NESTLE: a No-Code Tool for Statistical Analysis of Legal Corpus

Kyoungyeon Cho[1,*] **Seungkum Han**[1] **Young Rok Choi**[1] **Wonseok Hwang**[1,2,*,†]
[1]LBox [2]University of Seoul
{kycho, hsk2950, yrchoi, wonseok.hwang}@lbox.kr

Abstract

The statistical analysis of large scale legal corpus can provide valuable legal insights. For such analysis one needs to (1) select a subset of the corpus using document retrieval tools, (2) structure text using information extraction (IE) systems, and (3) visualize the data for the statistical analysis. Each process demands either specialized tools or programming skills whereas no comprehensive unified "no-code" tools have been available. Here we provide NESTLE, a no-code tool for large-scale statistical analysis of legal corpus. Powered by a Large Language Model (LLM) and the internal custom end-to-end IE system, NESTLE can extract any type of information that has not been predefined in the IE system opening up the possibility of unlimited customizable statistical analysis of the corpus without writing a single line of code. We validate our system on 15 Korean precedent IE tasks and 3 legal text classification tasks from LEXGLUE. The comprehensive experiments reveal NESTLE can achieve GPT-4 comparable performance by training the internal IE module with 4 human-labeled, and 192 LLM-labeled examples.

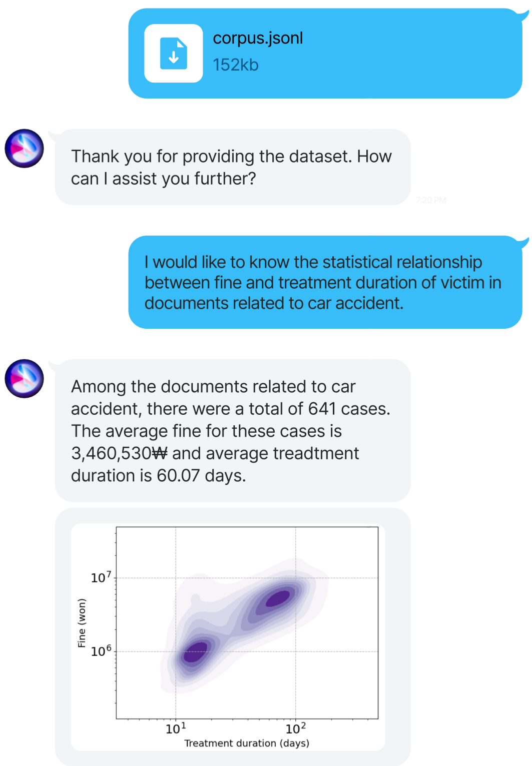

Figure 1: Illustration of NESTLE.

1 Introduction

Legal documents include a variety of semi-structured information stemming from diverse social disputes. For instance, precedents include factual information (such as blood alcohol level in a driving under the influence (DUI) case or loss in an indemnification case) as well as a decision from the court (fine, imprisonment period, money claimed by the plaintiff, money approved by the court, etc). While each document contains detailed information about specific legal events among a few individuals, community-level insights can be derived only by analyzing a substantial collection of these documents. For instance, the consequence of the subtle modification to the statute might only become evident through a comprehensive statistical analysis of the related legal corpus. Indeed a recent study shows that how the revision of the Road Traffic Act has changed the average imprisonment period in drunk driving cases by analyzing 24k Korean precedents (Hwang et al., 2022a).

Conducting a comprehensive statistical analysis on a legal corpus on a large scale may entail following three key steps: (1) choosing a subset of the corpus using retrieval tools, (2) structuralizing the documents using information extraction (IE) systems, and (3) visualizing the data for the statistical analysis. Each step requires either specialized tools or programming knowledge, impeding analysis for the majority of legal practitioners. Particularly dur-

* Equal contribution.
† Corresponding author

Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics System Demonstrations, pages 52–61
March 17-22, 2024

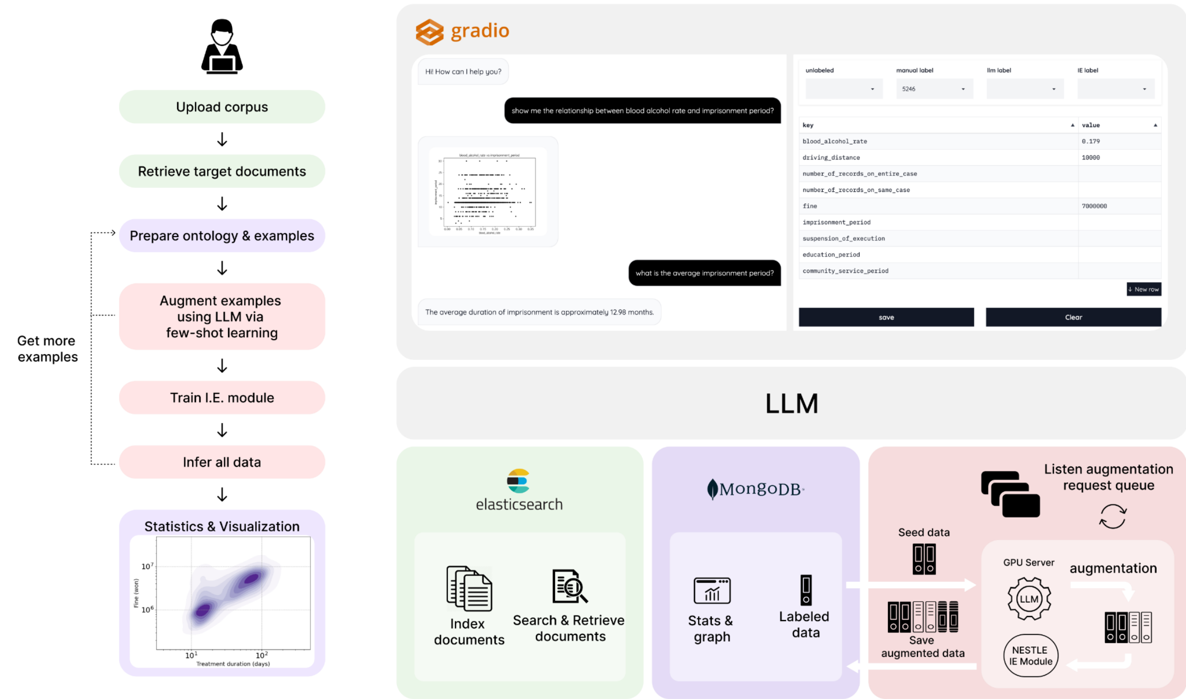

Figure 2: The workflow of NESTLE

ing text structuration, if the target information is not predefined in the ontology of the IE system, one needs to build their own system.

To overcome such limitation, we developed NESTLE[1], a no-code tool for statistical analysis of legal corpus. With NESTLE, users can search target documents, extract information, and visualize statistical information of the structured data via the chat interface, accompanied by an auxiliary GUI for the fine-level controls, such as hyperparameter selection, ontology modification, data labeling, etc. A unique design choice of NESTLE is the combination of LLM and an custom end-to-end IE system (Hwang et al., 2022a) that brings the following merits. First, NESTLE can handle custom ontology provided by users thanks to the end-to-end (generative) property of the IE module. Second, NESTLE can extract target information from the corpus with as few as 4 examples powered by the LLM. For given few examples, LLM builds the training dataset for the IE module under few-shot setting. Finally, the overall cost can be reduced by 200 times, and the inference time can be accelerated by 6 times compared to IE systems that rely exclusively on LLM, like ChatGPT, when analyzing 1 million documents.

We validate NESTLE on three legal AI tasks: (1) 4 Korean Legal IE tasks (Hwang et al., 2022a), (2) 11 new Korean Legal IE tasks derived from LBOX-OPEN dataset (Hwang et al., 2022b), and (3) 3 English legal text classification tasks from LEXGLUE (Chalkidis et al., 2022; Chalkidis, 2023; Tuggener et al., 2020; Lippi et al., 2018). The comprehensive experiments reveal NESTLE can achieves GPT-4 comparable performance with just 4 human-labeled, and 192 LLM-labeled examples. In summary, our contributions are as below.

- We develop NESTLE, a no-code tool that can assist users to perform large scale statistical analysis of legal corpus from a few (4–8) given examples.

- We extensively validate NESTLE on 15 Korean precedent IE tasks and 3 English legal text classification while focusing on three real-world metrics: accuracy, cost, and time[2].

- We show NESTLE can achieve GPT-4 comparable accuracy but with 200 times lower cost and in six times faster inference compared to IE systems that solely rely on commercial LLM like ChatGPT for analyzing 1 million documents.

[1] NO-CODE TOOL FOR STATISTICAL ANALYSIS OF LEGAL CORPUS

[2] The demo is available from http://nestle-demo.lbox.kr. The part of the datasets (including 550 manually curated test set for few-shot IE tasks) will be available from https://github.com/lbox-kr/nestle

2 Related Works

Large Language Model as an Agent With rapid popularization of LLM (OpenAI, 2023; Touvron et al., 2023a,b; Anil et al., 2023; Anthropic, 2023; Taori et al., 2023; Zheng et al., 2023), many recent studies examine the capability of LLM as an agent that can utilize external tools (Liang et al., 2023; Li et al., 2023; Liu et al., 2023; Wang et al., 2023; Song et al., 2023; Zhuang et al., 2023; Tang et al., 2023; Patil et al., 2023; Qin et al., 2023; Viswanathan et al., 2023). There are few studies focusing on the capability of LLM as a data analysis agent. Zhang et al. develop Data-Copilot that can help users to interact with various data sources via chat interface. Ma et al. examines the capability of GPT-3 (CODEX, `code-davinci-002`) as few-shot information extractor on eight NER and relation extraction tasks and propose using LLM to rerank outputs from small language models. Ding et al. evaluate the capability of GPT-3 as a data annotators on SST2 text classification task and CrossNER tasks reporting that GPT-3 shows good performance on SST2. He et al. propose 'explain-then-annotate' framework to enhance LLM's annotation capability. Under their approach, GPT-3.5 achieves either super-human or human-comparable scores on three binary classification tasks.

Our work is different from these previous works in that we focus on building a no-code tool for "statistical analysis" of "corpus" where efficient, accurate, yet customizable methods of structuralization of large-scale documents are necessary. Our work is also different in that we focus on information extraction tasks from legal texts. Finally, rather than performing all IE via LLM, we focus on hybridization between commercial LLM and open-sourced small language model (SLM) by distilling knowledge of LLM to SLM. In this way, the API cost of using LLM does not increase linearly with the size of corpus enabling NESTLE to be applied to industrial scale corpus.

Viswanathan et al. recently proposes Prompt2Model allowing users to construct an NLP system by providing a few examples. Compared to Prompt2Model, NESTLE is specialized in large-scale IE task in legal domain and provides additional features like chat-based statistical analysis and GUI for fine-level control. Also NESTLE is rigorously validated on a variety of legal IE tasks.

Information Extraction from Legal Texts Previous studies build IE systems for legal texts using tagging-based methods (Cardellino et al., 2017; Mistica et al., 2020; Hendrycks et al., 2021; Habernal et al., 2022; Chen et al., 2020; Pham et al., 2021; Hong et al., 2021; Yao et al., 2022) or generative methods (Pires et al., 2022; Hwang et al., 2022a).

Our system is similar to (Hwang et al., 2022a) in that we use an end-to-end IE system and focus on statistical analysis of legal information. However our work is unique in that we present a no-code tool and explore hybridization of commercial LLM and open-sourced SLM to expand the scope of analysis to a large-scale corpus while focusing on three real-world metrics: accuracy, time, and cost.

3 System

NESTLE consists of three major modules: a search engine for document retrieval, a custom end-to-end IE systems, and LLM to provide chat interface and label data. Through conversations with the LLM, users can search, retrieve, and label data from the corpus. After labeling a few retrieved documents, users can structure entire corpus using the IE module. After that, users can conduct statistical analysis through the chat interface using the LLM. Internally, user queries are converted into executable logical forms to call corresponding tools via the "function calling" capability of ChatGPT. The overall workflow is depicted in Fig. 2.

Search Engine The search engine selects a portion of the corpus for statistical analysis from given user queries. Utilizing LLM like ChatGPT, we first extract potential keywords or sentences from user queries, then forward them to the search engine for further refinement and selection. Elasticsearch is used for handling large volumes of data efficiently.

IE Module To structure documents, users first generate a small set of seed examples via either a chat interface or GUI for fine-level control. Then LLM employs these seed examples to label other documents via few-shot learning. The following prompt is used for the labeling

`You are a helpful assistant for IE tasks. After reading the following text, extract information about` *FIELD-1*, *FIELD-2*, ..., *FIELD-n* `in the following JSON format. '`*FIELD-1*`: [value1, value2, ...],` *FIELD-2*`: [value1, value2, ...], ...,` *FIELD-n*`: [value1, value2, ...]'.`
TASK DESCRIPTION

Table 1: Performance of various models on KORPREC-IE task showing the F_1 scores for individual fields: BAC (blood alcohol level), Dist (travel distance), Vehicle (vehicle type), Rec (previous drunk driving record), Loss, Loss-A (aiding and abetting losses), Fine (fine amount), Imp (imprisonment type and period), Susp (execution suspension period), Educ (education period), Comm (community service period). The average scores (AVG) are calculated excluding DRUNK DRIVING task, as all models achieve high scores on it. Scores are based on test sets, each containing 100 examples per task.

Name	LLM module	IE module backbone size	# of training examples (per task)	# of LLM-labeled examples (per task)	AVG	DRUNK DRIVING			EMBZ	FRAUD		RULING-CRIMINAL				
					F_1	BAC	Dist	Rec	Loss	Loss	Loss-A	Fine	Imp	Susp	Educ	Comm
mt5-small[a]	-	0.3B	50	-	58.0	95.8	93.0	90.1	72.2	42.9	0	79.4	89.4	85.7	60.4	34.1
mt5-large[a]	-	1.2B	50	-	63.9	98.0	96.4	93.6	87.5	64.8	0	84.7	82.1	96.7	68.1	27.0
NESTLE-S_0	ChatGPT	0.3B	4[b]	92	62.2	98.0	95.3	93.0	70.1	52.2	0.0	71.2	96.5	93.6	76.7	37.5
NESTLE-S	ChatGPT	0.3B	4	192	64.7	98.0	95.3	89.8	77.3	56.5	0.0	77.4	96.5	98.9	57.1	54.2
NESTLE-L_0	ChatGPT	1.2B	4	92	71.8	97.4	94.7	93.0	84.9	65.3	0.0	86.7	97.9	98.9	82.4	57.9
NESTLE-L	ChatGPT	1.2B	4	192	77.3	98.0	95.3	91.7	87.0	68.0	11.8	88.9	97.9	97.8	94.5	72.7
NESTLE-L+	GPT-4	1.2B	4	192	83.6	-	-	-	90.5	71.2	38.1	89.2	95.8	98.9	96.4	88.9
NESTLE-XXL+	GPT-4	12.9B	4	192	80.4	-	-	-	92.5	72.6	28.6	92.3	96.6	96.8	88.9	75.0
ChatGPT	-	-	4	-	79.6	99.0	95.3	95.2	87.5	75.2	34.8	87.1	97.8	96.5	94.7	63.4
ChatGPT + aux. inst.	-	-	4	-	-	-	-	-	-	75.6	41.7	88.5	98.6	98.8	96.4	72.7
GPT-4	-	-	4	-	88.7	98.5	97.8	92.1	93.5	82.3	59.3	93.9	97.1	98.9	92.6	92.3
ISLA[a]	-	1.2B	–1,000	-	90.3	99.5	97.4	99.0	91.7	80.3	69.6	95.5	95.7	98.9	98.2	92.3

a: From (Hwang et al., 2022a).
b: 8 examples are used in RULING-CRIMINAL task.

INPUT TEXT 1 , PARSE 1

INPUT TEXT 2 , PARSE 2

. . .

INPUT TEXT n , PARSE n

INPUT TEXT [3]

The generated examples are used to train the IE model. We use open-sourced language model multilingual T5 (mt5) (Xue et al., 2021) as a backbone. mt5 is selected as (1) it provides checkpoints of various scale up to 13B, and (2) previous studies show Transformers with encoder-decoder architecture perform better than decoder-only models in IE tasks (Hwang et al., 2022a,b). The model has also demonstrated effectiveness in distilling knowledge from LLM for QA tasks (Li et al., 2022). The trained model is used to parse remaining documents retrieved from previous step.

4 Demo

In this section, we provide the explanation for our demo. The video is also available at `https://youtu.be/twkpjYJrvI8`

Labeling Interface Users can upload their data (unstructured corpus) using an upload button. Alternatively, they can test the system with examples prepared from 7 legal domains by selecting them through the chat interface. Each dataset comes with approximately 1500 documents and 20 manually labeled examples. After loading the dataset, users can view and perform manual labeling on documents using the dropdown menu where the values of individual fields (such as blood alcohol level, fine amount, etc.) can be labeled or the new fields can be introduced. The changes are automatically saved to the database.

IE Module Interface Users can select options such as model size, number of training epochs, and number of training examples within the IE Module Interface. The training of IE module typically takes from 40 minutes to an hour, depending on the parameters above. The data is automatically augmented by LLM when the number of manually labeled examples is less than the specified number of training examples above.

Statistical Analysis Interface Using the chat interface from the second tab of our demo, users can perform various statistical analyses such as data visualization and calculation of various statistics. Users can also retrieve a target document upon request.

5 Experiments

All experiments are performed on NVIDIA A6000 GPU except the experiments with mt5-xxl where eight A100 GPUs are used. The IE module of NESTLE is fine-tuned with batch size 12 with learning rate 4e-4 using AdamW optimizer. Under this condition, the training sometimes becomes unstable. In this case, we decrease the learning rate to 3e-4. The

[3]The original prompt is written in Korean but shown in English for the clarity.

high learning rate is purposely chosen for the fast training. The training are stopped after 60 epochs (NESTLE-S), or after 80 epochs (NESTLE-L, NESTLE-L+). In case of NESTLE-XXL, the learning rate is set to 2e-4 and the model is trained for 20 epochs with batch size 8 using deepspeed stage 3 offload (Ren et al., 2021). For efficient training, LoRA is employed in all experiments (Hu et al., 2022) using PEFT library from Hugging face (Mangrulkar et al., 2022). In all evaluations, the checkpoint from the last epoch is used.

For the data labeling, we use ChatGPT version `gpt-3.5-turbo-16k-0613` and GPT-4 version `gpt-4-0613`. In all other operations with LLM, we use the same version of ChatGPT except during normalization of numeric strings such as imprisonment period and fines where `gpt-3.5-turbo-0613` is used. We set temperature 0 to minimize the randomness as IE tasks do not require versatile outputs. The default values are used for the other hyperparameters. During the few shot learning, we feed LLM with the examples half of which include all fields defined in the ontology while the remaining half are selected randomly.

6 Results

We validate NESTLE on 15 Korean precedent IE tasks and 3 English legal text classification tasks.

15 Korean precedent IE tasks are further divided into two categories: KORPREC-IE which consists of 4 tasks from criminal cases previously studied in (Hwang et al., 2022a) and LBOXOPEN-IE, which is generated from LBOXOPEN (Hwang et al., 2022b) using the factual descriptions from 7 criminal cases and 4 civil cases. In all tasks, a model needs to extract a legally important information from factual description or ruling of cases such as blood alcohol level, fraud loss, fine, and imprisonment period, the duration of required hospital treatment for injuries, etc.

Three classification tasks are EURLEX, LEDGAR, and UNFAIR-ToS from LEXGLUE (Chalkidis et al., 2022; Tuggener et al., 2020; Lippi et al., 2018). EURLEX dataset consists of a pair of European Union legislation (input) and corresponding legal concepts (output) from the EuroVoc Thesaurus. In LEDGAR task, a model needs to classify the paragraphs from contracts originated from US Securities and Exchange Commission fillings. Similarly, UNFAIR-ToS is a task of predicting 8 types of unfair contractual

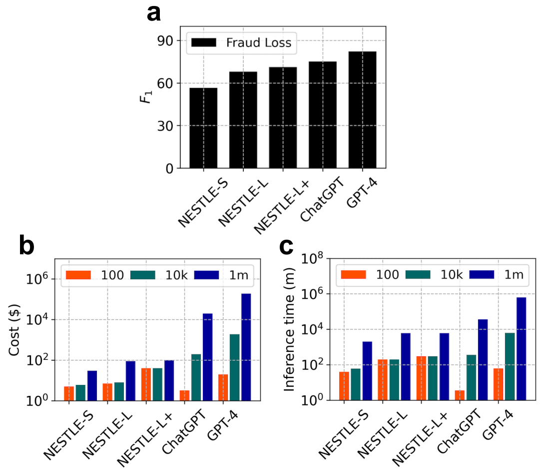

Figure 3: Trade-off analysis on FRAUD task focuses on three real-world metrics: (a) accuracy, (b) cost, and (c) time.

terms for given individual sentences from 50 Terms of Service. These 3 classification tasks are used to demonstrate NESTLE on common (English) legal AI benchmark and also to show NESTLE can be applied to general AI tasks that can be represented in text-to-text format (Raffel et al., 2020).

NESTLE shows competent performance with only four examples We first validate NESTLE on KORPREC-IE that consists of four tasks: DRUNK DRIVING, EMBEZZLEMENT, FRAUD, and RULING-CRIMINAL. With four seed examples and 92 LLM-labeled examples, we train mt5-small (Xue et al., 2021). The result shows that our method already achieves + 4.2 F_1 on average compared to the case trained with 50 manually labeled examples (Table 1, 1st vs 3rd rows, 5th column).

NESTLE can achieve GPT-4 comparable performance To enhance the accuracy of NESTLE, we scale both the quantity of labeled examples by LLM and the size of the backbone of NESTLE's end-to-end IE module. With a greater quantity of LLM-labeled examples (from 92 to 192), NESTLE achieves +2.5 F_1 on average (3rd vs 4th rows) while the labeling time increases (for example, from 2.4 minutes to 10.6 minutes in FRAUD task). With a larger backbone (from mt5-small (0.3B) to mt5-large (1.2B)), NESTLE's shows +9.6 F_1 (3rd vs 5th rows). With both, NESTLE shows +15.1 F_1 (3rd vs 6th rows). However, both the labeling time and the training time increase (for example, from 15 minutes to 170 minutes in FRAUD task).

If the accuracy of teacher model (ChatGPT) is low, the performance of student (mt5) may be

Table 2: Performance of GPT-4 and NESTLE-L on the seven criminal IE tasks from LBOXOPEN-IE. F_1 scores are shown: nRec (the number of identical criminal records), nRec-A (the number of criminal records), Waiver (the victim's intent to waive punishment), Injury (the extent of injuries), and Gender (the victim's gender).

Name	AVG	Indecent Act.[1]			Obstruction[2]		Traffic injuries [3]				Drunk driving [4]				Fraud [5]	Injuries [6]		Violence [7]		
	F_1	nRec	nRec-A	Waiver	nRec	nRec-A	nRec	nRec-A	Waiver	Injury	nRec	nRec-A	BAC	Dist	Loss	Injury	Gender	nRec	nRec-A	Gender
GPT-4	81.1	88.2	85.7	83.1	78.7	82.6	55.6	66.7	68.4	96.0	88.2	88.2	100	99.0	94.9	94.1	81.6	47.1	61.3	81.6
NESTLE-L	78.1	88.9	76.5	52.9	71.8	57.1	73.4	78.0	71.9	95.8	71.8	64.9	100	96.9	81.0	96.9	75.0	64.9	71.8	93.6

1: Indecent act by compulsion (강제추행), 2: Obstruction of performance of official duties (공무집행방해), 3: Bodily injuries from traffic accident (교통사고처리특례법위반(치상), 4: Drunk driving (도로교통법위반(음주운전)), 5: Fraud (사기), 6: Inflicting bodily injuries (상해), 7: Violence (폭행)

Table 3: Performance of GPT-4 and NESTLE-L on the four civil IE tasks from LBOXOPEN-IE. F_1 scores for individual fields are shown: Dom (the event domain such as real estate, fire incident, etc), Ctr (the type of contract), Exp (the amount of money that plaintiffs spent), Loan (the sum of money borrowed by the defendant), and Relat (the relation between plaintiff and defendant).

Name	AVG	Indem[1]			Loan[2]		UFP[3]			LFD[4]		
	F_1	Dom	Ctr	Exp	Loan	Relat	Dom	Ctr	Relat	Dom	Ctr	Relat
GPT-4	83.1	97.0	90.4	95.8	73.2	93.3	93.9	64.9	59.4	92.8	73.9	79.1
NESTLE-L	71.5	73.4	63.9	82.9	59.2	30.5	82.4	78.0	83.7	87.4	64.4	81.0

1: Price of indemnification (구상금), 2: Loan (대여금), 3: Unfair profits (부당이득금), 4: Lawsuit for damages (손해배상(기))

Table 4: F_1 scores of ChatGPT and NESTLE-L on EURLEX, LEDGAR, and UNFAIR-ToS from LEXGLUE were evaluated using 1,000 random samples from their original test sets, following (Chalkidis, 2023). The number of manually labeled examples (n_{train}) and the number of LLM-labeled examples (n_{LLM}) are shown in the 2nd and 3rd columns respectively..

Name	n_{train}	n_{LLM}	EURLEX		LEDGAR		UNFAIR-ToS	
			μ-F_1	m-F_1	μ-F_1	m-F_1	μ-F_1	m-F_1
ChatGPT[a]	8	-	24.8	13.2	62.1	51.1	64.7	32.5
ChatGPT[b]	32	-	33.0	18.3	68.3	55.6	88.3	57.2
NESTLE-L	32	192	34.1	16.7	58.8	41.5	91.5	51.4

a: `gpt-3.5-turbo-0301`. From (Chalkidis, 2023).
b: `gpt-3.5-turbo-16k-0613`.

bounded by it. To check the upper bound of the achievable accuracies, we measure the few-shot performance of ChatGPT. NESTLE-L and ChatGPT shows only 2.3 F_1 difference on average (6th vs 9th rows, 5th column) indicating the student models may approach the upper bound. To improve NESTLE further, we replace ChatGPT with GPT-4. Although the labeling time and cost increase roughly by 10 times, the average scores increase by +6.3 F_1 (Table 1 6th vs 7th rows). Notably, this score is higher than ChatGPT by +4.0 F_1 (7th vs 9th rows).

Next we attempt to scale the backbone of the IE module from mt5-large to mt5-xxl (12.9B). Note that unlike commercial LLMs, the IE module can be trained on multiple GPUs for efficient training and indeed the total training time decreases by 70 minutes even compared to a smaller model (NESTLE-L) by changing GPU from a single A6000 GPU to eight A100 GPUs. However, we could not observe noticeable improvement in F_1.

NESTLE can be generalized to other datasets Although we have validated NESTLE on KORPREC-IE, the dataset mainly consists of numeric fields from criminal cases. For further validation, we build LBOXOPEN-IE from LBOXOPEN (Hwang et al., 2022b). LBOXOPEN-IE consists of 7 tasks from criminal cases (Table 2) and 4 tasks from civil cases (Table 3). Compared to KORPREC-IE, the target fields are more diverse including non-numeric fields such as a contract type, plaintiff and defendant relation, victims' opinion, incident domain, etc as well as numeric fields such as the extent of injury, number of previous criminal records, loan, and more.

We use NESTLE-L and measure the performance on manually curated 550 examples (50 for each task). NESTLE-L achieves a GPT-4 comparable performance in 7 criminal tasks (Table 2, 78.1 vs 81.1) and lower performance in 4 civil tasks (Table 3, 71.5 vs 83.1). This implies NESTLE can be used to glimpse the statistical trend of specific information included in a corpus, but some care must be taken as their accuracies range between ∼70 and ∼90. To overcome this limitation, NESTLE also offers a GUI for rectifying the LLM-augmented examples and collecting more examples manually. In general, higher accuracy can be achieved by utilizing a specialized backbone in the IE module for the target tasks, alongside a more robust LLM, which is a direction for our future work.

Finally, the further validation on three English legal text classification tasks from LEXGLUE shows NESTLE-L can achieve ChatGPT comparable performance (Table 4, 2nd vs 3rd rows).

7 Analysis

We have shown that NESTLE can extract information with accuracies comparable to GPT-4 on many tasks. In this section, we extend our comparison of NESTLE to commercial LLMs focusing on two additional real-world metrics: cost and time. As a case study, we select FRAUD task from KORPREC-IE where all models struggled (Table 1, 11th and 12th columns, Fig. 3a). We calculate the overall cost by summing up (1) manual labeling cost, (2) API cost, and (3) training and inference cost. The manual labeling cost is estimated from the cost of maintaining our own labeling platform (the cost of employing part-time annotators is considered). The API cost is calculated by counting input and output tokens and using the pricing table from OpenAI. The training and inference cost is calculated by converting the training and inference time to dollars based on Lambdalabs GPU cloud pricing. Note that the API cost increases linearly with the size of the corpus when using commercial LLM. On the other hand, in NESTLE, only the inference cost increases linearly with the size of the corpus. The results show that, for 10k documents, the overall cost of NESTLE-L is only 4% of ChatGPT and 0.4% of GPT-4 (Fig. 3b). For 1 million documents, the overall cost of NESTLE-L is 0.5% of ChatGPT and 0.05% of GPT-4 (Fig. 3b). This highlights the efficiency of NESTLE. Similarly, the estimation of overall inference time for 1 million documents reveals NESTLE-L takes 83% or 99% less time compared to ChatGPT or GPT-4 respectively[4].

8 Conclusion

We develop NESTLE, a no-code tool for statistical analysis of legal corpus. To find target corpus, structure them, and visualize the structured data, we combine a search engine, a custom end-to-end IE module, and LLM. Powered by LLM and the end-to-end IE module, NESTLE enables unrestricted personalized statistical analysis of the corpus. We extensively validate NESTLE on 15 Korean precedent IE tasks and 3 English legal text classification tasks while focusing on three real-world metrics: accuracy, time, and cost. Finally, we want to emphasize that although NESTLE is specialized for legal IE tasks, the tool can be easily generalized to various NLP tasks that can be represented in a text-to-text format.

[4]The further detailed comparison is available from `https://github.com/lbox-kr/nestle`

9 Ethical Considerations

The application of legal AI in the real world must be approached cautiously. Even the arguably most powerful LLM, GPT-4, still exhibits hallucinations (OpenAI, 2023) and its performance in the real world legal tasks is still limited (Shui et al., 2023; Zhong et al., 2023; Martinez, 2023). This may imply that AI systems offering legal conclusions should undergo thorough evaluation prior to being made accessible to individuals lacking legal expertise.

NESTLE is not designed to offer legal advice to general users; instead, it aims to assist legal practitioners by providing statistical data extracted from legal documents. Furthermore, to demonstrate the extent to which NESTLE can be reliably used for analysis, we conducted extensive validation on 15 IE tasks. While NESTLE shows generally high accuracy, our experiments reveal that NESTLE is not infallible, indicating that the resulting statistics should be interpreted with caution.

All the documents used in this study consist of Korean precedents that are redacted by the Korean government following the official protocol (Hwang et al., 2022b).

Acknowledgements

We thank Gene Lee for his critical reading of the manuscript, Minjoon Seo for his insightful comments, Paul Koo for his assistant in preparing the figures, and Min Choi for her assistant in preparing the demo.

References

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. 2023. Palm 2 technical report.

Anthropic. 2023. Introducing claude. `https://www.anthropic.com/index/introducing-claude`.

Cristian Cardellino, Milagro Teruel, Laura Alonso Alemany, and Serena Villata. 2017. Legal NERC with ontologies, Wikipedia and curriculum learning. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 254–259, Valencia, Spain. Association for Computational Linguistics.

Ilias Chalkidis. 2023. Chatgpt may pass the bar exam soon, but has a long way to go for the lexglue benchmark. *SSRN*.

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.

Yanguang Chen, Yuanyuan Sun, Zhihao Yang, and Hongfei Lin. 2020. Joint entity and relation extraction for legal documents with legal feature enhancement. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1561–1571, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Boyang Li, Shafiq Joty, and Lidong Bing. 2023. Is GPT-3 a good data annotator? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11173–11195, Toronto, Canada. Association for Computational Linguistics.

Ivan Habernal, Daniel Faber, Nicola Recchia, Sebastian Bretthauer, Iryna Gurevych, Christoph Burchard, et al. 2022. Mining legal arguments in court decisions. *arXiv preprint arXiv:2208.06178*.

Xingwei He, Zhenghao Lin, Yeyun Gong, A-Long Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, and Weizhu Chen. 2023. Annollm: Making large language models to be better crowdsourced annotators.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. *NeurIPS*.

Jenny Hong, Derek Chong, and Christopher Manning. 2021. Learning from limited labels for long legal dialogue. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 190–204, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Wonseok Hwang, Saehee Eom, Hanuhl Lee, Hai Jin Park, and Minjoon Seo. 2022a. Data-efficient end-to-end information extraction for statistical legal analysis. In *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 143–152, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Wonseok Hwang, Dongjun Lee, Kyoungyeon Cho, Hanuhl Lee, and Minjoon Seo. 2022b. A multi-task benchmark for korean legal language understanding and judgement prediction. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Minghao Li, Feifan Song, Bowen Yu, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. 2023. Api-bank: A benchmark for tool-augmented llms.

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, Wenhu Chen, and Xifeng Yan. 2022. Explanations from large language models make small reasoners better.

Yaobo Liang, Chenfei Wu, Ting Song, Wenshan Wu, Yan Xia, Yu Liu, Yang Ou, Shuai Lu, Lei Ji, Shaoguang Mao, Yun Wang, Linjun Shou, Ming Gong, and Nan Duan. 2023. Taskmatrix.ai: Completing tasks by connecting foundation models with millions of apis.

Marco Lippi, Przemyslaw Palka, Giuseppe Contissa, Francesca Lagioia, Hans-Wolfgang Micklitz, Giovanni Sartor, and Paolo Torroni. 2018. CLAUDETTE: an automated detector of potentially unfair clauses in online terms of service. *CoRR*, abs/1805.01217.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. 2023. Agentbench: Evaluating llms as agents.

Yubo Ma, Yixin Cao, YongChing Hong, and Aixin Sun. 2023. Large language model is not a good few-shot information extractor, but a good reranker for hard samples!

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, and Sayak Paul. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Eric Martinez. 2023. Re-evaluating gpt-4's bar exam performance.

Meladel Mistica, Geordie Z. Zhang, Hui Chia, Kabir Manandhar Shrestha, Rohit Kumar Gupta, Saket Khandelwal, Jeannie Paterson, Timothy Baldwin, and Daniel Beck. 2020. Information extraction from legal documents: A study in the context of common law court judgements. In *Proceedings of the The 18th Annual Workshop of the Australasian Language Technology Association*, pages 98–103, Virtual Workshop. Australasian Language Technology Association.

OpenAI. 2023. Gpt-4 technical report.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. 2023. Gorilla: Large language model connected with massive apis. *arXiv preprint arXiv:2305.15334*.

Nhi Pham, Lachlan Pham, and Adam L. Meyers. 2021. Legal terminology extraction with the termolator. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 155–162, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ramon Pires, Fábio C. de Souza, Guilherme Rosa, Roberto A. Lotufo, and Rodrigo Nogueira. 2022. Sequence-to-sequence models for extracting information from registration and legal documents. In *Document Analysis Systems*, pages 83–95, Cham. Springer International Publishing.

Yujia Qin, Shengding Hu, Yankai Lin, Weize Chen, Ning Ding, Ganqu Cui, Zheni Zeng, Yufei Huang, Chaojun Xiao, Chi Han, Yi Ren Fung, Yusheng Su, Huadong Wang, Cheng Qian, Runchu Tian, Kunlun Zhu, Shihao Liang, Xingyu Shen, Bokai Xu, Zhen Zhang, Yining Ye, Bowen Li, Ziwei Tang, Jing Yi, Yuzhang Zhu, Zhenning Dai, Lan Yan, Xin Cong, Yaxi Lu, Weilin Zhao, Yuxiang Huang, Junxi Yan, Xu Han, Xian Sun, Dahai Li, Jason Phang, Cheng Yang, Tongshuang Wu, Heng Ji, Zhiyuan Liu, and Maosong Sun. 2023. Tool learning with foundation models.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Jie Ren, Samyam Rajbhandari, Reza Yazdani Aminabadi, Olatunji Ruwase, Shuangyan Yang, Minjia Zhang, Dong Li, and Yuxiong He. 2021. Zero-offload: Democratizing billion-scale model training.

Ruihao Shui, Yixin Cao, Xiang Wang, and Tat-Seng Chua. 2023. A comprehensive evaluation of large language models on legal judgment prediction. *arXiv preprint arXiv:2310.11761*.

Yifan Song, Weimin Xiong, Dawei Zhu, Cheng Li, Ke Wang, Ye Tian, and Sujian Li. 2023. Restgpt: Connecting large language models with real-world applications via restful apis.

Qiaoyu Tang, Ziliang Deng, Hongyu Lin, Xianpei Han, Qiao Liang, and Le Sun. 2023. Toolalpaca: Generalized tool learning for language models with 3000 simulated cases.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models.

Don Tuggener, Pius von Däniken, Thomas Peetz, and Mark Cieliebak. 2020. LEDGAR: A large-scale multi-label corpus for text classification of legal provisions in contracts. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1235–1241, Marseille, France. European Language Resources Association.

Vijay Viswanathan, Chenyang Zhao, Amanda Bertsch, Tongshuang Wu, and Graham Neubig. 2023. Prompt2model: Generating deployable models from natural language instructions.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2023. Voyager: An open-ended embodied agent with large language models.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Feng Yao, Chaojun Xiao, Xiaozhi Wang, Zhiyuan Liu, Lei Hou, Cunchao Tu, Juanzi Li, Yun Liu, Weixing Shen, and Maosong Sun. 2022. Leven: A large-scale chinese legal event detection dataset. *arXiv preprint arXiv:2203.08556*.

Wenqi Zhang, Yongliang Shen, Weiming Lu, and Yueting Zhuang. 2023. Data-copilot: Bridging billions of data and humans with autonomous workflow.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. 2023. Agieval: A human-centric benchmark for evaluating foundation models.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. 2023. Toolqa: A dataset for llm question answering with external tools.