Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2023) 000–000

Procedia
Computer Science

www.elsevier.com/locate/procedia

Seventh Information Systems International Conference (ISICO 2023)

Vietnamese Legal Text Retrieval

Nguyen Hoang Gia Khang[a], Nguyen Minh Nhat[a], Nguyen Quoc Trung[a], Truong Hoang Vinh[b]

[a] *Department of Information Technology, FPT University, Ho Chi Minh city, Vietnam*
[b] *Faculty of Information Technology, Ho Chi Minh City Open University, Vietnam*

Abstract

We will introduce the combination of two techniques: Sparse Retrieval and Dense Retrieval, while experimenting with different training approaches to find the optimal method for the Vietnamese Legal Text Retrieval task. Moreover, the Question Answering task was only built on the open domain of UIT-ViQuAD but shown promising results on the in-domain legal dataset. Finally, we also mentioned the data augmentation of legal documents up to 3GB to train the Phobert language model, improve this backbone with Condenser, Cocondenser in this paper. Furthermore, these techniques can be utilized for other information retrieval assignments in languages with limited resources.

Keywords: Sparse Retrieval; Dense Retrieval; Vietnamese Legal Text Retrieval; Question Answering; UIT-ViQuAD; Phobert language model.

1. Introduction

Vietnam's legal system has its own distinct and relatively complex features, from the Constitution to local-level regulations and directives. Therefore, traditional search methods or online tools integrated into legal websites can make it difficult to retrieve legal information or research laws. Our goal is to develop a system consisting of two main parts: "Vietnamese Legal Text Retrieval" and "Question Answering". In the first part, we will focus on analyzing and using the Condenser architecture to perform queries combined with Sparse Retrieval, which is BM25, to find specific laws. In the second part, "Question Answering," we will extract the most relevant information from those laws for users' queries. We also optimized data processing for training in the two main tasks because legal data requires high precision, but sometimes the datasets used cannot guarantee it, so we need to reprocess them for specific tasks.

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.
E-mail address: khangnhgse150829, nhatnmse150958, trungnq46@fpt.edu.vn; vinh.th@ou.edu.vn

2. Background and Related Work

2.1. Sparse Retrieval

Sparse retrieval methods are mainly based on relevant, similarity between two documents, or between a query and documents based on keywords that appear in both. Sparse Retrieval is famous for its classic algorithms including algorithms like BM25 [1], TFIDF [2], of which BM25+ [3] is the best algorithm in this approach. The biggest disadvantage of these traditional information retrieval methods rely on keyword matching and simple scoring functions, which can lead to suboptimal results. The grammatical structure of the sentence will not be considered

2.2. Dense Retrieval

Using a pre-trained deep neural language model, it is possible to encode questions or supporting documents into a continuous latent semantic embedding vector,allowing for more accurate similarity matching and ranking of relevant passages. Dense passage retrieval is helpful because it allows for the efficient retrieval of relevant passages or documents in response to a given query. However, dense passage retrieval uses deep neural language models to encode queries and documents into continuous latent semantic vectors, allowing for more accurate similarity matching and ranking of relevant passages. The success in recent researches such as [4] [5] or [6] is contributed by dense retrieval approaches.

2.3. Cross-encoder approaches

The cross-encoder consists of a backbone language model, which can be BERT [7], RoBERTa [8], or any other transformer encoder model. It is able to capture global interactions between a query and a document [9][10]. The input consists of a pair of a query and a document, which are passed through the backbone language model to generate a joint representation that captures the relationship between the two inputs. However, the cross-encoder approach also has some drawbacks, such as higher computational complexity and longer training times compared to the dual-encoder approach. Nevertheless, its advantages in capturing global interactions between inputs make it a powerful approach for dense retrieval in information retrieval and related fields.

2.4. Dual-encoder approaches

The dual-encoder approach involves two backbone language models, typically also transformer encoder models. One model is trained to encode queries, while the other is trained to encode documents. The dual-encoder approach maps input queries and output targets to a shared vector space, where the inner products of the query and target vectors can be used as a reliable similarity function. In practice, dual encoders accomplish the ability to scale to a large number of targets through two mechanisms: sharing weights among targets by means of a parametric encoder, and employing a scoring function based on inner products that is efficient. Therefore, it is a potential research topic of [11],[12]

2.5. Sequence-to-Sequence (Seq2Seq) for question answering

In an extractive question answering task, the goal is to identify the answer span within a given context that best answers a question. While encoder models are currently the preferred choice for extractive question answering, sequence-to-sequence (Seq2Seq) is also proved its great performance in extractive question answering via [13] [14] [15]. Recently, a novel method for applying Transformer models to extractive question answering tasks has been proposed [16]

3. Methods

The Vietnamese Legal Text Retrieval system uses a Retriever module that transforms a question into an embedding vector representation and compares it to a knowledge base of legal documents using similarity scores such as dot product, cosine similarity, and Euclidean distance. Relevant legal documents are then retrieved, either in the form of full text or just titles, but this approach can have limitations. To address this, a question answering model has been developed to extract the main idea from returned circulars and decrees text, improving the efficiency and user experience of legal searches.

3.1. Legal Text Retrieval

3.1.1. Processing Data

The search engine uses the ranking algorithm BM25 to determine the relevance of a group of documents to a given question. It ranks documents based on the search terms in each record and creates negative sentence pairs to train Sentence Transformer in 3.1.5. The negative training samples consist of the most relevant articles to the query, but may not be the correct label. Data is divided into positive and negative samples to train Contrastive Learning [17], which pulls the correct relevant sentence close to an anchor and pushes wrong samples away. Positive samples are from the dataset's answer data (documents that match a given query), while negative samples are created by using BM25 to extract 50 or 20 sentences close to a given query. Data is preprocessed before being fed into the lexical matching algorithm. Preprocessing techniques include word segmentation with Pyvi and text conversion to uppercase. The Retriever module is trained in two rounds based on [18]. Round one uses negative pairs extracted by BM25 to train Sentence Transformer with Contrastive loss. Round two uses the Sentence Transformer trained in round one to predict False Negative samples and continue to train Contrastive loss on new data.

3.1.2. In-domain LegalPhoBERT

The backbone language model we used is PhoBERT [19], a transformer encoder model for the Vietnamese language. Although PhoBERT is trained on an open-domain dataset like Vietnamese Wikipedia, it may not perform well in specific domains such as the legal domain. To improve the model's ability to encode the semantics of legal documents accurately, we followed the approach of [20] by fine-tuning RoBERTa on 4GB of legal text data. In our case, we fine-tuned PhoBERT on the Masked Language Modeling (MLM) task using two versions of the legal dataset, one with 300MB and the other with 3GB. We chose this approach not only to match our resource capabilities but also to observe the performance difference between the two dataset sizes. The dataset used for fine-tuning was collected similarly to [20] but with less data. In this way, we transformed an open-domain PhoBERT into an in-domain legal language model, which is particularly useful when pre-training a full PhoBERT on a large legal dataset is not feasible. The new legal PhoBERT is then used to train the condenser in the next section.

3.1.3. Pretrain Condenser

After LegalPhoBERT have been finetuned with MLM, it is taken to train Condenser [21]. Transfrormer model is used in semantic textual similarity takes [CLS] token, a representative of all word embedding in a sequence. However, [21] show that each token in the sequence, including the [CLS], only pays heed when receiving information from other tokens. The [CLS] token in the Transformer architecture plays a crucial role in knowledge aggregation, but its effectiveness is determined by attention patterns. According to a study, the [CLS] token is not attended to by other tokens in most middle layers, but it exhibits similar attention patterns to other tokens. The Condenser model is introduced to address this issue by collecting information into the [CLS] token using the Masked Language Modeling (MLM) objective, resulting in a dense representation. By doing so, the Transformer architecture can be effectively utilized for language modeling tasks, making it more robust to errors.

3.1.4. Pretrain Cocondenser

Checkpoint of Condenser from the previous Section 3.1.3 is then trained on coCondenser approach. The creation of dense retrieval models through pre-trained language models can be difficult due to the high computational and human resource requirements for training and dataset processing. Dense retrieval models often face two main challenges: noise and large batch sizes during training. One approach to address these issues is RocketQA [10], which removes hard negative pairs during training and increases batch sizes. However, this method may not be feasible for those with limited resources. To overcome these challenges, [22] proposed a solution that trains the backbone language model to have local anti-interference abilities and creates dense retrieval using a corpus-level contrastive learning objective. The coCondenser pre-training technique uses Condenser pre-training and corpus-level contrastive learning to generate an information-rich [CLS] representation, allowing the model to perceive a wider range of observations in an unsupervised manner. This one-time pre-training process is independent of end task queries and enables the resulting model to be applied to various queries without retraining.

3.1.5. Build Sentence Transformer

The transformer model creates token-level embeddings, but we need sentence-level embeddings for text similarity tasks. Sentence-BERT (SBERT) was developed to provide sentence embeddings that outperform previous state-of-the-art models for standard semantic textual similarity (STS) tasks. Many other sentence transformer models have been developed using similar ideas and loss functions to optimize embeddings for related and dissimilar sentences. The backbone of our SBERT is LegalPhoBERT that is improved with coCondenser, an unsupervised corpus-aware language model pre-training method exaplained in the previous Section 3.1.4 , and it is trained with a siamese network using contrastive learning instead of triplet networks. Moreover, the architecture of our SBERT model is designed as dual-encoder. Sampling data for training, workflow and loss function we follow [23]

3.2. Question Answering

The article discusses the use of question answering models in legal domain. There are two types of question answering models: extractive and abstractive. Extractive question answering is preferred in legal domain because it provides accurate answers based on given knowledge without changing or paraphrasing the answer. Given a query sentence and one or more support documents, a generative approach using a ViT5 [24] backbone is recommended to extract a pieces of information from given support documents. While encoder models are currently the preferred choice for extractive question answering, sequence-to-sequence (Seq2Seq) models can also have advantages in certain scenarios:

- Non-contiguous answer spans: Seq2Seq models are well-suited for tasks where the answer span is not a contiguous sequence of words. This is because they are capable of producing an output sequence that is not constrained to a fixed length.
- Additional context: Seq2Seq models can incorporate additional context beyond the given passage to answer a question. This can be useful for tasks that require a broader understanding of the topic or require reasoning beyond the information contained in the input passage.
- Language Generation: Seq2Seq models are capable of generating natural language output, which can be useful for tasks that require the system to generate an answer in a specific format or style.

Morevover, we notice that predicting 2 kinds of number is not associated to the idea of question answering task. We pass a query sentence and its relevant knowledge, then expect the model to return the answers existing in one of given contexts. start position and end position are not originally the idea of using language model to create an answer, answer a question like human. Model is better to predict a span of text where each token is generated based on the previous one until complete the answer. Predicting the beginning of the supposed answer and its ending token, then slice the span of text is not appropriate. With the hundreds of legal data, it is impossible to perform question answering when using ViT5. Therefore, we train extractive question answering on open-domain dataset first, then inference on the small dataset we have from legal contest. We decide to literally train ViT5 to perform reading comprehension ability. We also follow [20] that re-use the backbone language model to train question answering task but it does not perform well. Result and examples are explained in detail in later Section.

4. Experiments and Results

4.1. Legal Text Retrieval

Dataset: The data we used for the "Vietnamese Legal Text Retrieval" section comes from the "Legal Text Retrieval" dataset in the "ZALO AI Challenge 2021", which includes up to 3200 legal articles. In addition, we collected legal data from Vietnamese sources such as "Lawnet.vn" and "vbpl.vn", with a total of nearly 145,000 legal documents spanning from October 2018 to January 2023. After processing steps to filter out noise and remove duplicate words in sentences, we obtained nearly 3GB of legal data.

Data Processing: For collecting data from Vietnamese law websites, we follow the procedure of standardizing punctuation and using pre-built Vietnamese text forms. The creation of combined and pre-built Vietnamese character sets has facilitated the use of Vietnamese characters on computers, including all the characters in the Vietnamese alphabet, along with tone and punctuation marks. While both sets are crucial in supporting Vietnamese on computers and mobile devices, users must use the correct characters and punctuation marks to avoid confusion or loss of meaning. In Vietnamese, punctuation marks play a significant role in distinguishing the meanings of words. Incorrect use of punctuation can lead to misunderstandings or loss of meaning. Therefore, using punctuation correctly is crucial in Vietnamese.

Table 1. The results of legal text retrieval versions

The metrics	Training data	F1 Score(val)
SB-Condenser-100MB	100MB	0.61
SB-Condenser-300MB-Lite	300MB	0.63
SB-Condenser-300MB-Full	300MB	0.63
SB-Condenser-3GB	3GB	0.66

The result: with regard to the summary table 1, we introduce four versions of a conference we tested based on the Condenser architecture to solve the problem of "Vietnamese Legal Text Retrieval".

- **In the first version (SB-Condenser-100MB):** we use 100MB of data from the Zalo Legal Text 2021 competition. We use 100 MB of data to pretrain Masked Language Model, although it is not much, it helps the language model understand the characteristics of words in the legal field linked together. In the following rounds, Condenser and Coconder, we continue to use this data to understand the context of each sentence and the links between legal terms and the separate content of each term.
- **In the second version (SB-Condenser-300MB-Lite):**, we added 200MB of data collected from reputable legal websites in Vietnam to supplement the initial 100MB data. First, we used the 300MB data to fine-tune the Phobert language model. Then, we used the checkpoint obtained from fine-tuning to train in subsequent rounds, with 100MB of data retained for training in each round.
- **In the third version (SB-Condenser-300MB-Full):**, we used 300MB of data similar to experiment 2, but this time we trained all rounds with the 300MB data, including Pretrain Masked Language Model, Pretrain Condenser, Pretrain Cocondenser. For the final round, Sentence Transformer, we reused 100MB because the task was to combine Sparse Retrieval and Dense Retrieval on the domain dataset to answer questions.
- **In the fourth version (SB-Condenser-3GB):**,we added 2.9 GB of data collected from reputable legal websites in Vietnam to supplement the initial 100MB data. First, we used the 3GB data to fine-tune the Phobert language model. Then, we used the checkpoint obtained from fine-tuning to train in subsequent rounds, with 100MB of data retained for training in each round.

We trained a total of 4 versions on different amounts of data: 100MB, 300MB, and 3GB. In particular, for the 300MB data, we split it into two training methods. First, with the SB-Condenser-300MB-Lite version, we used 300MB of training data for pretraining the masked language model, while in subsequent iterations, we used the original 100MB data in the Zalo domain for the SB-Condenser-300MB-Full experiment, training with 300MB for all steps. The results of the two 300MB experiments exceeded 0.63, which opened up a new training approach where we can apply the training method of the SB-Condenser-300MB-Lite version to save time and training resources. For the 3GB experiment, we applied the training method of the 300MB-Lite version and achieved an F1 score of over 0.66.

Table 2. The F2 Score of legal text retrieval versions

Version	Metric	Score
SB-Condenser-300MB-Lite	F2	0,699
SB-Condenser-300MB-Full	F2	0,649
SB-Condenser-3GB	F2	0.723

In terms of the table 2, we evaluated our trial version using the F2 Score evaluation method. In the field of law, Recall plays a crucial role as it indicates the proportion of predictions that match the labels. Our highest F2 Score result was achieved with the SB-Condenser-3GB version, with a score of 0.723. The Lite version trained on 300MB achieved a score of 0.699, while the Full version achieved a score of 0.649.

4.2. Question Answering

Table 3. The result of question answering version (Vqa-ViT5)

Version	Metric	Score
Vqa-ViT5	F1	0,646
Vqa-ViT5	EM	0,41
Vqa-ViT5	rougeL	0,66

Dataset: In order to address the problem of limited training data for legal question answering, the authors trained their legal Phobert model with UIT-ViQuAD [25], a collection of 23,000 questions and answers created by humans using passages from 174 Vietnamese Wikipedia entries. By doing this, the extractive question answering model can first learn reading comprehension skills before being applied or performed inference on the 520 legal questions and 1377 articles from the Automated Legal Question Answering Competition (ALQAC 2022). After checking 520 questions, we found that there are 9 questions contain the answer is not a pieces of information that can be extracted from a given question's corresponding articles. Because we trained our Vqa-ViT5 to perform extractive question answering task, we discard these unsuitable question and retain 511 samples

The results: We used three evaluation methods, namely F1 Score, Exact Match, and ROUGE, to evaluate the performance of our Vqa-ViT5 experiment, which was trained on the dataset UIT-ViQuAD [25] with data spanning various domains. For evaluation, we used 511 pairs of legal questions and answers from the ALQAC-2022 competition, in contrast to the commonly used approach of training on legal data and using a small set of validation data to evaluate results. Table 3 shows our Vqa-ViT5 model trained on the comprehensive dataset and using all 511 questions achieved an F1 Score of 0.646. The more accuracy-demanding evaluation method, Exact Match (EM), yielded a score of 0.41. [20] gives their result about 90% on ALQAC-2022. However, they mentions that their F1 score is on the dev set which they randomly pick 15% of the official dataset. It means their validation dataset (about 78 data) is smaller than us (511 data). Moreover, our Vqa-ViT5 is not trained on ALQAC-2022 dataset at all but it still gives acceptable result. Table 4 shows some examples from Vqa-ViT5, at the first question, model predict correctly answer in token-level, obviously its EM is 1.0. Take a look at the second question, our model predicts a span of text that includes the actual label, although the prediction is not incorrect, but EM metric still return 0.0, and most of inference result is the same with this situation, that why we said the result is kind of acceptable but the EM score is not pretty high. Therefore In Table 4, we also compute rougL score in order not to discard acceptable result like EM score because rougL score measures the similarity between a machine-generated summary or translation and a reference summary or translation by computing the longest common subsequence (LCS) between them. The LCS is the longest sequence of words that appears in both the generated and reference summaries. However, result of rougeL score is just approximate to indicate that there are some semantically correct predicted answers are discarded by EM, rougeL metric is not popularly used for extractive question answering task because of answer's representation.

Table 4. Vqa-ViT5 example result table

question:	Chiếm đoạt di vật của tử sĩ có thể bị phạt tù lên đến bao nhiêu năm? (Appropriating relics of martyrs can be punished with up to how many years?)
context:	Tội chiếm đoạt hoặc hủy hoại di vật của tử sỹ ...thì bị phạt tù từ 02 năm đến **07 năm**: a) Là chỉ huy hoặc sĩ quan; b) Chiếm đoạt hoặc hủy hoại di vật của 02 tử sỹ trở lên. (The crime of appropriating or destroying the relics of martyrs ... shall be punishable by imprisonment from 02 years to **07 years**: a) Being a commander or officer; b) Appropriating or destroying relics of 02 or more martyrs.)
prediction:	07 năm (07 years)
label:	07 năm (07 years)
question:	Người đã nhận làm gián điệp, nhưng không thực hiện nhiệm vụ và sau đó tự nguyện báo cáo với cơ quan nhà nước có thẩm quyền, thì họ sẽ không bị kết tội về hành vi gián điệp? (A person who has accepted to act as a spy, but fails to perform the assigned tasks and confesses and honestly declares to the competent state agency, shall be exempt from any responsibility for espionage charges?)
context:	Tội gián điệp 1. Người nào có một trong các hành vi... a) Hoạt động tình báo, ... nếu tự nguyện báo cáo với cơ quan nhà nước có thẩm quyền một cách thành thật, thì sẽ không bị xử lý trách nhiệm **hình sự** về tội này. (Crime of espionage 1. Any person who commits one of the acts... a) Intelligence activities, ... sincerely declares to the competent state agency, shall be exempt from responsibility **criminal** about this crime.)
prediction:	miễn trách nhiệm hình sự (exempt from criminal liability)
label:	hình sự (Criminal)

5. Conclusion

In this papper, we presented the development of the "Vietnamese Legal Text Retrieval" model through multiple training steps, along with the use of data to find the optimal training approach. We also used common evaluation methods, F1 and F2 Score, to assess the performance of the model. In addition, we developed an improved Question Answering model for query retrieval and experimented with training on multi-domain datasets, resulting in highly promising results compared to top-rated articles on the same topic.

In the future, we will enhance the data to train the mentioned models and reduce the noise level of the collected data. Moreover, building a large data repository for quick and comprehensive querying of current laws will increase the feasibility of applying this topic and expand it to other languages.

Appendix A. Evaluation Methods

F2 Score: being a metric that combines precision and recall. It is calculated as follows:

$$F_2 = (1+\beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall} \tag{A.1}$$

- where β is a parameter that adjusts the relative weight of precision and recall.
- To calculate F2 score, we first need to calculate precision and recall from the confusion matrix.

Precision: being the ratio of true positives to the sum of true positives and false positives:

$$precision = \frac{TP}{TP + FP} \tag{A.2}$$

Recall: being the ratio of true positives to the sum of true positives and false negatives:

$$recall = \frac{TP}{TP + FN} \tag{A.3}$$

References

[1] Stephen Robertson, Hugo Zaragoza, and Michael Taylor. Simple bm25 extension to multiple weighted fields. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 42–49, 2004.

[2] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[3] Stephen E Robertson and Steve Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pages 232–241. Springer, 1994.

[4] Qiao Jin, Andrew Shin, and Zhiyong Lu. Lader: Log-augmented dense retrieval for biomedical literature search, 2023.

[5] Avirup Sil, Jaydeep Sen, Bhavani Iyer, Martin Franz, Kshitij Fadnis, Mihaela Bornea, Sara Rosenthal, Scott McCarley, Rong Zhang, Vishwajeet Kumar, Yulong Li, Md Arafat Sultan, Riyaz Bhat, Radu Florian, and Salim Roukos. Primeqa: The prime repository for state-of-the-art multilingual question answering research and development, 2023.

[6] Qiuhong Zhai, Wenhao Zhu, Xiaoyu Zhang, and Chenyun Liu. Contrastive refinement for dense retrieval inference in the open-domain question answering task. *Future Internet*, 15(4):137, 2023.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[8] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[9] Ruiyang Ren, Shangwen Lv, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. PAIR: Leveraging passage-centric similarity relation for improving dense passage retrieval. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics, 2021.

[10] Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2010.08191*, 2020.

[11] Nicholas Monath, Manzil Zaheer, Kelsey Allen, and Andrew McCallum. Improving dual-encoder training through dynamic indexes for negative mining, 2023.

[12] Xuan Fu, Jiangnan Du, Hai-Tao Zheng, Jianfeng Li, Cuiqin Hou, Qiyu Zhou, and Hong-Gee Kim. Ss-bert: A semantic information selecting approach for open-domain question answering. *Electronics*, 12(7):1692, 2023.

[13] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2020.

[14] Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model?, 2020.

[15] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering, 2021.

[16] Peng Xu, Davis Liang, Zhiheng Huang, and Bing Xiang. Attention-guided generative models for extractive question answering, 2021.

[17] Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. *CoRR*, abs/2012.09740, 2020.

[18] Nhat-Minh Pham, Ha-Thanh Nguyen, and Trong-Hop Do. Multi-stage information retrieval for vietnamese legal texts, 2022.

[19] Dat Quoc Nguyen and Anh Tuan Nguyen. Phobert: Pre-trained language models for vietnamese. *arXiv preprint arXiv:2003.00744*, 2020.

[20] Hieu Nguyen Van, Dat Nguyen, Phuong Minh Nguyen, and Minh Le Nguyen. Miko team: Deep learning approach for legal question answering in alqac 2022. In *2022 14th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1–5. IEEE, 2022.

[21] Luyu Gao and Jamie Callan. Condenser: a pre-training architecture for dense retrieval. *arXiv preprint arXiv:2104.08253*, 2021.

[22] Luyu Gao and Jamie Callan. Unsupervised corpus aware language model pre-training for dense passage retrieval. *arXiv preprint arXiv:2108.05540*, 2021.

[23] Yizhi Li, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. More robust dense retrieval with contrastive dual learning. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 287–296, 2021.

[24] Long Phan, Hieu Tran, Hieu Nguyen, and Trieu H. Trinh. Vit5: Pretrained text-to-text transformer for vietnamese language generation, 2022.

[25] Kiet Van Nguyen, Duc-Vu Nguyen, Anh Gia-Tuan Nguyen, and Ngan Luu-Thuy Nguyen. A vietnamese dataset for evaluating machine reading comprehension, 2020.