Legal Knowledge and Information Systems
G. Sileno et al. (Eds.)

doi:10.3233/FAIA230976

Giving Examples Instead of Answering Questions: Introducing Legal Concept-Example Systems

Tomer LIBAL [a,1], Aleksander SMYWIŃSKI-POHL [b,2]
[a] *Department of Computer Science, Luxembourg University, Luxembourg*
[b] *Computer Science Institute, AGH University of Krakow, Poland*

Abstract. Question-Answering Systems (QASs) have seen a big development in recent years and various attempts have been made to extend them to the legal domain. Nevertheless, the needs and methodology of legal research relies often less on getting answers and more on finding positive and negative examples for certain legal concepts. In this paper, we introduce a sub-category within QASs that focuses on such legal tasks and design a methodology for the automated production of such systems.

Keywords. Question-Answering Systems, Legal Research, Automatic Questions Generation, GDPR

1. Introduction

Question Answering Systems (QASs) are prominent in making knowledge accessible. Systems such as, for example, in health care (for a survey, see [1]), have seen rising popularity. Similar attempts have also been made in the legal domain. The COLIEE competition [3] had until recently the Legal Question-Answering challenge, which in the last competitions has been subsumed by Task 3 (Statute Law Retrieval Task) and Task 4 (Legal Textual Entailment Data Corpus).

Martinez-Gil [2] gives a comprehensive survey of the different approaches and tools and concludes that no solution has managed to provide high accuracy, interpretability, and performance. The reason being that accuracy and inerpretability are mainly obtained by investing considerable human effort, for example those based on ontologies (e.g. [3]) and linked-data (e.g. [4]).

An important challenge for legal professionals is legal research [5] due to the ever changing statute and case law. As a consequence of their volume and dynamics, a legal

[1] Supported by the Luxembourg National Research Fund under grant C22/IS/17228828/ExAlLe
[2] Supported by the Polish National Centre for Research and Development – Pollux Program under Grant WM/POLLUX11/5/2023 titled „Examples based AI Legal Guidance". We gratefully acknowledge Polish high-performance computing infrastructure PLGrid (HPC Centers: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2023/016304.
[3] `https://sites.ualberta.ca/~rabelo/COLIEE2023`

QAS must meet the performance property. Legal QASs in general should also meet the accuracy and interpretability properties.

The problem addressed in this paper is how the three properties can be obtained within one system, for the purpose of enhancing the legal research process. In the next section, we first introduce a new approach to QASs, called Concept-Example Systems (CESs). We then follow by briefly surveying the current state-of-the-art in legal QASs with regards to this analysis and introduce an automated methodology for their generation. We finish with a conclusion and future work section.

2. Legal Concept-Example Systems

2.1. Motivation and Definition

In order to motivate the importance of Concept-Example Systems (CESs), we refer the reader to the description by Sanderson and Kelly in their "A practical Guide to Legal Research" [5] and especially of their running example

According to them, an important step of legal research is the search for case law by the use of keywords. In this paper, we propose performing legal research by using CESs. The legal expert, instead of using keywords, is using legal concepts and context. The CES pre-index relevant sentences from case decisions and demonstrates to the user positive and negative examples for the concept in the specific context.

More formally, a Concept-Example System (CES) differs from a QAS in two aspects:

- The starting point for generating a question-answer pair is a legal concept and not a question.
- We do not look for specific answers but for as many extracts from court judgements which resolve the concept, under a specific context, either positively or negatively.

The reason for using a concept as a starting point is that concepts are the basic elements of legal research. Nevertheless, for interpretability, it is important that the user can easily assess the relevancy of the concept and context and we have therefore opted for translating the concept, legislation and context into a question.

Once questions are being identified, our goal is to automatically extract from judgement as many paragraphs as possible which can serve as either a positive or a negative example for the concept and under the context defined in the question.

2.2. Related work

Following the analysis given in [2] and the recent developments in neural QAS, it is safe to say that Legal QAS fall into one of three categories:

1. knowledge-based systems (using ontologies and linked data), e.g. [3,4],
2. retrieve, re-rank and interpret systems, e.g. [6],
3. large langue model based systems, e.g. [7].

An example of the first approach is given in [3]. The authors manually build an ontology for criminal law and supplement it with rules. Answering questions is possible thanks to an inference engine, which draws conclusions based on the interpretations of the legal concepts and application of the rules.

The example of the second approach is given in [6], where the authors describe a system helping engineers in China to find legislation regulating the design and development of buildings. The system is based on a combination of sparse retrieval and deep learning. When the user asks a question, the system searches for relevant law in a database containing the relevant legislation using TF-IDF term weighting scheme [8]. Then for top-n returned passages (the system does not have a re-ranking module) a Chinese BERT-like model is applied.

ChatGPT [7] is an example of a system of the third type. Even though these system achieve state-of-the-art results in legal QA[4], they have one very important limitation – they are not explainable and they have a tendency to hallucinate, i.e. they can make up facts. In the USA, two lawyers were recently fined for citing non-existent cases reported by ChatGPT [10].

We claim that the system presented in the next two sections achieves a similar accuracy level to the ones mentioned above while being more explainable and scalable.

2.3. Methodology

Our methodology for finding the relevant examples can be divided into the following steps:

1. Given a legal concept, generate a number of questions that ask if the concept was applicable in a given context. The precise number and form of the questions is variable.
2. Once a question is generated, we use a binary classification model to find sentences in court cases which answer the question. We use the predicted probability not only to sort these sentences, but also to filter sentences below a defined threshold. As a result this step might generate an empty set, meaning that in our corpus there are no relevant sentences. This prevents the system from presenting passages that are irrelevant, but also limits the possibility of false classification of the sentences.
3. In the last step, we extend the sentence to include the preceding and the following sentences forming a three-sentence passage and ask an instruction-following model to judge if the example gives a positive or a negative answer to the generated question. We use that prediction to classify the passage as a positive or a negative example.

2.3.1. Question generation

In order to generate questions, we first have to choose relevant concepts. This raises an interesting connection with hybrid AI solutions, as the concepts could be extracted from knowledge representation databases such as legal ontologies. For the purpose of

[4][9] claims that GPT 4.0 scores among 10% of the top test takers for the Uniform Bar Exam, but the paper is a pre-print and „the authors" do not provide the most important technical details of the system

this paper, the authors have read the relevant legal articles and have chosen 11 concepts according to their understanding.

In order to generate the questions from the concepts, we have approached ChatGPT[5] with the following prompt "For each of the following legal concepts from the GDPR, please give 4 specific and 1 general yes/no questions, which might be discussed by a court". The specific number and type of questions to generate is variable and depend on the legislation.

Since we focus on the GDPR, we assume that ChatGPT has a good knowledge of the relevant legislation. As the results show (cf. Table 1), this was a sensible assumption. However, for a solution that is independent from the knowledge of the language model (i.e. taking into account some amendments that were applied recently), we could provide the relevant provision as a part of the prompt or via further fine-tuning.

Legal concept	**Generated question**
Effective legal remedies for data subjects	Are there established procedures within the organization for data subjects to seek legal remedies?
Legally binding and enforceable instrument	Does the instrument in question meet the criteria for being legally binding under relevant laws?
Standard data protection clauses	Has the organization implemented Standard Contractual Clauses (SCCs) for data transfers?
Standard data protection clauses	Are there mechanisms in place to review and update the standard data protection clauses in response to legal or operational changes?

Table 1. Example questions generated by ChatGPT for the legal concepts taken from GDPR.

We note that the above process meets the interpretability and performance properties. The first because the outcome can be easily verified by humans and the third because the process is fully automatic and unsupervised.

For this step, as well as for the following ones, our initial experiments have shown that the method is not less accurate than current existing systems.

2.3.2. Sentence retrieval

The main element of the procedure is the retrieval of sentences within judgments relevant to the generated questions. In this initial work, we have implemented only a cross-encoder, which is typically used for re-scoring the results on a subset of documents. The cross-encoder is run for each combination of generated question and passage from the decision corpus.

Our goal for the cross-encoder is to score the likelihood that a certain passage contains information relating to the question. For this purpose, we are using a binary classification model by fine-tuning AlBERT [11] model (from the BERT family [12] of encoder-only transformer models) with the SQuAD 2.0 dataset [13].

We had to adapt the dataset for the task at hand, as in SQuAD most of the questions are answerable, while in our task, the vast majority of answers are irrelevant. To resolve this problem we have pre-processed SQuAD 2.0 by splitting each answer into individual sentences. This is followed by mapping the sentence answering the question as a positive example, while all the other sentences **from the same passage** are treated as the

[5] June 22 Version using GPT-4

negative examples for the same question. We call this approach an indirect contrastive learning, since the negative examples will include information which is highly relevant for the question, but they do not include the actual answer. As a result we obtain a dataset that includes a high number of negative examples, making it more representative of our problem.

Since the model outputs a probability score for the question – sentence pair, it is possible to sort the sentences according to the decreasing probability of the sentence containing the answer to the question. But since this is a binary classification model, by default pairs with probability below 0.5 will be judged as not containing the answer. This threshold can be increased to obtain a higher accuracy of the results.

Once again, the model satisfies two out of the three properties. The model is interpretable, since the answers are identical and contain references to the original phrases in judgements. It is also efficient as it is fully automatic and relatively simple (matching all questions with all sentences). The performance might be further improved by using a dense retriever [14] and approximate maximum product search with the help of e.g. Faiss [15].

2.3.3. Passage classification

For the last stage of the processing, we extend the sentences into contexts and classify them as positive or negative examples. Extending the questions is done by taking one preceding and one following sentence. It should be noted that finding more sophisticated approaches is left as a future work. To classify the extended context into one of the two classes we use an instruction following model, in that case Flan-T5 [7]. This is a T5-based model [16], fine-tuned to follow instructions on a large group of various English datasets. The prompt sent to the model is given in Figure 1.

```
Given the context: '<context>' answer the following question by
              reasoning step-by-step: <question>
```

Figure 1. The prompt used to ask Flan-T5 for classification of the examples.

The result of the prompt is an explanation of why the model believes that the answer is either positive or negative. It allows to group the examples into positive and negative ones by checking if the answer contains 'yes'. Once again, this makes the model interpretable on the human level, since the user is able to assess if the justification is sensible and eliminate inaccurate results. According to [17] Flan-T5 is a state-of-the-art instruction-following model among the models of similar size. Still, the accuracy of the model could be improved if a dedicated dataset, with binary QA pairs would be created. Creation of such dataset is left for a future research.

3. Conclusion

In this paper we introduced a sub-system of QAS and discussed its merits and a possible methodology for its creation.

The work introduced in this paper is an initial work towards high-quality CESs creation. As such, there are numerous ways to optimize the process, which we consider as future work. Due to lack of space, we mention but a few of them next.

In the paper we have used a subjective and biased method in order to evaluate the relationship between the score provided by our model and the actual relevance of examples to the questions. We plan on conducting an evaluation with the help of law students.

Another form of improvement would be the fine-tuning of pre-trained models with legal datasets for the purpose of sentence extraction.

We also plan on generating better and more numerous questions for each concept. Either via improving the prompt, using LLMs fine-tuned specifically on legal data or by using an entirely different approach.

Similarly, the selection of the relevant context for each answer (currently fixed on the preceding and following sentences) can be much improved.

Lastly, we would like to apply the method on a much larger set of relevant court case judgements. For example by using the one provided by `case.law`.

References

[1] Budler LC, Gosak L, Stiglic G. Review of artificial intelligence-based question-answering systems in healthcare. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2023;13(2):e1487.

[2] Martinez-Gil J. A survey on legal question–answering systems. Computer Science Review. 2023;48:100552.

[3] Fawei B, Pan JZ, Kollingbaum M, Wyner AZ. A methodology for a criminal law and procedure ontology for legal question answering. In: Semantic Technology: 8th Joint International Conference, JIST 2018, Awaji, Japan, November 26–28, 2018, Proceedings 8. Springer; 2018. p. 198-214.

[4] Filtz E, Kirrane S, Polleres A. The linked legal data landscape: linking legal data across different countries. Artificial Intelligence and Law. 2021;29(4):485-539.

[5] Sanderson J, Stamboulakis D, Kelly K. A Practical Guide to Legal Research. Lawbook Co.; 2021.

[6] Zhong B, He W, Huang Z, Love PE, Tang J, Luo H. A building regulation question answering system: A deep learning methodology. Advanced Engineering Informatics. 2020;46:101195.

[7] Chung HW, Hou L, Longpre S, Zoph B, Tay Y, Fedus W, et al. Scaling instruction-finetuned language models. arXiv preprint arXiv:221011416. 2022.

[8] Manning CD, Manning CD, Schütze H. Foundations of statistical natural language processing. MIT press; 1999.

[9] OpenAI. GPT-4 Technical Report; 2023.

[10] AP. Lawyers fined for filing bogus case law created by ChatGPT. CBS News. Available from: `https://www.cbsnews.com/news/chatgpt-judge-fines-lawyers-who-used-ai/`.

[11] Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations; 2019.

[12] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:181004805. 2018.

[13] Rajpurkar P, Jia R, Liang P. Know What You Don't Know: Unanswerable Questions for SQuAD. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2018. Available from: `http://dx.doi.org/10.18653/v1/P18-2124`.

[14] Karpukhin V, Oğuz B, Min S, Lewis P, Wu L, Edunov S, et al. Dense passage retrieval for open-domain question answering. arXiv preprint arXiv:200404906. 2020.

[15] Johnson J, Douze M, Jégou H. Billion-scale similarity search with GPUs. IEEE Transactions on Big Data. 2019;7(3):535-47.

[16] Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research. 2020;21(1):5485-551.

[17] Chia YK, Hong P, Bing L, Poria S. INSTRUCTEVAL: Towards Holistic Evaluation of Instruction-Tuned Large Language Models. arXiv preprint arXiv:230604757. 2023.