

A Large Language Model Approach to Educational Survey Feedback Analysis

Michael J. Parker^{*}, Caitlin Anderson, Claire Stone, YeaRim Oh
Harvard Medical School

^{*}Corresponding author(s). E-mail(s): michael_parker@hms.harvard.edu

Abstract

This paper assesses the potential for the large language models (LLMs) GPT-4 and GPT-3.5 to aid in deriving insight from education feedback surveys. Exploration of LLM use cases in education has focused on teaching and learning, with less exploration of capabilities in education feedback analysis. Survey analysis in education involves goals such as finding gaps in curricula or evaluating teachers, often requiring time-consuming manual processing of textual responses. LLMs have the potential to provide a flexible means of achieving these goals without specialized machine learning models or fine-tuning. We demonstrate a versatile approach to such goals by treating them as sequences of natural language processing (NLP) tasks including classification (multi-label, multi-class, and binary), extraction, thematic analysis, and sentiment analysis, each performed by LLM. We apply these workflows to a real-world dataset of 2500 end-of-course survey comments from biomedical science courses, and evaluate a zero-shot approach (i.e., requiring no examples or labeled training data) across all tasks, reflecting education settings, where labeled data is often scarce. By applying effective prompting practices, we achieve human-level performance on multiple tasks with GPT-4, enabling workflows necessary to achieve typical goals. We also show the potential of inspecting LLMs' chain-of-thought (CoT) reasoning for providing insight that may foster confidence in practice. Moreover, this study features development of a versatile set of classification categories, suitable for various course types (online, hybrid, or in-person) and amenable to customization. Our results suggest that LLMs can be used to derive a range of insights from survey text.

Keywords: Large Language Models (LLMs), survey analysis, GPT-4, GPT-3.5, ChatGPT, qualitative methodology

1 Introduction

Education feedback, much of it in the form of unstructured text comments from learners as part of survey responses, is considered an important aspect of course evaluation as well as facilitates course improvement [1]. This holds true regardless of whether a course is online, in-person, or in a blended or hybrid format [2–4].

During the COVID-19 pandemic, many educators shifted their courses online. This change necessitated updating knowledge of course design and teaching to incorporate rules of learning and engagement that share principles with those of in-person courses, but that also differ to some extent based on changes in the medium, types of course resources, teaching modality, and methods of course delivery. Even with a widespread return to in-person or hybrid learning, many of the tools and media from online teaching have persisted, such that learning about how to best design, teach, and deliver courses is a continual process, with a strong need for understanding how to make courses that have high learning value and are well-received.

In this context, collecting course feedback plays a critical role not only for educators, but also for course designers, educational administrators, and course providers (for example, organizations that create online courses for widespread use). Each of these roles has a set of specific questions and goals against which they seek to evaluate courses using results of feedback tools like surveys. For example, educators and course designers would like to know what content and modalities resonated with students or were received poorly, such that a course can be improved in a more rapid iterative cycle. For those involved in course delivery, understanding how the scheduling, timing, cost, ease of access, and other such factors affected the student experience can provide valuable information for process improvement. At a higher level, educational administrators often seek to evaluate their faculty as teachers, feeding into aspects such as teaching awards, promotion, or determination of the need for faculty development. With an eye toward long-term planning, course providers try to identify gaps in course content and formats to maximize learning value and engagement.

1.1 Types of tasks associated with analysis unstructured survey data

Using survey textual responses to explore these types of high level goals of stakeholders requires chaining together multiple NLP tasks in the form of workflows. Such workflows can be implemented with a small set of natural language processing (NLP) tasks, including classification, extraction, and sentiment analysis, that form composable building blocks for similar workflows.

Classification of comments may be single-label (binary or multi-class, the latter involving classifying into one of a set of tags) or multi-label (classification of each comment with one or more of a set of tags), and the tags (also called labels, classes, or categories) are frequently custom-chosen, reflecting the goals of a particular analysis. Often those doing the analysis have a specific objective or goal focus that they are investigating (e.g. suggestions for improvement), and text extraction is a useful technique for this purpose.

Sentiment analysis can be used to lend nuance and insight to the quantitative ratings that are gathered through Likert scales or “star” ratings.

A high-level breakdown of objectives and NLP tasks is shown in Table 1.

Table 1 NLP tasks that may be used for analysis of textual survey responses.

Objective	Question	NLP Tasks	Notes
High-level initial analysis	What did students say (and how did they feel about the course)?	Multi-label classification, inductive thematic analysis, sentiment analysis	Depends on whether analysis is top-down (using pre-determined labels or areas of interest) or bottom-up (deriving themes from scratch based on student comments)
Answering a focused question	What did students say about x (particular focus)?	Extraction	Results are amenable to multi-class classification or inductive thematic analysis
Quantification of textual survey responses	How many comments were there on each aspect?	Classification (binary, multi-label, or multi-class)	Helps the person performing analysis find themes of greater importance

1.2 Previous approaches and challenges in analyzing education feedback

Despite the high motivation to learn from education feedback, significant challenges remain. Prior to recent developments in machine learning, systematic analysis of feedback comments, needed for forming data-backed conclusions, required manual (human) annotation and classification or extraction of key passages, tasks which can be time-consuming, costly, and significantly lengthen the course improvement cycle. This type of manual analysis is still the primary approach in many settings. In more specialized courses or course platforms, those familiar with the use case (domain experts, in the form of course educators or those involved in other ways in course delivery) are needed for annotation or extraction, making the process even more difficult. In courses with many students and hence a large volume of feedback, common for both in-person courses as well as online courses, the time and/or cost of human annotation of feedback can be prohibitive.

Employing crowdworkers, for example via Amazon’s Mechanical Turk platform, reduces the cost and time of manual annotation. However, the quality of results may vary, particularly in cases where some degree of domain expertise is needed. Additionally, a recent study [5] provided evidence that a substantial fraction of crowdworkers used generative AI (LLMs) to assist with a summarization task, leading to a mix of results from humans and LLMs and raising doubt that crowdworkers will continue to be a reliable source of human annotations.

For feature extraction from text, techniques like TF-IDF, and Word2Vec have been applied for short text classification and sentiment analysis [6–9]. Topic modeling using latent semantic analysis or latent Dirichlet allocation has been useful for discovering

themes and trends in collections of student feedback [10–12]. For evaluating text, sentiment analysis techniques like CNN and Bi-LSTM models have been used to classify student evaluations [13, 14]. Overall, these techniques have shown utility for gaining insights from student feedback.

With the advent of recent machine learning (ML) techniques, great strides have been made in dealing with unstructured text. BERT (Bidirectional Encoder Representations from Transformers, [7]) and related models allow for transformation of text passages into numerical formats (high dimensional dense vectors called embeddings) that are then amenable to classification via conventional ML methods such as logistic regression. Good results have been achieved in certain contexts using such models [15]. Despite such advances, challenges remain that present obstacles to routine use of such models in practice.

Specialized ML models often require a “fine-tuning” process using labeled data (data that human annotators have classified) to best adapt to a specific use case. Depending on the amount of human labeling needed, this aspect may provide a stumbling block based on the time and effort involved. Although there are many examples of labeled datasets [16–19], real-world use cases often rely on custom labels for which there is no pre-existing labeled data for fine-tuning. Even supposing such fine-tuning takes place, there are additional barriers to practical use of this technology.

One such barrier is that multiple distinct AI models may be needed, depending on the range of tasks. The model that is suitable for classification may not be the same one that performs text extraction, and each model may need its own fine-tuning or adaptation.

Even for a core task like classification, there are a number of challenges. Difficulty of classification increases in situations where multiple labels may concurrently be assigned to the same survey comment, often leading to a degree of inter-rater disagreement even among highly-skilled human annotators who have high familiarity with the domain. Other challenges include data imbalance, multi-topic comments, and domain-specific terminology [20, 21].

In classifying unstructured textual feedback, data imbalance exists when the labels chosen are not attributable in equal proportions across a dataset; some labels may be comparatively rare. If there are few examples of particular labels, this scarcity can create difficulties in training machine learning models that classify new comments. If human labeling is being used as ground truth, rarity of certain labels may require labeling a larger set of feedback to enable training an ML classifier.

Another challenge is that of multi-topic comments. Depending on how feedback is collected and how open-ended the survey questions are, students may provide feedback that encompasses multiple topics (for example, “I found the quizzes incredibly difficult, but the teacher was great and I felt I got what I paid for. If I had had more time to complete the course, this would have been even better.”). Such multi-topic comments present a challenge for ML techniques based on embeddings (dense vector representations) derived from models such as BERT (or BERT related, such as Sentence-BERT, [22]), given that the embedding of a comment is related to the comment’s semantic meaning. A comment with multiple topics may have an embedding that doesn’t adequately localize to the semantic

“neighborhood” of any of the individual topics associated with that comment, decreasing the performance of downstream classifiers. Use of context-specific, specialized terms in the text data, known as domain-specific language, can also decrease the performance of ML techniques.

Deep learning models like BERT that perform feature selection by creating embeddings have been pre-trained on a large corpus of text, usually publicly accessible and mostly from the internet. Depending on the pre-training, terms specific to a specialized domain such as immunology or biomedical engineering may not have been seen during training, or seen only in very limited quantities. In those cases, the pre-trained model cannot adequately capture the semantics of such terms via its embeddings, again impacting the performance of downstream applications such as classification and clustering that may rely on those embeddings.

In sentiment analysis, pre-trained sentiment analysis models may not adapt well to settings where it is important to take into account the context. For example, in analyzing comments from biomedical science courses that cover cancer as a topic, learners’ comments may include the words ‘cancer’ or ‘oncology’ or ‘tumor’, simply as referring to parts of the curriculum. These comments may end up being classified as negative even by a state-of-the-art existing model, given that discussions of cancers and tumors in many training datasets (often from internet settings) may be in the context of negative emotions being expressed.

Finally, a common challenge is that of lack of interpretability of results coming from specialized machine learning models. Although there has been significant work on approaches like visualizing factors that contribute to a neural network-based model’s predictions, complex models may still be viewed as “black boxes” by downstream users in areas like education, with this perception potentially inhibiting usage.

1.3 LLM Background and Related Research

Education feedback analysis seeks to extract insights from open-ended written responses, such as student surveys or teacher evaluations, and automated techniques can be seen as a particular application of the broader field of natural language processing (NLP). The introduction of transformer-based neural network architectures in 2017 led to an explosion of new AI models for NLP with increasing capabilities. BERT (mentioned above) was developed shortly thereafter (2018), with multiple related models (e.g., RoBERTa) being further developed over the last five years, with effectiveness at various NLP tasks that often exceeded those of pre-transformer models. Such models have been applied to a wide range of tasks, both with fine-tuning and without.

Large language models are neural networks based on transformer architectures, including not only those in the BERT lineage but also other models such as GPT-2, GPT-3, T5, and many others, with tremendous scale in terms of the number of model parameters (billions and sometimes trillions) and the internet scale volume of text on which they are trained (billions or even trillions of tokens, with tokens being numerical representations of words or parts of words). BERT (the large variant) has approximately 345 million parameters and was trained on about 3.3 billion words; in comparison, GPT-3 has 175 billion

parameters and was trained on approximately 500 billion tokens (approximately 375 billion words). Many of the newer models have generative AI capabilities, with the ability to do tasks like summarization, translation, and generation of high-quality text output. As their scale has grown, the range of tasks of which they have shown to be capable has increased, along with a level of performance that has surprised many. With the recent popularization and wider spread availability of LLMs, in part due to ChatGPT, with its underlying GPT-3.5 and GPT-4 models, as well as other LLMs like Claude (Anthropic), Command (Cohere), Bard (Google), LLaMA (Meta), and a range of open-source models, interest has grown in applying these to use cases like analysis of short text comments such as are seen in Tweets [23], customer feedback, and education survey feedback [24, 25].

Multiple recent studies have examined using ChatGPT for text annotation and classification tasks, with mixed results based on variations in prompts, datasets, parameters, and complexity of tasks. Reiss [26] focused on sensitivity to the prompts and parameters used in classification, in the context of performing classification on a German dataset. Pangakis et al. [27] argues that researchers using LLMs for annotation must validate against human annotation to show that LLMs are effective for particular tasks and types of datasets, given that there is variation in the quality of prompts, the complexity of the data, and the difficulty of the tasks. Other studies ([28, 29]) demonstrate the potential for ChatGPT to perform text annotation or provide natural language explanations at levels approaching or matching those of humans.

1.4 Research Significance and Objectives

Exploration of the use cases for LLMs is in its relative infancy, and education is an important area of focus for LLM applications. A primary focus of recent related research has been on direct use of LLMs in teaching and learning, with less exploration of the capabilities in education feedback analysis. Education feedback surveys are a valuable source of information for evaluation and iterative improvement of course experiences, but remain difficult to process in a data-driven fashion, in part due to the manual labor associated with conventional analysis of the unstructured (text) responses component. Machine-learning approaches have shown promise in aiding analysis, but often require conditions that make their use less feasible to most educators, such as the need for fine-tuning and use of separate models for the natural language processing (NLP) tasks involved.

In this context, we:

- demonstrate a versatile approach that uses an LLM to perform multiple unstructured text analysis tasks on survey responses, including multi-label classification, multi-class classification, binary classification, extraction, inductive thematic analysis, and sentiment analysis.
- evaluate performance in a zero-shot approach across all tasks, a scenario that mimics many real-world practical use cases in the education setting.
- show the potential of LLMs to offer a form of insight into the trajectory (“reasoning”) of how they arrive at their answers, providing a degree of transparency that may help foster confidence in real world usage.

As part of the evaluation process, we also developed a set of classification categories that can be applied to a variety of course types (online, hybrid, or in-person), and which are amenable to customization depending on specific requirements.

2 Methodology

2.1 Survey data used for evaluation

2500 survey responses were selected at random from a larger set of survey responses received as end-of-course feedback on a range of biomedical science courses, including courses on genetics, immunology, and pharmacology. Additional survey comments were chosen as a development set that could be used for LLM prompt tuning. The courses all use a single, uniform end-of-course survey. In addition to quantitative ratings (e.g., net promoter scores) and optional demographic data, the survey included open-ended text responses to four questions/directives:

- “Please describe the best parts of this course.” [Q1]
- “What parts of the experience enhanced your learning of the concepts most?” [Q2]
- “What can we do to improve this course?” [Q3]
- “Please provide any further suggestions, comments, or ideas you have for this series.” [Q4]

On average, learners answered approximately two of the four questions. The shortest responses containing content were one word, and the longest responses were several paragraphs. Example survey responses are shown in Table 2.

Table 2 Example actual survey responses.

Q1 responses	Q2 responses	Q3 responses	Q4 responses
“The teachers they are incredible and their fascination about this topic make it more interesting.”	“the whole concept, the short videos with the explanations written down and then the interactive modules”	“Implement more checkpoints that review previous material throughout the course.”	“I really enjoyed the course and learned a lot of applicable information for my job. I would have like a little more time between new releases of information. It would also be nice to have a live question/answer session.”
“the structure of this course is just great. however i would love to have the chance to repeat all the modules as i am from a very different background.”	“The quizzes after each module made me think about the material I just learned.”	“The course was fantastic and informative. However, I had to rewatch the videos several times to write down everything that is said. I learn best by looking at the words. The videos should come with either a transcript or written words or some sort that convey the same information”	“A visit to meet the tutors and a summary discussion on location would be fabulous - I am aware not many people would make it, but a thought nonetheless.”

Survey responses were collected via Qualtrics, with minor processing with Pandas 2.0.1 for elimination of leading and trailing white space and automated removal of responses with no content (NA or None variants).

Survey responses were inspected manually and via named entity recognition (NER), running locally, to ensure that no private or sensitive information was transmitted to publicly available LLMs.

2.2 Development of course tagging system

The authors spent considerable time developing and testing a set of labels that would work well not only for online courses like those that the survey responses in this paper were a part of, but also other types of educational offerings. The label development process started with a much larger set of labels (71 total), based on the goals of those involved in course production and delivery. Given that each survey response could cover multiple topics, the task was to assign as many labels to each survey response as were applicable (a multi-label classification task). The four authors (all of whom have been involved in either course development or delivery for multiple years and can be considered domain experts in the course resources and processes) each labeled a test set of 2000 survey responses (from the same educational program overall, but distinct from the set of 2500 comments ultimately labeled), with resulting relatively low inter-rater agreement. Based on this experiment, tag categories were combined to arrive at a much smaller set of generalizable tags (see Table 3). In addition, best practices were followed to ensure generalizability [1, 30–32].

A one to three sentence description of each tag was created to provide guidance so that tags could be applied appropriately in testing rounds. The intent is also that others can adapt these same tags by modifying the description portion for their own purposes. The same descriptions that served as context for the human annotators were also used in the prompts for the LLMs in the multi-label classification task as a form of deductive thematic analysis.

We then iteratively tested the new, much smaller set of tags on several sets of 100 survey responses, with all four authors independently tagging the same entries, followed by examination of inter-rater agreement. This yielded good results. With this set of tags, we then independently labeled 2500 survey responses, and evaluated inter-rater agreement using Jaccard similarity coefficient between pairs of raters and averaged across all pairs of raters.

2.3 LLM processing

All LLM tasks were performed via calls to the OpenAI API endpoints. GPT-3.5 (model: gpt-3.5-turbo-0301) and GPT-4 (model: gpt-4-0314) were used for the multi-label classification task; all other tasks described used GPT-3.5 (model: gpt-3.5-turbo-0613) and GPT-4 (model: gpt-4-0613). All tests used a temperature of 0 with other parameters set to their default values, other than the functions parameter and the function_call parameter, which were set to specify the applicable function schema and the function name where applicable. Tests were run with calls to the models’ asynchronous endpoints, in

Table 3 Final tags and descriptions.

Tag	Description
course logistics and fit	course delivery (policy, support), cost, difficulty, time commitment, grading, credit, schedule, user fit, access, background (e.g., prereqs and appropriateness of course level).
curriculum	course content, curriculum, specific topics, course structure. This focuses on the content and the pedagogical structure of the content, including flow and organization. This also includes applied material such as clinical cases and case studies. Includes references to pre-recorded discussions between experts or between a doctor and a patient. Includes specific suggestions for additional courses or content.
teaching modality	video, visual, interactive, animation, step-by-step, deep dive, background builder (the format rather than the content/topic).
teaching	instructors, quality of teaching and explanations
assessment	quizzes, exams
resources	note taking tools, study guides, notepads, readings. Includes other potential static resources like downloadable video transcripts.
peer and teacher interaction	includes chances for the student to interact with another person in the course (teacher or student). This includes discussion forums, teacher-student or student-student interactions. Includes requests for live sessions with teachers or live office hours.
other	catch-all for the rarer aspects that we’ll encounter and also the ‘na’, ‘thank you’, etc. comments that don’t really belong in the above bins. Also for sufficiently general comments like ‘all the course was terrific’ that can’t be narrowed down to one of the other categories.

order to run many model calls in parallel for suitable tasks (e.g., classification of individual survey responses). “Function calling”, a capability specific to these models, was used to generate the JSON structured output for all tasks. Comments were run in batches that fit within the rate limits (tokens per minute) of each model. Prompts used (see Appendix A) involve function schemas, which count in the context limits, as well as the system and user messages to the model.

For the LLM approach to the multi-label classification task, the multi-class classification task for extracted excerpts, the binary classification task, and the sentiment analysis task, zero-shot chain-of-thought (CoT) prompting was used (where a model is prompted to reason step-by-step but without examples of such reasoning provided) [33, 34]. In addition to use of CoT enhancing the accuracy of the model output, the reasoning was included in the output to allow for error analysis and prompt tuning, as well as to allow inspection of the model’s reasoning, something potentially helpful for those using the results in practice. For sentiment analysis, we had the LLM output a sentiment classification based on the possible categories ‘negative’, ‘slightly negative’, ‘neutral’, ‘slightly positive’, and ‘positive’, along with its reasoning.

For the LLM approach to inductive thematic analysis of survey responses, a two-step approach was used. The first step involved prompting the LLM to derive themes representing feedback from multiple students and summarize the themes. This step was run in parallel on batches of survey responses that would fit within the model’s context window. The second step involved prompting the LLM to coalesce derived themes based

on similarity to arrive at a final set of themes and descriptions. These steps could be considered analogous to part of the human inductive thematic analysis qualitative analysis workflow [35].

Various prompting techniques were used in this study to improve the results. These include:

1. Zero-shot CoT - This technique involves asking the model to think step-by-step to arrive at a correct result and to provide its detailed reasoning. In the absence of providing examples of CoT reasoning in the prompt, this type of prompting is categorized as zero-shot.
2. Prompt tuning via inspection of CoT reasoning - In testing, error analysis was supplemented with inspection of CoT reasoning to help discern where prompts might need refinement. As prompts were updated, we observed corresponding changes in the output and the stated reasoning, with improvement in the development set metrics.
3. Additional descriptive context for labels - Given that there was no fine-tuning to allow the model to learn the appropriate context and meaning of labels, we added context to prompts in the form of definitions for each label and the types of elements for which each label applied.
4. Additional context through injection of the survey questions into the prompt - Inclusion of additional context, such as the survey question that a given comment is in reply to, may improve the performance of LLMs and was used in this study.
5. Use of function calling for reliable structured output - This technique is specific to the GPT-3.5 and GPT-4 models, for which the June 2023 checkpoint (0613) has been fine-tuned to enable structured output (e.g., JSON) when provided with information about a function schema that could be called with the output. For this study, in which thousands of rows of data were processed into structured output, the function calling capability vastly reduced the need for elaborate prompting to elicit structured output, as well as error-handling and parsing of variations in output formatting. We started this project well before the models had been fine-tuned for structured output and saw the benefits of greater reliability once these capabilities existed.
6. Memetic proxy, also known as the persona pattern [36, 37] - Asking the LLM to act as a certain persona, for example as an expert in survey analysis tasks, has been described as another way to improve results, potentially by helping the model access a portion of its memory that holds higher quality examples of the task at hand. Guiding the model to imitate correct examples is more likely to result in good answers than asking the model simply to produce results.

2.4 Other models

In addition to comparison to human ground truth labels, for multi-label classification, comparison was made to SetFit [38], a SentenceTransformers finetuning approach based on Sentence-BERT and requiring very little labeled data; for sentiment analysis, comparison was made to a publicly available RoBERTa-based model trained on 124M Tweets. These comparisons provide some context for the LLMs’ performance relative to recent specialized models.

2.5 Evaluation metrics

Scikit-learn 1.2.0 was used for statistical tests, along with numpy 1.23.5 and Pandas 2.0.1 for data analysis. Weights & Biases was used for tracking of model evaluation results. For the multi-label classification task, model results were compared to the human ground

truth labels. Two ways were used to arrive at ground truth labels aggregating results from multiple annotators: 1) using consensus rows: only the subset of survey responses (dataset rows) where all 4 annotators had majority agreement on all selected tags were kept; and 2) using consensus labels: all survey responses were kept but only labels with majority agreement were chosen as selected.

To fine-tune the SetFit model, we used a portion of each ground truth dataset (the first 20 examples for each label). Those examples were omitted from the test set, leaving 2359 rows in the consensus labels test set and 1489 rows consensus rows test set.

For each of the above scenarios, model results for multi-label classification were evaluated against aggregated human annotator results via the following metrics: 1) Jaccard similarity coefficient, comparing the model against aggregated human results for each row (survey response) and then averaged over all rows; 2) average precision per tag; 3) average recall per tag; 4) macro average precision, recall, and F1 score across all tags; 5) micro average precision, recall, and F1 score across all tags; 6) Hamming loss; and 7) subset accuracy.

For the binary classification task, accuracy, precision, recall, and F1 score were calculated, comparing the model results to one expert human annotator.

For the extraction task, extracted excerpts were evaluated by GPT-4 using a rubric created specifically for this task, examining performance on multiple aspects of performance, including the presence of excerpts that were not exact quotes from the original (part of the original extraction instructions), the completeness of capturing relevant excerpts, the presence of excerpts irrelevant to the initial goal focus, the inclusion of relevant context from the original comment, and several others. The results were also evaluated by human annotation to determine the presence of hallucinations (excerpts that were substantial changes from the original survey responses, rather than just changes in punctuation, spelling, or capitalization), with the percent of the total number of excerpts representing hallucinations being reported.

For the inductive thematic analysis task, there is not an accepted evaluation method given that this is a complex, compound task, and evaluation consisted of inspecting the derived themes and descriptions as well as inspecting the results of the associated multi-label classification step.

The sentiment analysis results of GPT-3.5 and GPT-4 were compared to those of a RoBERTa sentiment classifier trained on 124 million tweets [39, 40], as well as to results from a human annotator, with accuracy, precision, recall, and F1 scores reported for the prediction of sentiment as negative, neutral, or positive. Comparison was made by grouping ‘negative’ and ‘slightly negative’ into a single class, keeping ‘neutral’ as its own class, and grouping ‘positive’ and ‘slightly positive’ into a single class to allow for comparison across sentiment analysis methods. The RoBERTa classifier produced a dictionary with negative, neutral, and positive classes, with probabilities summing to 1.0. The class with the maximum probability score was chosen as the label for comparison to the human annotations.

3 Results

In this section, we first provide an overview of the rationale for using specific NLP tasks to accomplish different types of survey analysis goals, in order to provide motivation for the workflows that follow. We then demonstrate examples of potential workflows, applied to our real-world dataset, followed by presentation of examples of one model’s chain-of-thought reasoning, and finally show evaluations of the individual tasks involved in the workflows. For the examples, we use GPT-4 as the LLM; the evaluations compare GPT-4 and GPT-3.5 as well as the other models used.

3.1 Approach to LLM Workflows

The main types of workflows demonstrated support the goals shown in Table 1 of 1) high-level analysis, in which the desire is to understand the main categories and areas of emphasis across all student feedback, or 2) more focused analysis, e.g., answering specific questions about a particular aspect of a course. In both cases, quantification of results is a consideration, which is supported by classification tasks.

For initial, high-level analysis across the entire set of survey comments, we demonstrate two approaches: 1) inductive thematic analysis, a “bottom-up” approach supporting the use case where no predetermined labels (areas of interest) have been defined, similar to topic modeling, and 2) multi-label classification using predefined labels, a “top-down” approach, also referred to as deductive thematic analysis. When categories of interest are known in advance, multi-label classification is an appropriate first step, binning survey responses into relevant categories that provide a sense of the type of feedback learners are providing. These categories also provide groupings of comments for further focused analysis (e.g., via extraction), as well as allow for quantification based on the number of comments labeled with each category.

For focused analysis, in which there is a specific question or goal for the analysis, not necessarily known in advance, we demonstrate extraction as a key step, followed by either a classification step or thematic analysis. To provide output for further downstream analysis and quantification, multi-class classification can be used as a step, as demonstrated here with the generalizable set of labels used in this study, or with an adapted or fully customized version for one’s own use case. This step is shown used after extraction, given that short excerpts are more likely to be adequately classified with a single label versus multi-sentence comments. The output of other forms of classification (binary or multi-label) also lends itself well to quantification of results.

Sentiment analysis was applied as a final step for workflows where finding positive or negative excerpts was of interest, as demonstrated in the example related to the level of difficulty of the course.

Although the full model responses were in JSON format, only the relevant output text is shown for brevity and clarity.

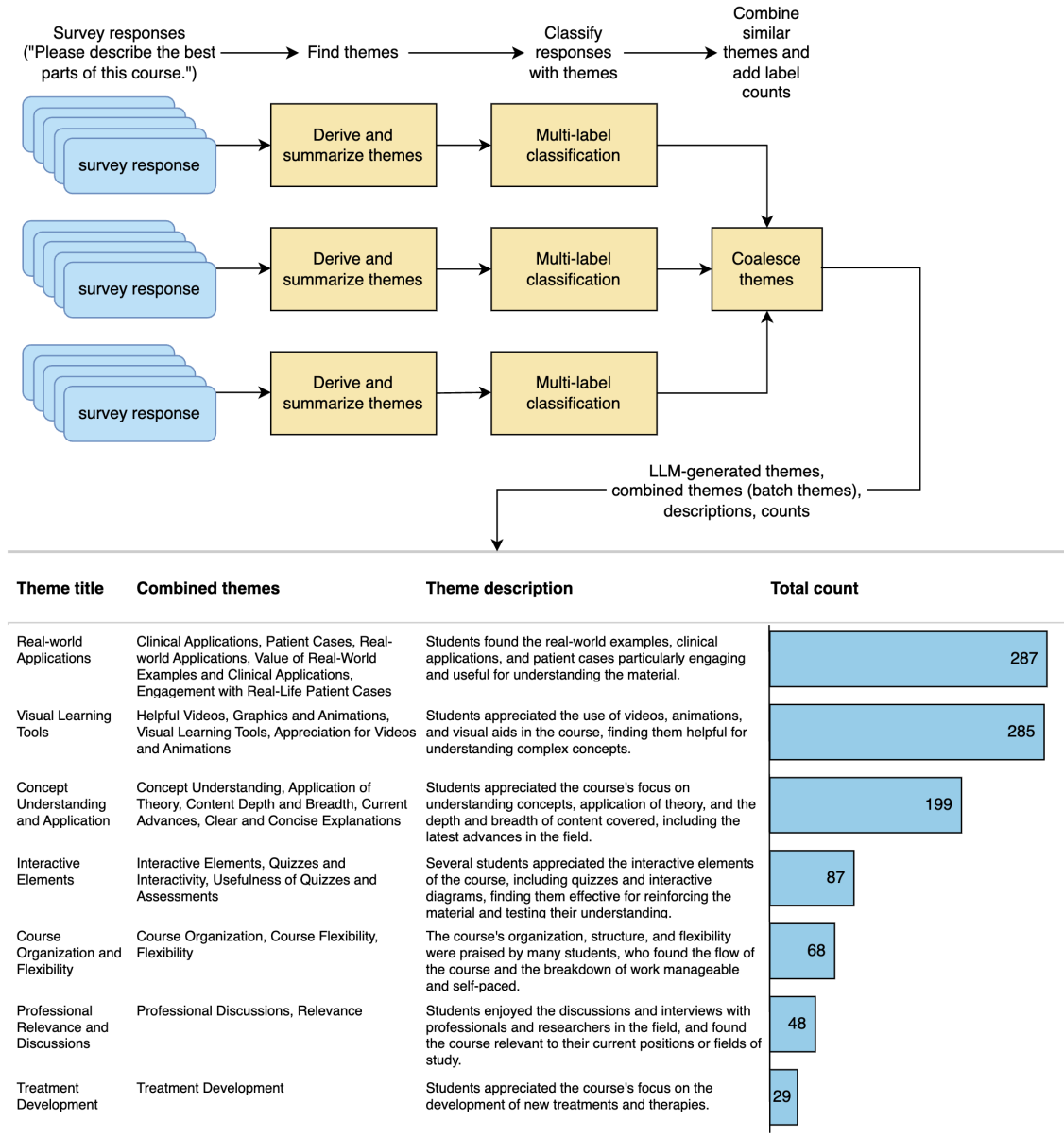
3.2 Workflow Examples

3.2.1 Example - High-level analysis by inductive thematic analysis (“bottom-up” approach)

A workflow for finding and summarizing the main themes (ideas expressed by multiple students) of survey responses is shown in Figure 1, and consists of three LLM steps: 1) themes are first derived and summarized for batches of comments, each of which is sized to fit within the context window of the model used; 2) comments are classified using the derived themes; and 3) sets of themes from these batches are coalesced to arrive at a final set of themes. Additionally, label counts are aggregated from the themes that were combined. In qualitative research, steps 1 and 3 are called inductive thematic analysis; this is similar to topic modeling, in that themes are inductively derived from comments. In general, depending on the input size (context window) for the model used (8K tokens in this example) and the number of comments being analyzed, dividing into batches and coalescing the themes from each batch may be unnecessary.

Results for running this process on the 625 comments from Q1 (‘Please describe the best parts of this course’) are shown in Figure 1. The number of comments that the LLM identified as corresponding to each theme is shown, along with the theme titles and descriptions.

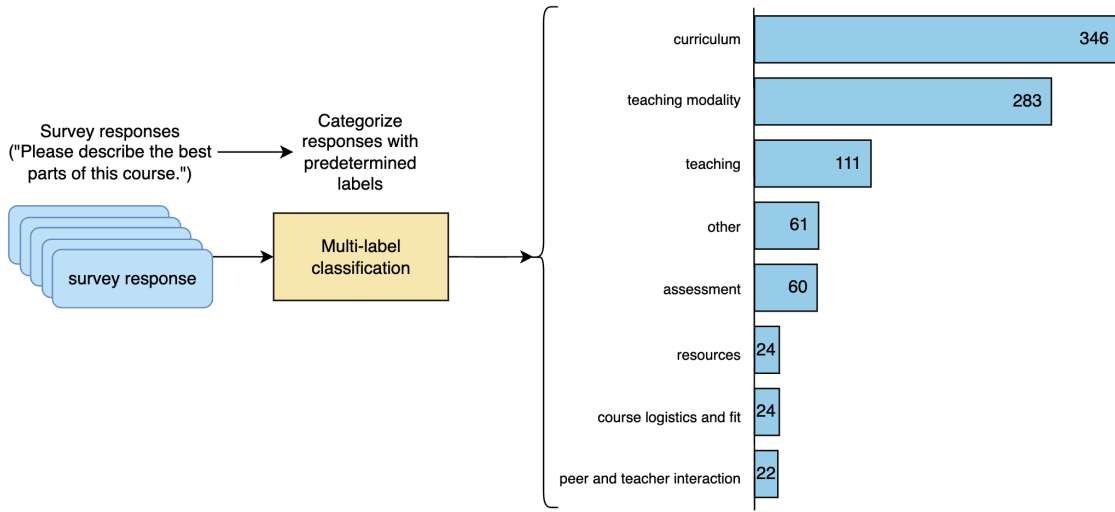
Fig. 1 Derivation of themes from student comments (results shown using GPT-4).



3.2.2 Example - High-level analysis by categorizing student comments (“top-down” approach)

Multi-label classification of survey responses, using the set of predetermined labels developed for this study (Table 3) was run on the 625 comments from Q1 (‘Please describe the best parts of this course’) and results are shown in Figure 2. The categorized comments can be used for analysis (for example, comparing the categorization of responses to ‘Please describe the best parts of this course’ to the categorization of responses to ‘What can we do to improve this course?’) or as a starting point for further downstream tasks.

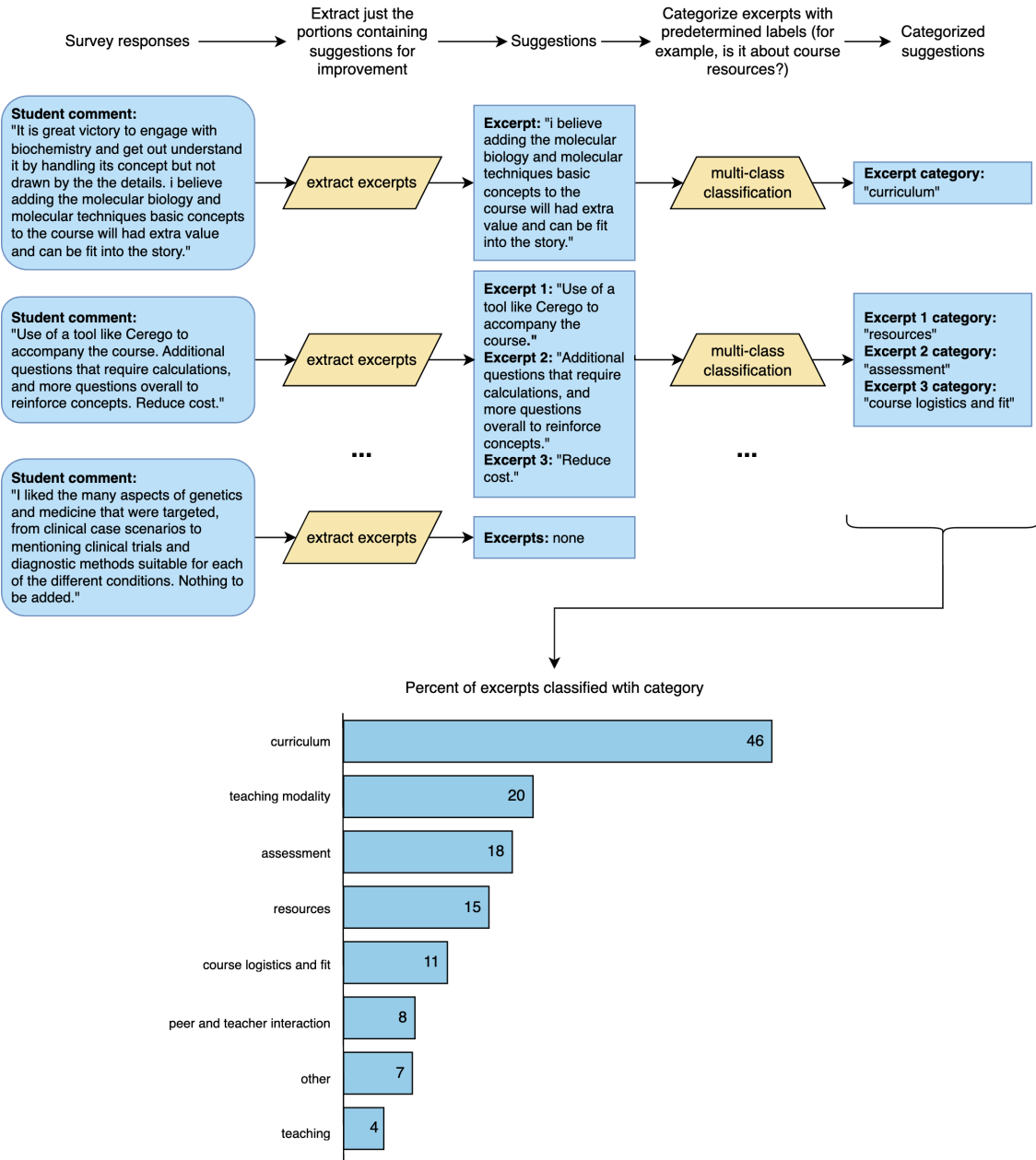
Fig. 2 Multi-label classification of student comments (results shown using GPT-4).



3.2.3 Example - Finding suggestions for improvement

A workflow for finding and quantifying suggestions for course improvement is shown in Figure 3, and consists of extraction of relevant excerpts, followed by multi-class classification, based on the labels in Table 3, to facilitate quantification as well as routing of comments to the appropriate stakeholders. Excerpts resulting from the extraction step were assumed to be focused enough that they could each be categorized with a single class from among the pre-existing labels in Table 3. Results for several representative real comments from the larger set of survey comments are shown in Figure 3. The model’s CoT reasoning for each step is shown elsewhere, but is omitted here for clarity.

Fig. 3 Finding suggestions for improvement from student comments (results shown using GPT-4).



3.2.4 Example - What other content or topics were students interested in seeing covered?

A common goal in analyzing student feedback is to better understand the gaps in course content, in order to decide whether to develop additional material or even new courses. To see if this type of information could reliably be derived from survey responses, we focused on responses to relevant survey questions (Q3 and Q4) for immunology courses with the workflow shown in Figure 4. Results for several representative real comments are shown. First, just the portions containing new content or topic area suggestions are extracted from the survey responses. Content suggestion themes are then derived and summarized from the excerpts; this is done in batches if they cannot be fit within a single prompt to the LLM (i.e., if there are too many excerpts to fit in the model’s maximum context size). Multi-class classification is performed on the excerpts with the themes from each batch. If thematic analysis is done in batches, sets of themes from these batches are then coalesced to arrive at a final set of content themes. The results suggest that GPT-4 is capable of finding content suggestions despite many being specific to the biomedical domain. This may be due to the volume and diversity of the model’s pre-training data (although this training mixture has not been disclosed). Immunology is used as an example, but the workflow is not specific to the type of course.

3.2.5 Example - What feedback did students give about the teaching and explanations?

Feedback about teachers and the quality of teaching and explanations in a course is a frequent objective of academic course surveys. Here, we show a workflow where multi-label classification has already been run as an initial step in high-level analysis, and we use the results of that classification as our initial filter to focus on the identified subset of comments related to teaching (corresponding directly to one of the pre-existing labels), with extraction used to further narrow the output of analysis. The workflow, shown in Figure 5, consisted of multi-label classification, using the pre-existing labels developed (Table 3) followed by extraction of relevant excerpts from the comments that were classified into the ‘teaching’ category (9% of total comments). If multi-label classification hadn’t previously been run, extraction could have been performed on the broader group of comments as the initial step. For our dataset, which includes numerous multi-topic comments, the extraction step was used to further filter the information to only content related to the goal. Results for several representative real comments (de-identified in pre-processing) from the larger set of survey comments are shown in Figure 5, including one where the model improperly filtered out the comment despite it containing a reference to the quality of explanations. An error such as the one shown could be considered somewhat subtle and highlights the need with zero-shot prompting of LLMs for clear specification of the goal of the extraction.

Fig. 4 Finding suggestions for new immunology content from student comments (results shown using GPT-4).

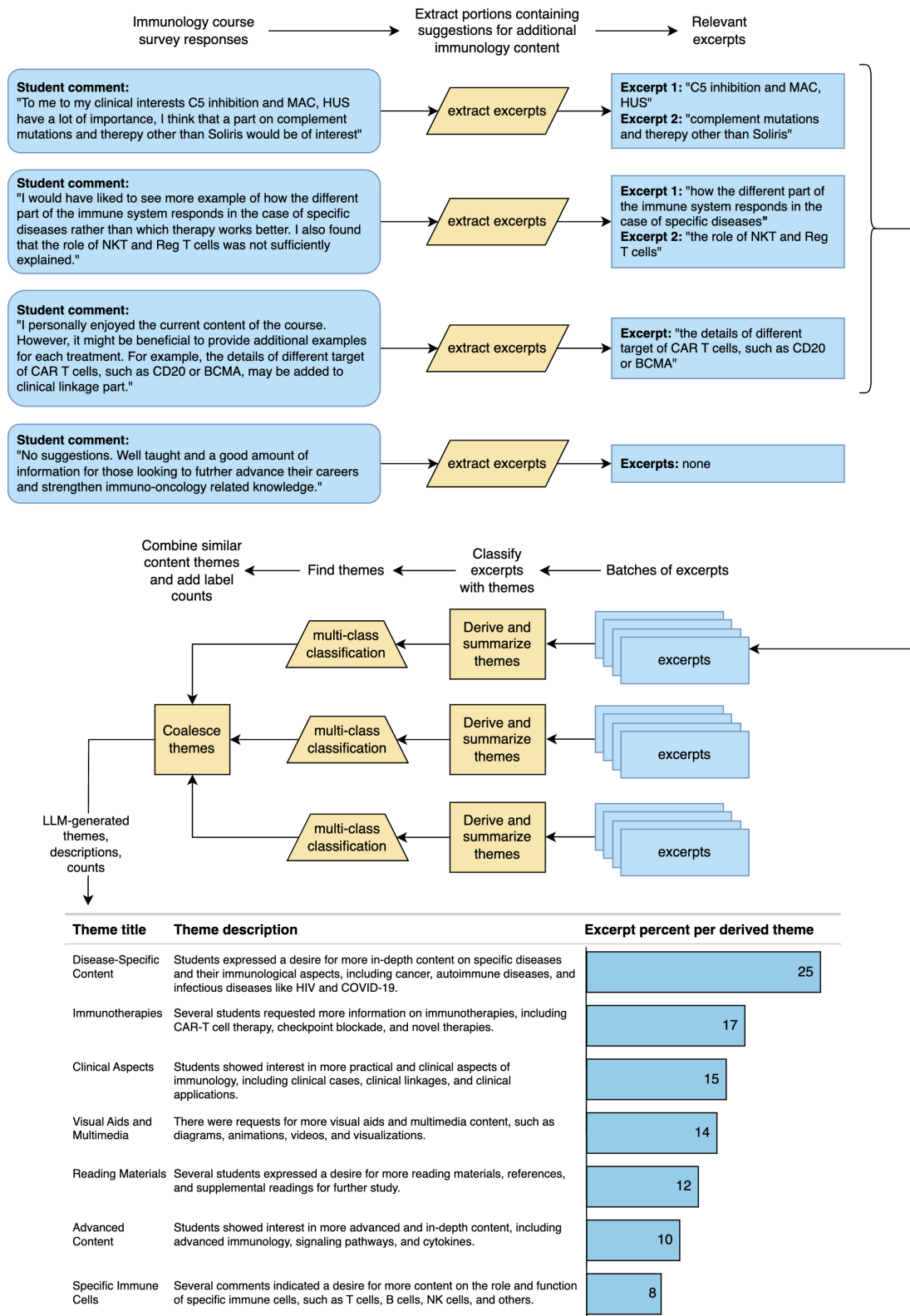
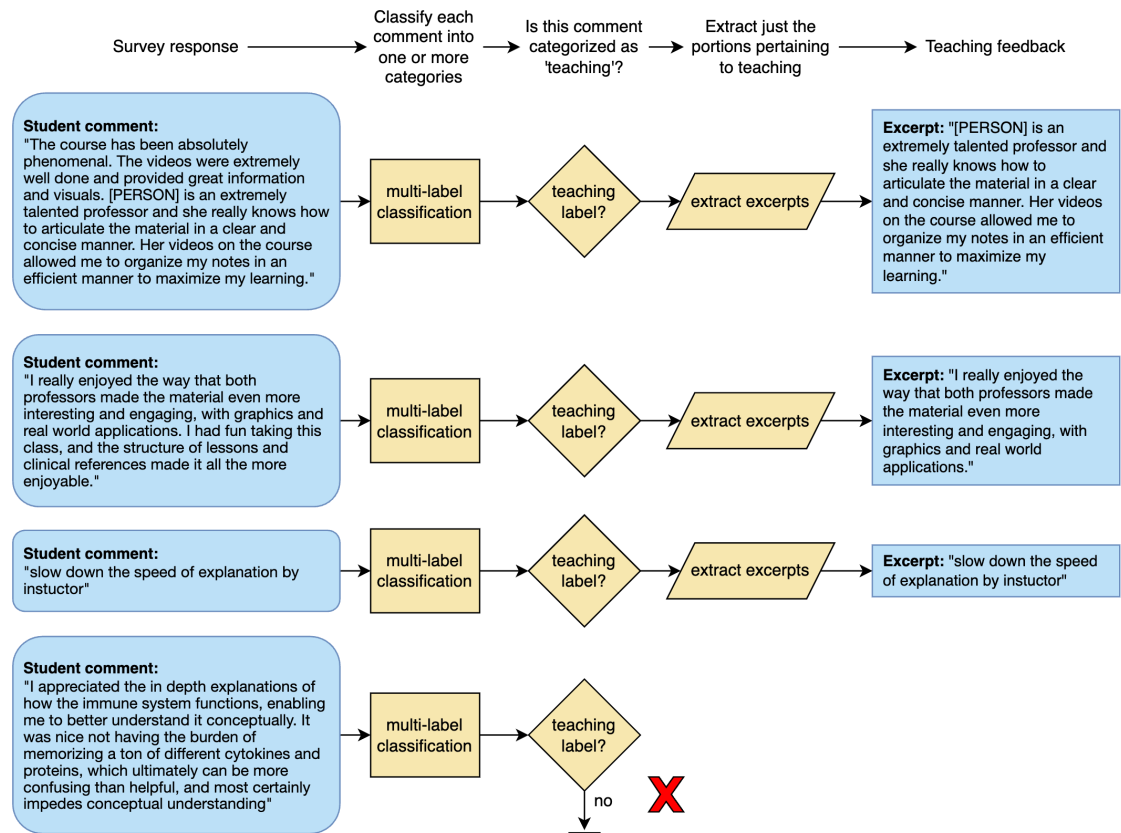


Fig. 5 Feedback about teaching and explanations (results shown using GPT-4). The red 'x' indicates an error by the model.



3.2.6 Example - How did students feel about the level of difficulty of the course?

Feedback about the level of difficulty of a course can help guide decisions on prerequisites and messaging about the intended target audience. Again, we show a workflow where multi-label classification, using the pre-existing labels developed (Table 3), has been run as an initial step in high-level analysis, and we use the results of that classification as an initial filter. In this case, the desired goal (level of difficulty) falls within the ‘course logistics and fit’ label but is not an exact match. As shown in Figure 6, after filtering to comments that were classified as ‘course logistics and fit’, a further binary classification step was applied to filter only to comments containing passages about the level of difficulty of the course; the binary classification step was optional, but significantly reduced the number of comments that needed to be processed with the more complex extraction task. Finally, extraction of the comment passages about level of difficulty and classification of sentiment were applied. Results for several representative real comments are shown in Figure 6. The results of the evaluating the sentiment analysis portion (see Section 3.4.4) suggest that sentiment analysis can be a challenging zero-shot task in areas such as biomedical online learning where the course context, the feedback context, and the inclusion of multiple topics in a single survey comment may differ significantly from the model’s public pre-training data distribution.

3.3 Chain-of-Thought Reasoning

The prompts for binary classification, multi-label classification, multi-class classification, sentiment analysis, and evaluation of extraction results all used zero-shot chain-of-thought (CoT) to enhance the quality of the results while maintaining the zero-shot conditions of this study. The CoT reasoning was included in the structured output, allowing for inspection. Only the reasoning from GPT-4 was consistently reliable, and examples are shown here.

Example results for binary and multi-class classification tasks are shown in Figure 7 and Figure 8, and reasoning for sentiment analysis is also shown in Figure 8. The reasoning, inspected manually over several hundred comments, is consistent with the classification results and appears to provide logical justification that is grounded in the contextual information (e.g., labels and descriptions) included as part of the prompts (see Appendix). This suggests that the CoT reasoning from GPT-4 meets a threshold of consistency and logic that allows for potential downstream use cases such as prompt tuning and insight into reasoning for end-users. Potential benefits and caveats of such uses are explored in the Discussion.

Figures 7 and 8 show the model’s CoT reasoning related to Example 3.2.3 (suggestions for improvement) and Example 3.2.6 (level of difficulty of the course) above.

Evaluation of the extraction task used a custom LLM evaluation (see Appendix), developed for this study. In order to refine the evaluation to align results with human preferences, we inspected the CoT reasoning along with the structured eval results for the separate development set of survey responses and made modifications to the evaluation prompts in an iterative fashion. An example of the CoT output for GPT-4 is shown in

Fig. 6 Finding feedback about level of difficulty (LLM: GPT-4).

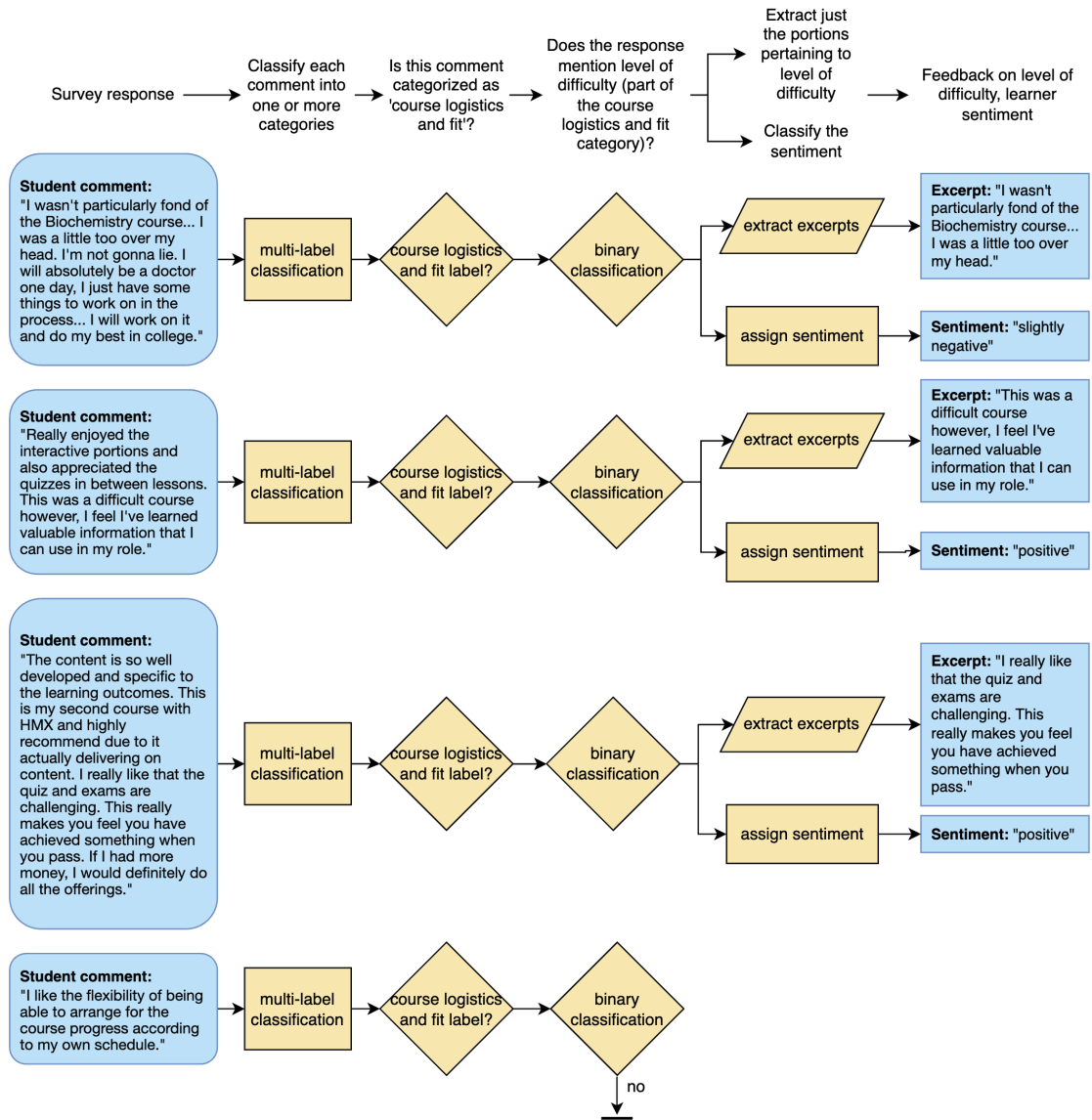


Figure 9. As prompts were altered based on human review, the eval results changed in a consistent fashion, suggesting that GPT-4 provided CoT reasoning may be useful in refining LLM evaluations.

Fig. 7 Examples of GPT-4 CoT reasoning for binary classification and multi-class classification related to the task of finding suggestions for improvement.

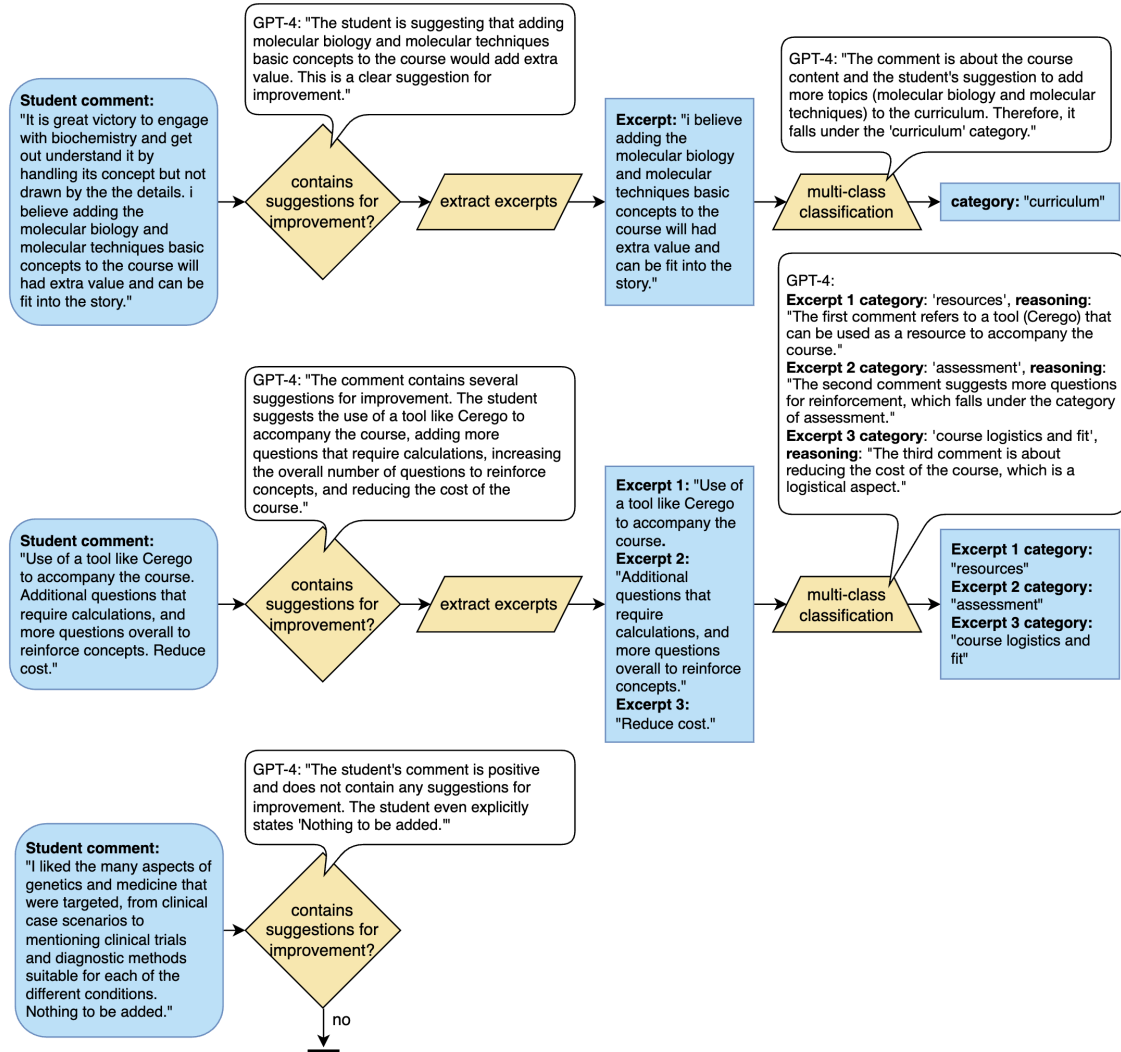


Fig. 8 Examples of GPT-4 CoT reasoning for multi-label classification, binary classification, and sentiment analysis related to the task of finding how students felt about the level of difficulty of the course.

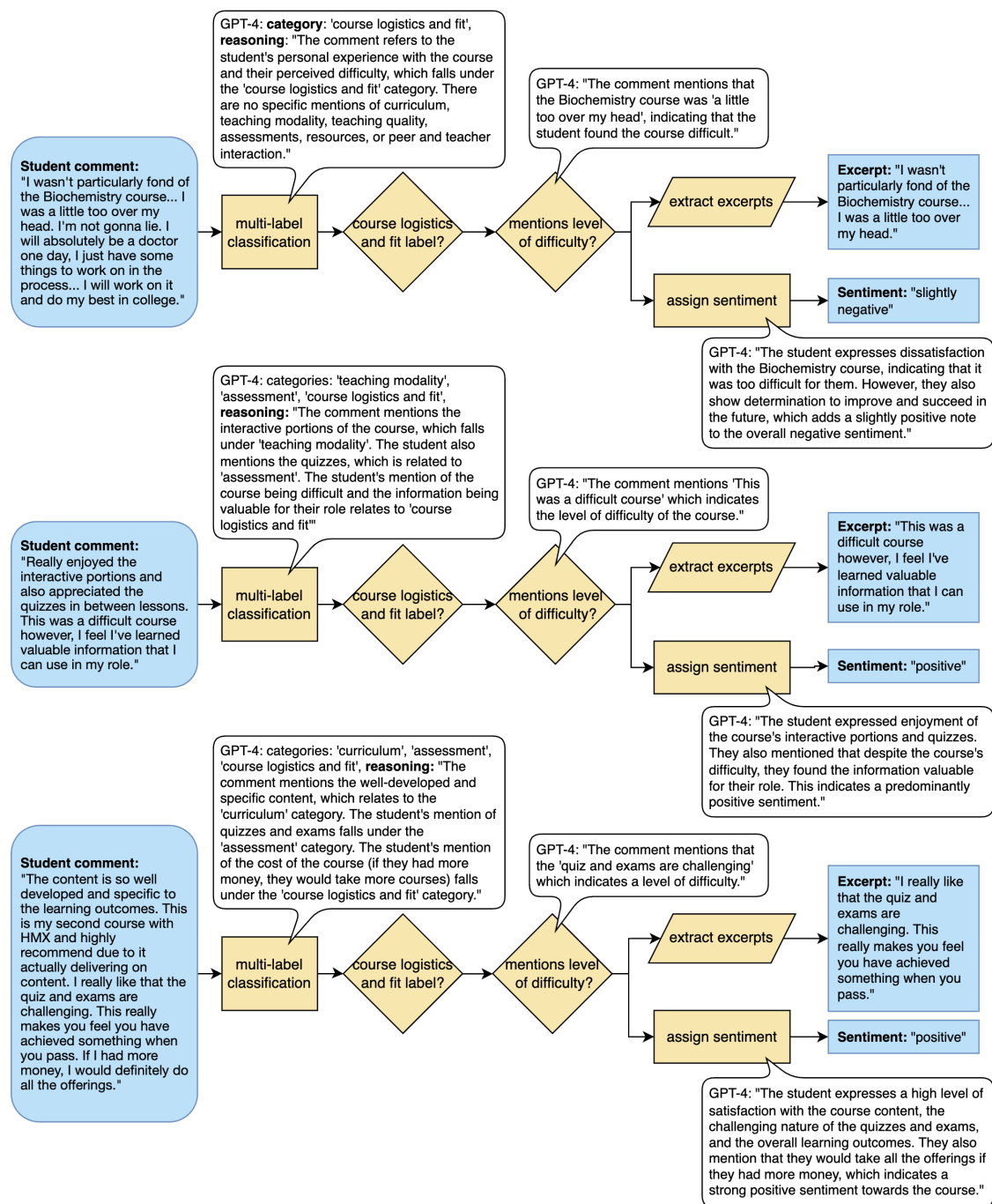
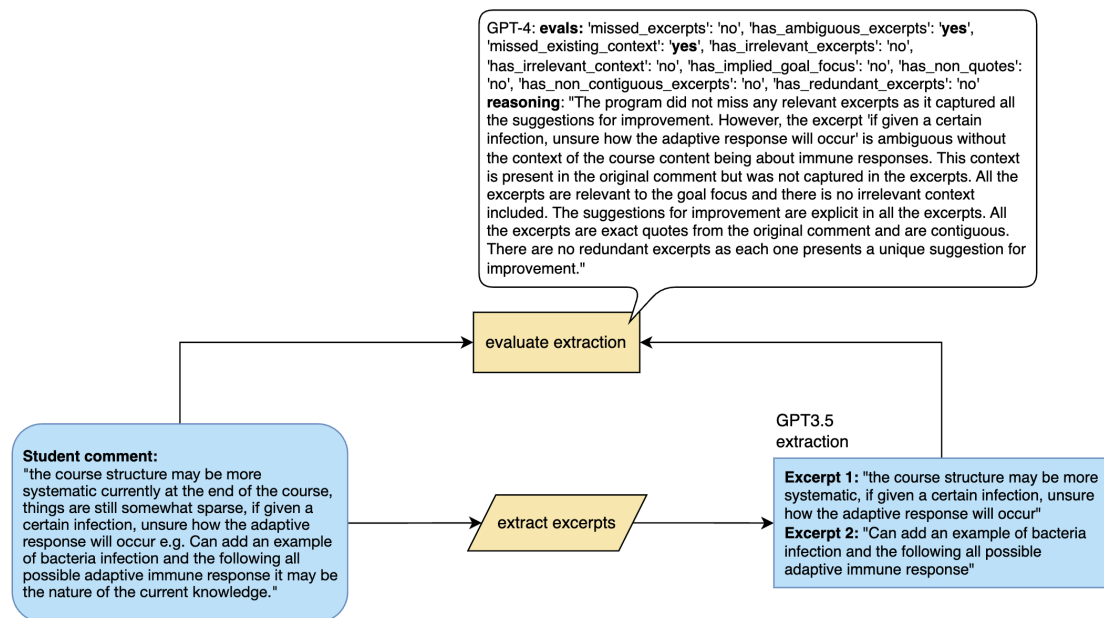


Fig. 9 Example of GPT CoT reasoning for extraction evaluation.



3.4 Individual NLP Task Evaluations

To better assess the reliability of workflows such as those shown in the examples, we evaluated the individual tasks, including multi-label classification, binary classification, extraction, and sentiment analysis.

3.4.1 Multi-label classification metrics

The difficulty of multi-label classification tasks varies widely [15], depending on the content to which the labels are being applied, the design of the labels (for example, the clarity of their specification and the potential for overlap), and the number of labels. To put the LLM results in context, we show the inter-rater agreement for application of the eight-label set (Table 3) to our dataset and also compare the LLM results to SetFit, another classification technique.

Inter-rater agreement: 1413 (57%) of 2500 rows had all 4 human raters in agreement across all selected labels and 1572 (63%) had majority (3 of 4) agreement on all selected labels. The average Jaccard similarity coefficient including all 2500 rows (averaged across the six unique pairings of four human raters for all rows) was 81.24% (Table 4), suggesting that this was a challenging task even for expert human annotators who developed the custom label set in close collaboration. GPT-4 agreement with human annotators is shown; the average across all pairings including GPT-4 was 80.60%.

Table 4 Inter-rater Jaccard similarity coefficients, including human annotators and GPT-4 as another rater/annotator (human pairs average = 81.24%; all pairs average = 80.60%).

	annotator 1	annotator 2	GPT-4	annotator 3	annotator 4
annotator 1	-	81.27	80.18	83.37	82.35
annotator 2	81.27	-	79.40	80.84	78.42
GPT-4	80.18	79.40	-	80.74	78.22
annotator 3	83.37	80.84	80.74	-	81.18
annotator 4	82.35	78.42	78.22	81.18	-

LLM and SetFit evaluation: In addition to evaluating the GPT models, we also performed multi-label classification using SetFit (Tables 5 and 6).

Table 5 Evaluation on consensus rows, with majority agreement on all tags (1572 rows for LLMs, 1489 rows for SetFit).

Model	Jaccard	Average precision	Macro average			Micro average		
			Precision	Recall	F1	Precision	Recall	F1
GPT-4	92.97	93.91	89.88	90.59	89.78	93.66	93.26	93.46
GPT-3.5	72.61	74.79	69.34	82.18	72.63	72.36	84.48f	77.96
SetFit	73.86	78.01	84.37	57.59	66.85	91.92	71.43	80.39

For the consensus rows evaluation, the zero-shot results for GPT-4 are similar to what might be expected of fine-tuned classifiers [15]. The other models have strengths and

Table 6 Evaluation on all rows using consensus labels (2500 rows for LLMs, 2359 rows for SetFit).

Model	Jaccard	Average precision	Macro average			Micro average		
			Precision	Recall	F1	Precision	Recall	F1
GPT-4	80.17	81.53	73.91	88.38	79.69	78.32	89.70	83.63
GPT-3.5	63.00	65.18	60.42	79.79	65.75	60.31	83.45	70.02
SetFit	62.72	67.52	73.22	53.08	59.61	79.40	65.14	71.57

weaknesses, with SetFit having relatively high precision and lower recall, and GPT-3.5 following the converse pattern. The overall results for SetFit and GPT-3.5, focusing on Jaccard coefficient and F1 scores, are similar. The results emphasize 1) the fact that fine-tuning is desirable when feasible, approaching the performance of powerful LLMs like GPT-3.5 even with a few-shot fine-tuning approach; and 2) the quality of the zero-shot performance of GPT-4.

3.4.2 Binary classification metrics

1250 comments were classified as to whether or not they contained ‘suggestions for improvement’, and results were compared against one expert human annotator. Binary classification could be considered the simplest of the evaluated NLP tasks, and both LLM models exhibited good performance (Table 7).

Table 7 Binary classification task performance.

Model	Accuracy	Precision	Recall	F1
GPT-4	95.20	96.20	95.39	95.79
GPT-3.5	90.16	89.01	93.35	91.14

3.4.3 Extraction evaluation

Using ‘suggestions for improvement’ as an example target of extraction, comments were first classified via GPT-4 as containing the target or not (see binary classification task above). Of the 1250 comments, 716 were labeled as containing suggestions for improvement. These comments were then run through extraction to find the individual excerpts. The excerpts for each comment were scored by applying a custom evaluation rubric with nine questions (Table 24) via GPT-4. Only GPT-4 was capable of applying the evaluation reliably. The extracted excerpts were also examined by a human annotator to determine the percentage of the 716 rows that contained hallucinations in the excerpts, as defined by substantial edits or complete fabrication of additional language not present in the original comment. Table 8 shows the error rate across all target-containing comments for each model’s extraction results for all categories where error rates were not close to 0 and the percent hallucinations as determined by human annotation.

The GPT-4 model included some ambiguous excerpts; however, those were most commonly due to lack of context in the comment itself, rather than the model failing to

Table 8 Error rate (%) of extraction for ‘suggestions for improvement’ from comments classified as containing ‘suggestions for improvement’ (worst-performing metrics from rubric and human annotation for hallucinations).

Model	Missed Excerpts	Ambiguous Excerpts	Missed Existing Context	Irrelevant Excerpts	Implied Goal Focus	Non Quotes	Redundant Excerpts	Hallucinations
GPT-4	2.37	4.61	0.28	0.14	3.07	0.00	0.28	0.00
GPT-3.5	7.82	4.75	0.84	0.84	2.79	6.01	2.79	3.91

extract that context. GPT-4 followed directions very closely, and its results did not contain hallucinations. In contrast, the output of GPT-3.5 contained hallucinations at a rate of about 4% and edits to comments at a rate of about 6%. GPT-3.5 also missed relevant excerpts significantly more frequently than GPT-4. Additional prompt tuning may reduce the rate of these errors; nonetheless, the results suggest that a degree of caution should be applied in using GPT-3.5 for extraction.

3.4.4 Sentiment analysis metrics

Using GPT-4 and GPT-3.5, comments related to course suggestions and improvement (Q3 and Q4) were classified as ‘negative’, ‘slightly negative’, ‘neutral’, ‘slightly positive’, or ‘positive’. Table 9 shows accuracy, and macro precision, recall, and F1 scores for three models; comparison was made by grouping ‘negative’ and ‘slightly negative’ into a single negative class, keeping ‘neutral’ as its own class, and grouping ‘positive’ and ‘slightly positive’ into a single positive class.

Table 9 Classification of comments as negative, positive, or neutral relative to human annotator.

Model	Accuracy	Precision (macro)	Recall (macro)	F1 (macro)
GPT-4	80.86	82.65	80.28	80.78
GPT-3.5	65.17	73.68	66.44	64.88
twitter-roberta-base-sentiment-latest	66.69	71.38	64.86	61.10

GPT-4 is substantially better on each metric than the other models; however, the results are lower than what has been seen for fine-tuned models on in-domain datasets, indicating that the sentiment expressed in student course feedback may differ from the range of sentiment expressed in the internet training data of these models. The negative class was the most challenging for all models, suggesting that negative course feedback may differ significantly from negative internet feedback.

3.5 LLM Cost and Time

The cost of using the OpenAI APIs for GPT-4 and GPT-3.5 depends on the number of prompt tokens and number of completion tokens. For the final prompts and tasks used in this study, the average price of running 100 comments is shown in Table 10 for each model for different tasks (cost as of June 2023). These provide an approximate gauge given

that comments vary in length. Total API cost for this study including prompt tuning was approximately \$300.

Table 10 Cost per 100 comments for GPT-4 and GPT-3.5.

Task	GPT-4	GPT-3.5
binary classification	\$0.93	\$0.04
multi-label classification	\$2.63	\$0.12
multi-class classification	\$2.13	\$0.10
text extraction	\$1.10	\$0.05
text extraction evaluation	\$3.01	\$0.13
sentiment analysis	\$1.17	\$0.05
inductive thematic analysis	\$0.13	\$0.006

The time for model calls for GPT-4, the slower of the OpenAI models, was approximately 10 seconds for running 100 comments in parallel for most tasks listed. For the extraction evaluation, it took approximately 1 minute to run 100 comments in parallel. For batches, sleep intervals were also incorporated to stay conservatively within maximum token rates. A small percentage of API calls received errors and automatic retries were used after wait intervals.

4 Discussion

Analysis of education feedback, in the form of unstructured data from survey responses, is a staple for improvement of courses. However, this task can be time-consuming, costly, and imprecise, hampering the ability for educators and other stakeholders to make decisions based on insights from the data. The objective of this research was to demonstrate the capability of recent LLMs to perform a range of relevant natural language processing tasks that aid in this process, using a zero-shot approach that would be feasible for many educators, and to determine through evaluation whether the quality is acceptable for practical use in educational settings. Additionally, we proposed that chain-of-thought reasoning (CoT), used to improve the accuracy of results, can also potentially be useful for providing a degree of insight into the model’s stated logic for those using the results. Being able to peer into the “thinking” of the LLM may provide confidence, increase adoption, and reduce the perception of the models as “black box” algorithms.

This study’s evaluation of specific tasks with real-world data is not meant to be a benchmark or to show that performance exceeds fine-tuned or few-shot prompted models; rather its purpose was threefold: 1) demonstrate that the model results for the most capable models are viable for these types of use cases for education feedback surveys; 2) determine if there were steps that were particularly difficult for LLMs (weak links in the chain, as it were) that might benefit from specialized models or be amenable only to the best-performing LLMs; and 3) better understand the overall strengths and limitations of this approach.

Some of the aspects from our work that we discuss below are:

- the quality of the results compared to expert human annotators
- the need for prompting techniques and prompt tuning
- the tasks where few-shot prompting might have the highest impact
- the possibility of using CoT reasoning for purposes beyond enhancing model results
- the viability of composing LLM workflows from NLP tasks for the purpose of survey feedback analysis

4.1 Individual LLMs’ results varied significantly, and GPT-4 performed on par with expert humans even with zero-shot prompting

Our tasks and dataset were drawn from real-world data and actual use cases, motivated by common goals of those evaluating unstructured educational feedback. Some of the tasks could be considered challenging, with ambiguity involved, even for expert annotators. For example, as we gathered human ground truth data through annotating the sample dataset for multi-label classification, we found that inter-rater agreement metrics, shown previously, reflected the difficulty of the task. There were significant differences in the performance of models, with only the most capable model tested, GPT-4, reaching a level that was indistinguishable from any single expert human annotator in its multi-label classification results, based on Jaccard similarity coefficient (see Table 4). The human-like level of GPT-4 extended to other tasks and can be seen in examples of the reasoning results as well (Figures 7 and 9). In addition, it outperformed label-efficient fine-tuned classifiers like SetFit. For workflows that chain together two or more NLP tasks, like those examined in this study, it is important that the performance on each task is reliable enough such that errors do not accrue in the process of obtaining a final result.

4.2 LLM results were highly prompt-dependent

Even within the most capable models, we observed that prompting techniques and prompt tuning made a significant difference. There is considerable literature on effective methods of prompting. There is an interplay of prompting techniques with the behavior of instruction-tuned models in a way that may or may not fully elicit the capabilities of each model, with prompts being seen as a form of hyperparameter to the model and with responses changing depending on updates to model training [41].

Although this study focused on zero-shot prompting to reflect realistic use cases in educational settings, the results suggest that there are tasks that likely could benefit from few-shot prompting to reach a level necessary for inclusion in workflows. For sentiment analysis, the models’ zero-shot results differed from human annotation particular for negative comments. What constitutes negative feedback for an online course is subjective; for comments that are critical of certain course aspects but still adopt a civil tone, an educator may still choose to count those as negative. Therefore, providing examples in the prompt to help the model calibrate may be helpful, given potential differences between the data and the model’s internet pre-training. For the example of finding what other content or topics students were interested in seeing covered, even GPT-4 failed to distinguish suggestions for changes that were focal and focused on existing course content from those that were for new content or topics. In cases like this, where it might be difficult

to fully specify the objective sufficiently in abstract terms, providing few-shot examples could also be beneficial.

4.3 Inspecting a model’s chain-of-thought reasoning may have multiple uses

We have shown examples of GPT-4’s CoT results that provide apparent insight into the model’s reasoning or trajectory. We use the word “apparent” because it is possible that the LLM is imitating plausible reasoning rather than providing insight into how it actually arrived at its answer; however, this distinction may be immaterial given that 1) GPT-4’s reasoning was logical and highly consistent with the chosen label, excerpt, or extraction evaluation results, displaying elements of causal reasoning; and 2) when prompts were changed, reasoning results changed accordingly. This has been discussed in recent work; GPT-4 has been shown to score highly on evaluations of causal reasoning [42]. In Peng et al. [43], GPT-4 was used for evaluation of other LLMs and was able to provide consistent scores as well as detailed explanations for those scores. Whether or not the stated reasoning is actually how GPT-4 arrived at the answer, we observed that it was useful as an additional signal for prompt tuning in conjunction with metrics and for providing a logical justification for each response. While specialized non-LLM models can provide signals like confidence scores in individual classes, they lack more detailed explanations of results; we believe that seeing a version of logical reasoning behind complex output can foster confidence and reduce the perception of these models as black boxes. Furthermore, it is important to consider that having human annotators reliably provide consistent, logical justification for each decision is prohibitive for datasets of any appreciable scale.

4.4 LLMs form the backbone of a versatile and composable approach

We demonstrate good performance on individual tasks on a real-world dataset, allowing the composition of those tasks into useful, entirely LLM-based workflows,. At the point of writing, only GPT-4 shows sufficient levels of performance on all tasks to provide confidence in the results of a multi-step workflow. However, given the rapid pace of improvement of other models, it is to be expected that these multi-step processes will be more broadly feasible in the near future.

The real impact of this new paradigm is scalability in comparison to specialized models or annotation by human domain experts; with minor modification of categories and prompts, rather than time-consuming expert annotation or fine-tuning models, the same workflows can be used for other types of courses and potentially for other types of surveys.

5 Limitations and Future Research

The data used in this study was from a specific domain (online biomedical science courses) and was in English. Greater performance could potentially have been seen with additional prompting techniques, for example through the use of self-consistency [44], reflection

(iterative self-refinement, [45], and few-shot learning; these were out-of-scope for the zero-shot premise of this article but are worth exploring.

Other than the comparison to SetFit and to the RoBERTa sentiment analysis model, we limited our exploration to recent OpenAI models; future work may expand this to include other models such as Claude v2, Command, and Llama 2. Use of open source models may have certain advantages, including greater stability, for example, based on having control over any changes that may impact the model’s responses, e.g., instruction fine-tuning or reinforcement learning from human feedback (RLHF).

The ability to compose survey analysis workflows is also amenable to the use of agents [46–48]. An educator or other stakeholder analyzing survey feedback should be able to state a goal or intent to an LLM-powered agent, with the agent picking and running tasks as a chain to get the desired analysis. Such an agent could also incorporate non-LLM tools, for example if a fine-tuned model is available that excels on a given task and is well-matched to the dataset at hand. Ideally, users of such tools should be able to operate by stating intent rather than tuning prompts or fine-tuning specialized models. Future work may incorporate these concepts.

Acknowledgments. We wish to thank members of the HMX team for their contributions to creating and administering the surveys used in this study.

Declarations

Competing interests

The authors have no relevant financial or non-financial interests to disclose.

Ethics and conflict of interest statements

This study was determined not to be human subjects research by the Harvard Medical School Office of Human Research Administration.

Author contributions

Conceptualization, methodology, software, analysis, and drafting of the manuscript were performed by Michael J. Parker. Development of labels, annotation/labeling for multi-class classification, and refinement of the manuscript were performed by all authors (equal contributions).

Availability of data and materials

The prompts used in this study are shared in the appendix. Function schemas are shared in supplementary material. To help preserve the anonymity of students and of feedback about teachers, the survey responses are not included in an open-access repository. The data may be provided upon request to the authors and approval of the university research ethics board.

References

- [1] Brennan, J., Williams, R.: Best Practices and Sample Questions for Course Evaluation Surveys. <https://www.advance-he.ac.uk/knowledge-hub/collecting-and-using-student-feedback-guide-good-practice>. Accessed: 2023-8-21 (2004)
- [2] Alhija, F.N.-A., Fresko, B.: Student evaluation of instruction: What can be learned from students' written comments? *Studies in Educational Evaluation* **35**(1), 37–44 (2009) <https://doi.org/10.1016/j.stueduc.2009.01.002>
- [3] Onan, A.: Sentiment analysis on massive open online course evaluations: A text mining and deep learning approach. *Comput. Appl. Eng. Educ.* **29**(3), 572–589 (2021) <https://doi.org/10.1002/cae.22253>
- [4] Aldeman, M., Branoff, T.J.: Impact of course modality on student course evaluations. In: 2021 ASEE Virtual Annual Conference Content Access. ASEE Conferences, Virtual Conference (2021). <https://peer.asee.org/37275.pdf>
- [5] Veselovsky, V., Ribeiro, M.H., West, R.: Artificial artificial artificial intelligence: Crowd workers widely use large language models for text production tasks (2023) [arXiv:2306.07899](https://arxiv.org/abs/2306.07899) [cs.CL]
- [6] Onan, A.: Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks. *Concurr. Comput.* **33**(23) (2021) <https://doi.org/10.1002/cpe.5909>

- [7] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding (2018) [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) [cs.CL]
- [8] Deepa, D., Raaji, Tamilarasi, A.: Sentiment analysis using feature extraction and Dictionary-Based approaches. In: 2019 Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp. 786–790 (2019). <https://doi.org/10.1109/I-SMAC47947.2019.9032456>
- [9] Zhang, H., Dong, J., Min, L., Bi, P.: A BERT fine-tuning model for targeted sentiment analysis of Chinese online course reviews. *Int. J. Artif. Intell. Tools* **29**(07n08), 2040018 (2020) <https://doi.org/10.1142/S0218213020400187>
- [10] Unankard, S., Nadee, W.: Topic detection for online course feedback using lda. In: Popescu, E., Hao, T., Hsu, T.C., Xie, H., Temperini, M., Chen, W. (eds.) *Emerging Technologies for Education. SETE 2019. Lecture Notes in Computer Science. Lecture Notes in Computer Science*, vol. 11984. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-38778-5_16
- [11] Cunningham-Nelson, S., Baktashmotlagh, M., Boles, W.: Visualizing student opinion through text analysis. *IEEE Trans. Educ.* **62**(4), 305–311 (2019) <https://doi.org/10.1109/TE.2019.2924385>
- [12] Perez-Encinas, A., Rodriguez-Pomeda, J.: International students’ perceptions of their needs when going abroad: Services on demand. *Journal of Studies in International Education* **22**(1), 20–36 (2018) <https://doi.org/10.1177/1028315317724556>
- [13] Sindhu, I., Muhammad Daudpota, S., Badar, K., Bakhtyar, M., Baber, J., Nurunnabi, M.: Aspect-Based opinion mining on student’s feedback for faculty teaching performance evaluation. *IEEE Access* **7**, 108729–108741 (2019) <https://doi.org/10.1109/ACCESS.2019.2928872>
- [14] Sutoyo, E., Almaarif, A., Yanto, I.T.R.: Sentiment analysis of student evaluations of teaching using deep learning approach. In: *International Conference on Emerging Applications and Technologies for Industry 4.0 (EATI’2020)*, pp. 272–281. Springer, Uyo, Akwa Ibom State, Nigeria (2021). https://doi.org/10.1007/978-3-030-80216-5_20
- [15] Meidinger, M., Aßenmacher, M.: A new benchmark for NLP in social sciences: Evaluating the usefulness of pre-trained language models for classifying open-ended survey responses. In: *Proceedings of the 13th International Conference on Agents and Artificial Intelligence. SCITEPRESS - Science and Technology Publications, Online* (2021). <https://doi.org/10.5220/0010255108660873>
- [16] Papers with Code - Machine Learning Datasets. <https://paperswithcode.com/datasets?task=text-classification>. Accessed: 2023-8-21
- [17] Hugging Face – The AI community building the future. https://huggingface.co/datasets?task_categories=task_categories:zero-shot-classification&sort=trending. Accessed: 2023-8-21
- [18] Kastrati, Z., Imran, A.S., Kurti, A.: Weakly supervised framework for Aspect-Based sentiment analysis on students’ reviews of MOOCs. *IEEE Access* **8**, 106799–106810 (2020) <https://doi.org/10.1109/ACCESS.2020.3000739>

- [19] Kastrati, Z., Arifaj, B., Lubishtani, A., Gashi, F., Nishliu, E.: Aspect-Based opinion mining of students' reviews on online courses. In: Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence. ICCAI '20, pp. 510–514. Association for Computing Machinery, New York, NY, USA (2020). <https://doi.org/10.1145/3404555.3404633>
- [20] Shaik, T., Tao, X., Li, Y., Dann, C., McDonald, J., Redmond, P., Galligan, L.: A review of the trends and challenges in adopting natural language processing methods for education feedback analysis. *IEEE Access* **10**, 56720–56739 (2022) <https://doi.org/10.1109/ACCESS.2022.3177752>
- [21] Edalati, M., Imran, A.S., Kastrati, Z., Daudpota, S.M.: The potential of machine learning algorithms for sentiment classification of students' feedback on MOOC. In: Intelligent Systems and Applications, pp. 11–22. Springer, Amsterdam (2022). https://doi.org/10.1007/978-3-030-82199-9_2
- [22] Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using siamese BERT-networks (2019) [arXiv:1908.10084](https://arxiv.org/abs/1908.10084) [cs.CL]
- [23] Törnberg, P.: ChatGPT-4 outperforms experts and crowd workers in annotating political twitter messages with Zero-Shot learning (2023) [arXiv:2304.06588](https://arxiv.org/abs/2304.06588) [cs.CL]
- [24] Jansen, B.J., Jung, S.-G., Salminen, J.: Employing large language models in survey research. *Natural Language Processing Journal* **4**, 100020 (2023)
- [25] Masala, M., Ruseti, S., Dascalu, M., Dobre, C.: Extracting and clustering main ideas from student feedback using language models. In: Artificial Intelligence in Education: 22nd International Conference, AIED, Proceedings, Part I, pp. 282–292. Springer, Utrecht, The Netherlands (2021). https://doi.org/10.1007/978-3-030-78292-4_23
- [26] Reiss, M.V.: Testing the reliability of ChatGPT for text annotation and classification: A cautionary remark (2023) [arXiv:2304.11085](https://arxiv.org/abs/2304.11085) [cs.CL]
- [27] Pangakis, N., Wolken, S., Fasching, N.: Automated annotation with generative AI requires validation (2023) [arXiv:2306.00176](https://arxiv.org/abs/2306.00176) [cs.CL]
- [28] Gilardi, F., Alizadeh, M., Kubli, M.: ChatGPT outperforms Crowd-Workers for Text-Annotation tasks (2023) [arXiv:2303.15056](https://arxiv.org/abs/2303.15056) [cs.CL]
- [29] Huang, F., Kwak, H., An, J.: Is ChatGPT better than human annotators? potential and limitations of ChatGPT in explaining implicit hate speech (2023) [arXiv:2302.07736](https://arxiv.org/abs/2302.07736) [cs.CL]
- [30] Best Practices and Sample Questions for Course Evaluation Surveys. <https://assessment.wisc.edu/best-practices-and-sample-questions-for-course-evaluation-surveys/>. Accessed: 2023-8-21
- [31] Medina, M.S., Smith, W.T., Kolluru, S., Sheaffer, E.A., DiVall, M.: A review of strategies for designing, administering, and using student ratings of instruction. *Am. J. Pharm. Educ.* **83**(5), 7177 (2019) <https://doi.org/10.5688/ajpe7177>
- [32] Course Evaluations Question Bank. <https://teaching.berkeley.edu/course-evaluations-question-bank>. Accessed: 2023-8-21

- [33] Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y.: Large language models are Zero-Shot reasoners (2022) [arXiv:2205.11916](#) [cs.CL]
- [34] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models, 24824–24837 (2022) [arXiv:2201.11903](#) [cs.CL]
- [35] Braun, V., Clarke, V.: Using thematic analysis in psychology. *Qual. Res. Psychol.* **3**(2), 77–101 (2006) <https://doi.org/10.1191/1478088706qp063oa>
- [36] Reynolds, L., McDonell, K.: Prompt programming for large language models: Beyond the Few-Shot paradigm (2021) [arXiv:2102.07350](#) [cs.CL]
- [37] White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., Schmidt, D.C.: A prompt pattern catalog to enhance prompt engineering with ChatGPT (2023) [arXiv:2302.11382](#) [cs.SE]
- [38] Tunstall, L., Reimers, N., Jo, U.E.S., Bates, L., Korat, D., Wasserblat, M., Pereg, O.: Efficient few-shot learning without prompts (2022) [arXiv:2209.11055](#) [cs.CL]
- [39] cardiffnlp/twitter-roberta-base-sentiment-latest - Hugging Face. <https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest>. Accessed: 2023-8-21 (2022)
- [40] Loureiro, D., Barbieri, F., Neves, L., Anke, L.E., Camacho-Collados, J.: TimeLMs: Diachronic language models from twitter (2022) [arXiv:2202.03829](#) [cs.CL]
- [41] Chen, L., Zaharia, M., Zou, J.: How is ChatGPT’s behavior changing over time? (2023) [arXiv:2307.09009](#) [cs.CL]
- [42] Kıcıman, E., Ness, R., Sharma, A., Tan, C.: Causal reasoning and large language models: Opening a new frontier for causality (2023) [arXiv:2305.00050](#) [cs.AI]
- [43] Peng, B., Li, C., He, P., Galley, M., Gao, J.: Instruction tuning with GPT-4 (2023) [arXiv:2304.03277](#) [cs.CL]
- [44] Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., Zhou, D.: Self-Consistency improves chain of thought reasoning in language models (2022) [arXiv:2203.11171](#) [cs.CL]
- [45] Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegrefe, S., Alon, U., Dziri, N., Prabhume, S., Yang, Y., Gupta, S., Majumder, B.P., Hermann, K., Welleck, S., Yazdanbakhsh, A., Clark, P.: Self-Refine: Iterative refinement with Self-Feedback (2023) [arXiv:2303.17651](#) [cs.CL]
- [46] Weng, L.: LLM Powered Autonomous Agents. <https://lilianweng.github.io/posts/2023-06-23-agent/>. Accessed: 2023-8-21 (2023)
- [47] Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., Cao, Y.: ReAct: Synergizing reasoning and acting in language models (2022) [arXiv:2210.03629](#) [cs.CL]
- [48] Shen, Y., Song, K., Tan, X., Li, D., Lu, W., Zhuang, Y.: HuggingGPT: Solving AI tasks with

A Appendix

We organize the appendix into three sections:

- Additional metrics for multi-label classification, including scores for individual classes, as well as Hamming loss and subset accuracy
- Text extraction rubric full metrics
- Prompts, showing the details of how LLMs were called, along with an example function schema showing how structured output is obtained for OpenAI model calls for the binary classification task

A.1 Additional metrics for multi-label classification

A.1.1 Consensus rows - 1572 rows dataset (1489 for SetFit)

Precision, recall, and F1 score are shown for each tag in multi-label classification for the consensus rows condition, along with macro averages for each metric, in Table 11 for GPT-4, Table 12 for GPT-3.5, and Table 13 for SetFit. Hamming loss and subset accuracy are shown in Table 14.

Table 11 Individual label scores for multi-label classification with GPT-4, consensus rows.

Tag	Precision	Recall	F1 score
course logistics and fit	89.83	64.63	75.18
curriculum	95.89	91.48	93.64
teaching modality	97.78	96.70	97.24
teaching	76.04	91.25	82.95
assessment	97.50	93.41	95.41
resources	92.31	97.30	94.74
peer and teacher interaction	77.08	92.50	84.09
other	92.60	97.47	94.97
Macro average	89.88	90.59	89.78

Table 12 Individual label scores for multi-label classification with GPT-3.5, consensus rows.

Tag	Precision	Recall	F1 score
course logistics and fit	72.88	52.44	61.00
curriculum	76.79	95.23	85.02
teaching modality	94.99	93.68	94.33
teaching	27.64	95.00	42.82
assessment	80.49	79.04	79.76
resources	63.30	93.24	75.41
peer and teacher interaction	64.00	80.00	71.11
other	74.66	68.84	71.63
Macro average	69.34	82.18	72.63

Table 13 Individual label scores for multi-label classification with SetFit, consensus rows.

Tag	Precision	Recall	F1 score
course logistics and fit	34.78	33.33	34.04
curriculum	92.31	72.99	81.52
teaching modality	94.82	92.84	93.82
teaching	84.21	24.62	38.10
assessment	97.70	56.29	71.43
resources	91.43	50.00	64.65
peer and teacher interaction	80.00	55.17	65.31
other	99.71	75.49	85.93
Macro average	84.37	57.59	66.85

Table 14 Hamming loss and subset accuracy for multi-label classification, consensus rows.

Model	Hamming loss	Subset accuracy
GPT-4	0.0194	0.8849
GPT-3.5	0.0710	0.5948
SetFit	0.0508	0.6797

A.1.2 Consensus labels - 2500 rows dataset (2359 for SetFit)

Precision, recall, and F1 score are shown for each tag in multi-label classification for the consensus labels condition, along with macro averages for each metric, in Table 15 for GPT-4, Table 16 for GPT-3.5, and Table 17 for SetFit. Hamming loss and subset accuracy are shown in Table 18.

Table 15 Individual label scores for multi-label classification with GPT-4, consensus labels.

Tag	Precision	Recall	F1 Score
course logistics and fit	74.81	57.71	65.16
curriculum	82.86	86.38	84.58
teaching modality	78.35	95.94	86.26
teaching	56.70	87.59	68.83
assessment	87.36	94.82	90.94
resources	70.95	95.49	81.41
peer and teacher interaction	60.58	94.03	73.68
other	79.64	95.10	86.68
Macro average	73.91	88.38	79.69

A.2 Extraction rubric (full metrics)

Full metrics are shown for the text extraction evaluation rubric for GPT-4 and GPT-3.5 in Table 19.

Table 16 Individual label scores for multi-label classification with GPT-3.5, consensus labels.

Tag	Precision	Recall	F1 Score
course logistics and fit	70.19	41.71	52.33
curriculum	66.33	91.81	77.02
teaching modality	75.43	93.82	83.62
teaching	23.14	92.41	37.02
assessment	80.30	80.79	80.55
resources	48.75	87.97	62.73
peer and teacher interaction	62.07	80.60	70.13
other	57.12	69.22	62.59
Macro average	60.42	79.79	65.75

Table 17 Individual label scores for multi-label classification with SetFit, consensus labels.

Tag	Precision	Recall	F1 Score
course logistics and fit	33.04	24.52	28.15
curriculum	82.46	66.48	73.61
teaching modality	71.92	92.83	81.05
teaching	66.67	23.53	34.78
assessment	96.91	52.16	67.82
resources	79.31	40.71	53.80
peer and teacher interaction	56.82	53.19	54.95
other	98.59	71.22	82.70
Macro average	73.22	53.08	59.61

Table 18 Hamming loss and subset accuracy for multi-label classification, consensus labels.

model	Hamming loss	Subset accuracy
GPT-4	0.0503	0.7168
GPT-3.5	0.10235	0.4628
SetFit	0.07238	0.5774

Table 19 Error rate (%) of extraction for ‘suggestions for improvement’ from comments classified as containing ‘suggestions for improvement’ (all metrics from rubric and human annotation for hallucinations).

Metric	GPT-4	GPT-3.5
Missed Excerpts	2.37	7.82
Ambiguous Excerpts	4.61	4.75
Missed Existing Context	0.28	0.84
Irrelevant Excerpts	0.14	0.84
Irrelevant Context	0.00	0.14
Implied Goal Focus	3.07	2.79
Non Quotes	0.00	6.01
Non Contiguous Excerpts	0.00	0.14
Redundant Excerpts	0.28	2.79
Hallucinations	0.00	3.91

A.3 Prompts and example function schema

All elements of the prompts for GPT-4 and GPT-3.5 API calls are shown in Tables 20, 21, 22, 23, 24, 25 and 26.

A.3.1 Multi-label classification

Table 20 Prompt for multi-label classification with GPT-4 and GPT-3.5. ‘tags’ and ‘comment’ are values that get inserted into Python f-strings. ‘comment’ is the survey response.

prompt - system message	<p>You are a highly-skilled assistant that classifies student course feedback comments. You respond only with a JSON object.</p> <p>You will be provided with a comment from a student course feedback survey. Categorize the comment with as many of the following categories as apply:</p> <p>{tags}</p> <p>Think step-by-step to arrive at a correct classification. Include your reasoning behind every assigned category in the output.</p>
tags	<p>Category: “course logistics and fit” Description: course delivery (policy, support), cost, difficulty, time commitment, grading, credit, schedule, user fit, access, background (e.g., prereqs and appropriateness of course level).</p> <p>Category: “curriculum” Description: course content, curriculum, specific topics, course structure. This focuses on the content and the pedagogical structure of the content, including flow and organization. This also includes applied material such as clinical cases and case studies. Includes referencesto pre-recorded discussions between experts or between a doctor and a patient. Includes specific suggestions for additional courses or content.</p> <p>...(continued for all categories/descriptions shown in Table 3)</p>
prompt - user message	{comment}

A.3.2 Multi-class classification

Table 21 Prompt for multi-class classification with GPT-4 and GPT-3.5. ‘tags’ and ‘excerpt_list’ are values that get inserted into Python f-strings. ‘excerpt_list’ is a Python list of extracted excerpts corresponding to a single survey response. ‘goal_focus’ is the directive that was used for text extraction, e.g., ‘suggestions for improvement’.

prompt - system message	<p>You are a highly-skilled assistant that categorizes {goal_focus} from student course feedback comments. You respond only with a JSON array.</p> <p>You will be provided with an array of excerpts from a student course feedback survey. Categorize each excerpt with exactly one of the following categories:</p> <p>{tags}</p> <p>Think step-by-step to arrive at a correct classification. Include your reasoning in the output.</p>
tags	<p>Category: “course logistics and fit” Description: course delivery (policy, support), cost, difficulty, time commitment, grading, credit, schedule, user fit, access, background (e.g., prereqs and appropriateness of course level).</p> <p>Category: “curriculum” Description: course content, curriculum, specific topics, course structure. This focuses on the content and the pedagogical structure of the content, including flow and organization. This also includes applied material such as clinical cases and case studies. Includes references to pre-recorded discussions between experts or between a doctor and a patient. Includes specific suggestions for additional courses or content.</p> <p>...(continued for all categories/descriptions shown in Table 3)</p>
prompt - user message	{comment}

A.3.3 Binary classification

Table 22 Prompt for binary classification with GPT-4 and GPT-3.5. ‘delimiter’, ‘question’, ‘goal_focus’, and ‘comment’ are values that get inserted into Python f-strings. ‘comment’ is the survey response.

prompt - system message	<p>You are a highly-skilled assistant that classifies student course feedback comments based on whether or not they contain {goal_focus}. You answer only with a JSON object and nothing else.</p> <p>You will be provided with a comment from a student course feedback survey. The comment will be delimited by {delimiter} characters. Each original comment was in response to the question: “{question}”. Classify the comment as either containing {goal_focus} or not containing {goal_focus}. Think step-by-step to arrive at a correct classification. Include your reasoning in the output.</p> <p>Only use yes or no for the classification. If something is N/A, use no.</p>
delimiter	Delimiter characters used to clearly delineate the comment (for example, “####”)
goal_focus	The focus of the binary classification; examples include “suggestions for new topics or content” and “feedback about the teaching or explanations”
question	The survey question that the learner’s comment is in response to
prompt - user message	{delimiter}{comment}{delimiter}

A.3.4 Extraction

Table 23 Prompt for text extraction with GPT-4 and GPT-3.5. ‘delimiter’, ‘question’, ‘goal.focus’, and ‘comment’ are values that get inserted into Python f-strings.

prompt - system message	<p>You are a highly-skilled assistant that extracts {goal.focus} from student course feedback comments. You respond only with a JSON array.</p> <p>You will be provided with a comment from a student course feedback survey. The comment will be delimited by {delimiter} characters. Each original comment was in response to the question: "{question}". However, your task is to only select excerpts which pertain to the goal focus: "{goal.focus}". Excerpts should only be exact quotes taken from the comment; do not add or alter words under any circumstances.</p> <p>If you cannot extract excerpts for any reason, for example if the comment is not relevant to the question, respond only with an empty JSON array: []</p> <p>If there are relevant excerpts, ensure that excerpts contain all relevant text needed to interpret them - in other words don't extract small snippets that are missing important context.</p> <p>Before finalizing excerpts, review your excerpts to see if any consecutive excerpts are actually about the same suggestion or part of the same thought. If so, combine them into a single excerpt.</p>
goal.focus	The focus of the binary classification; examples include "suggestions for new topics or content" and "feedback about the teaching or explanations"
question	The survey question that the learner's comment is in response to
prompt - user message	{delimiter}{comment}{delimiter}

A.3.5 Extraction evaluation

Table 24 Prompt for text extraction evaluation with GPT-4. ‘goal_focus’, ‘question’, ‘comment’, and ‘excerpt_list’ are values that get inserted into Python f-strings. ‘comment’ is the survey response. ‘excerpt_list’ is a Python list of extracted excerpts corresponding to a single survey response.

prompt - system message	<p>You are a highly-skilled assistant that evaluates how well a computer program has extracted {goal_focus} from student course feedback comments. You respond only with a JSON object.</p> <p>You will be provided with the original comment from a student course feedback survey and a list of the extracted excerpts. Each original comment was in response to the question: “{question}”. The computer program’s task was to select exact quote excerpts which pertain to the goal focus: “{goal_focus}”.</p> <p>Your task is to evaluate how well the computer program did by answering the following questions:</p> <ul style="list-style-type: none"> - Did the program miss any relevant excerpts? Ignore spelling, punctuation, and capitalization changes. (yes or no) - Did the program extract any excerpts that are ambiguous because of lack of context? (yes or no) - If context is missing, did the needed context exist in the original comment but not get captured in the excerpts? If all needed context was present in the excerpts, answer ‘na’. (yes, no, or na) - Did the program extract any irrelevant excerpts? (yes or no) - Did the program extract any irrelevant text that is not needed to interpret the excerpts? (yes or no) - Did the program extract any excerpts where {goal_focus} is only implied but not explicit? (yes or no) - Did the program produce any excerpts that are not exact quotes from the original comment? (yes or no) - Did the program extract any excerpts that are not contiguous (i.e., consecutive) in the original comment? (yes or no) - Are there two or more excerpts that are actually about the same idea and which could have been combined or had one dropped? (yes or no) <p>Think step-by-step to arrive at correct answers. Include your reasoning behind every question in the output.</p>
goal_focus	The focus of the binary classification; examples include “suggestions for new topics or content” and “feedback about the teaching or explanations”
question	The survey question that the learner’s comment is in response to
prompt - user message	<p>Comment: {comment}</p> <p>List of excerpts: {excerpt_list}</p>

A.3.6 Sentiment analysis

Table 25 Prompt for sentiment analysis with GPT-4 and GPT-3.5. ‘tags’ and ‘comment’ are values that get inserted into Python f-strings. ‘comment’ is the original student comment.

prompt - system message	<p>You are a highly-skilled assistant that classifies student course feedback comments. You respond only with a JSON object.</p> <p>You will be provided with a comment from a student course feedback survey. Categorize the comment with one of the following categories: {tags}</p> <p>Think step-by-step to arrive at a correct classification. Include your reasoning behind every assigned category in the output.</p>
tags	<p>Category: “positive” Description: The comment is predominantly pleased or happy with the course experience.</p> <p>Category: “slightly positive” Description: The comment is slightly pleased or happy.</p> <p>Category: “neutral” Description: The comment is neither positive or negative overall.</p> <p>Category: “slightly negative” Description: The comment is slightly displeased or unhappy.</p> <p>Category: “negative” Description: The comment is predominantly displeased or unhappy.</p>
prompt - user message	{comment}

A.3.7 Inductive thematic analysis

Table 26 Prompt for inductive thematic analysis with GPT-4 and GPT-3.5. Each ‘comment’ is a student survey response. ‘question’ is the survey question that the learner comments are in response to.

prompt - system message	<p data-bbox="528 472 1348 521">You are a highly-skilled assistant that derives the main themes from student course feedback comments. You respond only with a JSON array.</p> <p data-bbox="528 546 1348 672">You will be provided with a set of comments from a student course feedback survey. Each comment is surrounded by the delimiter {delimiter}. Each original comment was in response to the question: “{question}”. Your task is to derive the main themes from the comments. A theme is a short phrase that summarizes a piece of feedback that is expressed by multiple students.</p> <p data-bbox="528 696 1348 772">Respond with a JSON array of theme objects. Each theme object should have a ‘theme_description’ field which describes the theme in two sentences or less, and a ‘theme_title’ field (which gives a short name for the theme in 5 words or less).</p> <p data-bbox="528 797 1348 846">If you cannot find any themes, for example if there are no comments in the original list, respond only with an empty JSON array: []</p>
prompt - user message	list of {delimiter}{comment}{delimiter} joined by ‘\n’

A.3.8 Binary classification function schema

For each prompt, there was also an associated function schema that was used as part of function calling to guide the format of the output. An example of one function schema (for binary classification) is shown in Listing 1.

Listing 1 Function schema for binary classification.

```
{
  "name": "store_binary_classification_with_reasoning",
  "description": "Store the the binary classification and reasoning of
    an online course survey comment in a database.",
  "parameters": {
    "type": "object",
    "properties": {
      "classification": {
        "type": "string",
        "enum": ["yes", "no"],
        "description": "Whether or not the comment contains the
          goal focus.",
      },
      "reasoning": {
        "type": "string",
        "description": "The reasoning for the classification."
      }
    },
    "required": ["classification", "reasoning"],
  },
}
```