Automated Question Type Coding of Forensic Interviews and Trial Testimony in Child Sexual Abuse Cases

Zsofia A. Szojka, Suvimal Yashraj and Thomas D. Lyon

University of Southern California, Gould School of Law

Author Note

Zsofia A. Szojka https://orcid.org/0000-0002-4126-1261

Suvimal Yashraj

Thomas D. Lyon https://orcid.org/0000-0001-8179-759X

Zsofia A. Szojka is now at the School of Law and Criminology, University of Greenwich

We have no known conflict of interest to disclose. This project was supported by Eunice Kennedy Shriver National Institute of Child Health and Human Development Grant HD101617. Special thanks to Mansi Gaur, Alex Kavaldjiev and present and past Lyon Lab staff and research assistants.

Correspondence regarding this manuscript should be sent to Zsofia Szojka, Old Royal Naval College, Park Row, London SE10 9LS, United Kingdom. Email: z.a.szojka@greenwich.ac.uk

Abstract

Objectives

Question type classification is widely used as a measure of interview quality. However, question type coding is a time-consuming process when performed by manual coders. Reliable automated question type coding approaches would facilitate the assessment of the quality of forensic interviews and court testimony involving victims of child abuse.

Method

We examined whether a large language model (RoBERTa) trained on questions (N = 351,920) asked in forensic interviews (n = 1,435) and trial testimony (n = 416) involving 3- to 17-year-old alleged victims of child sexual abuse could distinguish among 1) invitations, 2) wh- questions, 3) option-posing questions and 4) non-questions.

Results

The model achieved high reliability (95% agreement; K = .93). In order to determine if disagreements were due to machine or manual errors, we re-coded inconsistencies between the machine and manual codes. Manual coders erred more often than the machine, particularly by overlooking invitations and non-questions. Correcting errors in the manual codes further increased the model's reliability (98% agreement, K = .97).

Conclusion

Automated question type coding can provide a time-efficient and highly accurate alternative to manual coding. We have made the trained model publicly available for use by researchers and practitioners.

Public Significance Statement

Question type coding is widely used as a measure of interview quality in forensic interviewing research and practice. A large language model achieved high reliability with manual coders in distinguishing among invitations, wh- questions, option-posing questions and non-questions, providing researchers with a time-efficient alternative to manual coding and allowing interviewers to monitor the quality of their questioning. We have made the trained model publicly available to researchers and practitioners.

Question type is widely used as a measure of the quality of questioning in forensic interviews involving child victims of sexual and physical abuse. Across forensic interviewing protocols, there is a wide consensus about the recommendation to maximize the use of broad open-ended recall prompts (also called invitations), resort to other recall prompts (wh-questions or directives) when eliciting specific details, and minimize the use of option-posing (yes-no and forced choice) questions (American Professional Society on the Abuse of Children [APSAC], 2012; Cyr, 2022; Lamb et al., 2018; Poole & Dickinson, 2024; Powell & Brubacher, 2020). In the past 25 years, researchers have frequently relied on the relative proportion of question types to examine the effectiveness of interview protocols and interviewer training (e.g., Cyr et al., 2021; Johnson et al., 2015; Lamb, 2016, Warren et al., 1999).

Question type distinctions are important to researchers and practitioners in several contexts. Examining interviewer and child behavior in the lab, in forensic interviews, and in court testimony, researchers use question type coding to investigate the productivity and accuracy of questions (Brown et al., 2013; Garcia et al., 2022), the utility of questions in eliciting specific types of content (Szojka et al., 2023), and the efficacy of interview training (Cyr et al., 2012). Trainee forensic interviewers are taught to recognize the different question types so that they can monitor their performance (Powell et al., 2013). Not only does the proportion of open-ended questions provide a simple and clear measure of their adherence to best practice guidelines (Lamb et al., 2018), examining suboptimal questions also allows interviewers to identify topics that may trigger their use of option-posing prompts, such as clarifying sexual touch or the timing of the events (Guadagno et al., 2013), and practice re-phrasing them in an open-ended manner. Furthermore, the courts and expert witnesses often consider question-type in evaluating the admissibility of forensic interviews and the quality of questioning (Lyon & Dente, 2012; Vieth, 2021).

Currently, question type coding is a time-consuming and labor-intensive process performed by manual coders. Automated question type coding would allow researchers, interviewers, and other practitioners to review the quality of questioning in forensic interviews and trial testimony more accurately and efficiently. Moreover, the process could lead to greater coordination and consistency among research labs conducting research into forensic interviewing.

Question Type Confusion

Assessment of question types is difficult. In our experience, research assistants need extensive training in order to become reliable question type coders, and many experience coding drift, in which their performance deteriorates over time. This is analogous to the performance of trainee interviewers, who through intensive training can increase their use of invitations, but whose performance eventually returns to baseline without refresher training (Lamb et al., 2002).

A particularly difficult distinction for novice coders is between invitations and wh-questions. Invitations are broad open-ended questions (often using the word “happen”) that place no restrictions on the type of information to report, whereas what/where/who/why/how questions provide some specification of the detail requested. Powell and colleagues (2013) examined the types of errors commonly made by trainee coders, including post-graduate research assistants, online forensic interviewing trainees, and police officers. One of the most common errors they found was coding invitations as wh- questions (what the authors called “specific cued-recall” questions), suggesting that coders failed to recognize that questions with phrases such as “what happened” or “tell me more about [child generated-detail]” could be quite specific and yet constitute invitations. On the other hand, they also found that coders sometimes treated any “tell me” questions as invitations, failing to recognize that a question

like “tell me where it happened” is a “where” question and thus should be coded as a wh-question. The belief that “tell me” signals an invitation is echoed in practice, with trainees often taught to begin their questions with “tell me” in order to make them more open-ended (Oxburgh, et al., 2010). Henderson and colleagues (2020) coined the term “faux invitations” to refer to “tell me” questions that were not proper invitations, either because they contained a wh- question, or a yes-no question (e.g., “tell me if it was dark”). They found that one in four “tell me” questions in forensic interviews were faux invitations (see also Wolfman et al., 2016). Similarly, Yi and Lamb (2018) trained Korean police officers to use the NICHD protocol and found that the most common errors were misclassifying wh- questions as invitations (20%) and invitations as wh- questions (10%).

Prior Efforts to Automate Question Type Coding

Researchers have begun to identify the potential benefits of automated question type coding. Training manual coders and performing the coding itself takes a great deal of time, inspiring some attempts at automation (Garcia, 2022). Some researchers have developed automated question type coding as part of programs in which human interviewers were trained to develop interviewing skills by interacting with child avatars (Haginoya et al., 2023; Hassan et al., 2023; Røed et al., 2023). In this context, automated question type coding allows the avatar to respond appropriately, giving more productive responses when trainees ask invitations, and becoming reticent when trainees ask option-posing questions. Moreover, the avatar programs are able to provide explicit feedback to interviewers regarding their performance at the end of the interview, breaking down their questions into more and less preferred types. Approaches to automation vary, ranging from rule-based categorization (Garcia, 2022), to n-gram approaches analyzing contiguous word sequences (Haginoya et al., 2023), to training large language models that use machine learning to understand and

generate human language based on extensive text data (Hassan et al., 2023; Røed et al., 2023).

In an unpublished dissertation, Garcia (2022) aimed to categorize the question type of the transition prompts in 341 forensic interviews. The transition prompt is the question in which the interviewer moves from rapport building to substantive questioning, typically asking a question such as "tell me why you came to talk to me today." Garcia adopted a rule-based approach that distinguished between yes-no questions (classified as "closed") and other questions (classified as "open"). The program initially identified all questions with the words "can you" and "do you" as closed, and all other questions as open. In the first iteration of the program, there was 97% agreement between manual coders and machine coding (K = .93). Examining the disagreements, the author added four phrases signaling closed questions, and adjusted one of the phrases signaling open questions, ultimately achieving 100% accuracy in classifying the questions.

Although the program was ultimately very successful, the work required to automate the identification of yes-no questions in a relatively small sample of transition prompts highlights the difficulties of using rule-based approaches to automatic question type coding. Notably, 100% success was obtained by re-testing the model on the original sample, rather than a new test sample. Although more comprehensive than the first iteration of the model, the updated rules would still miss some types of yes-no questions, such as declarative questions, in which the interviewer makes a statement with a rising intonation ("You're here to talk about your dad?") (Klemfuss et al. 2014). The difficulties are compounded when the categorization system is expanded to several different question types, rather than exclusively yes-no questions, and to all utterances in an interview, rather than a single prompt.

Three studies have evaluated automated question type coding as part of their avatar child interviewee programs. Haginoya and colleagues (2023) trained a machine learning algorithm on questions (N = 8,754) asked in prior human-avatar interviews. Unlike the rule-based framework of Garcia's (2022) study, the authors used a machine learning approach. The model utilized a gradient boosting framework ("XGBoost") that iteratively refines its performance by identifying inaccuracies in initial attempts at classification and making corrections. The model classified questions into 11 types based on N-gram patterns, which entailed examining different lengths of consecutive word combinations (e.g., a 2-gram would break down "what happened next" into "what happened" and "happened next"). The authors then grouped the 11 types of questions into "recommended" and "not recommended" questions. Invitations and wh- questions ("directives") were classified as "recommended," and option-posing questions were classified as "not recommended." Furthermore, any type of question could be classified as "not recommended" based on other problems, including questions that were compound, contained difficult vocabulary, were grammatically ambiguous, requested temporal information, or encouraged fantasy. The authors obtained moderate agreement between manual coded and machine coded judgments regarding recommended and not-recommended questions (72%; K = .49).

A model's reliance on N-gram patterns restricts its ability to grasp context beyond immediate word combinations. Large language models have been trained to understand and analyze language in greater depth and with more subtlety. Hassan and colleagues (2023) trained the GPT-3 davinci model on 58 transcripts of trainee-avatar interviews (N = 2,745 questions). The transcripts had been manually coded for 14 different types of questions, which the authors then grouped into open-ended and closed questions, with invitations and wh- questions ("descriptive questions") among those coded as open-ended and option-posing questions among those coded as closed. They then tested the model on another group of 40

interviews (N = 2,415 questions), and obtained 87% agreement (K = .55, based on data reported in the paper). The high percentage rate of agreement coupled with only moderate reliability reflects the fact that 83% of the questions in the training data were open-ended, suggesting that the model could perform well by simply classifying all questions as open-ended. Sensitivity (the true positive rate) for closed-ended questions was only 65%, which means that the model mistakenly classified 35% of the closed-ended questions as open-ended.

For purposes of analyzing interview quality, it is ideal for models to be able to distinguish among invitations, wh- questions, and option-posing questions. Invitations should be distinguished from wh- questions because they are consistently more productive, at least among children five and older (Hershkowitz et al., 2012). Option-posing questions should be distinguished from other questions because they increase the likelihood of false positives (Peterson and Grant, 2001) and false negatives (Henderson et al., 2022) and, particularly among younger children, elicit unelaborated responses (Szojka & Lyon, 2024). In what appears to be the most successful model to date, Røed and colleagues (2023) trained the GPT-3 davinci model on data from 700 10-minute interviews conducted by trainees with adults posing as children (Powell et al., 2016). They then examined the performance of the model in coding 120 human interviews with a child avatar (N = 3,206 questions), distinguishing among invitations (which they called “open-ended questions”), wh- questions (“cued recall”) and option-posing questions (“closed questions”). Although the authors did not report sensitivities (true positive rates), specificities (true negative rates), or the relative frequencies of question types in the test sample, the model achieved strong agreement (87% agreement for invitations, 87% agreement for wh- questions, and 85% agreement for yes-no questions; K = .80).

Question Type Coding of Forensic Interviews and Trial Testimony

The machine learning approaches discussed above have exclusively relied on mock interviews for their training and test data (Haginoya et al., 2023; Hassan et al., 2023; Røed et al., 2023), which is understandable given their goal to create avatars that can realistically respond in mock interviews. In order to be maximally useful in research and practice with actual interviews, however, models could be trained and tested on data from real forensic interviews and court testimony, where factors such as varying levels of interviewer training may result in variability in the types of questions asked in different contexts.

Moreover, actual forensic interviews and testimony in court contain many utterances that are best classified as non-questions (Friend et al., 2024 [22%, forensic interviews]; Wylie et al., in press [19%, testimony]). Non-questions include introductory comments (e.g., "My job is to talk to children and make sure that they are safe"), instructions (e.g., "You can sit here, ok?") (Cyr & Lamb, 2009; Orbach et al; 2000), supportive statements (e.g., "You are being very clear") (Blasbalg et al., 2021), and echoes of children's statements (Friend et al., 2024).

Non-questions also include utterances that encourage the child to continue their narrative, such as "uh-huh." These are variously known as facilitators (Hershkowitz, 2002), back-channel utterances (Peterson et al., 1999), and minimal encouragers (Hassan et al., 2023). Because these utterances encourage children to continue responding to the previous question, rather than ask a new question, they are typically excluded from question type coding (Friend et al., 2024; Lamb et al., 2018). Lamb and colleagues (2018) noted that facilitators constituted 10-15% of interviewers' utterances in several studies, and these were samples in which invitations were quite rate; facilitators are likely still more common in higher quality interviews in which longer narrative responses from children give rise to more opportunity for facilitator use. In contrast, Hassan and colleagues' model (2023) included

facilitators (“minimal encouragers”) as open-ended questions, which is appropriate for programming avatars to respond to questions but may distort analyses of forensic interviews.

In highly structured interviews, one might manually remove interviewer utterances in the introductory stages of interviews without too much difficulty, but many interviews and court transcripts are not so tidy, complicating this process. At any rate, non-questions occur throughout interviews and testimony. Because automated question type models are most useful when they require minimal manual pre-processing, the ability to reliably recognize non-questions would increase the efficacy of machine-learning approaches.

The Present Study

We examined whether a large language model (RoBERTa) trained and tested on questions from forensic interviews and trial testimony in child sexual abuse cases (N = 351,920 utterances) could reliably distinguish among four types of utterances: 1) invitations, 2) wh- questions, 3) option-posing questions, and 4) non-questions.

Our analysis plan had three steps: (1) We calculated inter-rater reliability between machine and manual coders using various metrics, including Cohen’s Kappa; (2) For inconsistent codes, we manually double-coded the questions to determine whether inconsistencies were due to machine or manual inaccuracies; (3) We updated the original inconsistent manual codes with the corrected manual codes, and then recalculated reliability to give a more accurate measurement of the reliability of the machine model.

Method

Participants

The sample consisted of 351,920 interviewer/attorney utterances (including questions and non-questions) extracted from transcripts of forensic interviews (n = 1,435) with 3- to 17-

year-old children (M = 7.81) conducted in Los Angeles County between 2004 and 2022, and trial testimony (n = 416) of 4- to 17-year-old children (M = 11.85) in child sexual abuse criminal cases in Los Angeles County between 1997 and 2001. The interviews were transcribed and anonymized for training purposes. The court transcripts were obtained pursuant to the California Public Records Act (Cal. Government Code 6250, 2016). The use of archived interviews and court transcripts for research purposes was approved by the University of Southern California Institutional Review Board as exempt (45 CFR Section 46.014(d)(4)(ii)).

Materials and Procedure

Manual Question Type Coding

For the initial coding, coders had been trained to achieve high reliability (Kappa equal to or greater than .80) on categorizing interviewers' and attorneys' utterances as 1) invitations, 2) wh- questions, 3) yes-no questions, 4) forced-choice questions (questions that ask "x or y?"), 5) non-questions, and 6) other/unclassifiable. "Do you know" questions were coded as yes-no questions due to young children's tendency to answer them with unelaborated "yes" and "no" responses (Evans et al., 2014; 2017). Each interviewer turn was considered a single utterance. When multiple questions were posed in a single turn, the last question asked was coded, based on research finding that children most often respond to the last question in multi-part questions (Katz & Hershkowitz, 2012). Re-coding occurred until coders achieved reliability on a sample of 1,000 question-answer pairs and an additional sample of 400 lines specifically chosen to include questions and responses that were particularly difficult to code. The full dataset for the present study was coded by over 50 undergraduate and graduate research assistants trained by the third author and their staff, using a consistent coding scheme and similar training methods across the span of 15 years.

For the present study, we collapsed question types into four categories: 1) invitations, 2) wh-questions, 3) option-posing questions (yes-no questions and forced-choice questions), and 4) non-questions. Questions that had been manually coded as other/unclassifiable (0.7%) were excluded. The proportion of the different question types in the sample, broken down by source (forensic interview or trial testimony) is in Table 1.

Table 1. The proportion of invitations, wh- questions, option-posing questions, and non-questions in the forensic interviews and trial testimony

	Forensic Interviews (*n* = 236,598)	Trial Testimony (*n* = 115,322)	Total Sample (*n* = 351,920)
Invitations	9%	2%	6%
Wh- questions	44%	26%	38%
Option-posing questions	25%	61%	38%
Non-questions	22%	11%	18%

Training the Model

As demonstrated in Table 1, there was an imbalance in the number of questions between different question types. Therefore, we used stratified splitting to randomly divide the dataset into 1) a training and validation set (85%; *n* = 299,566) and 2) a test set (15%; *n* = 52,354), while preserving the relative proportion of different classes. We used RoBERTa, a large language model designed with approximately 125 million adjustable parameters enabling it to learn and understand text and make accurate predictions.

We used cross-validation with hyperparameter tuning to train the RoBERTa model to ensure the model generalizes well to new data. Hyperparameters are settings or configurations provided to a machine learning model prior to training, which can be adjusted

to control the learning process and model behavior. Cross-validation involves partitioning the training data into multiple subsets, or folds, and training the model multiple times, each time using a different fold as the validation set and the remaining folds as the training set. The three key hyperparameters optimized are learning rate, batch size, and epochs. Learning rate controls how much the model's parameters are adjusted during training after each step. A smaller learning rate means the model learns more slowly but can converge more precisely, while a larger learning rate speeds up learning but may overshoot the optimal solution. Batch size refers to the number of samples used in one update; smaller sizes lead to frequent updates, larger sizes offer more stability but require more memory. Epochs signify the number of times the model sees the entire dataset; more epochs allow more learning, but too many can cause overfitting.

First, a range of values was created for each hyperparameter to be tuned. This forms a grid of possible combinations of hyperparameters. For each combination, one full three-fold cross validation cycle was conducted. This cycle involved randomly splitting the training dataset (n = 299,566) into three equal parts using stratified splitting and training the RoBERTa model three times. In the first run, the first two splits served as the training set and the third split as the training test set. In the second run, the first and third splits served as the training set and the second split as the training test set. In the third run, the second and third splits served as the training set and the first split as the training test set. For each run, the model's accuracy was calculated, and the final accuracy was determined by averaging the accuracies from all three runs. This process was repeated for each hyperparameter combination, with the combination yielding the highest average accuracy selected as the optimal hyperparameter set. Using this optimal set, the RoBERTa model was then trained on the entire training set to obtain the final model. The test set (n = 52, 534), which had not been used in training, was then run on the final model.

Analysis Plan

We used the test set (n = 52,354) to assess inter-rater reliability between the machine codes and the manual codes. Inter-rater reliability was measured using percentage agreement and Cohen's Kappa. Sensitivity (the likelihood of correctly detecting questions that belong in a specific category, also known as the true positive rate) and specificity (the likelihood of correctly rejecting questions that do not belong in a specific category, also known as the true negative rate) were calculated separately for each question type. We present confusion matrices to summarize the performance of the model by displaying the counts of true positive, true negative, false positive, and false negative predictions.

Our analysis plan had three steps: (1) We calculated inter-rater reliability between the machine and manual coders, including percentage agreement, Kappa, and sensitivity (true positive rate) and specificity (true negative) for each question type. This provided an initial measure of the efficacy of the machine coding, assuming the manual coding was always correct. (2) In order to assess whether inconsistencies between the machine codes and the original manual codes reflected inadequacies in the machine model or resulted from human error, two blind coders re-coded every question in which the machine and manual codes were inconsistent. We then calculated two reliability measures. First, we calculated reliability between the machine codes and the re-coded manual codes for the inconsistent sample. Second, we calculated reliability between the original manual codes and the re-coded manual codes. Comparing these reliabilities enabled us to determine whether the inconsistencies were more often due to machine error (as reflected by the first reliability measure) or more often due to manual error (as reflected by the second reliability measure). (3) To minimize human error, we replaced the original manual codes with the re-coded manual codes for the inconsistent subsample. Reliability was calculated between the machine codes and the

manual codes (consisting of the re-coded codes for the inconsistent subsample and the original codes for all other lines) for the entire test sample.

Results

Step 1

The machine achieved high reliability with the manual codes; 95% agreement; K = .93, SE = .001. The confusion matrix is presented in Table 2, and the sensitivities and specificities of the machine's performance are in Table 3. The machine codes had high sensitivity, meaning the machine correctly detected at least 95% of questions belonging to each question type, and high specificity, meaning the machine correctly rejected at least 98% of questions that did not belong to each question type.

Table 2. Confusion matrix summarizing agreements and disagreements between the machine codes and the manual codes

		Machine codes				
		Invitation	Wh-question	Option-posing question	Non-question	Row totals
Manual codes	Invitation	**3,207**	91	4	44	3,346
	Wh- question	175	**18,921**	458	180	19,734
	Option-posing question	7	460	**18,548**	419	19,434
	Non-question	64	138	319	**9,319**	9,840
	Column totals	3,453	19,610	19,329	9,962	**52,354**

* Figures in bold signify agreement between the machine codes and the manual codes

Table 3. The sensitivity and specificity of the machine codes in relation to the manual codes

	Sensitivity [95% CI]	Specificity [95% CI]
Invitation	96% [.95, .97]	99% [.99, 1]
Wh-question	96% [.96, .96]	98% [.98, .98]
Option-posing question	95% [.95, .96]	98% [.97, .98]
Non-question	95% [.94, .95]	98% [.98, .99]

Step 2

Disagreements between the machine codes and the manual codes could have reflected mistakes by the machine or by the manual coders. Two experienced research assistants re-coded the small percentage (4.5%) of questions in which the machine codes were inconsistent with the manual codes (n = 2,360). The research assistants were blind to both the machine codes and the manual codes. Agreement between the two blind coders was K = .73, with all disagreements resolved by a third blind coder. The re-coders categorized 20 questions as other/unclassifiable, and these were excluded from comparisons between the machine and the manual codes.

Because the re-coded manual codes only included questions that were inconsistently coded by the machine and the manual coders, they could match the machine model's codes, the original manual codes, or neither code. The confusion matrix for the machine codes/re-coded manual codes is in Table 4, and the confusion matrix for the original manual codes/re-coded manual codes is in Table 5. The sensitivities and specificities for the machine codes and the manual codes, with the re-coded manual codes as the benchmark, are in Table 6.

In this sub-sample of questions in which there was initial disagreement between the machine and the manual coders, the machine outperformed the original manual coders on

every question type (Table 6). Particularly remarkable was the machine's superior performance in identifying invitations and non-questions. The reliability between the machine codes and the re-coded manual codes, 59% agreement, K = .43, SE = .02, was higher than between the original manual codes and re-coded manual codes, 38% agreement, K = .12, SE = .01.

Table 4. Confusion matrix summarizing agreements and disagreements between the machine codes and the re-coded manual codes for the subsample of inconsistently coded questions.

		Machine codes				
		Invitation	Wh-question	Option-posing question	Non-question	**Row totals**
Re-coded manual codes	Invitation	**191**	48	6	23	268
	Wh-question	36	**370**	216	103	676
	Option-posing question	5	238	**464**	158	800
	Non-question	13	31	90	**348**	600
	Column totals	245	687	775	633	**2,340**

* Figures in bold signify agreement between the machine codes and the re-coded manual codes.

Table 5. Confusion matrix summarizing agreements and disagreements between the manual codes and the re-coded manual codes for the subsample of inconsistently coded questions.

		Manual codes				
		Invitation	Wh-question	Option-posing question	Non-question	**Row totals**
Re-coded manual codes	Invitation	**61**	152	3	52	268
	Wh- question	53	**330**	233	109	725
	Option-posing question	4	252	**381**	228	865
	Non-question	21	74	260	**127**	482
	Column totals	139	808	877	516	**2,340**

* Figures in bold signify agreement between the original manual codes and the re-coded manual codes

Table 6. A comparison of the sensitivity and specificity of the manual codes and the machine codes in relation to the re-coded manual codes for the subsample of inconsistently coded questions

	Manual sensitivity [95% CI]	Machine sensitivity [95% CI]	Manual specificity [95% CI]	Machine specificity [95% CI]
Invitation	23% [.18, .28]	71% [.65, .77]	96% [.95, .97]	97% [.97, .98]
Wh-question	46% [.42, .49]	51% [.47, .55]	70% [.68, .73]	80% [.78, .82]
Option-posing question	44% [.41, .47]	54% [.50, .57]	66% [.64, .69]	79% [.77, .81]
Non-question	26% [.22, .31]	72% [.68, .76]	79% [.77, .81]	85% [.83, .86]

Step 3

We replaced the original manual codes with the re-coded manual codes in the test sample. This minimized human error, ensuring that the remaining inconsistencies reflected true differences between the machine codes and manual codes. The confusion matrix is in Table 7. The reliability of the machine codes increased, 98% agreement, $K = .97$, $SE = .001$, and, as shown in Table 8, the sensitivities and specificities of the machine codes approached 100%.

Table 7. Confusion matrix summarizing agreements and disagreements between the machine codes and the manual codes, after replacing the original manual codes with the re-coded manual codes.

		Machine codes				
		Invitation	Wh-question	Option-posing question	Non-question	Row totals
Manual codes after replacing unreliable codes	Invitation	**3,398**	48	6	23	3,475
	Wh-question	36	**19,291**	216	103	19,646
	Option-posing question	5	238	**19,012**	158	19,413
	Non-question	13	31	90	**9,666**	9,800
	Column totals	3,452	19,608	19,324	9,950	**52,334**

* Figures in bold signify agreement between the machine codes and the manual codes after replacing unreliable codes

Table 8. The sensitivity and specificity of the machine codes in relation to the manual codes, after replacing the original manual codes

	Sensitivity [95% CI]	Specificity [95% CI]
Invitation	98% [.97, .98]	100% [1, 1]
Wh-question	98% [.98, .98]	99% [.99, .99]
Option-posing question	98% [.98, .98]	99% [.99, .99]
Non-question	99% [.98, .99]	99% [.99, .99]

Discussion

This study examined whether a large language model trained on questions asked of children in forensic interviews and trial testimony (N = 351,920) could accurately distinguish among invitations, wh- questions, option-posing questions, and non-questions. Reliability between the codes generated by the model and the manual codes was high (95% agreement, K = .93), with sensitivities (true positive rates) at or above 95% and specificities (true negative rates) at or above 98% for all four question types. Re-coding of the disagreements between the machine codes and the manual codes revealed that human error generated most of the inconsistencies. Therefore, correcting errors in the manual codes resulted in still higher reliability between the machine codes and the manual codes (98% agreement, K = .97), with sensitivities at or above 98% and specificities at or above 99%. In addition to achieving high reliabilities, the model improves upon previous automated approaches to question type coding by using actual forensic interviews and trial testimony for the training, validation, and testing of the model; expanding the number of question types (including non-questions); adding measures of sensitivity and specificity; and identifying errors in the original manual codes.

Machine Codes Improved Sensitivity

One of the novel characteristics of our approach was to identify the 4.5% of questions in which the machine codes were different from the manual codes, and to re-code those disagreements in order to determine the source of the disagreement. Doing so revealed that the machine codes exhibited higher sensitivity (true positives) and specificity (true negatives) than the manual codes in every question category.

The difference between the machine codes and the manual codes was particularly stark with respect to correctly identifying invitations. Among the disagreements, the machine coding identified 71% of the invitations, whereas the manual coding identified only 23%. Examination of the confusion matrix comparing the re-coded manual codes to the original manual codes showed that the original manual codes misclassified 57% (152/268) of the invitations as wh- questions. Anecdotally, these were often cued invitations (e.g., “Ok tell me about that and what happened when [suspect] touched your peepee”; “And tell me everything that happened the day you got your dog”), suggesting that the specificity of the questions misled coders. This is reminiscent of the research finding that trainees sometimes miscoded invitations as wh- questions (Powell et al., 2013; Yi & Lamb, 2018), and Powell and colleagues’ (2013) observation that coders sometimes failed to recognize that invitations could include precise detail. On the other hand, neither the machine nor the manual coding misidentified wh- questions as invitations, a problem observed among some trainees (Powell et al., 2013; Yi & Lamb, 2018). Among the disagreements, both the machine codes (97%) and the manual codes (96%) showed high specificity for invitations, meaning that they correctly recognized when questions were not invitations, even when they disagreed about the correct code.

Limitations and Future Research

Although our reliability assessment and re-coding enabled us to identify errors when the machine and manual coders disagreed, we could not identify errors committed by both. These mutual errors would most likely occur if the manual coders systematically erred, such that the training data led the model to make the same mistakes. This presents a challenge for any model trained on manually coded data and is analogous to the emergence of biases in large language models trained on manually produced data (Navigli et al., 2023). Optimal training data would utilize double-coded questions with differences resolved by an expert coder (He & Schonlau, 2020). Although double-coding of a dataset as large as ours would be impractical, it may be possible to train future models on less data, particularly when using more sophisticated large language models.

Although this model was adept at distinguishing among invitations, wh- questions, option-posing questions, and non-questions, future work can make finer distinctions among question types. Invitations can be categorized as general or initial invitations (e.g., “Tell me everything that happened”) on the one hand, and cued invitations on the other, with the latter themselves divisible into breadth prompts (e.g., “What happened next?”) and depth prompts (e.g., “You mentioned X. Tell me more about that.”) (Danby & Sharman, 2023). Option-posing questions subsume a number of question types, including yes-no and forced-choice questions, and these too could be distinguished in future models. Furthermore, future models could analyze the appropriateness of different types of questions based on where they appear in the interview. For example, interviewers are advised to exhaust initial recall through invitations before moving to wh-questions (Lyon & Henderson, 2021), and option-posing questions are considered less risky when “paired” with a follow-up open-ended prompt (Lamb et al., 2018).

Consistent with prior research utilizing machine coding of question-type (Garcia, 2022; Hassan et al., 2023; Røed et al., 2023), we did not attempt to separately identify

suggestive questions. Observational research has found that the great majority of suggestive questions were option-posing questions (Andrews et al., 2016), and much of the classic experimental work examining suggestibility used option-posing questions, including forced-choice questions (e.g., “When Sam Stone ripped the book, did he do it because he was angry, or by mistake?”; Leichtman & Ceci 1995) and yes-no questions phrased as tag questions (“That is naughty, isn’t it?”; Thompson et al., 1997). Regardless of their suggestiveness, option-posing questions are generally recognized as less productive and more prone to false positive and false negative errors (Lamb et al., 2018). Nevertheless, future work can determine if machine coding can distinguish between non-suggestive and suggestive questions, which are usually option-posing but can also be phrased as open-ended questions (e.g., “how did he touch you?” asked before the child disclosed touch).

Future models may also be trained to categorize the types of responses elicited by questions. At the most basic level, models could distinguish between unelaborated and elaborated responses to option-posing questions and identify unproductive or unresponsive answers such as “don’t know” responses, requests for clarification, and silence. Response type coding could indicate whether details were child-generated and flag potential interviewer-child miscommunications.

Implications for Research and Practice

Automated question type coding substantially decreases the time and effort required to question type code forensic interviews and court transcripts. Coding a single forensic interview typically takes less than a minute for the machine-generated model, whereas in our experience manual coders need between 30 minutes and several hours, depending on interview length, to complete the same task. Furthermore, automated models can improve the accuracy of question type coding, as demonstrated by the superior specificity and sensitivity

of the machine-generated coding compared with the manual codes in identifying all four question types. They have promise for other contexts in which question type is important, including questioning of suspects (Hershkowitz & Lamb, 2024) and adults (Webster et al., 2021).

These results have implications for both research and practice. In research, using an automated model to code for question type allows researchers to focus their time and effort on developing new coding classifications, including codes for specific content, such as clothing placement (Stolzenberg & Lyon, 2017) or sexual touch (Burrows et al., 2017; Szojka et al., 2023). Widespread availability of an automated model can help researchers identify differences in their assessment of question type, and ultimately lead to greater consistency across labs in classification. Automated question type coding may be especially useful for practitioners, as it provides them with the opportunity to review the quality of their questioning without investing substantial time and effort in manually coding interviews. Furthermore, they may benefit the most from the greater accuracy of machine coding since they are unlikely to have received the training and supervision of coders involved in research.

As noted in the introduction, automated question type coding models are part of innovative approaches to forensic interviewer training using AI technology to create a virtual child programmed to respond as a real child (e.g., Hassan et al., 2023; Roed et al., 2023). Initial versions of this model relied on an operator to manually code interviewers' questions and trigger a predetermined response (Haginoya et al., 2021); however, more recent studies were able to automate this step using large language models (Hassan et al., 2023; Roed et al., 2023). Our model's superior reliability and ability to identify non-question utterances make it a valuable asset for researchers aiming to fine tune AI-based interviewer training programs.

Conclusion

We trained a large language model to automate the classification of question types in forensic interviews and trial testimony regarding child sexual abuse. The model achieved high reliability in distinguishing among invitations, wh- questions, option-posing questions, and non-questions. Discrepancies between machine and manual codes revealed that the machine codes were more accurate, resulting in further improvements in the model's reliability when manual errors were corrected. Automated question type coding increases the speed and accuracy of classification and promotes consistency among researchers and practitioners. In order to facilitate these goals, we have made our model publicly available on an online repository, and the supplemental materials include a step-by-step guide for using the model.

References

Andrews, S. J., Ahern, E. C., Stolzenberg, S. N., & Lyon, T. D. (2016). The productivity of wh-prompts when children testify. *Applied Cognitive Psychology*, 30(3), 341-349. https://doi.org/10.1002/acp.3204

APSAC Taskforce. (2012). *Practice guidelines: Forensic interviewing in cases of suspected child abuse.* American Professional Society on the Abuse of Children.

Blasbalg, U., Hershkowitz, I., Lamb, M. E., & Karni-Visel, Y. (2021). Adherence to the revised NICHD protocol recommendations for conducting repeated supportive interviews is associated with the likelihood that children will allege abuse. *Psychology, Public Policy, and Law, 27(2),* 209-220. https://doi.org/10.1037/law0000295

Brown, D. A., Lamb, M. E., Lewis, C., Pipe, M.-E., Orbach, Y., & Wolfman, M. (2013). The NICHD investigative interview protocol: An analogue study. *Journal of Experimental Psychology: Applied*, 19(4), 367-382. https://doi.org/10.1037/a0035143

Burrows, K. S., Bearman, M., Dion, J., & Powell, M. B. (2017). Children's use of sexual body part terms in witness interviews about sexual abuse. *Child Abuse and Neglect, 65,* 226-235. https://doi.org/10.1016/j.chiabu.2017.02.001

Cyr, M. (2022). *Conducting interviews with child victims of abuse and witnesses of crime: A practical guide.* Routledge.

Cyr, M., Dion, J., Gendron, A., Powell, M., & Brubacher, S. (2021). A test of three refresher modalities on child forensic interviewers' posttraining performance. *Psychology, Public Policy, and Law*, 27(2), 221-230. https://doi.org/10.1037/law0000300

Cyr, M., Dion, J., McDuff, P., & Trotier-Sylvain, K. (2012). Transfer of skills in the context of non-suggestive investigative interviews: Impact of structured interview protocol and feedback. *Applied Cognitive Psychology, 26(4),* 516-524. https://doi.org/10.1002/acp.2822

Cyr, M., & Lamb, M. E. (2009). Assessing the effectiveness of the NICHD investigative interview protocol when interviewing French-speaking alleged victims of child sexual abuse in Quebec. *Child Abuse & Neglect, 33*(5), 257-268. https://doi.org/10.1016/j.chiabu.2008.04.002

Danby, M. C., & Sharman, S. J. (2023). Open-ended initial invitations are particularly helpful in eliciting forensically relevant information from child witnesses. *Child Abuse & Neglect, 146*, 1-13. https://doi.org/10.1016/j.chiabu.2023.106505

Evans, A. D., Stolzenberg, S. N., Lee, K., & Lyon, T. D. (2014). Young children's difficulty with indirect speech acts: Implications for questioning child witnesses. *Behavioral Sciences & the Law, 32*(6), 775-788. https://doi.org/10.1002/bsl.2142

Evans, A. D., Stolzenberg, S. N., & Lyon, T. D. (2017). Pragmatic failure and referential ambiguity when attorneys ask child witnesses 'do you know/remember' questions. *Psychology, Public Policy, and Law, 23*(2), 191-199. https://doi.org/10.1037/law0000116

Friend, O.W., Nogalska, A.M., & Lyon, T.D. (2024). The utility of direct questions about actions with the hands in child forensic interviews. *Psychology, Public Policy, & Law, 30(2),* 121-131. https://doi.org/10.1037/law0000426

Garcia, F. J. (2022). *Techniques to elicit disclosure of child sexual abuse in investigative interviews.* (Doctoral dissertation, Griffith University). https://doi.org/10.25904/1912/4905

Garcia, F. J., Powell, M. B., Brubacher, S. P., Eisenchlas, S. A., & Low-Choy, S. (2022). The influence of transition prompt wording on response informativeness and rapidity of

disclosure in child forensic interviews. *Psychology, Public Policy, and Law, 28*(2), 255-266. https://doi.org/10.1037/law0000347

Guadagno, B. L., Hughes-Scholes, C. H., & Powell, M. B. (2013). What themes trigger investigative interviewers to ask specific questions when interviewing children? *International Journal of Police Science & Management, 15*(1), 51-60. https://doi.org/10.1350/ijps.2013.15.1.301

Haginoya, S., Ibe, T., Yamamoto, S., Yoshimoto, N., Mizushi, H., & Santtila, P. (2023). AI avatar tells you what happened: The first test of using AI-operated children in simulated interviews to train investigative interviewers. *Frontiers in Psychology, 14*, 1133621. https://doi.org/10.3389/fpsyg.2023.1133621

Hassan, S. Z., Sabet, S. S., Riegler, M. A., Baugerud, G. A., Ko, H., Salehi, P., ... & Halvorsen, P. (2023). Enhancing investigative interview training using a child avatar system: a comparative study of interactive environments. *Scientific Reports, 13(1),* 20403.https://doi.org/10.1038/s41598-023-47368-2

He Z., Schonlau M. (2020). Automatic coding of open-ended questions into multiple classes: whether and how to use double coded data. *Survey Research Methods, 14*, 267-287. https://doi.org/10.18148/srm/2020.v14i3.7639

Hershkowitz, I. (2002). The role of facilitative prompts in interviews of alleged sex abuse victims. *Legal and Criminological Psychology, 7*(1), 63-71. https://doi.org/10.1348/135532502168388

Hershkowitz, I., & Lamb, M. E. (2024). Interviewing young offenders about child-on-child sexual abuse. *Development and Psychopathology,* 1-17. https://doi.org/10.1017/S095457942400066X

Hershkowitz, I., Lamb, M.E., Orbach, Y., Katz, C. and Horowitz, D. (2012), The development of communicative and narrative skills among preschoolers: Lessons from forensic interviews about child abuse. *Child Development, 83*(2), 611-622. https://doi.org/10.1111/j.1467-8624.2011.01704.x

Henderson, H. M., Lundon, G. M., & Lyon, T. D. (2022). Suppositional wh-questions about perceptions, conversations, and actions are more productive than paired yes–no questions when questioning maltreated children. *Child Maltreatment, 28*(1), 55-65. https://doi.org/10.1177/10775595211067208

Henderson, H. M., Russo, N., & Lyon, T. D. (2020). Forensic interviewers' difficulty with invitations: Faux invitations and negative recasting. *Child Maltreatment, 25*(3), 363-372. https://doi.org/10.1177/107755951989559

Johnson, M., Magnussen, S., Thoresen, C., Lønnum, K., Burrell, L.V., & Melinder, A. (2015). Best practice recommendations still fail to result in action: A national 10-year follow-up study of investigative interviews in CSA cases. *Applied Cognitive Psychology, 29,* 661-668. https://doi.org/10.1002/acp.3147

Katz, C., & Hershkowitz, I. (2012). The effect of multipart prompts on children's testimonies in sexual abuse investigations. *Child Abuse & Neglect, 36*(11-12), 753-759. https://doi.org/10.1016/j.chiabu.2012.07.002

Klemfuss, J. Z., Quas, J. A., & Lyon, T. D. (2014). Attorneys' questions and children's productivity in child sexual abuse criminal trials. *Applied Cognitive Psychology, 28*(5), 780-788. https://doi.org/10.1002/acp.3048

Lamb, M. E., Orbach, Y., Hershkowitz, I., Horowitz, D., & Abbott, C. B. (2007). Does the type of prompt affect the accuracy of information provided by alleged victims of abuse in forensic

interviews? *Applied Cognitive Psychology, 21*(9), 1117-1130. https://doi.org/10.1002/acp.1318

Lamb, M. E. (2016). Difficulties translating research on forensic interview practices to practitioners: Finding water, leading horses, but can we get them to drink? *American Psychologist, 71*(8), 710-718. https://doi.org/10.1037/amp0000039

Lamb, M. E., Brown, D. A., Hershkowitz, I., Orbach, Y., & Esplin, P. W. (2018). *Tell me what happened: Questioning children about abuse* (2nd ed.). John Wiley and Sons.

Lamb, M. E., Sternberg, K. J., Orbach, Y., Esplin, P. W., & Mitchell, S. (2002). Is ongoing feedback necessary to maintain the quality of investigative interviews with allegedly abused children? *Applied Developmental Science, 6*(1), 35-41. https://doi.org/10.1207/S1532480XADS0601_04

Leichtman, M. D., & Ceci, S. J. (1995). The effects of stereotypes and suggestions on preschoolers' reports. *Developmental Psychology, 31*(4), 568-578. https://doi.org/10.1037/0012-1649.31.4.568

Lyon, T. D., & Henderson, H. M. (2021). Increasing true reports without increasing false reports: Best practice interviewing methods and open-ended wh- questions. *American Professional Society on the Abuse of Children Advisor, 33*(1), 29-39

Lyon, T. D., & Dente, J. (2012). Child witnesses and the Confrontation Clause. *Journal of Criminal Law & Criminology, 102,* 1181-1232.

Navigli, R., Conia, S., & Ross, B. (2023). Biases in large language models: origins, inventory, and discussion. *ACM Journal of Data and Information Quality, 15*(2), 1-21. https://doi.org/10.1145/3597307

Orbach, Y., Hershkowitz, I., Lamb, M. E., Sternberg, K. J., Esplin, P. W., & Horowitz, D. (2000). Assessing the value of structured protocols for forensic interviews of alleged child abuse victims. *Child Abuse & Neglect, 24*(6), 733-752. https://doi.org/10.1016/S0145-2134(00)00137-X

Oxburgh, G., Myklebust, T., & Grant, T. (2010). The question of question types in police interviews: A review of the literature from a psychological and linguistic perspective. *International Journal of Speech, Language and the Law: Forensic Linguistics, 17(1)*, 46-66. https://doi.org/10.1558/ijsll.v17i1.45

Peterson, C., & Grant, M. (2001). Forced-choice: Are forensic interviewers asking the right questions? *Canadian Journal of Behavioural Science, 33*(2), 118-127. https://doi.org/10.1037/h0087134

Peterson, C., Jesso, B., & McCabe, A. (1999). Encouraging narratives in preschoolers: An intervention study. *Journal of Child Language, 26(1)*, 49-67. https://doi.org/10.1017/S0305000998003651

Poole, D. A., & Dickinson, J. (2024). *Interviewing children: The science of conversation in forensic contexts* (2nd ed.). American Psychological Association.

Powell, M. B., Benson, M. S., Sharman, S. J., Guadagno, B., & Steinberg, R. (2013). Errors in the identification of question types in investigative interviews of children. *International Journal of Police Science & Management, 15*, 144-156. https://doi.org/10.1350/ijps.2013.15.2.308

Powell, M. B., & Brubacher, S. P. (2020). The origin, experimental basis, and application of the standard interview method: An information-gathering framework. *Australian Psychologist, 55*(6), 645-659. https://doi.org/10.1111/ap.12468

Powell, M. B., Guadagno, B., & Benson, M. (2016). Improving child investigative interviewer performance through computer-based learning activities. *Policing and Society, 26*(4)*, 3*65-374. https://doi.org/10.1080/10439463.2014.942850

Røed, R. K., Baugerud, G. A., Hassan, S. Z., Sabet, S. S., Salehi, P., Powell, M. B., Riegler, M. A., Halvorsen, P., & Johnson, M. S. (2023). Enhancing questioning skills through child avatar chatbot training with feedback. *Frontiers in Psychology, 14*. https://doi.org/10.3389/fpsyg.2023.1198235

Stolzenberg, S. N., & Lyon, T. D. (2017). 'Where were your clothes?' Eliciting descriptions of clothing placement from children alleging sexual abuse in criminal trials and forensic interviews. *Legal and criminological psychology, 22*(2), 197-212. https://doi.org/10.1111/lcrp.12094

Szojka, Z. A., & Lyon, T. D. (2024). Children's elaborated responses to yes-no questions in forensic interviews about sexual abuse. *Child Maltreatment, 29*(4), 637-647. https://doi.org/10.1177/10775595231220228

Szojka, Z. A., Moussavi, N., Burditt, C., & Lyon, T. D. (2023). Attorneys' questions and children's responses referring to the nature of sexual touch in child sexual abuse trials. *Child Maltreatment, 28*(3), 438-449. https://doi.org/10.1177/10775595231161033

Thompson, W. C., Clarke-Stewart, K. A., & Lepore, S. J. (1997). What did the janitor do? Suggestive interviewing and the accuracy of children's accounts. *Law and Human Behavior, 21*, 405-426. https://doi.org/10.1023/A:1024859219764

Vieth, V. I. (2021). The forensic interviewer at trial: Guidelines for the admission and scope of expert testimony concerning a forensic interview in a case of child abuse (Revised and Expanded). *Mitchell Hamline Law Review, 47*(3), 849-890.

Warren, A. R., Woodall, C. E., Thomas, M., Nunno, M., Keeney, J. M., Larson, S. M., & Stadfeld, J. A. (1999). Assessing the effectiveness of a training program for interviewing child witnesses. *Applied Developmental Science, 3*(2), 128-135. https://doi.org/10.1207/s1532480xads0302_6

Webster, W.S., Oxburgh, G.E., & Dando, C.J. (2021). The use and efficacy of question type and an attentive interviewing style in adult rape interviews. *Psychology, Crime & Law, 27*(7), 656-577. https://doi.org/10.1080/1068316X.2020.1849694

Wolfman, M., Brown, D., & Jose, P. (2016). Talking past each other: Interviewer and child verbal exchanges in forensic interviews. *Law and Human Behavior, 40(2),* 107-117. http://dx.doi.org/10.1037/lhb0000171

Wylie, B. E., Bruer, K. C., Williams, S., Evans, A. D. (in press). Lawyer questioning practices in Canadian courtrooms. *Canadian Journal of Behavioural Science/Revue canadienne des sciences du comportement.* Advance online publication. https://doi.org/10.1037/cbs0000413

Yi, M., & Lamb, M. E. (2018). How accurately do police officers identify the types of questions used in investigative interviews with child victims? *Korean Journal of Forensic Psychology, 9(3)*, 117-135. https://doi.org/10.53302/kjfp.2018.11.9.3.117

Online Supplemental Materials: Instructions for installing and running the automated question type coding model

Prerequisites

Python3 and an Integrated Development Environment (IDE) like Visual Studio Code (recommended) to run Python script.

Downloading and setting up Visual Studio Code (VS Code) for Python

Step 1: Download Visual Studio Code

- Go to the official Visual Studio Code website https://code.visualstudio.com/ using your web browser.
- Click on the "Download for Windows" button if you are using Windows or the corresponding button for your operating system.

Step 2: Install Visual Studio Code

- Once the installer is downloaded, open the downloaded file to start the installation process.
- Follow the on-screen instructions to install VS Code. You might need administrative privileges for installation.

Step 3: Install Python Extension for VS Code

- Open Visual Studio Code after installation.
- Go to the Extensions view by clicking on the square icon on the left sidebar or using the shortcut Ctrl+Shift+X (Windows/Linux) or Cmd+Shift+X (macOS).
- Search for "Python" in the Extensions Marketplace.
- Click on the "Install" button next to the official Python extension offered by Microsoft.

Step 4: Set up Python Interpreter

- Open your Python file or create a new one in VS Code.
- If prompted at the bottom-right corner, click on "Select Python Interpreter." If not prompted, use the shortcut Ctrl+Shift+P (Windows/Linux) or Cmd+Shift+P (macOS) to open the command palette.
- Type "Python: Select Interpreter" and select the Python interpreter you want to use. If it's not listed, navigate to the Python executable path.

Step 5: Write and run Python code

- Write or open your Python script in VS Code.
- To run the Python file:
- Use the shortcut Ctrl+S to save the file.
- Use the shortcut Ctrl+F5 (Windows/Linux) or Cmd+Option+N (macOS) to run the Python script without debugging.

- You can also click on the “Run Python File in Terminal” button in the top right corner of the editor.

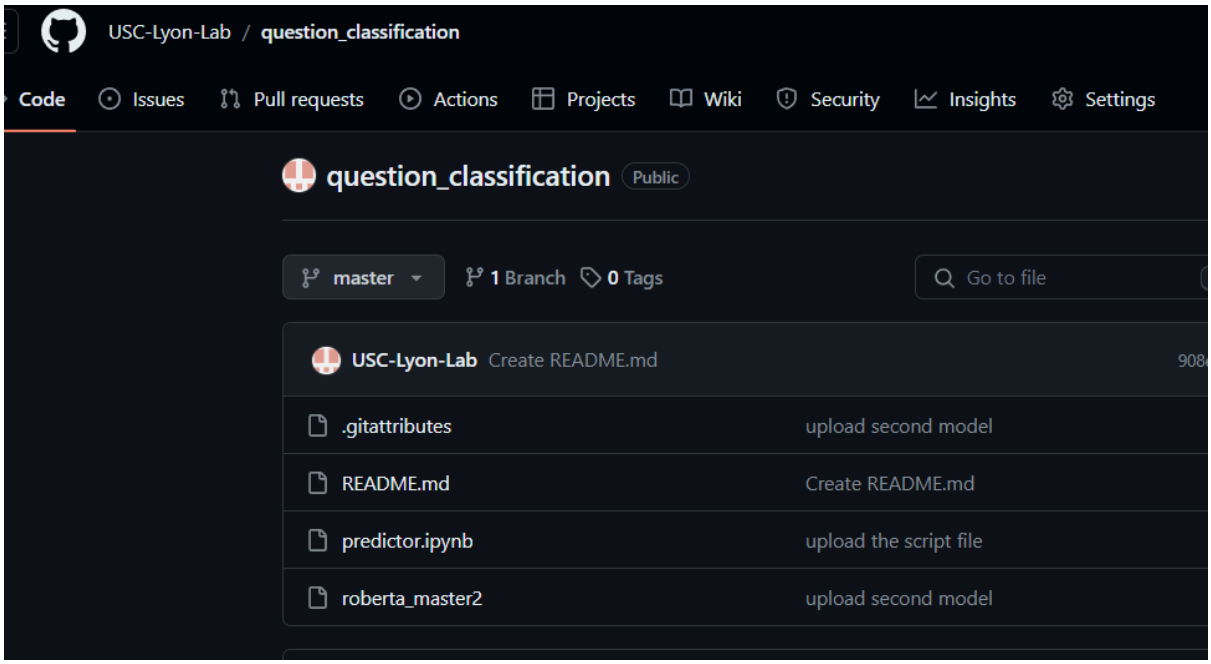
Running the question classification model

Step 1: Download the model

- Use the github link below to download the model and script file:
- https://github.com/usc-lyon-lab/question_classification
- Model file name - roberta_master2
- Script file name - predictor.ipynb

Step 2: Create the folder structure for running the model

- Make a new folder on your computer and put both of the above files in this folder.
- Place the csv file that needs to be classified in the same folder.
- See below the required folder structure.



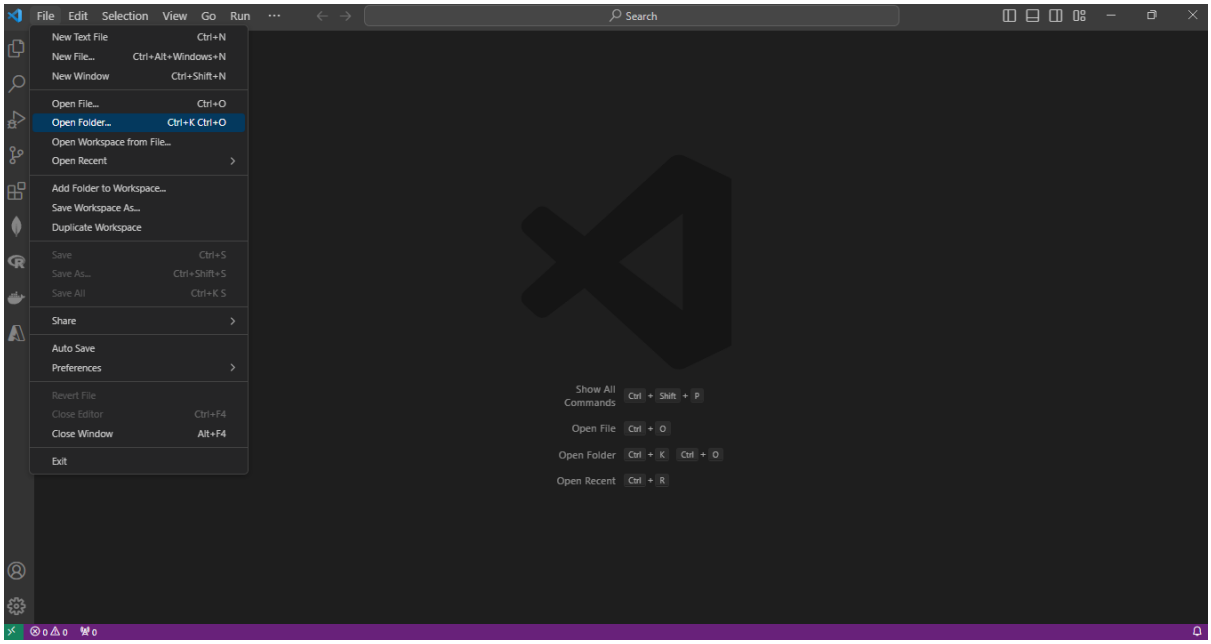
Step 3: Set up the csv file

- Make sure there is a column named “questions” in the csv file that needs to be classified. Please note, the column text is case sensitive and there should not be any spaces before or after the column name.

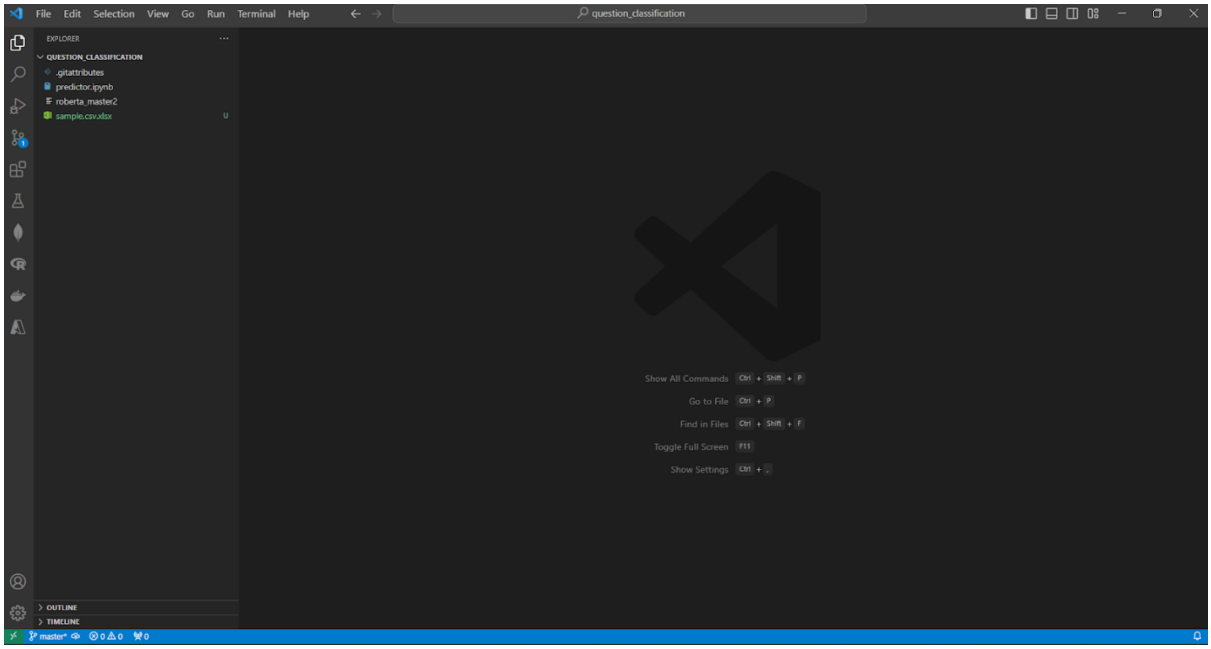
	A	B	C	D	E
1	**questions**				
2	Okay. And tell me all about what helped you say that.				
3	No? Ok.				
4	Okay. What was in that drawer?				
5	Did he ever hurt your arm?				
6	Mk. And what did you say to the police?				
7	Did she come in before he was doing this?				
8	Did you have lunch together?				
9	Ok, and who pulled down his pants?				
10	Ok. Well.				
11	Are you absolutely sure that happened?				
12	Where were you when you were chasing the two boys?				
13	Mm-hmm (yes).				
14	Mm-hmm, mm-hmm, I understand.				
15	No. He didn't put it inside your pants?				
16					

Step 4: Open Visual Studio

- Open visual studio and click on “File”-> “Open Folder” and browse to the above folder

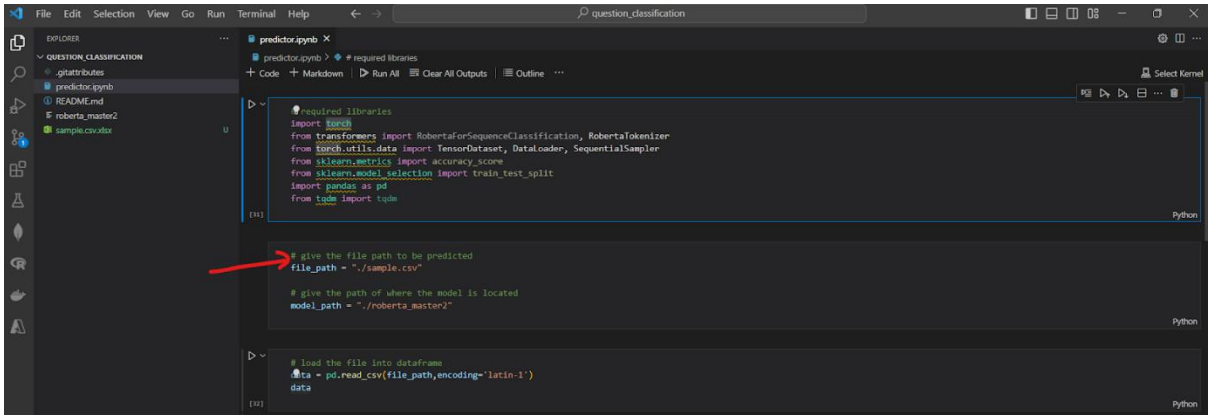


- Once opened, we can see all the content of the folder on the left side in Visual Studio.



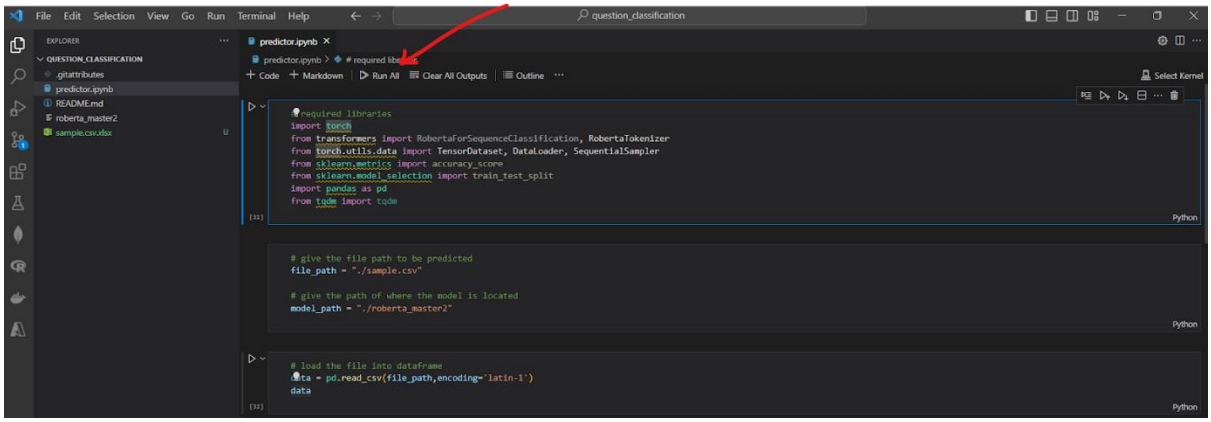
Step 5: Open the predictor file

- Open the predictor.py file in visual studio by clicking on the file name shown on the left window.
- In the second cell of the predictory.py file, there is a variable called “file_path”. Update the value of this variable to your csv file name. Eg: if your csv file name is “roberta.csv” then update the value: file_path = “./roberta.csv”



Step 6: Run the script

- Click on the “Run All” button on top to run the script.



- Once the script finishes, the csv file will be updated with a new column called "predictions."

	A	B	C	D	E
1	**questions**	predictions			
2	Okay. And tell me all about what helped you say that.	3			
3	No? Ok.	0			
4	Okay. What was in that drawer?	2			
5	Did he ever hurt your arm?	1			
6	Mk. And what did you say to the police?	2			
7	Did she come in before he was doing this?	1			
8	Did you have lunch together?	1			
9	Ok, and who pulled down his pants?	2			
10	Ok. Well.	0			
11	Are you absolutely sure that happened?	1			
12	Where were you when you were chasing the two boys?	2			
13	Mm-hmm (yes).	0			
14	Mm-hmm, mm-hmm, I understand.	0			
15	No. He didn't put it inside your pants?	1			
16					