appendix Figure 3.

7 Related Work

7.1 Transformers for Text Classification

Pretrained language models (PLMs) based on vanilla self-attention, such as BERT (Devlin et al., 2019) and its variants (He et al., 2021; Liu et al., 2019; Ma et al., 2021; Alsentzer et al., 2019), have achieved state-of-the-art (SOTA) results in regular text classification tasks. However, with their input typically limited to 512 tokens, truncation becomes necessary when handling long texts (Ding et al., 2020). Such truncation might cause the text to lose a significant amount of valuable information, thereby affecting the model's performance. Therefore, long document modeling serves as a solution to applying pretrained models to lengthy texts.

7.2 Long Document Modeling

To enable transformers to accept longer sequences, two primary approaches have been employed in long document modeling: efficient transformers (e.g., sparse attention transformers) and hierarchical transformers (Dong et al., 2023). Hierarchical transformer models (Li et al., 2023a; Ruan et al., 2022; Chalkidis et al., 2023) rely on chunking the text into slices of equal size and obtaining the document representation based on the representations

[1] https://platform.openai.com/docs/guides/gpt/chat-completions-api

Model	Diabetes				MIMIC				ECtHR				SCOTUS			
	F1-μ	F1-m	AUC-μ	AUC-m	F1-μ	F1-m	AUC-μ	AUC-m	F1-μ	F1-m	AUC-μ	AUC-m	F1-μ	F1-m	AUC-μ	AUC-m
LAMKIT	73.4	49.3	88.4	74.5	69.5	63.7	93.3	91.2	73.0	65.0	96.0	94.7	78.5	67.8	97.1	94.9
w/o MK	72.1	47.6	88.2	72.3	68.5	61.9	92.8	90.5	72.0	62.7	95.5	93.9	76.7	66.3	97.0	93.3
w/o LaV	71.5	42.1	87.5	72.7	68.4	62.9	93.0	90.8	71.5	64.2	95.6	94.3	77.6	66.6	97.1	93.1
w/o MK and LaV	69.9	46.6	85.3	71.1	66.3	60.0	92.3	89.9	70.4	61.3	94.9	93.4	76.0	63.9	96.4	93.6

Table 4: Ablation performance of LAMKIT modules in F1 and AUC, both micro (μ) and macro (m), shown in percentages.

Model	ECtHR		SCOTUS	
	F1-μ	F1-m	F1-μ	F1-m
ChatGPT	51.1	47.7	49.9	42.0

Table 5: F1 metrics (in %) of ChatGPT on Legal Data.

of these slices, ensuring that the model's input does not exceed the limit in each instance. For example, HiPool (Li et al., 2023a) employs Transformers for sentence modeling and then uses Graph Convolutional Neural Networks for document information modeling. HiStruct+ (Ruan et al., 2022) encodes the hierarchical structure information of the document and infuses it into the hierarchical attention model. Due to the full-rank attention mechanism in transformer models leading to quadratic computational complexity, efficient transformers (Beltagy et al., 2020; Zaheer et al., 2020; Choromanski et al., 2021; Kitaev et al., 2020; Wang et al., 2020; Zhang et al., 2023) aim to use sparse attention or low-rank methods to reduce the complexity and minimize context fragmentation caused by segmentation. For instance, to reduce computational complexity from $O(n^2)$ to $O(n)$, Longformer (Beltagy et al., 2020) employs a mix of local attention (through a sliding window) and global attention on certain special tokens. Similarly, BigBird (Zaheer et al., 2020) incorporates both these attention mechanisms and introduces an additional random attention strategy. Both models have expanded their input limits to 4096 tokens. However, they do not perform well on documents of all lengths.

Prior research (Li et al., 2023a) has noted that document lengths differ among datasets, and model performance can be inconsistent across corpora with varying lengths. Studies (Dai et al., 2022) have also shown that segmenting documents inevitably leads to issues of context fragmentation. However, no previous work has centered on the aforementioned two inherent issues of long document models: context fragmentation and generalizability across varying text lengths. In this study, we propose a novel approach Length-Aware Multi-Kernel Transformer (*LAMKIT*). By using multi-kernel encoding (MK), LAMKIT obtains multi-perspective context representations to mitigate the context fragmentation issue caused by using a unique chunk size. LAMKIT also enhances model robustness for documents of varying lengths through its Length-Aware Vectorization (LaV) module. This LaV module encodes length information hierarchically, using segment position embedding at the segment level and length vectors from the MK outputs at the document level.

8 Conclusion

In this study, we posit that for long document classification tasks, the length of the text might be a pivotal determinant for model performance. Our exploratory experiments demonstrate that the current state-of-the-art models display inconsistent results across samples of differing lengths, suggesting their lack of robustness and affirming our hypothesis.

To address this issue and the inherent problem of context fragmentation in long-text models, we propose Length-Aware Multi-Kernel Transformer. Through extensive experiments, LAMKIT consistently outperforms all baseline models across four standard long document classification benchmarks. Moreover, we follow our exploratory experiments to examine model robustness over varying document lengths. We also conduct ablation studies on two modules. The results show that LAMKIT exhibits better robustness and stability across different lengths.

Additionally, the case study on ChatGPT (OpenAI, 2022) reveals that large language models do not outperform task-specific models in long-text classification. Furthermore, due to input length constraints of large language models, our experiments are limited to zero-shot, posing challenges in harnessing their in-context learning strengths via few-shot(Brown et al., 2020). The source code for this study have been included in the supplementary attachment.

Limitations

LAMKIT has a flexibility to be applicable on other tasks by changing its prediction layer, while we experiment it on the text classification task. Dong et al. demonstrated the importance of long document modeling in other NLP scenarios. We plan to explore this direction for a more comprehensive understanding on long document modeling.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

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

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Ilias Chalkidis. 2023. ChatGPT May Pass the Bar Exam Soon, but Has a Long Way to Go for the LexGLUE Benchmark. *SSRN Electronic Journal*.

Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2019. Neural legal judgment prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4317–4323, Florence, Italy. Association for Computational Linguistics.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Ilias Chalkidis, Manos Fergadiotis, Dimitrios Tsarapatsanis, Nikolaos Aletras, Ion Androutsopoulos, and Prodromos Malakasiotis. 2021. Paragraph-level rationale extraction through regularization: A case study on European court of human rights cases. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 226–241, Online. Association for Computational Linguistics.

Ilias Chalkidis, Nicolas Garneau, Catalina Goanta, Daniel Katz, and Anders Søgaard. 2023. LeXFiles and LegalLAMA: Facilitating English multinational legal language model development. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15513–15535, Toronto, Canada. Association for Computational Linguistics.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022. LexGLUE: A benchmark dataset for legal language understanding in English. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4310–4330, Dublin, Ireland. Association for Computational Linguistics.

Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J Colwell, and Adrian Weller. 2021. Rethinking attention with performers. In *International Conference on Learning Representations*, Vienna, Austria.

Xiang Dai, Ilias Chalkidis, Sune Darkner, and Desmond Elliott. 2022. Revisiting transformer-based models for long document classification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7212–7230, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Loic Kwate Dassi and Loic Kwate. 2021. Legal-bigbird: An adapted long-range transformer for legal documents. In *Proceedings of the 35th International Conference on Neural Information Processing Systems, Black in AI Workshop*. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ming Ding, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Cogltx: Applying bert to long texts. In *Advances in Neural Information Processing Systems*, volume 33, pages 12792–12804, Vancouver, British Columbia, Canada. Curran Associates, Inc.

Zican Dong, Tianyi Tang, Lunyi Li, and Wayne Xin Zhao. 2023. A survey on long text modeling with transformers.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736, Seattle, United States. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. {DEBERTA}: {DECODING}-{enhanced} {bert} {with} {disentangled} {attention}. In *International Conference on Learning Representations*.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3:160035.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. In *International Conference on Learning Representations*, Addis Ababa, Ethiopia.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Irene Li, Aosong Feng, Dragomir Radev, and Rex Ying. 2023a. HiPool: Modeling long documents using graph neural networks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 161–171, Toronto, Canada. Association for Computational Linguistics.

Xianming Li, Zongxi Li, Xiaotian Luo, Haoran Xie, Xing Lee, Yingbin Zhao, Fu Lee Wang, and Qing Li. 2023b. Recurrent attention networks for long-text modeling. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3006–3019, Toronto, Canada. Association for Computational Linguistics.

Yikuan Li, Ramsey M Wehbe, Faraz S Ahmad, Hanyin Wang, and Yuan Luo. 2023c. A comparative study of pretrained language models for long clinical text. *Journal of the American Medical Informatics Association*, 30(2):340–347.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Renze Lou, Kai Zhang, and Wenpeng Yin. 2023. Is prompt all you need? no. a comprehensive and broader view of instruction learning.

Weicheng Ma, Kai Zhang, Renze Lou, Lili Wang, and Soroush Vosoughi. 2021. Contributions of transformer attention heads in multi- and cross-lingual tasks. 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)*, pages 1956–1966, Online. Association for Computational Linguistics.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. 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 1101–1111, New Orleans, Louisiana. Association for Computational Linguistics.

OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. `https://openai.com/blog/chatgpt/`. Accessed: 2023-07-24.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, pages 8024–8035. Curran Associates, Inc.

Guanghui Qin, Yukun Feng, and Benjamin Van Durme. 2023. The NLP task effectiveness of long-range transformers. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3774–3790, Dubrovnik, Croatia. Association for Computational Linguistics.

Qian Ruan, Malte Ostendorff, and Georg Rehm. 2022. HiStruct+: Improving extractive text summarization with hierarchical structure information. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1292–1308, Dublin, Ireland. Association for Computational Linguistics.

Adam Rule, Steven Bedrick, Michael F. Chiang, and Michelle R. Hribar. 2021. Length and Redundancy of Outpatient Progress Notes Across a Decade at an Academic Medical Center. *JAMA Network Open*, 4(7):e2115334–e2115334.

Daniel Saggau, Mina Rezaei, Bernd Bischl, and Ilias Chalkidis. 2023. Efficient document embeddings via self-contrastive bregman divergence learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12181–12190, Toronto, Canada. Association for Computational Linguistics.

Harold J. Spaeth, Lee Epstein, Andrew D. Martin, Jeffrey A. Segal, Theodore J. Ruger, and Sara C. Benesh. 2020. Supreme Court Database, Version 2020 Release 01. `http://Supremecourtdatabase.org`. Accessed: [2021-01-01].

Amber Stubbs, Michele Filannino, Ergin Soysal, Samuel Henry, and Özlem Uzuner. 2019. Cohort selection for clinical trials: n2c2 2018 shared task track 1. *Journal of the American Medical Informatics Association*, 26(11):1163–1171.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, Long Beach, California, United States. Curran Associates, Inc.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. 2021. A label attention model for icd coding from clinical text. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, pages 3335–3341. International Joint Conferences on Artificial Intelligence Organization. Main track.

Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. 2020. Linformer: Self-attention with linear complexity.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021. Hi-transformer: Hierarchical interactive transformer for efficient and effective long document modeling. 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 848–853, Online. Association for Computational Linguistics.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: Transformers for longer sequences. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, volume 33, pages 17283–17297, Red Hook, NY, USA. Curran Associates Inc.

Xuanyu Zhang, Zhepeng Lv, and Qing Yang. 2023. Adaptive attention for sparse-based long-sequence transformer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8602–8610, Toronto, Canada. Association for Computational Linguistics.

A Experimental Details

For all baseline models, we maintain the same model architecture and optimization parameters as described in their respective papers. For Longformer (Beltagy et al., 2020), Bigbird (Zaheer et al., 2020), and BERT(Devlin et al., 2019), we fine-tune the pre-trained models obtained from huggingface transformers (Wolf et al., 2020) library based on their given configurations and produce predictions. For H-BERT(Dai et al., 2022), we train using the code released by the authors and obtain our results.

For our proposed *LAMKIT* model. The kernel sizes are set to $\{32, 64, 128\}$ in the ECTHR dataset and $\{128, 256, 512\}$ in the other three datasets. The kernel stride is set by default to be equal to the kernel size. To make the results reproducible, we set the random seed in training to 1. For the MIMIC-III and Diabetes datasets, we employ pretrained Roberta-PM-M3-base (Lewis et al., 2020) as our multi-kernel encoder. For SCOTUS and ECtHR, we opt for pretrained Legal-BERT-base (Chalkidis et al., 2020). Both encoders have 12 layers, 12 attention heads, and hidden states of 768 dimensions. Additionally, we set a Transformer (Vaswani et al., 2017) encoder with 1 layer, 12 attention heads, and 768-dimensional hidden states as the length encoder, and another with 2 layers, 12 attention heads, and 768-dimensional hidden states as the document encoder. The dropout between the two linear layers of the classifier is set at 0.1. Due to our limited computational resources, we empirically set the learning rate and tried two batch sizes: 32 and 16. Each experiment is set with a maximum of 20 training epochs and an early stopping patience of 3. We utilize the AdamW (Loshchilov and Hutter, 2019) optimizer, with a weight decay of 0.01. To expedite model convergence, we make use of 16-bit float point numbers (half-precision). Finally, we select the best-performing model based on F1-micro on the validation set. The chosen hyperparameters for the model are presented in table 6.

All experiments are conducted on a device equipped with an NVIDIA 3090 GPU with 24GB

Dataset	Learing Rate	Batch Size	Kernel Size		
MIMIC	3.5e-5	16	128	256	512
ECtHR	1.0e-5	32	32	64	128
SCOTUS	3.5e-5	16	128	256	512
Diabetes	2.5e-5	16	128	256	512

Table 6: Chosen hyperparameters for LAMKIT.

memory, running the Ubuntu system, and utilizing the PyTorch (Paszke et al., 2019) framework.

B Prompt Template of Case Study

For ChatGPT (OpenAI, 2022), we set the temperature to 0, and the Top P sampling value to 1. The prompt template is shown in Figure 3.

Data	Long Document Input [X]	Template T + Input[X]	Output [Y]
ECtHR	The applicants are former members……had in fact been fleeing the State forces.	*Task Definition:* **Given the following facts from a European Court of Human Rights (ECtHR) case.** *Test Instance:* **Input [X]** *Labels Presentation:* **Which article(s) of ECHR have been violated, if any, out of the following options:** **Article 2** **…..** **Article 1** Output: [Y]	[Article 2, Article 3]
SCOTUS	Messrs. Thomas J. Hughes, of Detroit…… Charles River Bridge v. Proprietors of Warren Bridge	*Task Definition:* **Given the following opinion from the Supreme Court of USA (SCOTUS):** *Test Instance:* **Input [X]** *Labels Presentation:* **Which topics are relevant out of the following options:** **Criminal Procedure** **……** **Civil Rights** Output: [Y]	[Criminal Procedure]

Figure 3: The best performing zero-shot template of the legal data.