The Internet and Higher Education 65 (2025) 101001

Contents lists available at ScienceDirect

The Internet and Higher Education

journal homepage: www.elsevier.com/locate/iheduc

Transforming online learning research: Leveraging GPT large language models for automated content analysis of cognitive presence

Daniela Castellanos-Reyes [a,*], Larisa Olesova [b], Ayesha Sadaf [c]

[a] Department of Teacher Education and Learning Sciences, North Carolina State University, Center for Technology and Innovation, 1010 Main Campus Dr, Suite 300, Raleigh, NC 27606, United States of America
[b] School of Teaching and Learning, College of Education, University of Florida, PO Box 117048, Gainesville, FL 32611, United States of America
[c] Department of Educational Leadership, Cato College of Education, University of North Carolina at Charlotte, 9201 University City Blvd, Charlotte, NC 28223, United States of America

ARTICLE INFO

Keywords:
Cognitive presence
Online learning
Artificial intelligence in education
Automated content analysis
Discussion boards
Large language models
GPT

ABSTRACT

The last two decades of online learning research vastly flourished by examining discussion board text data through content analysis based on constructs like cognitive presence (CP) with the Practical Inquiry Model (PIM). The PIM sets a footprint for how cognitive development unfolds in collaborative inquiry in online learning experiences. Ironically, content analysis is a resource-intensive endeavor in terms of time and expertise, making researchers look for ways to automate text classification through ensemble machine-learning algorithms. We leveraged large language models (LLMs) through OpenAI's Generative Pre-Trained Transformer (GPT) models in the public API to automate the content analysis of students' text data based on PIM indicators and assess the reliability and efficiency of automated content analysis compared to human analysis. Using the seven steps of the Large Language Model Content Analysis (LACA) approach, we proposed an AI-adapted CP codebook leveraging prompt engineering techniques (i.e., role, chain-of-thought, one-shot, few-shot) for the automated content analysis of CP. We found that a fine-tuned model with a one-shot prompt achieved moderate interrater reliability with researchers. The models were more reliable when classifying students' discussion board text in the Integration phase of the PIM. A cost comparison showed an obvious cost advantage of LACA approaches in online learning research in terms of efficiency. Nevertheless, practitioners still need considerable data literacy skills to deploy LACA at a scale. We offer theoretical suggestions for simplifying the CP codebook and improving the IRR with LLM. Implications for practice are discussed, and future research that includes instructional advice is recommended.

1. Introduction

Cognitive presence (CP) refers to learners' knowledge construction through discourse and reflection (Garrison et al., 2001; Swan & Ice, 2010). It is well-established that CP provides a framework for instructors in online higher education to guide meaning-making in collaborative discourse (Castellanos-Reyes, 2020a; Martin et al., 2022; Swan, 2021). Yet, examination of CP traditionally happens through labor-intensive content analysis of students' cognitive processes. Such content analysis requires vast expertise in leveraging frameworks like the Practical Inquiry Model (PIM) that sets a footprint for how cognitive development unfolds in online text-based discussions (Sadaf et al., 2021). Furthermore, content analysis of CP is a resource-intensive endeavor in terms of time and human capital (Gašević et al., 2019), which has made researchers look for ways to automate text classification through ensemble machine learning algorithms (e.g., Barbosa et al., 2021; Ferreira et al., 2020; Zou et al., 2021). Nevertheless, automated classification of students' discussion text data is still unreachable for instructors to provide real-time feedback to help students reach higher levels of CP (Lindgren et al., 2020; Rourke et al., 2001). Given cognitive presence's critical role in supporting complex reasoning in distance learning, it is urgent to provide instructors with the tools that assist in the classification of students' discourse accurately and efficiently.

To enable instructors to provide real-time feedback to scaffold students' CP development, we explored how generative Artificial Intelligence (AI) tools can assist instructors in classifying students' written

* Corresponding author at: North Carolina State University, Center for Technology and Innovation, 1010 Main Campus Drive, Suite 300, Centennial Campus, Raleigh, NC, United States of America.
E-mail addresses: dcastel2@ncsu.edu (D. Castellanos-Reyes), lolesova@coe.ufl.edu (L. Olesova), asadaf@uncc.edu (A. Sadaf).

https://doi.org/10.1016/j.iheduc.2025.101001
Received 13 August 2024; Received in revised form 10 February 2025; Accepted 10 February 2025
Available online 12 February 2025

interactions in asynchronous text-based distance learning environments. Specifically, we propose investigating to what extent the Large Language Model (LLM) - Content Analysis (LACA) compares to instructors' and researchers' manual examination of online learners' text-based interactions in terms of reliability (i.e., correct classification) and efficiency (i.e., resource investment). LLMs have the potential to assist knowledge workers, like online instructors, in laborious tasks (Platt & Platt, 2023), like facilitating discussion boards. Among the multiple LLMs available, we argue that the OpenAI suite of large language models through the public API can help decrease the hand-coding time of students' text-based CP interactions, given its capability to create output in seconds. The findings of this study will be beneficial for researchers and instructors who want to address students' needs immediately and will ultimately streamline the benefits of examining online learning experiences through the CP lens.

2. Literature review

2.1. Cognitive presence and the practical inquiry model

The Community of Inquiry (CoI) framework, developed by Garrison et al. (2001), identifies three presences essential for a meaningful educational experience in oline learning: Cognitive, social presence, and teaching Presence. CP is a core element of CoI framework that guides the design and implementation of online learning environments through a social-constructivist approach to learning (Akyol & Garrison, 2011). Cognitive presence is defined as "the extent to which learners are able to construct and confirm meaning through sustained reflection and discourse" (Garrison et al., 2001, p. 11). It represents ways to support and maintain a purposeful community of learners (Garrison, 2017).

CP is operationalized through the Practical Inquiry Model (PIM) based on phases of Dewey's (1933) reflective thinking and a collaborative inquiry process (Garrison et al., 2001). The PIM is a conceptual framework that guides the design and facilitation of online learning experiences and emphasizes practical, real-world problem-solving. The PIM involves four key phases of cognitive presence: (1) Triggering—becoming aware of a problem through initiating the inquiry process; (2) Exploration—exploring a problem by searching for relevant information, engaging in reflection, and sharing explanations; (3) Integration—constructing meaning from various resources and offering a possible solution, and (4) Resolution—applying or defending potential solutions with a new thought or idea. These four phases provide a practical means to judge the nature and quality of critical reflection and discourse in a community of inquiry (Akyol & Garrison, 2011). Much of the research on the PIM investigating the distribution patterns of cognitive presence in online discussions has found resolution as less and exploration as the dominant phase during the inquiry process (Sadaf & Olesova, 2017; Chen & Chang, 2017; Gašević et al., 2015).

Garrison (2017) argued that to establish deep and meaningful learning in an online educational experience, effective learning must take into consideration both the internal cognitive processes (e.g., reflection) as well as the external contextual elements (e.g., shared discourse) that precipitate and shape thinking. CP ensures that learners engage in critical thinking and deep learning, moving beyond surface-level understanding. The PIM emphasizes practical application, preparing learners to tackle real-world challenges in their field of study or profession. Both CP and the PIM promote collaborative learning through meaningful discourse and shared exploration.

2.2. Content analysis of cognitive presence

Analysis of CP usually involves systematically classifying online discussion text data to identify the elements of CP (Garrison et al., 2001). Researchers analyze online discussion posts to find indicators of the categories as outlined in the PIM described above (Garrison et al., 2001) – a process called content analysis. Content analysis is "a research technique for making [...] valid inferences from texts [...] to the contexts of their use" (Krippendorff, 2004, p. 18). CA helps with a deeper understanding of the text data and how it can be interpreted through coding schemes. There are two main coding schemes to capture CP in online learners' discussions: Garrison et al. (2001) and Shea et al. (2010).

Garrison et al. (2001) developed a systematic procedure for assigning data (transcript segments) into the four phases of the PIM with descriptors, indicators, and examples that could serve as a framework and direction for future research on CP in asynchronous online discussions. In their work, Garrison et al. (2001) created an efficient and reliable assessment that provides information about the quality of the CP in asynchronous online discussions. Garrison et al. (2001) developed a set of categories into which segments of messages were coded. Garrison et al. (2001) proposed researchers use the code-down approach (i.e., to the earlier phase) if the distinction between two phases was unclear, and code-up (i.e., to the later phase) if evidence of multiple phases was present. In their first examination of their proposed coding scheme, the coding decisions of the two coders were evaluated for interrater reliability using Holsti's (1969) coefficient of reliability (CR) and Cohen's (1960) kappa (k). Their results for three training sessions were $CR =$ 0.45, 0.65, and 0.84; and $k = 0.35$, 0.49, and 0.74. Researchers explained the first two low interrater reliability due to the "latent projective" nature of the four phases, which is common for these types of studies (Garrison et al., 2001).

A decade later, Shea et al. (2010) revisited Garrison et al.'s (2001) initial coding scheme to clarify wording in the descriptions of each phase. Specifically, Shea et al. (2010) removed "divergent" and "convergent" words from the exploration and integration category descriptions. Furthermore, they deleted the "brainstorming" indicator from the exploration because arguing that it could not be differentiated from the "information exchange" indicator. They computed measures of Inter-rater reliability (IRR) among four researchers after analyzing ten online discussions of five modules in two online courses. IRR was computed first by using Cohen's (1960) kappa (k), which was low. Then, they computed Holsti's (1969) coefficient of reliability (CR). After calculating initial inter-rater reliability, researchers met to negotiate disagreements. The negotiated CR was the following: module 1 (0.98 for course 1 and 0.99 for course 2); module 2 (0.98 and 0.99); module 3 (0.98 and 0.97); module 4 (0.99 and 0.99), and module 5 (1.00 and 1.00).

Online learning research has traditionally leveraged coding schemes based on the PIM for CP identification in discussion board text data (e.g., Kaczkó & Ostendorf, 2023; Oriogun, 2009; Shea et al., 2010; Tirado-Morueta et al., 2016). Nevertheless, content analysis is a strenuous task demanding considerable time and expertise, creating a need for automation by integrating technological developments like machine learning algorithms. Ba et al. (2023) argue that it is essential for instructors to coach online learners in their development of CP. However, the lack of efficient tools that allow novice instructors without prior training on CP to identify the PIM phases hinders their ability to focus on scaffolding students (2023). Automated approaches can considerably reduce the effort required for content analysis.

Automated content analysis (ACA) appeared in response to the arduous task of implementing content analysis and involved some combination of natural language processing and machine learning algorithms. Yet, it requires sophisticated expertise for the arduous task of data pre-processing and the feature engineering of machine learning algorithms, the interpretation of results, and algorithm evaluation. Furthermore, training a machine learning model for content analysis demands a vast time investment that is virtually impossible to deploy for instructors' everyday tasks like course facilitation and scaffolding (Richardson et al., 2021).

Despite ACA's complexity, several researchers have attempted the automatic classification of CP (e.g., Hayati et al., 2019; Kovanovic, 2017; McKlin, 2004). Specifically, researchers feed pre-processed discussion board text data into various supervised algorithms, which are later evaluated based on classification accuracy. Text pre-processing involves significant efforts typical of natural language processing (NLP), like removing stop words and slang, spelling correction, stemming (i.e., removing words' prefixes and/or suffixes), and lemmatization (i.e., removing word inflections). Datasets for supervised classification algorithms are typically split between a training and a testing dataset. This step is essential so that researchers can feed the algorithm (i.e., training dataset), and then, evaluate its performance (i.e., testing dataset).

In the early 2000s, McKlin (2004) pioneered the implementation of machine learning algorithms for analyzing CP. They used artificial neural networks (ANN), also regarded as deep learning, to compare human coders' classification with ANN classification, finding a Cohen's k agreement of 0.70. Building on McKlin's early exploration, Kovanovic (2017) developed a random forest classifier that achieved a 70.3 % accuracy and a Cohen's $k = 0.60$. After these early attempts of automatic cognitive presence identification, researchers focused on binary classification of content analysis using machine learning algorithms. For example, Hayati et al. (2019) used NLP in combination with multiple binary classifiers to assess online learners' CP. They found that a Naive Bayes classifier algorithm to be the most accurate when compared to a support vector machine algorithm (SVM) and logistic regression (Cohen's $k = 0.68$).

Most recent developments in CP automated content analysis flourished with the availability of algorithms developed explicitly for language processing, like the Bidirectional Encoder Representations from Transformers (BERT) and the availability of larger datasets that increased the amount of information used for training algorithms, and eventually classification accuracy. For instance, Lee et al. (2022) used 1493 messages and achieved an F1 score of 0.76 (i.e., F1 is a measure of model precision). Likewise, Ba et al. (2023) also found that using BERT for cognitive presence classification substantially improved the classification accuracy from previous classifications by obtaining a Cohen's k of 0.76. The studies mentioned above illustrate the potential of using machine learning classifiers for automated content analysis of cognitive presence.

Assuming that automated content analysis of CP serves instructors and, eventually, online learners, it is essential to leverage user-friendly and just-in-time tools that require little to no programming knowledge. Although not focused on CP or online learning, researchers have used tools leveraging large-language models like GPT by OpenAI and Bard by Google AI to automate content analysis. Large-language models are revolutionizing the world due to their user-friendly interface and apparent accuracy. These models do not require programming skills, but instead prompt engineering focused on elaborated and specific instructions to give to the model. For example, Xiao et al. (2023) used LLMs (i.e, GPT-3) to classify children's curiosity-driven questions based on complexity and syntactic structure achieving a Cohen's k of 0.60. Researchers also used the LLM of OpenAI to analyze other text data like tweets or news articles (Chew et al., 2023), achieving researcher-model agreements as high as 0.97 when using Gwet's AC1 as their preferred reliability measure. Similar efforts have also been made with interview data suggesting the potential of LLMs to brainstorm new codes and as a confirmatory analysis tool (Tai et al., 2024). In this study, we propose leveraging the use of large-language models to automate content analysis of cognitive presence and support instructors' scaffolding efforts.

3. This study

The ubiquitous characteristic of discussion board conversations in online environments has made them the utmost data source for distance learning researchers for decades. Numerous researchers have used content analysis to explore distance learners' sense of community and, most importantly, the extent to which they create meaning, in other words, their CP (Sadaf et al., 2021; Garrison et al., 2001). Nevertheless, the content analysis endeavor demands a vast deployment of efforts and resources from training personnel to use highly technical coding schemes, requiring more than one researcher. These barriers make it unfeasible for instructors new to the distance learning field and instructors looking to examine their students' discussion-board interactions on the go. Although previous efforts have been made to automate the content analysis of discussion boards (e.g., Ba et al., 2023; Hayati et al., 2019; Kovanovic, 2017; McKlin, 2004) to expedite the process, this work is still out of reach of the larger population of educational researchers and distance learning instructors because it exploits sophisticated machine-learning algorithms.

Given the potential of LLMs to automate content analysis with little to no programming experience and considerably less effort (Chew et al., 2023; Xiao et al., 2023), this study explored how the GPT family of models through the OpenAI API could be used to analyze online discussion boards and what potential limitations exist in this approach. To our knowledge, no previous work has explored large-language models to investigate cognitive presence. Therefore, it is critical for the distance and online learning community to understand the potential of leveraging Generative AI in content analysis of distance learning. This study addresses this gap by investigating the reliability and efficiency of LLMs as part of the OpenAI suite in combination with prompt engineering to classify students' text-based interactions in discussion boards. The following research questions guided our inquiry:

- **RQ1:** How **reliable** are large language models in classifying students' process through the four phases of the Practical Inquiry Model of Cognitive Presence within inquiry-based discourse based on different prompts (i.e., one-shot vs few-shot)?
- **RQ2:** How **efficient** are large language models for the automated content analysis of Cognitive Presence in terms of time and financial cost compared to trained researchers?
- **RQ3:** What **additional insights** does automated content analysis using large language models reveal regarding Cognitive Presence in students' discussion boards?

4. Methods

This section describes the steps in the GPT-assisted content analysis of online discussions to examine CP. It starts by describing the dataset and context, then the Large Language Model (LLM)—Content Analysis (LACA) analytical approach, and finally, the steps taken in this study.

4.1. Dataset and context

The data included three week-long discussion threads among students in an online graduate course in instructional design offered over a sixteen-week semester and delivered via the learning management system Canvas. Given that previous research reports the lack of integration and resolution phases in discussion board data (Sadaf et al., 2021; Cleveland-Innes & Garrison, 2021), we purposefully chose the three discussion prompts because they used case-based instruction which research facilitates cognitive presence and critical thinking (e.g., Richardson et al., 2012; Sadaf & Olesova, 2017; Richardson & Ice, 2010; Zhang et al., 2024). Choosing case-based instruction discussions allowed us to obtain sufficient data from the four phases of CP for model training.

The discussion prompts were designed to guide students through the four phases of the PIM via case-based learning. Specifically, students examined a case using four discussion questions designed based on the PIM by integrating the four phases. The weekly discussion threads were divided into two. By midweek, students answered questions focused on triggering event (e.g., What do you think are the main instructional design challenges in the case?) and exploration phases [e.g., What are your thoughts about what this instruction should look like? (Hint: The Falance article helped bring some of these principles down to a practical level)]. In the second part of the week, students focused on integration (e.g., Outline a potential solution) and resolution (e.g., Justify your response: Explain why and how your strategy is the best solution in this case) targeted questions. The three case studies were about K-12 instruction, training in higher education, and project management, respectively.

Student data was anonymized, and discussion posts were segmented into paragraphs as a unit of analysis. The dataset had a total of 180 individual discussion posts, and the average length of the discussion posts was 169 words. The paragraph ($n = 293$) was chosen as a unit of analysis. The average length of the analysis unit was 98 words and 118 tokens, which are common sequences of characters in words. Shortening the units of analysis from message to paragraph excerpts allowed us to identify more variation in the phases of the PIM and reduced the presence of multiple phases simultaneously. Students' decisions on paragraph length were followed during data pre-processing. The use of the API guaranteed that none of the anonymized student data would be used by OpenAI to train future models. This study collected and analyzed data after receiving approval from the Institutional Review Board of our institution.

4.2. Analytical approach

We used content analysis as a methodological approach (Krippendorff, 2004), specifically, we adapted the Large Language Model (LLM) - Content Analysis (LACA) process developed by Chew et al. (2023) that uses the GPT family of models through the OpenAI public API. Chew et al.'s (2023) procedure distances from traditional content analysis by integrating LLMs as part of the co-creation of the codebook, including reliability tests between large language models and human codes, and eventually, replacing the manual coding with large language models.

The LACA process incorporates eight major steps. Step 1 focuses on the *human development of categories* or codes of interest. It is recommended to build on literature to create coding schemes or use previously created coding schemes to understand the text. In step 2, researchers engage in iterative co-development of a codebook with an LLM with definitions of codes and categories, any necessary instructions, and examples per code. Such a co-development process requires the usage of prompt engineering to give clear instructions to the LLM. Prompt engineering refers to the set of techniques used to design, refine, and optimize input prompts that efficiently communicate users' purpose with LLMs (Ekin, 2023). For example, Chew et al.'s (2023) recommend that LLMs have better performance when complex sentences are used (e.g., Include messages that…) instead of only categories and definitions. Further, Xiao et al. (2023) report that codebook-based prompts (i.e., providing the code definition) had a better performance than example-based approaches (i.e., providing only an example) in LLM-assisted content analysis. Other recommended prompt engineering techniques for LACA are requesting justification as part of the prompt (Chew et al., 2023; Xiao et al., 2023), role prompting (e.g., You are a qualitative coder…), and chain-of-though prompting to indicate procedures to the LLM step by step (Dunivin, 2024; Wei et al., 2023).

In step 3, a subset of the data is drawn at random for coding and statistical testing in a similar fashion to the data splitting procedures for the training and testing of machine learning algorithms. In step 4, the LLM is used to predict codes in the sample. Chew et al. (2023) clarified that coding a smaller sample assumes that coding a census of all data is infeasible. In step 5, the interrater reliability (IRR) among researchers' codes and the LLM codes is calculated and evaluated against a benchmark to ensure that codes are consistently used and to quantify coder agreement. If such a benchmark is not attained, steps 1 to 4 can be repeated. When a satisfactory IRR is achieved, a larger sample is drawn at random for final coding in step 6. It is essential to have a balance of categories coded between the random sample drawn in Step 3 and subsequent larger samples. In step 7, the LLM and codebook previously created are used to predict the codes in the larger sample with the goal of quantifying reliable coding while considering variability in the larger sample. Finally, in step 8, statistical analysis could be done using proportions and confidence intervals. Chew et al. (2023) validated their procedure with four datasets ranging from tweets and news articles to blog posts.

4.3. Adapted LACA steps (Chew et al., 2023)

The major differences between Chew et al.'s (2023) LACA process and ours were in steps 4, 5, and 8, where we implemented a default parameter model using only the codebook and a fine-tuned model based on the codes that researchers have agreed upon. Specifically, in step 4, we calculated the IRR between researchers only; in step 5, we fine-tuned the LLM based on the codes agreed upon; and in step 8, we evaluated both models using IRR. Our adapted LACA process incorporated eight major steps: 1) human development of categories or codes of interest, 2) co-development of an AI-adapted cognitive presence codebook with LLM for code refinement and LLM prompt engineering, 3) subsetting a training data set at random from the final dataset, 4) human coding and IRR calculation, 5) fine-tuning the LLM, 6) subsetting the testing dataset (i.e., larger than the training dataset) and human coding, 7) prediction of codes using LLM, and finally 8) evaluation of the LLM coding.

4.3.1. Step 1: Human development of categories

We used Garrison et al. (2001) foundational descriptors of the four phases of the PIM for the conceptualization of the constructs and built on Shea et al.'s (2010) coding scheme of CP for operationalization. In the following step, we modified the existing protocols to focus on the four major categories of the PIM and their respective indicators and sociocognitive processes.

4.3.2. Step 2: AI-adapted cognitive presence codebook co-development with LLM and prompt engineering

We used full imperative sentences rather than categories (i.e., triggering event) when developing the AI-adapted CP codebook. The codes were mutually exclusive to better evaluate the LLM performance despite the nuances in text data and phase interpretation could appear. Table 1 shows the AI-adapted CP codebook used in this study.

We used a codebook-based approach for our prompt engineering. Specifically, we developed two scenarios based on the coding scheme as part of the prompt engineering process: a one-shot prompt and a few-shot prompt. In the one-shot prompt, we instructed the LLM to classify students' discussion board text based on the definition and one example per each PIM phase. The few-shot prompt included imperative phrases using the indicators and sociocognitive processes of the original coding scheme (e.g., Include messages that…) in addition to the instruction, definition, and example.

We included justification as part of the prompt by instructing the LLM to provide a rationale for the coding decision (i.e., print the most applicable code and the rationale for coding decision). Furthermore, we used role prompting and chain-of-thought prompting to indicate to the LLM the task to perform step-by-step and constrain the output. Co-development and refinement of the AI-adapted CP codebook was done through the ChatGPT user interface. We followed Xiao et al.'s (2023) guidelines to develop the prompts and the codebook. The prompts used in the study are in Table 2.

4.3.3. Step 3: Training dataset creation

We created a subset at random from the total pool of data to train the LLM. This random procedure was followed to look for a balanced testing dataset. Specifically, 15 % of the text data was drawn at random, which corresponded to 45 paragraphs that were our unit of analysis. The remaining data was set apart for human content analysis with the final codebook and for testing the Generative AI prediction of codes and reasoning.

Table 1
AI-adapted codebook for cognitive presence based on the practical inquiry model (Garrison et al., 2001; Shea et al., 2010).

Structure of Code book	Text used in Large Language Models
Triggering Event Phase	Code: Triggering event.
Definition	A triggering event is evocative and stimulates curiosity about a concept or problem. It addresses a dilemma or elicits questions that a student wants to address from their previous experience or previous studies. It assesses the state of learners' knowledge and generates unintended but constructive ideas.
Indicators & Socio-cognitive process Included in Few-Shot Prompt	Include messages in which students recognize problems by presenting background information before posing a question. Include messages in which students display a sense of puzzlement by asking questions or messages that take the discussion in a new direction.
Example	Example: "It has been argued that the only way to deliver effective distance education is through a systems approach. However, this approach is rarely used. Why do you think that is?"
Exploration Phase	Code: Exploration
Definition	Exploration is inquisitive and focuses on understanding the nature of a problem, then searching for relevant information and possible explanations. In the exploration phase, students do group activities like brainstorming and private activities like literature searches. Students also manage and monitor this phase of divergent thinking in such a way that it begins to be more focused.
Indicators & Socio-cognitive process Included in Few-Shot Prompt	Include messages that show agreement or disagreement with previous ideas (e.g., "good point" or "I agree") without substantiated elaboration. Exclude messages that show personal experience that is substantiated. Include messages in which many different ideas/themes are presented in one message. Include information exchange (e.g., personal narratives, descriptions of facts from articles or websites) that add to a point but do not systematically defend/justify/develop the argument. Include messages about offering suggestions for consideration where explicitly characterize the message as exploration, for example, with phrases like "just thinking out loud," "here's a thought," "what if," and "how about." Include messages in which students leap to conclusions by offering unsupported opinions.
Example	Example: "One reason I think it is seldom used is that it is too complicated to get a cooperation. Another may be the mindsets of those in change practices."
Integration Phase	Code: Integration
Definition	Integration is tentative. In the integration phase, students focus and structure their messages by making meaning. In the integration phase, students make decisions about integrating ideas. Students are showing their understanding and misconceptions.
Indicators & Socio-cognitive process Included in Few-Shot Prompt	Include references to previous messages followed by the substantiated agreement or disagreement (e.g., "I agree/disagree because…"). Include messages that build on or add to others' ideas. Include messages that present justified, developed, defensible, yet tentative hypotheses. Include messages that integrate information from one or more sources (i.e., textbook, articles, personal experience, other posts from peer contributions). Include messages that explicit characterizations of a solution by a participant.
Example	Example: "We also had trouble getting cooperation. Often the use of new tools requires new organizational structures. We addressed these issues when implemented a systems approach, and I think that's why we were successful."
Resolution Phase	Code: Resolution
Definition	Resolution of the dilemma or problem is presented. In the resolution/application phase, students reduce the complexity of an idea/problem/issue by constructing a meaningful framework or discovering a contextually specific solution.
Indicators & Socio-cognitive process Included in Few-Shot Prompt	Include messages in which students test a solution by direct or vicarious application in the real world. Include messages in which students defend why a problem was solved in a specific manner.
Example	Example: "How we solved this problem was…"

Note. No citations were included in the prompt. The one-shot prompt excluded the indicators and socio-cognitive processes. This codebook differs from traditional codebooks in content analysis by integrating the code, definition, indicators, and examples vertically rather than in separate columns.

Table 2
Prompt used for large language model content analysis of cognitive presence.

Prompting Technique	Prompt Text
Role prompting	You are a qualitative coder who is annotating a paragraph from an online students' discussion board post based on the Practical Inquiry Model.
Chain-of-thought prompting	To code this text, do the following: First, read the codebook and the text. Next, decide which of the four codes is most applicable and explain your reasoning for the coding decision.
Justification	Finally, print the most applicable code and your reason for the coding decision in about 75 words.
Format standardization	Use the following format: Code: "Triggering event" or "Exploration" or "Integration" or "Resolution" Reasoning: Codebook: {Codebook from Table 1}

4.3.4. Step 4: Human coding of training dataset and IRR calculation

The first two authors coded the training text data from Step 3 in two stages. At stage one, they coded 2.5 % ($n = 7$ paragraphs) of the data together to negotiate misunderstandings about the codebook. At stage two, they independently coded chunks of 2.5 % (n = 7 paragraphs) of the remaining training dataset until coding 15 % ($n = 45$ paragraphs) of the text data. At each coding interval, the researchers discussed disagreements based on the PIM codebook to ensure the categories were understood and consistently applied to the discussion board data. Researchers specifically utilized the coding scheme by Shea et al. (2010) to clarify potential overlaps between the conceptually similar phases of the PIM (e.g., exploration and integration) by coding the most salient phase in a mutually exclusive way. When potential overlaps of coding appeared, the researchers followed Garrison et al.'s (2001) recommended heuristics of coding down when the PIM phase was unclear and coding up if multiple phases were evident. The researchers classified each paragraph and wrote a brief reasoning when coding. Researchers' reasoning was unified and standardized using the codebook created in Step 4. Both human coders had experience applying the PIM coding scheme for research purposes and in their teaching.

Inter-rater reliability (IRR) was calculated between human coders iteratively in the training dataset, and common understanding was established once a weighted Cohen's k larger than 0.8 was reached between human coders. We used a weighted Cohen's k to account for proximity and similarity between some phases of the PIM (i.e., exploration and integration) and to penalize extreme miscoding (e.g., triggering event instead of resolution). We used the R package 'irr' version 0.84.1 developed by Gammer et al. (2019) for this purpose. We chose cubic weights because they impose a stronger penalty for bigger errors than squared weights do. This helps ensure that serious disagreements are adequately addressed, which is crucial for accurate phase identification in PIM. For the four PIM phases, our cubic weight values were 0, 1, 8, and 27, reflecting increasing levels of disagreement. This approach highlights the significance of larger errors while still recognizing smaller, nuanced similarities.

4.3.5. Step 5: Fine-tuning the LLM

After a common understanding was established, a LLM model was fine-tuned using the codebook and prompts designed in Step 2 and the subset dataset from Step 3 through the public OpenAI API. The training process of the LLM is called fine-tuning. Specifically, the dataset contained the prompt, discussion board input text, and expected completion (i.e., the correct code and reasoning from Step 4). We used a GPT Turbo model (GPT-turbo-0125) because it focuses on understanding and generating natural language or code and is optimized for chat. The fine-tuning process was done in the OpenAI platform for both the one-shot and few-shot prompts. We designed a conservative GPT model using a temperature of 0 to reduce the variability in responses, limiting the tokens to 100, setting the probability of using the most likely tokens to 1 (i.e., top P), and not penalizing the repetition in responses.

4.3.6. Step 6: Creation of testing dataset subset and human coding

The remaining 85 % of the dataset was set apart for testing GPT fine-tuned model.

Both researchers coded halves of the remaining dataset in preparation for the last step of model evaluation. The testing sample was comprised of 248 units of analysis.

4.3.7. Step 7: Prediction of codes using LLM with a default parameter model and a fine-tuned model

The prediction of codes of students' discussion board data was made in two rounds. First, the one-shot prompt and few-shot prompts developed in Step 2 were used to do the LACA of the entire dataset using a default parameter model as a baseline. Second, the process was repeated using the fine-tuned model in Step 5 only in the testing dataset set apart in Step 6. All predictions were made using the public OpenAI API.

4.3.8. Step 8: Evaluation of the LLM coding against human coding

We used IRR, time, and cost to evaluate the LLM coding. IRR was computed between the human codes made in Step 6 and generated codes using the LLMs. First, the interrater reliability between researchers' coding and the LLMs was calculated using Cohen's k. Additionally, we measured percent agreement and visualized the reliability of the models using a confusion matrix. Time invested in human and LLM coding was tracked throughout the steps to compare the cost and time investment between human coders and the LLMs and for efficiency calculation.

5. Results

The aim of this paper is to explore how Large-Language Model Assisted Content Analysis can efficiently and reliably automate the content analysis of CP compared to human work. In this section, we present the results of the comparison between the LLMs used, both the default parameter model and the fin-tune model, and human coding based on reliability and efficiency and the potential for automated content analysis.

5.1. Reliability

We measured the reliability of our LACA approach using LLMs based on Cohen's Kappa. First, the baseline IRR between both researchers was a weighted Cohen's k of 0.84, which indicated a very strong agreement (Landis & Koch, 1977). Then, we computed four sets of Cohen's k between the human coders and the ChatGPT model based on the prompt and whether the model was fine-tuned or not. The highest Cohen's k value between human coders and the LLM was at the moderate level (k = 0.59) when using the fine-tuned model with a one-shot prompt.

For the default parameter model, both codebooks achieved a fair agreement with human coders. The LACA approach achieved a fair agreement with a weighted Cohen's k of 0.36 with human coders when using the few-shots prompt that included the indicators of the CP. The one-shot prompt that did not include the CP indicators achieved a weighted Cohen's k of 0.39, increasing the IRR by 0.03. Such an increase is not negligible given that a Cohen's k is 0.41 or above is considered moderate (Landis & Koch, 1977).

Similar to the default parameter model, the LACA approach using a fine-tuned model with a one-shot prompt achieved a higher weighted Cohen's k of 0.59 with human coders compared to the few-shot prompt, which only achieved a weighted Cohen's k of 0.52. The fine-tuned model with the one-shot prompt improved by 0.07, doubling the improvement that occurred in the default parameter model. The increase is noteworthy because a Cohen's k value that is 0.61 or greater is considered substantial (Landis & Koch, 1977). The results show that the one-shot approach performs better than a highly structured few-shot approach with both default parameter and fine-tuned models.

Table 3 presents the percentage of agreement between LLMs and the researchers by PIM phase. The results show that LACA performed better when classifying students' discussion board text in the Integration phase of the PIM. This holds true for both the default parameter and the fine-tuned LLMs. Specifically, the percept agreement was above 75 % in all models for the Integration phase, with the highest at 89.9 % in the few-shot default parameter model. Conversely, the LACA approach did not show accurate results for the Resolution or Triggering Event phases. The LLMs performed the worst for the Resolution phase in both default parameter models with a 0 % agreement with researchers but an average agreement of 13.3 % in the fine-tuned models. When taking a closer look at the fine-tuned models, the few-shot prompt performed slightly better when classifying text in the Resolution phase, suggesting that the description in the prompt could be improved to increase accuracy. The

Table 3
Percentage of agreement by PIM phases between LLMs and researchers.

	Default Parameter Models			Fine-Tuned Models	
	One-shot	Few-shot		One-shot	Few-shot
Triggering Event (n = 35)	8.57 %	14.3 %	Triggering Event (n = 33)	12.1 %	6.06 %
Exploration (n = 103)	32.4 %	10.7 %	Exploration (n = 81)	37 %	29.6 %
Integration (n = 119)	80.5 %	89.9 %	Integration (n = 104)	87.5 %	76.9 %
Resolution (n = 36)	0 %	0 %	Resolution (n = 30)	10 %	16.7 %
Overall (n = 293)	45.2 %	42 %	Overall (n = 248)	51.6 %	44.8 %

models showed more consistency in the Triggering Event phase, with at least one correct classification in all models and an average agreement of 10.18 %.

Fig. 1 shows a confusion matrix comparing the classification of the researchers' classification of four phases of the PIM with the fine-tuned model classification using a one-shot prompt style, which was the best-performing GPT model. The shading of the cells represents the frequency. The darker the cell, the more students' paragraphs were coded in such a category. The diagonal from the upper left corner shows the correct coding made by the fine-tuned model. Magenta color was used to represent true positives and green for misclassifications. A divergent palette was used to put equal emphasis on extreme values. Reflecting the percent agreement in Table 4, the fine-tuned model performed best in the integration phase, correctly coding 91 students' paragraphs.

Fig. 1 shows that the difficulty differentiating between the exploration and integration phases was greater compared to the rest of the misclassifications. Specifically, the model misclassified 48 students' paragraphs as integration when they were actually in the exploration phase, and conversely, 13 paragraphs as exploration when they were in the integration phase. The second misclassification patterned emerged between the phases of exploration and triggering event, in which the model misclassified 28 paragraphs in the exploration phase when they were in the triggering event phase and only one as triggering event when researched coded as exploration. The third greatest misclassification was between integration and resolution in which the model misclassified 22 paragraphs as integration when they were in the resolution phase.

Fewer misclassifications occurred between researchers' coding and GPT model classification between the following phases: exploration and triggering event ($n = 1$), resolution and exploration ($n = 5$), triggering event and integration (n = 1), and exploration and resolution ($n = 2$). There was greater reliability in extreme phases; for example, no misclassification occurred between the integration and resolution phases with the triggering event phase.

5.2. Efficiency

Table 4 presents detailed figures comparing the LACA approach to researchers coding in terms of efficiency as measured by resources invested (i.e., time and cost). As expected, the LACA approach is more efficient at the expense of accuracy, but our calculations explicitly detangle the extent of the cost-saving opportunity. The calculation of researchers' cost was based on the hourly rate for two assistant professors in the United States, as reported by ZipRecruiter (2024). Overall,

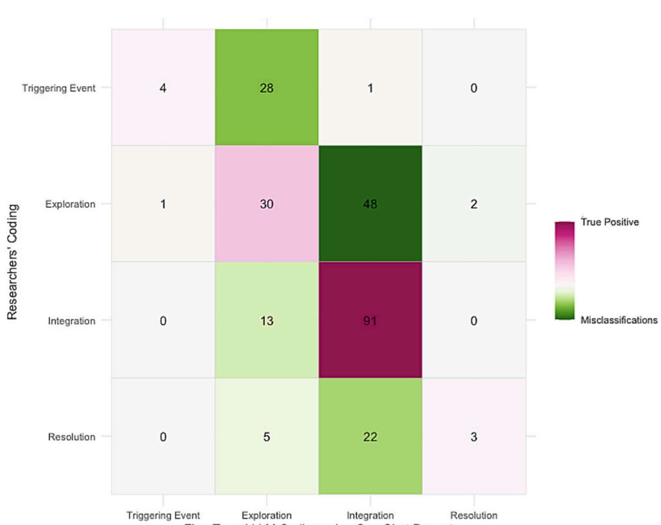

Fig. 1. Confusion matrix of the LACA of the four phases of the PIM using a trained GPT model with a one-shot prompt.

Table 4
Efficiency of large language model content analysis approach for cognitive presence analysis compared to researchers coding.

	Two Trained Researchers	Default Parameter Models		Fine-Tuned Models	
		One-shot	Few-shot	One-shot	Few-shot
Training Time				7 min 14 s	6 min 26 s
Coding Time	5 h 48 min	1 h 35 min	1 h 33 min	2 min 16 s	2 min 54 s
Training Cost[a]				69 cents	$1.04
Coding Cost[a]	$ 311	13 cents	21 cents	88 cents	60 cents

Note. Trained researchers' cost calculation is based on the average hourly rate in the United States (ZipRecruiter, 2024). Trained researchers' time of coding includes writing reasoning. [a] Cost is calculated in United States dollars.

the cost was reduced from hundreds of dollars to less than two dollars, with fine-tuned models being the most efficient. Specifically, the training for the fine-tuned models ranged between 69 cents to $1.04 dollars for the one-shot and the few-shot models, respectively. Both fine-tuned models' coding costs were under 90 cents. Regarding time, the fine-tuned model resulted in a more than 50 % reduction in time compared to the default parameter model. The most significant increase in efficiency was in both fine-tuned models, which were under ten minutes, including training and testing time. Specifically, training both models was under 8 min, and automated coding time was under 3 min.

5.3. Generative AI potential for CP automated content analysis

Including the reasoning as part of the coding in the LACA approach allowed us to investigate additional insights about the use of Generative AI specific to Cognitive Presence. Table 5 shows examples of agreement and rationale between researchers and the one-shot prompt LLMs. Perhaps the most noticeable change is that the reasoning written by researchers is shorter than the reasoning produced by the LLM. It is worth adding that although Table 5 displays researchers agreeing with both LLMs, the rationale shows that the classification could be done for different reasons. Albeit both rationales are correct in the Integration example, the researchers' rationale was "explicit characterizations of a solution," whereas both LLMs' argument was around "showing their understanding and misconceptions." Finally, the output of the fine-tuned model shows that standardization in the reasoning improves accuracy but decreases the specificity and variety in the rationale.

6. Discussion

We examined how LACA assisted CP classification in online learners' discussion text data. Specifically, discussion posts collected from an online course delivered through case-based instruction were examined using multiple GPT algorithms for the automated content analysis of CP. This study's results give insight into CP analysis beyond classification and identify how the existing CP codebook could be reduced to increase content analysis efficiency.

To address our first research question, we compared two GPTs models to the researchers' content analysis of discussion board data to inspect reliability. Despite the fact that previous researchers explored the automated content analysis of CP using machine learning algorithms achieving high agreement between humans and machine learning (Ba et al., 2023; Hayati et al., 2019; McKlin, 2004); the results from our GPT best-performing fine-tuned model using a one-shot prompt achieved only moderate agreement. Yet, our results are comparable to those of Kovanovic (2017) exploration using a random forest classifier. Given the potential of LLMs to decrease the complexity of the modeling process,

Table 5
Examples of coding reasons for agreement between researchers and the Large Language Model Content Analysis approach using the one-shot prompt for both default parameter and the fine-tuned models.

Example Paragraph per PIM phase	Researchers	One-shot Default Parameter Model	One-shot Fine-Tuned Model
Triggering Event "I noticed the case said the teachers were using the techniques 'for the most part.' Do you think it would be helpful to understand why they weren't using their previous trainings to fidelity?"	Include messages in which students recognize problems by presenting background information before posing a question.	The student's question about why the teachers were not using their previous training to fidelity indicates a triggering event. The student is curious about the discrepancy between the teachers' training and their actual practices, which prompts them to seek understanding and address a potential dilemma. This question reflects a desire to explore the reasons behind the teachers' behavior and suggests a curiosity to delve deeper into the topic.	The student asks a question based on the information provided in the scenario.
Exploration "I'm curious what types of learning checks you think would be most useful? The simulations mentioned on p. 43 of Clark & Mayer and p. 593 in Deepika et al. spring to mind for me. If participants can effectively apply their learning to a scenario, that suggests they are more likely to enact the desired behavior changes."	Exploration – Include messages in which the students is offering suggestions for consideration where explicitly characterize the message as exploration, for example, with phrases like "just thinking out loud," "here's a thought," "what if," and "how about."	The student is in an exploratory phase, seeking to understand the types of learning checks that would be most useful. They mention specific examples from readings and connect them to the idea of participants applying their learning to scenarios. This phase involves searching for relevant information and possible explanations, which aligns with the student's inquiry about learning checks and their effectiveness.	The student is inquisitive and focuses on understanding the nature of a problem, then searching for relevant information and possible explanations.
Integration "I think that it could have been beneficial for David to pause and at least take time to create a Gantt chart for the project, as described by Morrison et al. on page 418. The project was incredibly time-sensitive, but taking this time on the front end to plan and be intentional could have paid off in the long run. A Gannt chart would allow David and his team to determine which milestones are dependent on others and best allot resources to reach their goal by the deadline, while not missing any crucial steps of the design process."	Include messages that are explicit characterizations of a solution by a participant.	The student is reflecting on the situation presented in the text and suggesting a specific solution based on the information provided by Morrison et al. They are structuring their message by proposing the use of a Gantt chart to address the time-sensitive project effectively. This integration phase involves making decisions about integrating ideas and showing understanding of the problem at hand.	The student is showing their understanding and misconceptions.
Resolution "This lesson is an example that will be drafted by teachers in Oakes' online, asynchronous, and synchronous professional development. Here, each teacher in the groups would have to agree on a topic that they all teach, or plan to teach, and draft this constructivist approach lesson. The teachers would then need to present this to their students after the students understand the basic concepts behind photosynthesis and the environmental factors that can affect photosynthesis as described in the Cynefin framework of knowledge management under the "known" and	Include messages in which students defend why a problem was solved in a specific manner.	NA	The student presents a solution to a problem.

(continued on next page)

Table 5 (*continued*)

Example Paragraph per PIM phase	Researchers	One-shot Default Parameter Model	One-shot Fine-Tuned Model
"knowable" domains (Aylward & Cronjé pg. 508–510). This known domain also applies to the topic as it describes a "cause and effect model. The teachers will allow the students to brainstorm in groups (preferably of 2–3). As for the constructivist outcomes, there are no wrong answers (as long as they are realistic), and there could be many possible answers, which is why the students have the option of listing more in the chart. As the groups brainstorm their ideas, it will be important that the teacher monitors groups and asks questions that engage students in lower-level ideas and higher-order thinking."			

Note. There are no examples in which the default parameter GPT model and the researchers agreed on the Resolution phase.

we pose that the LACA approach used in this study could potentially increase access to automated algorithms to more researchers.

We speculate that using Cohen's *k* to calculate IRR could potentially yield lower reliability scores despite high agreement. Specifically, Chew et al. (2023) noted the "high agreement, low reliability" (p. 7) paradox when using Cohen's *k*, where even a few disagreements can result in significant discrepancies in IRR values. However, we argue that our approach using Cohen's *k* provides a more robust measure of IRR when applying LLMs for automated content analysis compared to earlier pioneering research that relied on the LACA approach. Further, foundational studies in CP in computer-mediated communication have used alternatives to Cohen's kappa, as "symmetrical imbalances in the marginal distributions of the coding table can lead to high levels of observed agreement but a very low kappa" (Shea et al., 2010, p. 12). In such a case, Shea et al. (2010) used Holsti's Coefficient of Reliability, which emphasizes percent agreement. Based on foundational empirical studies suggesting percent agreement as a viable alternative to Cohen's *k*, we included it in this study to provide a comprehensive view of the effectiveness of LLM applied to CP facilitation in higher education. Despite these challenges, we argue that Cohen's *k* remains a more conservative and rigorous approach to evaluating LACA effectiveness, as it provides a more stringent assessment of accuracy. This conservatism is crucial to avoid overestimating the effectiveness of LLMs in educational research.

Regarding IRR, we speculate that simplifying the CP codebook by eliminating the indicators improves the differentiation between neighboring phases (e.g., exploration and integration) of the PIM and could eventually improve the IRR with LLM. For example, when comparing the one-shot vs the few-shot prompt, our results showed that the one-shot prompt performed best in both fine-tuned and default parameter models possibly because the one-shot prompt did not include the indicators for CP. Our results agree with previous work that evaluates one-shot prompts to other prompts (e.g., Bezirhan & Von Davier, 2023; Xiao et al., 2023). Altogether, the results from the one-shot prompt have profound implications because we can suggest that the decade-long CP codebook could be simplified by excluding indicators without sacrificing clarity.

We found misclassifications in the LACA approach of the one-shot fine-tuned model between neighboring phases. Specifically, between exploration and integration, triggering event and exploration, and integration and resolution. The misclassification between exploration and integration phases could be due to the ambivalence between the categories was previously discussed in the literature, resulting in Shea et al.'s (2010) modification of the original codebook by removing the words "divergent" from the exploration phase and "convergent" from the integration phase to improve clarity. Misclassification between triggering event and exploration phases could be explained by the small sample of triggering event paragraphs in the training set. Yet, it is worth saying that the triggering event and resolution phases were not misclassified from each other. Therefore, the codebook is descriptive and distinctive enough between both seemingly extreme phases. The misclassifications between integration and resolution phases indicate potential ambiguity between such categories or due to the small sample size of resolution events previously reported in the literature (Sadaf et al., 2021; Cleveland-Innes & Garrison, 2021). Just as Shea et al.'s refinement of the codebook in 2010 enhanced clarity, we invite online learning researchers to propose small but significant improvements in the prompt that could increase Cohen's *k* from fair to moderate. We speculate that LLMs could serve in the quest of evaluating small tweaks to the codebook.

When answering our second research question, the LACA approach using the LLMs as part of the OpenAI suite seems more efficient than human coding in terms of both time and financial resources at first glance if one refers to the ChatGPT interface as we used in Step 2 for the co-development of the AI-adapted CP codebook. However, we urge researchers not to jump to conclusions based solely on such figures because the LACA approach goes beyond GPT integration and requires prior knowledge of data processing and programming languages to use the OpenAI API to automate the analysis of large batches of data (steps 5 and 7). In other words, the efficiency analysis of the LACA model does not account for the hours of training necessary to gain such knowledge. Furthermore, it is hard to calculate how much time it takes for a researcher to acquire such expertise and the deliberate practice required to implement the automation. We speculate that default parameter models with highly refined codebooks could be more efficient when compared to the hidden cost of implementing fine-tuned models. For example, default parameter models in OpenAI Playground allow for enough model specification (e.g., setting temperature parameters) in a more user-friendly way that could more easily reach practitioners than fine-tuned models.

Nevertheless, fine-tuned models could be more efficient for practitioners regarding financial costs. The pricing of the GPT model family is based on the tokenization of the text-data processing. Tokenization in this context means chunking the text data into a sequence of letters that often appear together. As an example, the codebook had 660 tokens, and including the instructions in the few-shot prompt was 783 tokens. The model used in this study charged 50 cents for one million tokens of input and $1.50 for one million tokens of output. Cost seems minimal, but as words accumulate in 1) the codebook (system input), 2) discussion board data (user input), and 3) the categorization (model output or completion) for default parameter models in each iteration, the cost could add up very quickly if practitioners are not careful. Yet, fine-tuned models are priced once, and then, practitioners are billed only for the discussion board data (data input) and classification (output or completion). Therefore, fine-tuned models could be not only more accurate but also a better financial option for when using vast data. We invite the field to further this investigation by creating GPTs and Assistants as stand-alone applications with embedded training datasets. No-programming-based tools could increase the reachability of CP analysis by increasing and setting content analysis at the hands of any instructor.

This study using LACA provides additional insights into the CP codebook and researchers' work on CP. First, we made syntax modifications to the widely validated CP codebook (Garrison et al., 2001; Shea et al., 2010) during prompt engineering. Specifically, we transformed descriptors of indicators (e.g., leaps to conclusions) and socio-cognitive processes (e.g., offers unsupported opinions) of CP into imperative sentences (e.g., Include messages in which students leap to conclusions by offering unsupported opinions) to produce an AI-adapted CP codebook. This narrative transformation is essential for LLMs, for their performance is better when using sentences rather than phrases. Despite the existence of other CP codebooks (e.g., (Guo et al., 2021), we adhered to Garrison et al.'s (2001) foundational work and Shea et al.'s (2010) update to encourage conceptual replicability of this study in other contexts and with other LLMs like Gemini, LLaMA, or Claude. Further replication of the current study could also include more context-specific codebooks that could potentially remediate the LLMs' misspecification between neighboring phases mentioned above.

Second, this study included the written rationale per classification from researchers for the fine-tuned model. Table 5 showed that the researchers' reasoning was shorter than the model's output. This could be explained by the coding agreement process among researchers, which includes conversation but does not typically translate into written reasoning about why a text is classified in a particular manner. Consequently, the additional work for researchers in writing the rationale for coding was unusual. Furthermore, the instructions for the LLM explicitly asked for a brief rationale to be written in about 75 words. Regarding the content of the rationale, the default parameter models' output included more details about the coding rationale, which could be beneficial for novice researchers and instructors unfamiliar with the PIM and as validity checks. Paradoxically, the standardization of the reasoning in the fine-tuned models improved accuracy but decreased variability in the reasoning.

It is important to note that although there is a tendency to explain the phases of the PIM in a hierarchical structure from triggering event to resolution, such conceptualization lacks theoretical and empirical support. A better conceptualization of the phases of PIM is of *probability* rather than *hierarchy,* as Garrison et al. (2001) initially conceptualized them as "an idealized construction" (Shea et al., 2022, p. 150) that should not be interpreted linearly. Further, Shea et al. (2022) clarify that online learners alternate among the PIM phases while they make sense of the world through internal reflection and collaborative discourse. At an empirical level, Hrastinski et al. (2023) studied online learners' transitions among the four phases of the PIM and did not find evidence of the idealized transition among phases but rather iteration among them. Using First-Order Markov Models, they identified exploration as the predominant phase preceding transitions to other phases. Their findings align with Richardson et al. (2021), who noted that given that CP is a collaborative meaning-making process that emphasizes exploration and integration phases, but once resolution is achieved, there is little room for collaboration. Shea et al. (2022) speculate that students could achieve integration and resolution phases through well-designed prompts. Sadaf & Olesova, 2017 found that prompts designed with the PIM in mind lead to higher integration and resolution phases in online learners than generic prompts. In our study, prompts were crafted to guide students through the phases of the PIM. Therefore, we speculate that our sample dataset, with a prompt designed to produce a higher-than-average level of students' resolution phase output, strengthens the robustness of our insights.

Finally, we emphasize that LACA approaches are not intended to replace manual coding but can serve as valuable complementary tools for those new to qualitative coding and, especially, those unfamiliar with facilitating CP in the context of the PIM. For example, Gencoglu et al. (2023) demonstrated the importance of expert manual coding to validate the results of automated content analysis approaches, such as the application of topic modeling to qualitative instructor evaluations. Their findings highlight that manual coding cannot be overlooked because it helps researchers identify nuanced details in rich data sets, particularly when working with big data, in their case, a national sample (Gencoglu et al., 2023). When working with extensive text data, LACA approaches play a critical role as a researcher's tool to confirm theoretical dimensions (Tai et al., 2024), while manual coding keeps a culturally sensible understanding of the data's unique context. Further, manual coding supports researchers in identifying additional constructs in rich data that might not emerge through automated methods alone (Tai et al., 2024). In the context of CP, LACA approaches should not be seen as a replacement for manual coding for facilitating online experiences but rather support those instructors new to the phases of the PIM. For experienced instructors, LACA has the potential to augment their ability to analyze data efficiently while still preserving the depth and context that manual coding provides.

7. Implications for instructors, students, and practitioners in higher education

The purpose of this study was to investigate how an AI-adapted CP codebook could be used in conjunction with LLMs for a reliable and efficient classification of online learning processes through the four phases of the Practical Inquiry Model. The results of this work have implications for practice at the instructor, student, and practitioner level. Specifically, this work has a direct impact on four scenarios: 1) instructors who are not trained in distance education and qualitative research, 2) instructors who want to measure CP in their classroom, 3) students using the AI-adapted CP codebook as self-assessment, and 4) higher education stakeholders for faculty professional development.

First, this work aids instructors in higher education who are new to online teaching and qualitative research. Until now, the application of the CP codebook, and consequently the PIM, has been limited to trained researchers in distance learning and those within the humanities to support critical thinking in computer-mediated communication (Lim, 2018). The AI-adapted CP codebook intends to broaden the CP's application beyond research for those not trained in distance learning or qualitative inquiry, such as educators and researchers in STEM disciplines, where CP has had little application (Lim, 2018). Because the original CP codebook was adapted using prompt engineering, the AI-adapted CP codebook is more descriptive and precise in defining PIM phases. For example, imperative sentence structures like "Include messages that …" are used for the indicators of each PIM phase.

Second, this research can assist higher education instructors in gauging their students' CP in their online classrooms. Further, in discussions about LLMs in education, it is crucial to address their accuracy and the urgent need for a precision standard commonly understood by educators and practitioners alike. Without these standards, the potential for misuse is significant. We argue that the IRR is an intuitive measure of LLMs accuracy for LACA and indicates that applying the AI-adapted CP codebook would help any novice instructor utterly unfamiliar with the PIM gain a more accurate understanding of CP. This would, in turn, support instructors in helping students improve their participation in online discussions. Hence, with the AI-adapted codebook, instructors can identify which phase of the PIM students are in and frame their feedback in a way that students move through all the phases of the PIM and eventually reach the elusive *resolution* phase.

Third, this work segues the conversation from traditional applications of CP, which have centered on researchers, by empowering learners with an AI-adapted CP codebook that can serve as a self-assessment tool. Students can input their work into an LLM and employ the AI-adapted CP codebook to understand the quality of their work in light of the PIM. This approach aids in self-directed learning and serves as a guideline for effective participation in online discussions. It supports students in diagnosing their work, engaging in co-regulated learning, and assisting peers in constructing knowledge, thereby fostering a collaborative learning atmosphere (Garrison, 2022). With the AI-adapted CP codebook, students truly move from passive spectators to agents of their learning processes and knowledge co-creators with their peers. For example, in case-based learning scenarios, where students often facilitate discussions, the codebook can also assist them in their role as facilitators. By guiding themselves with the CP indicators and engaging in shared metacognitive practices, students can deepen their understanding and application of course content, optimizing their online learning experiences (Garrison, 2022). Ultimately, given the positive relationship between CP and students' outcomes (Martin et al., 2022), the proposed AI-adapted CP codebook has direct implications for improving learners' performance in online learning.

Fourth, the AI-adapted CP codebook has implications for higher education stakeholders who must provide faculty professional development in online teaching and learning. Specifically, the proposed codebook can inform faculty on the job and encourage the adoption of theory-based teaching practices through data-driven insights. When tasked with teaching online, instructors in higher education generally experience anxiety regarding the technological aspects of delivering instruction at a distance, which may drive them to focus on acquiring technological skills rather than pedagogical expertise (Richardson et al., 2023). Nevertheless, it is of utmost importance for administrators in higher education to provide theory-informed professional development that synthesizes best practices for teaching and learning online for new instructors (Richardson & Alsup, 2015; Hodges et al., 2020). At the institutional level, we posit that a tool like the AI-adapted CP codebook can bridge the theory-practice gap by going beyond describing what CP means to give instructors a tool for understanding students' CP processes by identifying PIM phases.

8. Limitations and future research

Our sample was limited to the uniqueness of the context of case-based instruction and the discussion prompts designed based on the PIM. The text data produced by students could be different from the general lecture-type courses. Specifically, in case-based discussions and PIM-based prompts, students are required to have stronger argumentation skills, which could make learners produce richer and longer text excerpts. We want to note that students' decisions to use bold, italics, and bullet points were lost in data preprocessing. As usual, an instructor can account for such nuances, but using the LACA approach through LLMs cannot do so.

The prompt used in this study focused only on content analysis and rationale but did not include instructional advice for instructors, for example, by providing follow-up questions for students. Asking students probing questions is an essential component of supporting meaning-making, the ultimate goal of CP. Likewise, this study focused on a tool for instructors rather than a tool for learners. In our upcoming research, we are modifying the prompt by training the LLM to produce an open-ended question that could make the student move to another phase of the PIM in a Socratic seminar fashion (Castellanos-Reyes, 2020b). Specifically, we are using questions that foster critical thinking (Richardson et al., 2012) to guide metacognitive scaffolding in online learning (Richardson et al., 2021). Further, our work will focus on discerning how LLM instructional advice plays with diverse instructional strategies (i.e., case-based learning, debate, or open-ended questions) in online learning (Richardson & Ice, 2010).

Our main methodological limitation was our use of exclusive coding, which meant that each paragraph could have only one code. Using exclusive codes allowed us to have a better understanding of GPT accuracy and make direct comparisons to researchers' codes. However, we acknowledge the blurred boundary between exploration and integration and between exploration and resolution phases, respectively. For example, Shea et al.'s (2010) coding scheme explicitly discussed two codes that were on the borderline between phases. Specifically, "integration among group members" (p. 20) between exploration and resolution and "vicarious application to real world testing solutions" (p. 20) between integration and resolution. Future research could explore inclusive CP codes and their implication for LACA accuracy with more prompt engineering, for example, by integrating within the prompt Garrison et al. (2001) recommendation of coding one phase down when unsure which phase to assign and one phase up if multiple phases were clearly present. Nevertheless, including longer prompts in already extensive codebooks might decrease efficiency and increase costs for instructors. Another option could be to give LLMs the option not to assign any phase. For example, Chew et al. (2023) assert that LLMs might not have enough information in the codebook to make decisions about specific codes. Therefore, they recommend that future research could add the code "I do not know" and demand a justification from the LLM to explore such cases. However, emergent research consistently notes that LACA performs better with mutually exclusive codebooks (e.g., Chew et al., 2023; Dunivin, 2024).

Our work faces challenges regarding imbalance in the dataset, a common challenge in studies utilizing ML algorithms. Imbalanced datasets arise when one variable is significantly underrepresented in the training set (Boehmke & Greenwell, 2019). In our case, two of the CP phases (i.e., triggering event and resolution) were less prevalent in our training dataset. However, an equal representation of the four phases would present a distorted view of the distribution of naturally occurring phases of the PIM in text-based discussion boards. Future research could explore ML techniques, such as upsampling, to increase the representation of less common PIM phases. Another approach to expanding the dataset could be increasing granularity by shifting the unit of analysis from paragraphs, as used in the present study, to sentences, as suggested by Ba et al. (2023).

The lower number of resolution paragraphs could be attributed to the frequent critique of the PIM, where researchers lament the lack of students reaching the resolution phases (Sadaf et al., 2021; Cleveland-Innes & Garrison, 2021). Nevertheless, Garrison et al. (2001) clarified in the conceptualization of CP that the exploration phase is a more conducive environment for students, leading to more text production than the integration and resolution phases. Consequently, Richardson et al. (2021) concluded that the aim of CP is collaborative meaning-making that stimulates the exploration and integration phases, and that once students attain the resolution phase, there is limited scope for further collaboration. Future research could explore how LLMs could assist instructors in better understanding resolution phases.

Given the accelerated evolution of Generative AI models, it is challenging for researchers to foster high-fidelity reproducibility of AI-assisted content analysis. In other words, what LLMs helped us do at the time this research was done (March 2024) could vary within months. Yet, LLMs are probabilistic, implying that the results will change as more data is used to create them, imposing replicability issues. Nevertheless, researchers can refer to the specific model we implemented and take a conservative stance in LLM settings. For example, setting the temperature parameter to 0 to reduce variability in output, limiting tokens, and setting the probability of using the most likely token to 1. Achieving reliability in LLM probabilistic applications requires careful model setting, prompt engineering, and fine-tuning. We argue that it is through accumulated reproducible research that science can move the boundaries of knowledge forward. Therefore, we encourage educational researchers to keep applying the AI-adapted CP codebook in other contexts and disciplines.

Currently, the codebook developed in our research can be used in combination with mainstream LLMs, providing immediate utility for instructors seeking to enhance CP in their online classrooms. Despite this, the integration into institutional learning management systems may remain a barrier for under-resourced institutions due to technical and financial constraints (Richardson et al., 2023). One direct implication for instructors stems from our ongoing design-based research project, which builds on this study's findings to develop an open-source ChatBot using LLMs specifically designed to facilitate CP in educational settings. This tool will support instructors by providing automated, real-time analysis and feedback on student engagement levels during class discussions and activities. By leveraging the power of LLMs, the ChatBot will assist educators in identifying key areas where students may need additional guidance or resources, thereby enhancing the overall effectiveness of their teaching strategies. The ChatBot is expected to be completed by the end of 2025, and we anticipate that it will significantly streamline the process of coding CP, ultimately allowing instructors to focus more on pedagogical innovation and direct interaction with students. The present study provides the theoretical and technical basis for continuing our research agenda, which aims to empower educators to harness cutting-edge technology to foster a more interactive and reflective classroom environment conducive to CP.

9. Conclusions

This study addressed worldwide calls about speculating narratives on the use of AI in educational research (Bozkurt et al., 2023) and AI on a broader landscape by integrating LACA to automate content analysis of CP from discussion board data. Perhaps the most relevant question that this article addressed was not only about how AI could classify CP but also how – or if – AI implementation could change the experience of online learning instructors and students.

First, after comparing a one-shot vs. few-shot prompt with a default parameter model and fine-tuned model based on IRR, this study suggests that practitioners could leverage the AI-adapted codebook for CP with a one-shot prompt to increase the accuracy and reliability of the LACA approach. Given that CP centers on meaning-making as the ultimate purpose of online learning experiences, it is crucial to emphasize the collaborative cognitive process that occurs during discussion boards in case-based learning. Therefore, we speculate that practitioners could use the AI-adapted codebook for CP as a diagnostic tool for groups rather than singling out students to emphasize a community-centered approach in online teaching and learning.

Second, this study shows that LACA approaches offer many affordances to increase the efficiency of CP research. From a research management perspective, we found that using a simplified AI-adapted codebook for CP brings a significant cost savings opportunity and yields moderate reliability. Third, our results support previous work that identifies the fuzzy boundaries between the integration and exploration phases of the PIM, suggesting that potential simplification of the model is possible. Given the importance of scaffolding in online learning (Richardson et al., 2022), future work could go beyond classification and include instructional suggestions for practitioners fostering a Socratic seminar style (Castellanos-Reyes, 2020b).

Finally, using large-language models like those in the OpenAI rather than more traditional machine-learning algorithms could make automated content analysis more accessible to researchers and practitioners at large. Our approach to content analysis of CP through LLMs reflects the yearning to move the online learning field from reflective research for future improvement to real-time diagnostic and just-in-time support. Future research could expand on including all presences of the CoI framework and stand-alone AI-powered assistants that automate the process.

CRediT authorship contribution statement

Daniela Castellanos-Reyes: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Larisa Olesova:** Writing – review & editing, Writing – original draft, Validation, Investigation, Formal analysis, Data curation. **Ayesha Sadaf:** Writing – review & editing, Writing – original draft, Validation, Resources, Investigation.

Declaration of competing interest

None.

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Acknowledgments

DCR is grateful for Dr. Rathziel Roncancio Reyes' assistance with code debugging and Drs. Andy Parra-Martinez and Sandra Liliana Camargo Salamanca for their feedback on earlier drafts of this manuscript. We sincerely appreciate the time and thoughtful feedback provided by all the reviewers. In particular, we are deeply grateful to Reviewer 1 for their constructive and encouraging comments, which not only helped us improve the manuscript but also made the revision process especially rewarding.

Data availability

The authors do not have permission to share data.

References

Akyol, Z., & Garrison, R. (2011). Assessing metacognition in an online community of inquiry. *Internet and Higher Education, 14*(3), 183–190. https://doi.org/10.1016/j.iheduc.2011.01.005

Ba, S., Hu, X., Stein, D., & Liu, Q. (2023). Assessing cognitive presence in online inquiry-based discussion through text classification and epistemic network analysis. *British Journal of Educational Technology, 54*(1), 247–266. https://doi.org/10.1111/bjet.13285

Barbosa, A., Ferreira, M., Ferreira Mello, R., Dueire Lins, R., & Gasevic, D. (2021). The impact of automatic text translation on classification of online discussions for social and cognitive presences. In *LAK21: 11th international learning analytics and knowledge conference* (pp. 77–87). https://doi.org/10.1145/3448139.3448147

Bezirhan, U., & Von Davier, M. (2023). Automated reading passage generation with OpenAI's large language model. *Computers and Education: Artificial Intelligence, 5*, Article 100161. https://doi.org/10.1016/j.caeai.2023.100161

Boehmke, B., & Greenwell, B. (2019). *Hands-on machine learning with R* (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9780367816377

Bozkurt, A., Xiao, J., Lambert, S., Pazurek, A., Crompton, H., Farrow, R., … Jandrić, P. (2023). Speculative futures on ChatGPT and generative artificial intelligence (AI): A collective reflection from the educational landscape. *Asian Journal of Distance Education, 18*(1), 53–130.

Castellanos-Reyes, D. (2020a). 20 years of the Community of Inquiry framework. *TechTrends, 64*, 557–560. https://doi.org/10.1007/s11528-020-00491-7.

Castellanos-Reyes, D. (2020b). Socratic Seminar. In R. Kimmons (Ed.), *The Students's Guide to Learning Design and Research*. https://edtechbooks.org/studentguide/socratic_seminar.

Chen, H.-L., & Chang, C.-Y. (2017). Integrating the SOP2 model into the flipped classroom to foster cognitive presence and learning achievements. *Educational Technology & Society, 20*(1), 274–291. https://www.jstor.org/stable/jeductechsoci.20.1.274.

Chew, R., Bollenbacher, J., Wenger, M., Speer, J., & Kim, A. (2023). LLM-assisted content analysis: Using large language models to support deductive coding (arXiv: 2306.14924). *arXiv*. http://arxiv.org/abs/2306.14924.

Cleveland-Innes, M., & Garrison, D. R. (2021). *An introduction to distance education: Understanding teaching and learning in a New Era*. https://doi.org/10.4324/9781315166896-10

Dewey, J. (1933). *How we think*. Prometheus Books.

Dunivin, Z. O. (2024). Scalable qualitative coding with LLMs: Chain-of-thought reasoning matches human performance in some hermeneutic tasks (no. arXiv: 2401.15170). *arXiv*. https://doi.org/10.48550/arXiv.2401.15170

Ekin, S. (2023). *Prompt Engineering For ChatGPT: A Quick Guide To Techniques, Tips, And Best Practices*. https://doi.org/10.36227/techrxiv.22683919.v2

Ferreira, M., Rolim, V., Mello, R. F., Lins, R. D., Chen, G., & Gašević, D. (2020). Towards automatic content analysis of social presence in transcripts of online discussions. In

Proceedings of the tenth international conference on learning analytics & knowledge (pp. 141–150). https://doi.org/10.1145/3375462.3375495

Gammer, M., Lemon, J., Fellows, I., & Singh, P.. irr: Various Coefficients of Interrater Reliability and Agreement(Version 0.84.1) [R package]. https://CRAN.R-project.org/package=irr.

Garrison, R. (2017). *E-learning in the 21st century: A community of inquiry framework for research and practice*. Routledge. https://doi.org/10.4324/9781315667263

Garrison, R. (2022). Shared metacognition in a community of Inquiry. *Online Learning, 26*(1). https://doi.org/10.24059/olj.v26i1.3023

Garrison, R., Anderson, T., & Archer, W. (2001). Critical thinking, cognitive presence, and computer conferencing in distance education. *American Journal of Distance Education, 15*(1), 7–23. https://doi.org/10.1080/08923640109527071

Gašević, D., Adesope, O., Joksimović, S., & Kovanović, V. (2015). Externally-facilitated regulation scaffolding and role assignment to develop cognitive presence in asynchronous online discussions. *Internet and Higher Education, 24*, 53–65. https://doi.org/10.1016/j.iheduc.2014.09.006

Gašević, D., Joksimović, S., Eagan, B. R., & Shaffer, D. W. (2019). SENS: Network analytics to combine social and cognitive perspectives of collaborative learning. *Computers in Human Behavior, 92*(February 2017), 562–577. https://doi.org/10.1016/j.chb.2018.07.003

Gencoglu, B., Helms-Lorenz, M., Maulana, R., Jansen, E. P. W. A., & Gencoglu, O. (2023). Machine and expert judgments of student perceptions of teaching behavior in secondary education: Added value of topic modeling with big data. *Computers & Education, 193*, Article 104682. https://doi.org/10.1016/j.compedu.2022.104682

Guo, P., Saab, N., Wu, L., & Admiraal, W. (2021). The Community of Inquiry perspective on students' social presence, cognitive presence, and academic performance in online project-based learning. *Journal of Computer Assisted Learning, 37*(5), 1479–1493. https://doi.org/10.1111/jcal.12586

Hayati, H., Chanaa, A., Khalidi Idrissi, M., & Bennani, S. (2019). Doc2Vec & Naïve Bayes: Learners' cognitive presence assessment through asynchronous online discussion TQ transcripts. *International Journal of Emerging Technologies in Learning (iJET), 14*(08), 70. https://doi.org/10.3991/ijet.v14i08.9964

Holsti, O. R. (1969). *Content analysis for the social sciences and humanities*. Addison-Wesley Pub. Co; WorldCat.

Hodges, C., Moore, S., Lockee, B., Trust, T., & Bond, A. (2020). The Difference Between Emergency Remote Teaching and Online Learning. *Educause Review*.

Hrastinski, S., Stenbom, S., Saqr, M., Jansson, M., & Viberg, O. (2023). Examining the development of K-12 Students' cognitive presence over time: The case of online mathematics tutoring. *Online Learning, 27*(3). https://doi.org/10.24059/olj.v27i3.3481

Kaczkó, É., & Ostendorf, A. (2023). Critical thinking in the community of inquiry framework: An analysis of the theoretical model and cognitive presence coding schemes. *Computers & Education, 193*, 104662. https://doi.org/10.1016/j.compedu.2022.104662

Kovanovic, V. (2017). *Assessing cognitive presence using automated learning analytics methods*. doctoral dissertation. University of Edinburgh. ERA digital repository http://hdl.handle.net/1842/28759.

Krippendorff, K. (2004). *Content analysis: An introduction to its methodology* (2nd ed.). Sage.

Landis, J. R., & Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics, 33*(1), 159. https://doi.org/10.2307/2529310

Lee, J., Soleimani, F., Irish, I., Hosmer, I. V. J., Yilmaz Soylu, M., Finkelberg, R., & Chatterjee, S. (2022). Predicting cognitive presence in at-scale online learning: MOOC and for-credit online course environments. *Online Learning, 26*(1). https://doi.org/10.24059/olj.v26i1.3060

Lim, J. (2018). *Disciplinary differences in a community of inquiry (issue august)*. doctoral dissertation. Purdue University https://eric.ed.gov/?id=ED588742.

Lindgren, B.-M., Lundman, B., & Graneheim, U. H. (2020). Abstraction and interpretation during the qualitative content analysis process. *International Journal of Nursing Studies, 108*, Article 103632. https://doi.org/10.1016/j.ijnurstu.2020.103632

Martin, F., Wu, T., Wan, L., & Xie, K. (2022). A Meta-analysis on the community of inquiry presences and learning outcomes in online and blended learning environments. *Online Learning, 26*(1). https://doi.org/10.24059/olj.v26i1.2604

McKlin, T. (2004). *Analyzing cognitive presence in online courses using an artificial neural network*. Doctoral dissertation. Georgia State University. https://doi.org/10.57709/1059079.

Oriogun, P. K. (2009). Detecting Aspects of Critical Thinking by Cleaning Online Message Transcript Through Code-Recode. *American Journal of Distance Education, 23*(1), 34–50. https://doi.org/10.1080/08923640802661694

Platt, M., & Platt, D. (2023). Effectiveness of generative artificial intelligence for scientific content analysis. In *2023 IEEE 17th international conference on application of information and communication technologies (AICT)* (pp. 1–4). https://doi.org/10.1109/AICT59525.2023.10313167

Richardson, J., & Ice, P. (2010). Investigating students' level of critical thinking across instructional strategies in online discussions. *The Internet and Higher Education, 13*(1–2), 52–59. https://doi.org/10.1016/j.iheduc.2009.10.009

Richardson, J., Sadaf, A., & Ertmer, P. A. (2012). Relationship between Types of Question Prompts and Critical Thinking in Online Discussions. In Z. Akyol, & R. Garrison (Eds.), *Educational Communities of Inquiry* (pp. 197–222). Information Science Reference. https://doi.org/10.4018/978-1-4666-2110-7.ch011.

Richardson, J. C., & Alsup, J. (2015). From the classroom to the keyboard: How seven teachers created their online teacher identities. *International Review of Research in Open and Distance Learning, 16*(1), 142–167. https://doi.org/10.19173/irrodl.v16i1.1814

Richardson, J. C., Caskurlu, S., Castellanos-Reyes, D., Duan, S., Duha, M. S. U., Fiock, H., & Long, Y. (2021). Instructors' conceptualization and implementation of scaffolding in online higher education courses. *Journal of Computing in Higher Education, 11*. https://doi.org/10.1007/s12528-021-09300-3

Richardson, J. C., Caskurlu, S., Castellanos-Reyes, D., Duan, S., Duha, M. S. U., Fiock, H., & Long, Y. (2022). Instructors' conceptualization and implementation of scaffolding in online higher education courses. *Journal of Computing in Higher Education, 34*(1), 242–279. https://doi.org/10.1007/s12528-021-09300-3

Richardson, J. C., Castellanos Reyes, D., Janakiraman, S., & Duha, M. S. U. (2023). The process of developing a digital repository for online teaching using design-based Research. *TechTrends, 67*, 217–230. https://doi.org/10.1007/s11528-022-00795-w

Rourke, L., Anderson, T., Garrison, R., & Archer, W. (2001). Methodological issues in the content analysis of computer conference transcripts. *International Journal of Artificial Intelligence in Education, 12*, 8–22. https://doi.org/10.1145/1518701.1518791

Sadaf, A., & Olesova, L. (2017). Enhancing Cognitive Presence in Online Case Discussions With Questions Based on the Practical Inquiry Model. *American Journal of Distance Education, 31*(1), 56–69. https://doi.org/10.1080/08923647.2017.1267525

Sadaf, A., Wu, T., & Martin, F. (2021). Cognitive Presence in Online Learning: A Systematic Review of Empirical Research from 2000 to 2019. *Computers and Education Open, 2*, 100050. https://doi.org/10.1016/j.caeo.2021.100050

Shea, P., Hayes, S., Vickers, J., Gozza-Cohen, M., Uzuner, S., Mehta, R., Valchova, A., & Rangan, P. (2010). A re-examination of the community of inquiry framework: Social network and content analysis. *The Internet and Higher Education, 13*(1–2), 10–21. https://doi.org/10.1016/j.iheduc.2009.11.002

Shea, P., Richardson, J., & Swan, K. (2022). Building bridges to advance the Community of Inquiry framework for online learning. *Educational Psychologist, 57*(3), 148–161. https://doi.org/10.1080/00461520.2022.2089989

Swan, K. (2021). Teaching and learning in post-industrial distance education. In *An introduction to distance education: Understanding teaching and learning in a new era* (2nd ed.). Routledge https://www.routledge.com/An-Introduction-to-Distance-Education-Understanding-Teaching-and-Learning/Cleveland-Innes-Garrison/p/book/9781138054417.

Swan, K., & Ice, P. (2010). The community of inquiry framework ten years later: Introduction to the special issue. *Internet and Higher Education, 13*(1–2), 1–4. https://doi.org/10.1016/j.iheduc.2009.11.003

Tai, R. H., Bentley, L. R., Xia, X., Sitt, J. M., Fankhauser, S. C., Chicas-Mosier, A. M., & Monteith, B. G. (2024). An Examination of the Use of Large Language Models to Aid Analysis of Textual Data. *International Journal of Qualitative Methods, 23*. https://doi.org/10.1177/16094069241231168, 16094069241231168.

Tirado-Morueta, R., Maraver López, P., Hernando Gómez, Á., & Harris, V. W. (2016). Exploring social and cognitive presences in communities of inquiry to perform higher cognitive tasks. *Internet and Higher Education, 31*, 122–131. https://doi.org/10.1016/j.iheduc.2016.07.004

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2023). Chain-of-thought prompting elicits reasoning in large language models (arXiv:2201.11903). *arXiv*. http://arxiv.org/abs/2201.11903.

Xiao, Z., Yuan, X., Liao, Q. V., Abdelghani, R., & Oudeyer, P.-Y. (2023). Supporting qualitative analysis with large language models: Combining codebook with GPT-3 for deductive coding. In *28th international conference on intelligent user interfaces* (pp. 75–78). https://doi.org/10.1145/3581754.3584136

Zhang, Z., Zhang, E., Liu, H., & Han, S. (2024). Examining the association between discussion strategies and learners' critical thinking in asynchronous online discussion. *Thinking Skills and Creativity, 53*, 101588. https://doi.org/10.1016/j.tsc.2024.101588

ZipRecruiter. (2024). *Salary: Assistant professor of education (may, 2024) US*. ZipRecruiter. Retrieved May 22, 2024, from https://www.ziprecruiter.com/Salaries/Assistant-Professor-Of-Education-Salary.

Zou, W., Hu, X., Pan, Z., Li, C., Cai, Y., & Liu, M. (2021). Exploring the relationship between social presence and learners' prestige in MOOC discussion forums using automated content analysis and social network analysis. *Computers in Human Behavior, 115*(September 2020), Article 106582. https://doi.org/10.1016/j.chb.2020.106582