Using Large Language Models to Support Thematic Analysis in Empirical Legal Studies

Jakub DRÁPAL [a,b,1], Hannes WESTERMANN [c], and Jaromir SAVELKA [d]
[a] *Institute of State and Law of the Czech Academy of Sciences, Czechia*
[b] *Institute of Criminal Law and Criminology, Leiden University, the Netherlands*
[c] *Cyberjustice Laboratory, Université de Montréal, Canada*
[d] *School of Computer Science, Carnegie Mellon University, USA*

Abstract. Thematic analysis and other variants of inductive coding are widely used qualitative analytic methods within empirical legal studies (ELS). We propose a novel framework facilitating effective collaboration of a legal expert with a large language model (LLM) for generating initial codes (phase 2 of thematic analysis), searching for themes (phase 3), and classifying the data in terms of the themes (to kick-start phase 4). We employed the framework for an analysis of a dataset ($n = 785$) of facts descriptions from criminal court opinions regarding thefts. The goal of the analysis was to discover classes of typical thefts. Our results show that the LLM, namely OpenAI's GPT-4, generated reasonable initial codes, and it was capable of improving the quality of the codes based on expert feedback. They also suggest that the model performed well in zero-shot classification of facts descriptions in terms of the themes. Finally, the themes autonomously discovered by the LLM appear to map fairly well to the themes arrived at by legal experts. These findings can be leveraged by legal researchers to guide their decisions in integrating LLMs into their thematic analyses, as well as other inductive coding projects.

Keywords. Thematic analysis, empirical legal studies, criminal law, large language models, generative pre-trained transformers, GPT-4

1. Introduction

Empirical legal studies (ELS) is an approach to the study of law through empirical methods typical of economics, psychology, and sociology. Since law is a heavily text-based discipline ELS frequently focuses on text analytic methods, including deductive and inductive coding. Deductive coding focuses on applying a fixed set of codes to a dataset, whereas inductive coding leads to a simultaneous discovery of the codes from the data and their application. While investigations into various methods to support deductive coding have attracted much recent attention in AI & Law [1,2,3] very few studies focused on inductive coding [4]. One popular inductive coding method is "thematic analysis" [5].

[1] Corresponding Author: Jakub Drápal, drapalja@prf.cuni.cz. Work supported by Czech Grant Agency project "Sentencing disparities in the post-communist continental legal systems" n. 19-15077S.

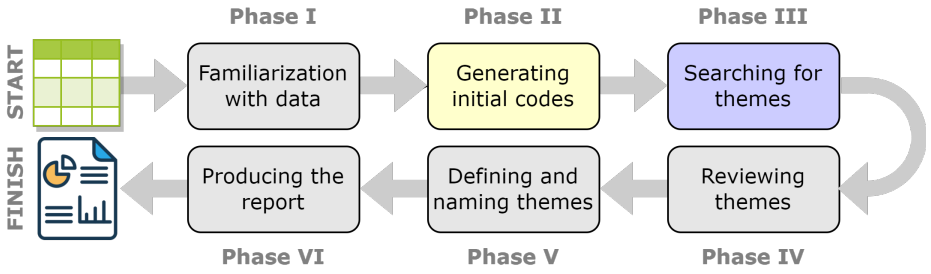

Figure 1. The six phases of thematic analysis, starting from the raw data and finishing with the scholarly report. This paper focuses on supporting phase II (yellow) and phase III (blue) of the analysis with LLMs.

While rarely explicitly acknowledged thematic analysis is widely used in ELS. Figure 1 shows the six phases of the analysis, starting from the raw data and finishing with the scholarly report on the studied phenomena. We propose a novel LLM-powered framework to support a subject matter expert in performing phases 2 and 3 of the analysis.

We employed the proposed framework in a thematic analysis of criminal courts' opinions, focused on the criminal offense of theft in Czechia. Criminal offense categories (e.g., theft, murder) are usually defined in statutory law while the individual criminal acts are described by courts when they apply the law to factual circumstances of the cases. An important question in criminal law and criminology is what behaviors are *actually* criminalized and whether it is done appropriately. Neither the statutory definitions of the offenses (they are too general and not "sociologically relevant" [6]) nor the descriptions of the factual circumstances from cases (too specific) can answer the question. To get insight into what behavior is criminalized and how effectively, it is necessary to identify shared features of criminal acts, generalize them into "typical crimes" [7,8] and arriving at behavioral-based categories [9,10]. This is akin to performing thematic analysis. While important, such analysis is an expensive and time-consuming endeavour. Hence, a (semi-)automated approach would be useful.

To assess the capability of a state-of-the-art LLM (GPT-4) to support selected stages of the thematic analysis, we investigated the following research questions:

(RQ1) How successfully can the LLM perform initial coding of the data?
(RQ2) To what degree can a subject matter expert improve the quality of the initial codes via natural language feedback?
(RQ3) How successfully can the LLM predict themes for the analyzed data points?
(RQ4) How successfully can the LLM autonomously discover themes and associate them with the analyzed data?

2. Related Work

There have been several studies exploring the use of *LLMs in thematic analysis*. De Paoli evaluated to what extent GPT-3.5 can carry out a full-blown thematic analysis of semi-structured interviews, finding that the LLM was indeed able to perform some of the steps while also cautioning about the methodological implications of using the approach [11]. Gao et al. developed a collaborative coding platform powered by GPT-3.5 that provides

code and code group suggestions to support the process of defining a codebook [12]. Gamieldien et al. used GPT-3.5 to generate codes for automatically clustered comments, finding that the produced codes were granular but not coherent, as similar clusters were assigned very different names [13].

There is a long tradition of studies identifying *patterns in criminal justice data*, including those focused on offense categories. The studies typically employed content analysis together with approaches such as factor, latent profile or cluster analyses. Santtila et al. identified 14 types of burglaries from the descriptions of crime scene behavior [14]. Higgs et al. collated descriptions of 700 sexual murderers to describe the overall patterns and motives underlying the offense [15]. Canter et al. performed a thematic classification of stranger rapes [16]. Gřivna and Drápal focused on criminal offenses involving computer data and systems (cybercrime) in the Czech Republic, identifying the most frequent types of such criminal behavior [17].

There is a similar tradition focused on *discovering stereotypical patterns in court opinions* in AI & Law. Ashley identified factors from trade secret law through reading cases and doctrine [18]. Similar analysis was performed by Gray et al. to discover typical factors of suspicion considered in auto stop cases [19]. Westermann et al. used the grounded theory approach (a close kin of thematic analysis) to discover relevant factors considered by judges in certain types of landlord-tenant disputes [20]. Notably, Salaun et al. used a topic modelling approach in the same domain to identify the factors automatically, finding that 33% of the discovered topics were relevant [4]. To our best knowledge, that study is the state-of-the-art attempt on inductive coding of legal texts. Our work differs by subscribing to a well-established framework (i.e., thematic analysis) and the use of GPT-4 that enables a subject matter expert to drive and influence the analysis through specified research questions and instructions.

There were multiple proposals of *frameworks focused on supporting legal experts* in deductive coding of legal texts. Branting et al. proposed manual annotation of factors in a small number of cases, which were then projected across a much larger dataset [2]. Westermann et al. described an approach where legal experts formulated sets of complex search terms (as classifiers) based on constantly updated dataset statistics [1]. Westermann et al. also proposed a framework utilizing sentence embeddings and similarity retrieval to support annotators in annotating legal documents [3]. Recently, Savelka et al. explored performing annotations with LLMs (GPT-3.5 and GPT-4) in zero-shot settings by providing the model with excerpts from annotation guidelines [21,22,23]. In this paper, we perform deductive coding when predicting the themes for the case facts descriptions (RQ3) as one of the steps in predominantly inductive coding-based analysis.

3. Dataset

In our experiments, we used a dataset of 785 facts descriptions from cases of Czech courts decided in 2017. From the Prosecution Service, we received 834 cases that found an adult defendant guilty of theft. In Czechia, theft also includes burglary and pickpocketing.[2] We slightly over-represented the most serious offenses to ensure a sufficient number of cases in the dataset. We removed 49 cases from the dataset because they were used in a pilot study or due to them containing errors. We extracted text describing the

[2]ICCS codes 0501 and 0502 except for 0502212 [9].

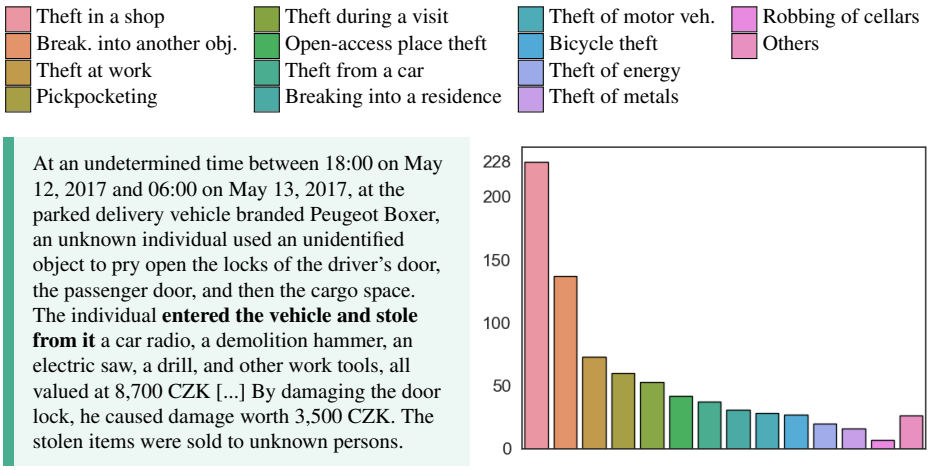

Figure 2. The categories from the theft types dataset (shown at the top) and their distribution (right). An example of case facts description from the *theft from a car* category is shown on the left.

facts. Each extracted text was anonymized and shortened or partially re-written if necessary. The resulting text snippets range from 73 to 29,695 characters in length (1Q 447, median 782, 3Q 1,462 characters). Figure 2 shows an example (automated translation).

A group of three law students under the supervision of one of the authors of this paper manually conducted an unstructured variant of thematic analysis.[3] The group arrived at 14 high-level themes focused on modus operandi and target of committed thefts (Figure 2). For each facts description a single theme was independently chosen by two students according to specified rules. The disagreements were resolved by one of the students following careful re-reading of the case. The distribution of the themes over the 785 facts descriptions included in the dataset is presented in Figure 2. The *theft in a shop* (29.0%) and *breaking into another object* (17.5%) are the most prevalent themes.

4. Proposed Framework

Model The framework relies on OpenAI's GPT-4 model's capabilities to perform complex NLP tasks in zero-shot settings [24]. We set the `temperature` of the model to 0.0, which corresponds to no randomness. Higher `temperature` leads to more creative, but potentially less factual, output. We set `max_tokens` to various values depending on the expected size of the output (a token roughly corresponds to a word). GPT-4 has an overall token length limit of 8,192 tokens, comprising both the prompt and the completion. We set `top_p` to 1, as is recommended when `temperature` is set to 0.0. We set `frequency_penalty` and `presence_penalty` to 0, which ensures no penalty is applied to repetitions and to tokens appearing multiple times in the output.

Resources We utilize the definition of thematic analysis and the individual phases from [5]. For example, the 15-point checklist of criteria for good thematic analysis have been adopted verbatim as well as selected excerpts defining the analysis, its flow and

[3] We did not rigorously adhere to the process described in [5].

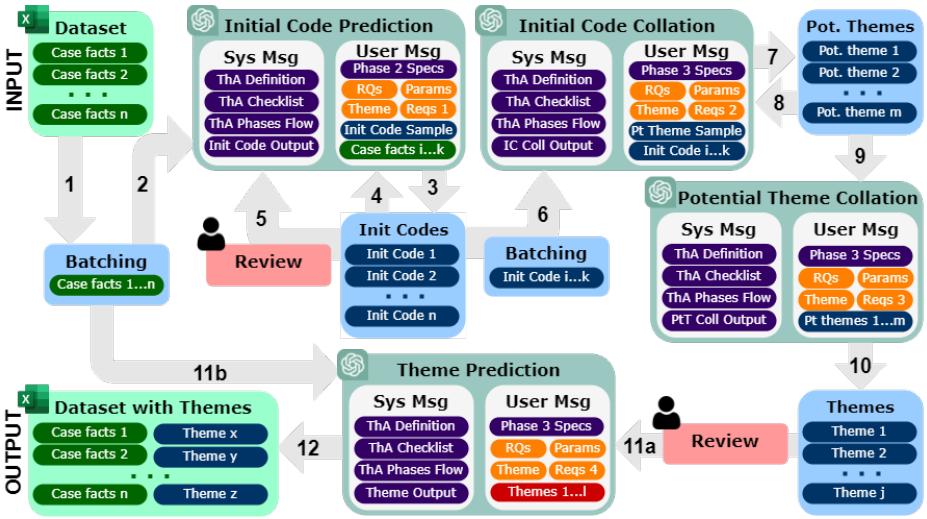

Figure 3. The diagram describes the proposed framework supporting phases 2 and 3 of thematic analysis. The data points are batched (1) to fit the LLM's prompt. Initial codes for each batch are predicted (2) in an iterative fashion utilizing the interim results (3 and 4). Expert feedback on the initial codes may be provided to trigger the re-generation of improved codes (5). The initial codes (batches) are collated into potential themes (6) in an iterative fashion utilizing the interim results (7 and 8). The potential themes are further collated into a compact list of high-level themes (9). The expert-reviewed themes (10) are predicted for the input data points (11a and 11b). The predictions of discovered themes for each data point are provided as the output (12).

the phases. Together with specifications of the expected outputs from various stages of the processing pipeline, these can be considered as *general* resources, i.e., invariant to the performed thematic analysis. In addition, the framework requires *context-specific* resources, i.e., different for each analysis. These include research questions, the parameters specifying the type of analysis (e.g., semantic/latent patterns, focus on a specific topic), the specification of what counts as a theme, and various sets of custom requirements.

Processing Flow The proposed framework is depicted in Figure 3. The analyzed dataset is automatically segmented into batches where as many data points as can be fitted into the model's prompt are batched together, using the `tiktoken` Python library.[4] The data points are processed from the shortest to the longest. If a single data point exceeds the size of the prompt it is truncated to fit the limit by taking its starting and ending sequence (both half the limit) and placing the "`[...]`" token in between them. Each batch is inserted into the user message, alongside the research questions, other context-specific information about the analysis as well as any custom requirements. We also include a random sample of the initial codes predicted in the batches preceding the current one. The system message consists of the general resources. Using the `openai` Python library,[5] the system and the user messages are then submitted to the LLM that generates a JSON file with the predicted initial codes. This is repeated until the whole dataset is labeled. The subject matter expert may review the predicted initial codes. Then, they may provide further custom instructions to the system as to what aspects of the data to focus on and

[4]Github: Tiktoken. Available at: `https://github.com/openai/tiktoken` [Accessed: 2023-04-30]

[5]GitHub: OpenAI Python Library. Available at: `https://github.com/openai/openai-python` [Accessed 2023-08-16]

Initial Codes Evaluation Scheme

1. $\neg$*How*: If the code does not address (even implicitly) how the theft happened; $\rightarrow$
2. $\neg$*What*: If the code does not address (even implicitly) what was stolen; $\rightarrow$
3. *Ok*: The code addresses how the theft happened and what was stolen. $\square$

Figure 4. The coding scheme employed in evaluating the quality of the initial codes generated by the system autonomously (RQ1) and after the expert feedback was provided (RQ2).

which to disregard. These instructions are appended to the custom requirements. This process can be repeated until the predicted initial codes match the expectations.

The predicted initial codes are collated into potential themes. This stage of the processing is similar to the preceding one with the notable difference that the system operates on the batches of initial codes instead of the raw data points. The most common 20 potential themes predicted in the batches preceding the current one are included in the user message. As a result, each data point gets associated with a candidate theme. The candidate themes, which could be many, are then further collated into a compact set of high-level themes. The whole set of candidate themes is provided in the user message and submitted to the LLM. While this may depend on the specific analysis there is most likely no need to supply these in batches as all of them are likely to fit in the prompt. The output of this stage are the high-level themes (with candidate themes as sub-themes).

The final stage of the pipeline is focused on labeling the data points with the discovered themes. Note that this step differs from the prediction of the initial codes or potential themes because, here, the LLM is used to predict the labels from the provided list of themes (i.e., to perform deductive coding). The result of the whole process is the original data points being associated with (semi-)automatically discovered themes. This artifact can be utilized by the subject matter expert as a starting point for the subsequent phase of the thematic analysis (reviewing themes).

5. Experimental Design

(RQ1) Autonomous Generation of Initial Codes The quality of the automatically generated initial codes was manually assessed by one of the authors (a subject matter expert). The analysis suggested that, while largely sensible, the codes overly focused on *what* was stolen, ignoring other aspects of the analysis (e.g., *how* the offense was committed). To gauge the extent of the issue, we evaluated whether codes address (even implicitly) how the theft happened and what was stolen (evaluation scheme shown if Figure 4). Each initial code was first analyzed with respect to $\neg$*How*, and if the issue was confirmed the analysis stopped, i.e., the code was not further considered for $\neg$*What*.

(RQ2) Generating Initial Codes with Expert Feedback Following the analysis of the autonomously generated initial codes, we formulated compact instructions for the system to mitigate the most commonly appearing issues. The feedback consisted of two parts: (i) *positive* (what to focus on) – target, modus operandi, seriousness, and intent; and (ii) *negative* (what to avoid) – multiplicity of the offense, degree of completion, co-responsibility, value of stolen goods. We also provided three examples of desirable initial codes such as, e.g. "*vehicle theft with forceful entry and disassembly of vehicles*". The

	¬*How*	¬*What*	*Ok*	¬*Ok*
Before expert feedback	104 (13.2%)	111 (14.1%)	570 (72.6%)	215 (27.4%)
After expert feedback	16 (2.0%)	72 (9.2%)	697 (88.8%)	88 (11.2%)

Table 1. Performance on the prediction of initial codes measured in terms of the evaluation scheme shown in Figure 4. The inclusion of expert feedback into the prompt results in notable improvements across the board.

instructions were included in the prompt (user message) as custom requirements. With the thus updated prompt, we repeated the generation of the initial codes. To assess the effects of the provided feedback, the newly generated codes were then manually coded using the same scheme as in the evaluation of RQ1, i.e., $\neg How \rightarrow \neg What \rightarrow Ok$ □.

(RQ3) Predicting Themes To evaluate the zero-shot performance of the LLM in predicting themes for the analyzed data points, we employed the theme prediction component of the pipeline to label each data point with one of the themes arrived at by human experts. Note that a factual description may contain multiple themes, e.g., *bicycle theft* and *theft from an open-access place*, whereas the experts were instructed to assign the most specific and salient one. To account for this phenomenon, we instructed the system to also assign each data point with three of the themes. Then, we measured the performance of the system on this task in terms of recall at 1 (R@1) and recall at 3 (R@3).

(RQ4) Automatic Discovery and Prediction of Themes To investigate the performance of the proposed pipeline on the end-to-end task of autonomously discovering themes from the provided data, and assigning each analyzed data point with one of the identified themes, we employed the successive components of the pipeline to: (i) generate initial codes (with expert feedback); (ii) collate the initial codes into potential themes; (iii) group the potential themes into a compact list of higher-level themes; and (iv) assign each data point with one of the high-level themes. We then compared the automatically assigned themes to the manual themes discovered and assigned by subject matter experts.

6. Results and Discussion

Table 1 reports the results of the experiments focused on the prediction of initial codes. After the first round, 72.6% of the 785 predicted codes were deemed reasonable, i.e., they described *how* and *what*. 13.2% of the codes appeared to lack the focus on *how*, and at least 14.1% did not seem to describe *what* was stolen. After the expert feedback was provided (Section 5), 88.8% of the codes were perceived as reasonable (+16.2% improvement). Table 2 compares example initial codes before and after the feedback, highlighting the improvements in coding the information of interest.[6] The results strongly suggest that the LLM can perform the initial coding of the data with reasonable quality (RQ1), and further improve the codes upon receiving feedback from a subject matter expert (RQ2). Hence, it appears that the proposed framework could become a valuable tool for supporting phase 2 of thematic analysis in ELS.

The performance on the task of predicting themes (specified upfront) for the individual facts descriptions is described in Table 3. The prediction was performed using the

[6] We admit a possible limitation of this experiment in that the author knew in which round the initial code was produced.

Before expert feedback	After expert feedback
Private theft of cash from a residential space	Theft of large quantity of cash from relative's home
Forced entry and theft involving an automobile	Burglary and theft of work tools from vehicle
Shoplifting - personal care items	Shoplifting of shaving equipment from drugstore

Table 2. Examples of autonomously generated initial codes (left) and the initial codes generated after the subject matter expert feedback was provided (right). The colors highlight improvements (red → green).

Manual Theme	R@1	R@3	Manual Theme	R@1	R@3
Theft in a shop	.95	.96	Breaking into a residence	.52	.71
Theft during a visit	.75	.91	Bicycle theft	.74	.96
Theft at work	.71	.86	Theft from a car	.70	.84
Breaking into another object	.35	.67	Robbing of cellars	.14	.57
Pickpocketing	.68	.87	Theft of motor vehicles	.50	.75
Theft from an open-access place	.21	.29	Theft of energy	1.0	1.0
Theft of metals	.69	.88	Others	.23	.73
Overall			**.66**	**.82**	

Table 3. Performance on the zero-shot prediction of themes discovered by subject matter experts from facts descriptions. R@1 is recall at 1 and R@3 is recall at 3.

list of 14 manually discovered themes (see Section 3). The overall R@1 of .66 and R@3 of .82 appear to suggest that the proposed approach is promising but clear limitations exist. This is largely consistent with prior related studies [21,22]. For some of the themes, e.g., *theft in a shop* or *theft of energy*, the automatic prediction worked remarkably well. However, there were also themes, e.g., *theft from an open-access place* or *robbing of cellars* where the performance was rather low. The promising results (RQ3) warrant investigations into the effects of providing expert feedback at this stage. This could either be done via providing additional custom instructions in the prompt and/or having the experts label a limited number of data points to be used in fine-tuning of the model.

The evaluation of the end-to-end performance in discovering and predicting themes (RQ4) is shown in Figure 5. The number of themes discovered by the LLM (8) was less than the number of themes arrived at by the legal experts (14). For example, the LLM-discovered *theft in commercial settings* maps to data points from manually discovered *theft at work* and *breaking into another object*. Some of the LLM-discovered themes appear to correspond to some of the manual ones (e.g., *theft in a shop* → *retail theft*). To further understand the mapping, we inspected the automatically generated potential themes from which the final 8 themes were automatically assembled. It appears that all of the manually identified themes could be mapped to one or more of the potential themes within a higher-level theme (e.g., *robbing of cellars* → *theft in residential areas*::*burglary of storage area*). Further, the LLM identified behaviors that were missed, perhaps in error, by human experts (e.g., *theft from gym*).

Some of the potential themes were so similar that they should have been likely collapsed together. Other potential themes were overly specific (e.g., *workplace theft* followed by various instances of what was stolen). Interestingly, the multiplicity of offending and/or stage of completion were present in some of the potential themes, despite specific instructions during the initial codes prediction not to focus on these aspects. Hence,

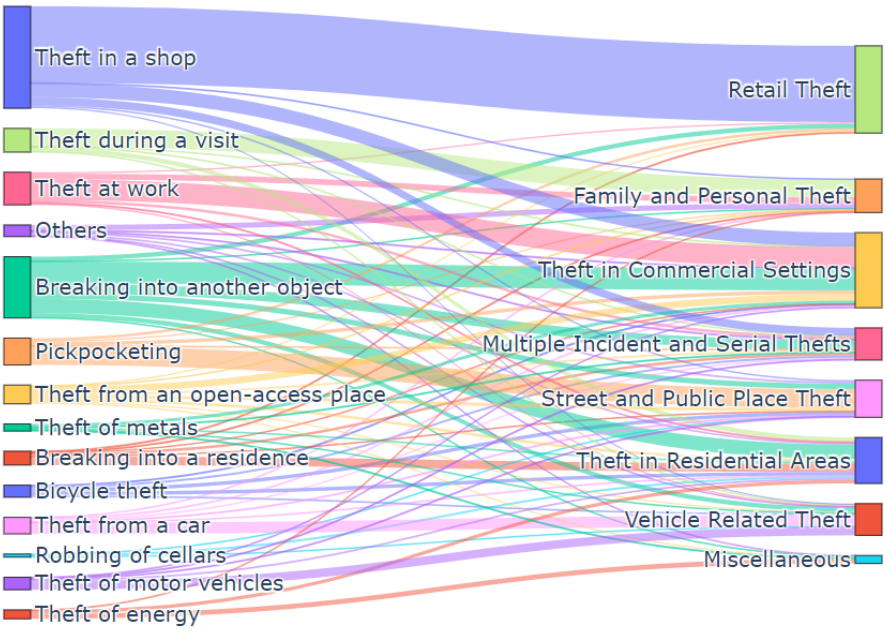

Figure 5. The graph shows mapping between themes discovered by subject matter expert (left) and the themes discovered by the proposed framework (right).

an additional expert intervention in predicting potential themes might be warranted. The analysis strongly suggests that the LLM performs well in the end-to-end task of discovering and predicting themes from the raw data (RQ4). However, subject matter expert interventions might be desirable at various stages of the processing to improve the quality of the resulting themes and their alignment with the research questions. This echos with the cautioning sentiments expressed by De Paoli [11] and Jiang et al. [25] who reported that researchers performing qualitative analysis require full agency over the process. Moreover, the black-box nature of the proprietary LLMs is especially problematic from this point of view.

7. Conclusions and Future Work

We proposed a novel LLM-powered framework supporting thematic analysis, and evaluated its performance on an analysis of criminal courts' opinions focused on the categories of thefts in Czechia. We found that the initial coding of data was performed with reasonable quality (RQ1), and further improved when expert feedback was provided (RQ2). The performance on zero-shot classification of the data (facts descriptions) in terms of themes (categories of theft) was promising (RQ3) but could likely benefit from expert feedback (future work). The evaluation of the end-to-end performance of the pipeline on discovering and predicting themes suggested viability of the proposed framework (RQ4) while highlighting the importance of subject matter expert supervision. Besides incremental improvements, the future work should focus on extending the support beyond

phases 2 and 3 of thematic analysis, and validating the findings of this study in other domains beyond court opinions and/or criminal law.

References

[1] Westermann H, Savelka J, Walker VR, Ashley KD, Benyekhlef K. Computer-Assisted Creation of Boolean Search Rules for Text Classification in the Legal Domain. In: JURIX; 2019. p. 123-32.

[2] Branting LK, Pfeifer C, Brown B, Ferro L, Aberdeen J, Weiss B, et al. Scalable and explainable legal prediction. Artificial Intelligence and Law. 2021;29:213-38.

[3] Westermann H, Savelka J, Walker VR, Ashley KD, Benyekhlef K. Sentence embeddings and high-speed similarity search for fast computer assisted annotation of legal documents. In: JURIX. vol. 334. IOS Press; 2020. p. 164.

[4] Salaun O, Gotti F, Langlais P, Benyekhlef K. Why Do Tenants Sue Their Landlords? Answers from a Topic Model. In: JURIX. vol. 362. IOS Press; 2022. p. 113.

[5] Braun V, Clarke V. Using thematic analysis in psychology. Qualitative research in psychology. 2006;3(2):77-101.

[6] Kitsuse JI, Cicourel AV. A note on the uses of official statistics. Soc Probs. 1963;11:131.

[7] Hornle T. Moderate and non-arbitrary sentencing without guidelines: the German experience. Law & Contemp Probs. 2013;76:189.

[8] Lappi-Seppälä T, Tonry M, Frase R. Sentencing and punishment in Finland: The decline of the repressive ideal. TONRY, Michael Why punish. 2001:239-54.

[9] UNODC, LXIV. UNODC, editor. International classification of crime for statistical purposes. Vienna: United Nations Office on Drugs and Crime; 2015.

[10] National Academies of Sciences, Engineering, and Medicine and others. Modernizing crime statistics: Report 1: Defining and classifying crime. National Academies Press; 2016.

[11] De Paoli S. Can Large Language Models emulate an inductive Thematic Analysis of semi-structured interviews? arXiv:230513014. 2023.

[12] Gao J, Guo Y, Lim G, Zhan T, Zhang Z, Li TJJ, et al. CollabCoder: A GPT-Powered Workflow for Collaborative Qualitative Analysis. arXiv preprint arXiv:230407366. 2023.

[13] Gamieldien Y, Case JM, Katz A. Advancing Qualitative Analysis: An Exploration of the Potential of Generative AI and NLP in Thematic Coding. Available at SSRN 4487768. 2023.

[14] Santtila P, Ritvanen A, Mokros A. Predicting burglar characteristics from crime scene behaviour. International Journal of Police Science & Management. 2004;6(3):136-54.

[15] Higgs T, Carter AJ, Tully RJ, Browne KD. Sexual murder typologies: A systematic review. Aggression and violent behavior. 2017;35:1-12.

[16] Canter DV, Bennell C, Alison LJ, Reddy S. Differentiating sex offences: A behaviorally based thematic classification of stranger rapes. Behavioral Sciences & the Law. 2003;21(2):157-74.

[17] Gřivna T, Drápal J. Attacks on the confidentiality, integrity and availability of data and computer systems in the criminal case law of the Czech Republic. Digital Investigation. 2019;28:1-13.

[18] Ashley KD. Reasoning with cases and hypotheticals in HYPO. International journal of man-machine studies. 1991;34(6):753-96.

[19] Gray M, Savelka J, Oliver W, Ashley K. Toward Automatically Identifying Legally Relevant Factors. In: Legal Knowledge and Information Systems. IOS Press; 2022. p. 53-62.

[20] Westermann H, et al. Using factors to predict and analyze landlord-tenant decisions to increase access to justice. In: Proceedings of ICAIL '19'; 2019. p. 133-42.

[21] Savelka J. Unlocking practical applications in legal domain: Evaluation of gpt for zero-shot semantic annotation of legal texts. arXiv preprint arXiv:230504417. 2023.

[22] Savelka J, Ashley KD, Gray MA, Westermann H, Xu H. Can GPT-4 Support Analysis of Textual Data in Tasks Requiring Highly Specialized Domain Expertise? arXiv preprint arXiv:230613906. 2023.

[23] Savelka J, Ashley KD. The Unreasonable Effectiveness of Large Language Models in Zero-shot Semantic Annotation of Legal Texts. Frontiers in Artificial Intelligence. 2023;6:1279794.

[24] OpenAI R. GPT-4 technical report. arXiv. 2023:2303-08774.

[25] Jiang JA, et al. Supporting serendipity: Opportunities and challenges for Human-AI Collaboration in qualitative analysis. Proceedings of the ACM on HCI. 2021;5(CSCW1):1-23.