Cross-domain Analysis on Japanese Legal Pretrained Language Models

Keisuke Miyazaki **Hiroaki Yamada** **Takenobu Tokunaga**
Tokyo Institute of Technology
{miyazaki.k.am@m, yamada@c, take@c}.titech.ac.jp

Abstract

This paper investigates the pretrained language model (PLM) specialised in the Japanese legal domain. We create PLMs using different pretraining strategies and investigate their performance across multiple domains. Our findings are (i) the PLM built with general domain data can be improved by further pretraining with domain-specific data, (ii) domain-specific PLMs can learn domain-specific and general word meanings simultaneously and can distinguish them, (iii) domain-specific PLMs work better on its target domain; still, the PLMs retain the information learnt in the original PLM even after being further pretrained with domain-specific data, (iv) the PLMs sequentially pretrained with corpora of different domains show high performance for the later learnt domains.

1 Introduction

Transformer-based pretrained language models (PLMs) such as BERT (Devlin et al., 2019) and its successors (Liu et al., 2019; Yang et al., 2019; Clark et al., 2020) achieved solid performance in various NLP tasks for a generic domain (Wang et al., 2018). Following their success, domain-specific PLMs have been proposed for science (Beltagy et al., 2019), medical (Alsentzer et al., 2019; Lee et al., 2019), financial (Yang et al., 2020; Loukas et al., 2022), and legal (Chalkidis et al., 2020) domains. These domain-specific PLMs are pretrained solely with the target domain corpora, or with both the generic and target domain corpora. The latter is a good option when the domain corpus size is limited. Gururangan et al. (2020) empirically proved that further pretraining a generic PLM using domain-specific corpora provided benefits; Chalkidis et al. (2020) confirmed this claim for the legal domain.

However, previous studies do not care the performance of the domain-adapted PLMs for a generic domain. The domain adaptation might degrade the model performance for a generic domain. The domain-adapted PLM should perform well in both the target domain and the domain in general. This requirement is essential for the legal domain, where the legal argumentation includes evidence descriptions cited from non-legal text such as web pages, books and SNS posts. The requirement is related to catastrophic forgetting. Ramasesh et al. (2022) recently showed that more steps and data for pretraining make a model robust against catastrophic forgetting. However, their findings are primarily in computer vision, and their experiments with PLMs are still preliminary. They focus on sequential fine-tuning of various size PLMs pretrained with a single domain corpus. On the other hand, we focus on pretraining PLMs with different domains through evaluation using corpora from 13 domains, including domains exclusive of training data. Also, compared with English, there are few findings in domain adaptation strategies of Japanese PLMs, despite several Japanese PLMs available for the generic (NICT, 2020; Tohoku NLP Group, 2021; NLP-Waseda, 2021), financial (Suzuki et al., 2021) and medical (Kawazoe et al., 2021) domains.

Further, despite its significance, no PLM study exists in the *Japanese legal* domain. In the recent COLIEE workshop, a competition on legal information extraction and entailment tasks, including the Japanese language, most high-scoring approaches utilise BERT-like PLMs (Rabelo et al., 2022) trained on Japanese Wikipedia text. Although there is an expectation that PLMs trained with Japanese legal corpora improve their performance, the insufficient size of publicly available corpora does not allow it. Further pretraining a generic PLM with available legal corpora is one of the promising adaptation strategies.

Against this backdrop, particularly considering the above-mentioned legal-domain peculiarity that both domain-specific and generic meanings are equally important, this paper reports the first comprehensive study on PLM adaptation strategies in

Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022, pages 274–281
November 20–23, 2022.

the Japanese legal domain and their performance across different domains through intrinsic evaluation.

2 Research Questions

Chalkidis et al. (2020) adopted two strategies for pretraining domain-specific PLMs: further pretraining (FP) an existing PLM with the domain corpus and pretraining a domain-specific PLM with the domain corpus from scratch (SC). Comparing these two strategies, we investigate the cross-domain performance of domain-specific PLMs, specialised in the Japanese legal domain. We set up the following research questions. **RQ1**: Is the FP/SC learning strategy effective and which is more effective? **RQ2**: Can the domain-adapted PLM learn the domain-specific meaning and distinguish it from the meaning of general usage? **RQ3**: Does the PLM performance change across the domain? **RQ4**: What is the best order of training data domains for pretraining?

3 Experimental Settings

3.1 Resources

Dataset We use the Japanese civil case judgment dataset (JD)[1], the Japanese Wikipedia dataset (WP)[2] and the Balanced Corpus of Contemporary Written Japanese (BCCWJ) (Maekawa et al., 2014). BCCWJ contains texts from 13 domains as shown in Table 4. Their data sizes are 5.4GB (JD), 3.2GB (WP) and 0.7GB (BCCWJ). Table 5 in the Appendix shows the dataset statistics. BCCWJ is used as a test dataset. JD and WP are split into training and test data at a ratio of 9:1, following the NVIDIA BERT implementation (NVIDIA, 2019).

Base PLM We use the BERT-base (WWM version) checkpoint by Shibata et al. (2019), which is pretrained with the Japanese Wikipedia dataset[3].

3.2 Preprocessing

The texts are divided into sentences and further into morphological units. The “short unit” (NINJAL, 2015) is used for BCCWJ, and the output of the morphological analyser JUMAN++ (Tolmachev and Kurohashi, 2018) is used for JD and WP as the morphological unit. The leading meta information, such as the case number, is removed from JD.

[1]provided by LIC Co., Ltd.

[2]version:20220520

[3]The Wikipedia dataset that Shibata et al. (2019) uses is an older dump than WP.

Setting	Strategy	Data size [%]	MLM	NSP
2-phase	FP	100	**0.805**	**0.992**
		50	0.801	0.991
		25	0.793	0.989
	SC	100	**0.789**	**0.991**
		50	0.785	0.991
		25	0.775	0.988
1-phase	FP	100	**0.806**	**0.990**
		50	0.788	0.987
		25	0.763	0.982
	SC	100	**0.785**	**0.989**
		50	0.755	0.984
		25	0.697	0.975
Baseline			0.703	0.687

Table 1: Accuracy of JLBERT family on the JD test set

The SC strategy uses the vocabulary of 32,000 tokens created from the domain corpus by BPE (Sennrich et al., 2016), and the FP strategy uses the vocabulary of the Base PLM for subword tokenisation.

3.3 Pretraining settings

We adopt the masked language modelling (MLM) and next sentence prediction (NSP) tasks to train the BERT model (Devlin et al., 2019). Following NICT (2020), Tohoku NLP Group (2021) and the NVIDIA BERT implementation (NVIDIA, 2019), we use two types of pretraining settings: two-phase (2-phase) and single-phase (1-phase) training. The 2-phase training limits the input token length to 128 in the first phase and enlarges it to 512 tokens in the second phase. The 1-phase training trains the model with the input token length limited to 512. The hyperparameters are the same for the 1-phase training setting and the second phase of the 2-phase training setting. We use the LAMB (You et al., 2020) optimiser. Table 6 in the Appendix shows the hyperparameters for the pretraining settings.

4 Experiments

4.1 RQ1: Pretraining strategies (FP vs SC)

We combine the two pretraining strategies (FP/SC) and the two pretraining settings (1/2-phase) to create four variants of PLMs, which we call the JLBERT family. We further pretrain the base PLM described in 3.1 using the JD dataset for the FP strategy. Only the JD dataset is used for the SC strategy. The model performance is measured through the intrinsic evaluation with the MLM and NSP tasks, i.e. the accuracy of those tasks on the JD test set. To investigate the impact of the training data size on the performance, we created the models with 25%, 50% and 100% of the JD dataset. The number of training steps in the 1-phase setting is reduced to 4,000 and 2,000 according to the dataset reduction, while the number of training steps in the 2-phase setting is fixed to 8,000. We also create a baseline model from the WP dataset using the SC strategy and the 1-phase setting. This baseline model is similar to the base PLM used in the FP strategy. However, the base PLM lacks the classifiers for solving the MLM and NSP tasks. Therefore, we create it from scratch.

Table 1 shows that pretraining with the domain-specific data increases the accuracy for both tasks against the baseline regardless of the pretraining strategies and settings. As the performance of NSP is almost saturated for all JLBERT models, we focus on the MLM performance hereafter. The FP strategy creates better models than the SC strategy, suggesting that out-of-domain data help than no data. This tendency becomes more significant when the domain-specific training data size is small. Increasing the training data size contributes to performance improvement. We need a larger JD dataset to see if the performance improvement has been saturated.

The training time for the first and second phases of the 2-phase setting was 28 and 18 hours, respectively, and 77 hours for the 1-phase setting, using four NVIDIA RTX A6000 GPUs. The 2-phase setting reduced the training time by 40% while retaining a comparable performance with the 1-phase setting. The model parameters learned in the first phase are applicable to inputs longer than 128 tokens, and the model needs to learn only position embeddings beyond 128 tokens in the second phase. It explains the speedup in the 2-phase setting.

4.2 RQ2: Domain specific meanings

RQ2 provides a microscopic analysis of PLMs looking at word meanings, whereas other RQs are macroscopic analysis using overall accuracy as metrics.

While recent PLM analysis researches focus on latent domains and concepts behind representations (Aharoni and Goldberg, 2020; Dalvi et al., 2022; Sajjad et al., 2022), we are interested in words themselves that have drastically different meanings across domains. For instance, "*akui* (maliciousness)" has quite a different meaning, "knowing a fact", in the certain legal context. Moreover, both meanings can simultaneously appear in a single document. We take "*akui*" as a probe word to investigate the domain-specific PLM can learn the domain-specific meaning and distinguish it from its ordinary meaning.

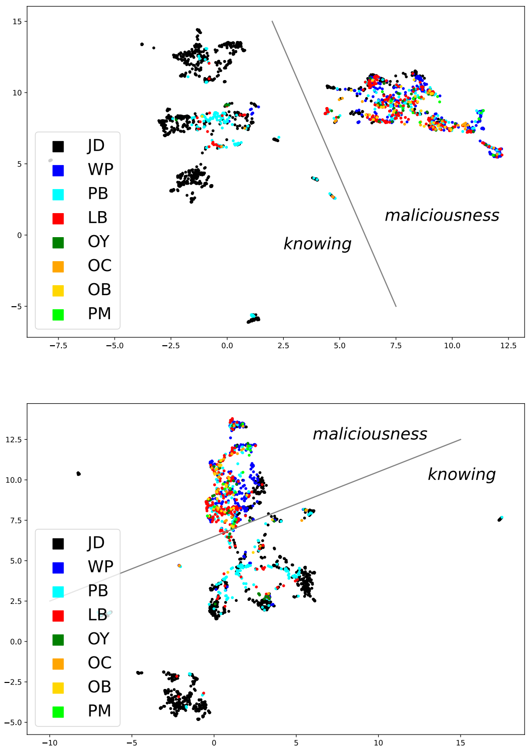

Figure 1: Contextualised embeddings for "*akui*" by JLBERT-2-phase-SC (top) and Base PLM (bottom). Only domains containing $\geq$ 10 occurrences of "*akui*" are depicted. The boundaries are manually annotated. The legend of domain acronyms is found in Table 4.

Following Reif et al. (2019), we collected 2,052 sentences containing "*akui*" from the JD (test), WP(test) and BCCWJ dataset and extracted the corresponding contextualised embedding for "*akui*" in each sentence. Figure 1 visualises the embedding distribution made by UMAP (McInnes and Healy, 2018). We used the base PLM (cf. 3.1), the JLBERT models made by the 2-phase setting and the FP or SC strategies to calculate embeddings. Figure 1 shows that "*akui*" from JD (black), PB (cyan) and LB (red), which would have the legal meaning, made clusters. PB and LB are both book domain, which potentially includes legal materials. These clusters are separable from other domain-mixture clusters. Besides, the boundary is more apparent for the domain-specific PLM.

We also apply the k-nearest neighbour (kNN)

#clusters	2-phase-SC	2-phase-FP	base PLM
2	**0.948** (.000)	0.945 (.000)	0.925 (.001)
3	**0.951** (.004)	0.945 (.000)	0.908 (.000)
4	**0.943** (.005)	0.943 (.002)	0.899 (.003)
5	**0.948** (.001)	0.944 (.004)	0.890 (.008)
6	**0.949** (.000)	0.944 (.003)	0.894 (.001)

Table 2: Global purity of clustered contextualised embeddings of "*akui*" with standard deviations in parentheses.

		Baseline	1-phase-FP	1-phase-SC
	WP (test)	**0.697**	0.589	0.596
	JD (test)	0.703	**0.806**	0.785
BCCWJ	LB	**0.534**	0.521	0.511
	OB	**0.520**	0.512	0.502
	OC	**0.501**	0.492	0.480
	OL	0.739	**0.827**	0.808
	OM	0.566	**0.587**	0.566
	OP	**0.584**	0.580	0.558
	OT	0.584	**0.585**	0.568
	OV	**0.345**	0.305	0.301
	OW	0.637	**0.669**	0.648
	OY	**0.478**	0.455	0.448
	PB	**0.556**	0.549	0.536
	PM	**0.527**	0.492	0.483
	PN	**0.546**	0.504	0.496
	micro avg.	0.538	0.529	0.517

Table 3: Domain-wise accuracy for MLM

clustering to the embeddings to calculate global purity, which indicates the majority's degree of dominance in a cluster. One of the authors[4] annotated the meaning of "*akui*" in the entire sentences for purity calculation. We run the kNN clustering with different numbers of clusters from two to six. The purity is calculated by averaging the results of ten clustering runs with different random seeds. Table 2 shows that the FP and SC strategies always result in higher purity than the base PLM, suggesting that the domain-specific models capture the different meanings of "*akui*" better than the generic model.

4.3 RQ3: Performance across domains

We investigate the model performance on the MLM task across different domains by comparing the baseline model described in 4.1, the JLBERT models made by the 1-phase setting and the FP or SC strategies. The test set includes WP (test), JD (test) and texts from 13 domains of BCCWJ. Table 3 shows that the JLBERT models are superior to the baseline model in law documents (OL), white papers (OW), and minutes of Parliament (OM). These domains contain legal content and follow a formal writing style, similarly to JD. Conversely, the baseline model works better in Yahoo! blog (OY), magazines (PM), newspapers (PN), and verses (OV) that are different in their writing styles from JD. We conclude that the domain-specific PLM degrades its performance outside the target domain but not significantly. Moreover, the FP model is consistently better than the SC model regardless of domains, suggesting that the FP model retains and leverages the information learnt from the WP data even after being pretrained with the JD data.

4.4 RQ4: Order of domain datasets

We compare the MLM performance of two domain-specific PLMs made by the 1-phase setting and the FP strategy, namely WP+JD and JD+WP. The WP+JD model is created by further pretraining the baseline model introduced in 4.1 with JD, while the JD+WP model is created by further pretraining the JLBERT-1-phase-SC model (cf. 4.1) with WP. WP+JD particularly works well in JD (Table 4). In addition, law documents (OL), white papers (OW), and minutes of Parliament (OM), which have a formal writing style similar to JD, also show high scores. On the other hand, JD+WP works well particularly in WP, and also does in newspapers (PN), magazines (PM), and verses (OV). These results indicate that the pretraining for the target domain should be put later in a sequence of pretraining phases to obtain a better domain-specific PLM.

5 Conclusion

This paper presents an empirical study of the pretrained language model specialised in the Japanese legal domain. Our findings are (i) the PLM built with general domain data can be improved by further pretraining with domain-specific data, (ii) domain-specific PLMs can learn domain-specific and general word meanings simultaneously and can distinguish them, (iii) domain-specific PLMs work better on its target domain; still, the PLMs retain the information learnt in the original PLM even after further pretraining with domain-specific data, (iv) the PLMs sequentially pretrained with different domain corpora show high performance for the later learnt domain. Although our findings might be limited in the Japanese legal domain, they provide clues and a basis for future research in other less-studied domains.

[4]The annotator has LL.B. and knowledge in the domain.

		Baseline	(a) WP+JD	Δ	(b) 1-phase-SC	(c) JD+WP	Δ	(a)-(b)	(c)-(a)
	WP (test)	0.697	0.606	-0.091	0.596	**0.718**	0.122	0.010	0.112
	JD (test)	0.703	**0.822**	0.119	0.785	0.694	-0.091	0.037	-0.128
BCCWJ	LB: Books in library	0.534	0.542	0.008	0.511	**0.545**	0.034	0.031	0.003
BCCWJ	OB: Bestseller	0.520	**0.534**	0.014	0.502	0.532	0.029	0.032	-0.003
BCCWJ	OC: Yahoo! Chiebukuro	0.501	**0.523**	0.023	0.480	0.494	0.014	0.043	-0.029
BCCWJ	OL: Law documents	0.739	**0.834**	0.095	0.808	0.741	-0.067	0.026	-0.093
BCCWJ	OM: Minutes of Parliament	0.566	**0.616**	0.050	0.566	0.546	-0.021	0.050	-0.070
BCCWJ	OP: Public relations paper	0.584	**0.606**	0.022	0.558	0.578	0.020	0.047	-0.028
BCCWJ	OT: Textbook	0.584	**0.599**	0.015	0.568	0.597	0.029	0.031	-0.002
BCCWJ	OV: Verse	**0.345**	0.328	-0.017	0.301	**0.345**	0.045	0.028	0.017
BCCWJ	OW: White paper	0.637	**0.679**	0.042	0.648	0.638	-0.009	0.032	-0.041
BCCWJ	OY: Yahoo! Blog	**0.478**	0.479	-0.001	0.448	0.484	0.036	0.031	0.005
BCCWJ	PB: Published books	0.556	**0.570**	0.014	0.536	0.563	0.027	0.034	-0.007
BCCWJ	PM: Magazine	0.527	0.519	-0.008	0.483	**0.534**	0.051	0.036	0.015
BCCWJ	PN: Newspaper	0.546	0.527	-0.020	0.496	**0.557**	0.062	0.031	0.031
BCCWJ	Micro average in BCCWJ	0.538	0.552	0.014	0.517	0.543	0.026	0.035	-0.009

Table 4: Accuracy for MLM: Impact of dataset order in pretraining

As we compared the PLM performance across different domains, we adopted intrinsic evaluation with domain-neutral tasks, MLM and NSP. As Gururangan et al. (2020) did, our future plan includes conducting extrinsic evaluation using downstream tasks like JGLUE (Kurihara et al., 2022).

Acknowledgements

The judgment documents data for this study was provided by LIC Co., Ltd. This work was partly supported by JST ACT-X Grant Number JPMJAX20AM.

References

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online. Association for Computational Linguistics.

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, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. 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.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Fahim Dalvi, Abdul Rafae Khan, Firoj Alam, Nadir Durrani, Jia Xu, and Hassan Sajjad. 2022. Discovering latent concepts learned in BERT. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

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.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Yoshimasa Kawazoe, Daisaku Shibata, Emiko Shinohara, Eiji Aramaki, and Kazuhiko Ohe. 2021. A clinical specific bert developed using a huge japanese clinical text corpus. *PLOS ONE*, 16(11):1–11.

Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. 2022. JGLUE: Japanese general language understanding evaluation. In *Proceedings of the Language Resources and Evaluation Conference*, pages 2957–2966, Marseille, France. European Language Resources Association.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

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. *ArXiv*, abs/1907.11692.

Lefteris Loukas, Manos Fergadiotis, Ilias Chalkidis, Eirini Spyropoulou, Prodromos Malakasiotis, Ion Androutsopoulos, and Georgios Paliouras. 2022. FiNER: Financial numeric entity recognition for XBRL tagging. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4419–4431, Dublin, Ireland. Association for Computational Linguistics.

Kikuo Maekawa, Makoto Yamazaki, Toshinobu Ogiso, Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. 2014. Balanced corpus of contemporary written japanese. *Lang. Resour. Evaluation*, 48(2):345–371.

Leland McInnes and John Healy. 2018. UMAP: uniform manifold approximation and projection for dimension reduction. *CoRR*, abs/1802.03426.

NICT. 2020. NICT BERT Japanese Pre-trained model. `https://alaginrc.nict.go.jp/nict-bert/index.html`. Accessed: 2022-7-13.

NINJAL. 2015. *Guide for using the Balanced Corpus of Contemporary Written Japanese, Version 1.1*. Center for Language Resource Development.

NLP-Waseda. 2021. nlp-waseda/roberta-base-japanese · Hugging Face. `https://huggingface.co/nlp-waseda/roberta-base-japanese`. Accessed: 2022-7-13.

NVIDIA. 2019. PyTorch/LanguageModeling/BERT · NVIDIA/DeepLearningExamples. `https://github.com/NVIDIA/DeepLearningExamples/tree/master/PyTorch/LanguageModeling/BERT`. Accessed: 2022-7-13.

Juliano Rabelo, Randy Goebel, Mi-Young Kim, Yoshinobu Kano, Masaharu Yoshioka, and Ken Satoh. 2022. Overview and discussion of the competition on legal information extraction/entailment (COLIEE) 2021. *Rev. Socionetwork Strateg.*, 16(1):111–133.

Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. 2022. Effect of scale on catastrophic forgetting in neural networks. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. 2019. Visualizing and measuring the geometry of bert. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Firoj Alam, Abdul Khan, and Jia Xu. 2022. Analyzing encoded concepts in transformer language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3082–3101, Seattle, United States. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Tomohide Shibata, Daisuke Kawahara, and Sadao Kurohashi. 2019. Improving the accuracy of japanese parsing with BERT. In *The Proceedings of the Twenty-fifth Annual Meeting of the Association for Natural Language Processing*, pages 205–208, Nagoya, Japan.

Masahiro Suzuki, Hiroki Sakaji, Masanori Hirano, and Kiyoshi Izumi. 2021. Construction and validation of a pre-trained language model using financial documents. In *Proceedings of JSAI Special Interest Group on Financial Infomatics (SIG-FIN) 27*, pages 5–10.

Tohoku NLP Group. 2021. cl-tohoku/bert-base-japanese-v2 · Hugging Face. `https://huggingface.co/cl-tohoku/bert-base-japanese-v2`. Accessed: 2022-7-13.

Arseny Tolmachev and Sadao Kurohashi. 2018. Juman++ v2: A practical and modern morphological analyzer. In *The Proceedings of the Twenty-fourth Annual Meeting of the Association for Natural Language Processing*, pages 917–920, Okayama, Japan.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the*

2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Yi Yang, Mark Christopher Siy Uy, and Allen Huang. 2020. FinBERT: A pretrained language model for financial communications. *CoRR*, abs/2006.08097.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. 2020. Large batch optimization for deep learning: Training BERT in 76 minutes. In *International Conference on Learning Representations*.

A Statistics of datasets

Dataset	Genre	#sents	#chars per sent	#morphs per sent
WP	Train	22,053,315	48.1	26.9
	Test	2,450,176	56.8	31.9
	Overall	24,503,491	48.9	27.4
JD	Train	21,411,914	77.0	46.8
	Test	2,378,943	76.6	46.2
	Overall	23,790,857	77.0	46.8
BCCWJ	LB	1,649,778	33.5	21.2
	OB	222,540	30.5	19.5
	OC	681,967	28.2	17.5
	OL	38,768	45.5	30.6
	OM	140,409	63.3	39.9
	OP	256,199	26.9	17.5
	OT	63,667	27.1	17.2
	OV	18,982	19.7	12.1
	OW	146,280	57.7	37.9
	OY	820,922	24.7	15.0
	PB	1,482,226	35.3	22.2
	PM	300,212	29.1	17.4
	PN	80,037	31.0	19.7
	Overall	5,901,987	32.7	20.6

Table 5: Statistics of preprocessed datasets

Table 5 shows the statistics of the datasets used in this study. These values are calculated after preprocessing (3.2). Comparing WP and JD, the numbers of recording sentences are almost the same. Therefore, when learning WP or JD under the same 1-phase condition in RQ4 (4.4), the number of epochs is also almost the same.

On the other hand, the number of characters and morphemes per sentence on JD is much higher than WP. Compared to WP, JD is not only a formal written document, but also has a long sentence. For this reason, it makes sense to create a JD-specific PLM to solve JD's downstream tasks.

B Pretraining hyperparameters

	2-phase phase1	2-phase phase2	1-phase
Accumulated batch size	32,768	16,384	16,384
Mini-batch size	64	8	64
Gradient accumulation	512	2,048	256
Training steps	7,038	1,563	8,000
Mini-batch inputs	3.6M	3.2M	2M
Warm-up steps	2,000	200	1,024
Warm-up rate	28.43%	12.80%	12.80%
Max length of tokens	128	512	512
[MASK] rate	0.15	0.15	0.15
Max [MASK]/sentence	20	80	80
Learning rate	0.006	0.004	0.004

Table 6: BERT pretraining hyperparameters

Table 6 shows the detailed settings of 1-phase and 2-phase (3.3). As shown in (3.3), the computing time for the first and second phases in the 2-phase setting was 28 and 18 hours, respectively, and 77 hours for the 1-phase setting, using four NVIDIA RTX A6000 GPUs. By changing the Mini-batch size in 2-phase phase 2 to 64, computing time will be shorter.

C Statistics of annotated "*akui*"

		knowing	malice	?	Sum
	JD (test)	882	200	6	1088
	WP (test)	0	317	0	317
BCCWJ	LB	19	203	1	223
	OB	0	28	0	28
	OC	0	35	1	36
	OL	2	1	0	3
	OM	0	6	0	6
	OT	0	1	0	1
	OV	0	3	0	3
	OY	4	38	1	43
	PB	130	154	3	287
	PM	0	15	0	15
	PN	0	2	0	2
	Sum	1037	1003	12	2052

Table 7: Statistics of annotated "*akui*"

Table 7 shows the statistics of annotated sentences which contain the word "*akui*". The "?" column shows sentences that cannot be classified into either "knowing a fact (technical usage in the legal domain)" or "malicious (general usage)".

According to our annotation, 200 out of 1088 sentences mean “malicious” in JD (test). Even in JD, which is a corpus of legal domain, “*akui*” does not always mean “knowing a fact” but also means “malicious”. For example, a legal argumentation includes evidence descriptions cited from non-legal text such as web pages, books and SNS posts. Moreover, both meanings can simultaneously appear in a single document. Thus, source of documents does not necessarily suggest which meaning “*akui*” has.