A Mixed Methods Analysis of Cognitive Engagement based on Video Position-Based Notes in Programming Learning

Xiaonan Wang
Graduate School of Intercultural Studies
Kobe University
Kobe, Japan
https://orcid.org/0000-0002-6709-2427

Yi Sun
Graduate School of Information Technology
Kobe Institute of Computing
Kobe, Japan
sun@kic.ac.jp

Yancong Su
School of Design and Art
Xiamen University of Technology
Xiamen, China
syc@xmut.edu.cn

Takeshi Nishida
Graduate School of Intercultural Studies
Kobe University
Kobe, Japan
tnishida@people.kobe-u.ac.jp

Kazuhiro Ohtsuki
Graduate School of Intercultural Studies
Kobe University
Kobe, Japan
ohtsuki@kobe-u.ac.jp

Hidenari Kiyomitsu
Graduate School of Intercultural Studies
Kobe University
Kobe, Japan
https://orcid.org/0000-0003-2828-9785

***Abstract*—This research full paper describes a study on Video Position-Based Notes (VPB-Notes). Video-based learning environments have become increasingly prevalent in programming education, offering flexibility and personalized pacing. The VPB-Notes which allow students to capture thoughts, questions, and observations at specific video timestamps, have emerged as a promising tool to promote active learning and engagement. However, the lack of standardization in VPB-Notes and the presence of mixed code and natural language poses significant challenges for learning analysis using traditional natural language processing techniques. This study aims to propose a novel analytical framework for understanding the cognitive processes and engagement levels reflected in VPB-Notes taken by students in asynchronous video-based programming courses. The research questions focus on uncovering the key characteristics and patterns of VPB-Notes and investigating the relationships between VPB-Notes features and learners' performance outcomes. Our study combines qualitative analysis of VPB-Notes from four programming courses with quantitative analysis using correlation and cluster analysis. The qualitative analysis reveals that VPB-Notes exhibit contextual relevance, conciseness, dynamic structure, and code integration with commentary. It also proposes an analytical framework for categorizing notes across four dimensions. The quantitative analysis suggests a hierarchy of note features, types, and initiative associated with academic performance. In contrast, cluster analysis highlights the complexity of note portfolios and the influence of personal learning styles. Despite limitations in the automation of note analysis and the need for further validation, this study contributes new insights to the growing body of knowledge on note-taking in online programming education, offering valuable guidance for instructional design and personalized learning support.**

Keywords—distance learning, programming education, video position-based note, mixed methods

This work was partly supported by JSPS KAKENHI Grant Number 19K03000.

I. Introduction

Video-based learning environments, such as flipped classrooms and on-demand courses, have revolutionized educational paradigms by addressing the need for flexibility and personalized pacing [1]. These platforms often emphasize accessibility and student autonomy, allowing learners to engage with material conveniently [2]. However, this learning mode also introduces challenges, notably the isolation and passive engagement students may experience during video interactions [3]. Various strategies to address the challenge, like the above, have been employed to enhance learner support and increase participation, such as incorporating interactive elements, implementing staged assessments, and guiding student interactions [4, 5]. Increasing engagement improves learning outcomes in these contexts [6, 7].

Traditional video-based learning analytics rely heavily on the collection of passive records, such as logs and clickstream data [8]. These are often referred to as 'passive learning data.' While some studies have shown that these data sources can offer valuable insights [9, 10], they do have their limitations. They struggle to capture the full spectrum of learner engagement and understanding. It is where 'active learning data' comes into play. It encompasses content that learners deliberately generate, such as questions posed, reports created, and other text data that reflect their thinking and engagement levels [11, 12]. The type of data offers unique advantages, providing a deeper understanding of learners' cognitive processes and knowledge construction, which can be a game-changer for instructional designers.

Note-taking has long been recognized as an active learning strategy that promotes engagement and facilitates knowledge acquisition. Extensive research has investigated the cognitive and metacognitive benefits of note-taking [13, 14], highlighting its role in organizing information, enhancing understanding, and promoting self-regulation [15]. In the context of asynchronous video learning environments, techniques for marking up videos have been employed to guide students' note-taking activities, such as providing annotation tools and prompts [16]. These interventions have shown positive effects on learner outcomes [17]. In the field of programming education, note-taking and code commenting are not just beneficial, they're crucial to understanding and problem-solving. Learners use notes to capture key concepts, document their thought processes, and annotate code snippets [18], which aids in developing computational thinking skills.

To better capture "active learning data", we developed an educational support system that provides learners with easy-to-use questioning and note-taking features [19]. The system allows students to link their notes directly to specific video frames, reducing the need for detailed descriptions. In our previous study, the approach significantly reduced the difficulty of learners asking questions and summarizing [20]. As a result, we received many notes labeled on the video timeline in our experiments, termed **Video Position-Based Note** (VPB-Note). Unlike traditional notes which prioritize depth and completeness, VPB-Note focuses on convenience and relevance. They allow learners to capture quick thoughts, ask questions, or highlight important points without extensive elaboration. However, it may need to conform to existing note-taking assessment standards emphasizing content expansion and structured organization.

Traditional note-taking assessment typically focuses on the completeness, clarity, and coherence of notes, expecting learners to provide detailed explanations and logically organized information [13, 21]. The brevity and lack of explicit structure in VPB-Notes can make applying these conventional assessment criteria challenging. Furthermore, VPB-Notes in programming education exhibit unique characteristics that distinguish them from notes in other domains. Programming-related VPB-Notes often contain code snippets, command usage, and domain-specific terminology interspersed with natural language explanations and questions. The blend of programming syntax and natural language creates a distinct note-taking style specific to the context of programming education. The code snippets and programming-specific vocabulary in VPB-Notes pose challenges for conventional natural language processing techniques, such as grammar analysis and keyword extraction.

The study aimed to answer the following research questions:

RQ1: What are the key characteristics and patterns of VPB-Notes in asynchronous video-based programming courses, and how do they differ from traditional note-taking in terms of content, structure, and style?

RQ2: How can VPB-Notes in a programming course be effectively analyzed to assess learners' understanding, engagement, and problem-solving strategies?

RQ3: What relationships exist between VPB-Notes features and learners' performance outcomes in programming courses, and how can these insights inform the development of personalized learning support systems?

Addressing the unique characteristics of VPB-Notes, the current study aims to provide valuable insights for machine learning techniques tailored to the educational context by combining qualitative analysis to redefine the characteristics of notes in the learning environment with quantitative analysis to explore correlations among note features.

The paper is organized as follows: Section II presents related work. Section III describes the methodology used in this study. Section IV uses qualitative analysis to characterize VPB-Note notes from programming courses in four different languages. Section V is a quantitative analysis investigating the association between VPB-Note note characterization and learning performance in a course. Section VI discusses these findings for video learning in programming education, and the limitations of this work. Finally, Section VII is conclusion.

II. Related Work

In recent years, video-based learning has gained significant traction, offering learners the flexibility to access educational content at their own pace and convenience. This shift has been supported by advancements in video technologies, also known as hyper-video technologies. Hyper-video technologies enable the integration of supplementary materials, such as text, images, and links, within the video itself, providing learners with a rich learning environment [22, 23]. The incorporation of elements has further enhanced the learning experience by increasing learner engagement [24].

One such technology is video annotation, which enables learners to add notes, comments, and highlights to specific points in a video. Prominent MOOC platforms such as Coursera and Udemy have introduced a video timeline-based note-taking feature that allow students to mark and take notes on the current video frame while watching a video lecture. Moreover, there are a number of video annotation tools developed by individuals and groups that are used in educational research. For example, Mitrovic et al. [16] developed the AVW system to promote learner comments on videos and foster collaborative learning through shared annotations. Dodson et al. [25] developed ViDex allows the learner to add textual annotations to videos or highlight intervals within videos or transcripts. Sidi et al. [26] categorize the function of students writing annotations on videos as a “hyper-video function,” where annotations are used to analyze active learning in the learning process.

Note-taking has long been considered a fundamental cognitive activity that plays a crucial role in the learning process. From a cognitive psychology perspective, note-taking involves a complex encoding process, including comprehension, selection, organization, and representation of information [27], and these activities enhance the processing of material and improve learning outcomes [28]. Traditional note-taking relies heavily on working memory [29]. When taking notes, we maintain a short-term memory buffer to acquire mental representations, select, and understand, as new information continuously flows in, the stored knowledge is updated and interacted with [13]. With the emergence of new note-taking methods such as VPB-Note, the memory buffering process has been greatly optimized. Katayama and Robinson [30] argue that the primary obstacle to high-quality notes is the cognitive overload experienced by students. By linking graphical (video frames) and textual information, VPB-Note directs learners' attention to relevant material, reducing irrelevant cognitive load in the learning process [31].

Furthermore, different note-taking methods may have varying effects on learning outcomes [32]. Makany et al. [33] compared the performance of traditional linear notes and non-linear notes, examining the efficiency of different note-taking styles. Kiewra et al. [34] provided an outline framework for notes and verified its contribution to improving test performance. Similarly, VPB-Note also exists in different forms. Vania et al. [17] introduced adaptive interventions to increase note-taking engagement by providing personalized recommendations to learners. The NoteStruct [35] encourages learners to first insert simple annotations on video content and provides the ability to add notes and integrate notes after viewing, highlighting the contribution of VPB-Note in enhancing metacognitive skills. In the learning environment we designed, VPB-Note is used to reduce the explanation cost in programming learning by lowering the difficulty of learners' questioning and expression, thereby increasing learning engagement.

The analysis of notes plays a pivotal role in understanding and supporting learning processes in educational contexts. By examining the content and structure of notes, researchers can gain deep insights into learners' cognitive processes, comprehension levels, and engagement with the material [36, 37]. The ICAP framework [37] defines cognitive engagement activities based on students' overt behaviors. Taskin et al. [38] used the ICAP framework as a foundation to link cognitive engagement activities with learners' annotations in videos, characterizing their engagement levels. Hecking et al. [39] investigated learners' engagement related to attention and thematic focus in active video watching tasks through lexical-semantic analysis of video comment texts. Sidi et al. [26] combined qualitative learning analytics to examine interactivity in video learning based on annotation content.

Content based analysis methods and cognitive frameworks provide support for analyzing annotations or notes in videos. However, the characteristics of VPB-Notes in programming learning environments pose challenges for standardizing and analyzing them using traditional natural language processing (NLP) tools. Traditional NLP tools like SpaCy and Stanford NLP rely heavily on well-formed syntactic structures to parse text. Most VPB-Notes, however, lack consistent dependency syntax and exhibit a high degree of syntactic irregularity. The inconsistency results in poor clarity and complexity when applying syntactic parsing, rendering these tools less effective. The results often fail to meaningfully capture the learner's intent or the instructional nuances contained within the notes.

Furthermore, keyword extraction methods such as TF-IDF and Text-Rank are typically not suitable for processing VPB-Notes. These techniques generally involve preprocessing steps like stop-word removal and tokenization, which can disrupt the integrity of code snippets embedded within the notes. In programming learning, such preprocessing could strip away crucial programming context and syntax, which are essential for understanding the content's full meaning and instructional value. The presence of code mixed with natural language and the specific references to video content require specialized analytical approaches to capture the characteristics to assess the effectiveness in supporting learning.

III. METHODOLOGY

The exploratory study had three main objectives: i) Identify the key characteristics of VPB-Notes in asynchronous video-based programming courses. ii) Develop a framework to analyze the unique characteristics of the VPB-Notes. iii) Investigate the relation between VPB-Note features and learners' performance in the courses, deriving insights to support the development of tailored machine learning techniques and inform personalized learning support systems in the educational context.

A. Design of System

This study employed a custom-designed learning support system tailored for video-based programming courses. The system is designed to enhance the asynchronous learning experience in online courses [20]. As illustrated in Figure 1, the system offers convenient video-interactive note-taking and questioning features, enabling students to add notes or raise questions at video timestamps while watching video lectures.

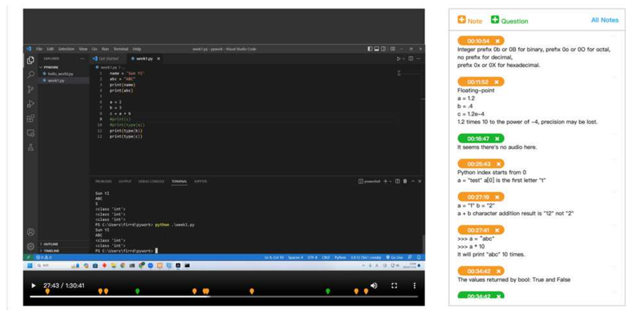

Fig. 1. Student Video Learning Interface, Consisting of Two Columns with the Video Area on the Left and Notes Area on the Right.

The primary focus of the system is to reduce the "explanation cost" of programming learning interactions through VPB-Notes, thereby minimizing the difficulty of asking questions and the complexity of note-taking during the learning process, ultimately leading to increased participation. As all notes are created based on the current video frame, students can directly refer to the code state in the video using terms such as "this" or "here" within their notes, effectively reducing the need for detailed descriptions. The approach ensures that the video content provides the necessary context for the notes, allowing students to focus on writing down what they deem important. Furthermore, the system offers a convenient method for reviewing learned material through notes linked to specific content, thus supporting the development of metacognitive skills crucial to the learning process. By facilitating self-reflection, students can effectively judge and evaluate the strategies employed. Moreover, the system enables teachers to review these notes and engage with learners' queries or notes via a dedicated instructor's interface.

B. Experiments and Data Collection

This study involved a diverse sample of students from various programming courses. In Section 4, the qualitative analysis utilized experimental data from four programming courses (Python, HTML, Java, and Database). The courses were all introductory ones designed for first and second-year university students. The student samples were drawn from two universities and included native and non-native English speakers. Diversity in the student sample may allow for the identifying of commonalities in the VPB-Notes of students from different backgrounds. All courses were offered in an on-demand format. Due to the voluntary nature of the experiments, each course provided approximately two hours of video lecture data, resulting in 2,101 VPB-Notes collected from 158 individuals.

Due to the excessive workload of manual annotation, Section 5 focuses on data from one of the experiments to perform a correlation analysis. The student sample consisted of 26 students enrolled in a second-year Python Foundation course at a university. The total length of the video lessons was approximately 90 minutes, and 359 VPB-Notes were collected. In the experiment for the Python Foundation course, students completed a stage test after watching the videos.

The participants were informed that their note-taking content would be analyzed, and their consent was obtained. All data used in the analysis was anonymized to protect the participants' privacy.

C. Analyzing of VPB-Notes

This study employs a mixed-methods approach, combining qualitative and quantitative analyses. The qualitative analysis (Section 4) examines the unique features of VPB-Notes in programming learning, such as contextual relevance, conciseness, dynamic structure, and the integration of code with commentary. We propose the analytical framework of VPB-Note, which categorizes notes across four dimensions: note type, initiative, cognitive processes, and features. The taxonomy analyzes VPB-Note characteristics and the students' underlying cognitive processes.

The quantitative analysis (Section 5) investigates the relationship between the note characteristics and learning outcomes using correlation analysis and cluster analysis. The correlation analysis calculates the Pearson correlation coefficients between each note feature and quiz scores to verify the hypothesized relationships between note characteristics and learning performance. The cluster analysis employs the K-Means clustering algorithm to group students based on their quiz scores. It examines the differences in note-taking characteristics among students with varying levels of academic performance.

IV. Qualitative Analysis

In the section, we delve into a qualitative analysis of the distinct features of VPB-Notes in a programming learning environment. We develop a classification framework based on the cognitive processes reflected by the note characteristics, which are not just theoretical constructs, but practical tools based on our findings, with direct implications for the field of programming learning and educational technology.

A. VPB-Notes in Programming Learning

As described in Section 2, Video Position-Based Notes (VPB-Notes) are increasingly used in research and practice. Learners' "active data" also provides new perspectives for learning analytics. In our study, the learning support system designed for asynchronous programming courses leverages VPB-Notes to reduce the "explanation cost" of learning interactions. It means that the complexities of questioning and note-taking are simplified, making the learning process more efficient. Within the environment, VPB-Notes exhibit unique characteristics.

Contextual Relevance. VPB-Notes are inherently contextual, as they correspond to the timestamps of video content. Learners can directly capture thoughts, questions, or observations directly related to the video segment they view. Such notes often include direct references to visual elements, such as specific code lines, error messages, or programming output seen in the video. Example:

- uses the 'range ()' function here to generate a sequence of numbers. It seems pretty handy for creating loops.

The learner has noted a specific observation about using the 'range()' function at a particular timestamp. The note directly refers to the code element seen in the video, making it highly contextual and precise. The direct correlation enhances the relevance and precision of the notes, making them more actionable during reviews.

Conciseness and Focus. Unlike traditional notes, VPB-Notes are concise. The medium itself partly drives this brevity—video content provides context for notes. The students focus primarily on capturing essential insights or queries that arise at the moment rather than elaborating on them extensively. In the experimental data we collected, the median note length is 48 characters, and 75% of the notes have 72.5 characters or fewer (SD = 66.59). Example:

- I keep getting an error message here...

Here, the learner makes a note of an error encountered in practice. Learners often point to their screens in face-to-face programming sessions and ask the instructor for help. That is because, for beginners, they usually do NOT know what is happening. Unfamiliar knowledge or code state makes it difficult for them to describe the problem. Therefore, we solved the problem in the learning environment we have designed. Using terms such as "this" or "here" in the notes allows students to easily pose questions without specifying the problem in detail. Those make VPB-Notes more streamlined. For instructors, the conciseness of VPB-Notes and the context provided by the video enable them to understand the learners' queries and provide targeted support.

Dynamic Structure. Regarding grammatical structure, VPB-Notes are less formal and more dynamic than traditional notes. They do not adhere to a rigid format but vary significantly based on the learner's style and the content type. Some notes might be simple bullet points, while others could be complex thoughts or code snippets. The structure is often influenced by the learner's immediate needs and cognitive strategies, reflecting personal learning styles and preferences. Example:

- The addition of strings is possible (without declaration), but multiplication is not. len(a) can get its length.
- a="1" b="two" a+b , '1two' output is a character splice.

Notes on the same topic show a dynamic structure of completely different expressions, with the former summarized in language and the latter illustrated with references to code examples. VPB-Notes in programming learning are more highly personalized, reflecting the individual's learning path, queries, and insights.

Integration of Code and Commentary. As can also be seen from the examples above, VPB-Notes in Programming Education uniquely blend code snippets with natural language comments. Such kind of integration is common among programming learners as it helps them to annotate their understanding or confusion about specific parts of the code.

The code snippet captures the essential parts of the function discussed in the video, while the comment reflects the learner's understanding of how the built-in functions make the code more concise. The integration of code and commentary is typical in programming VPB-Notes, as it helps learners reinforce their understanding and document their thought processes while working with code. In more examples, learners code snippets referencing the essential parts of the functions discussed in the video, and the comments reflect the learner's understanding of how to use the code.

These examples illustrate how VPB-Notes in programming education exhibit contextual relevance, conciseness, dynamic structure, and code integration with commentary. These characteristics make VPB-Notes challenging to standardize and analyze using traditional natural language processing (NLP) tools. The presence of code mixed with natural language and the specific references to video content require specialized analytical approaches to fully capture their richness and assess their effectiveness in supporting learning.

In the following subsections of this chapter, we qualitatively analyze and categorize VPB-Notes in programming education, aiming to provide a new way of evaluating and classifying them.

B. *Taxonomy of VPB-Note Behaviors*

In our previous study, we conducted a comprehensive analysis of the collected notes, categorizing them into four main types: Code-Note, Summary-Note, Tag-Note, Cryptic-Note, and Question [20]. This initial exploration yielded significant findings, revealing a strong correlation between Code-Note and Summary-Note and learning outcomes (scores). These early insights not only laid the groundwork for our subsequent research but also underscored the pivotal role of note-taking strategies in asynchronous video programming courses.

Upon meticulous examination of the Summary-Note and Code-Note subsets from our earlier research, we discovered a multitude of nuanced characteristics and some inherent ambiguities in the original categorization. This complexity in note categorization not only underscores the depth and thoroughness of our research but also urgently highlights the need for a more nuanced understanding of note-taking behaviors.

Summary-Note demonstrates varying degrees of initiative. In the Summary-Note classification, some notes had nearly identical expressions within similar time intervals. After comparing these notes with the video content, we found that these notes were merely copying sentences from the presentation slides or paraphrasing the instructor's words. In contrast, other notes demonstrated students' active summarization efforts.

Code-Note Complexity. Similarly, the Code-Note category included pure code snippets without any explanations and code accompanied by the students' reflections and interpretations. The observation necessitates that Code-Note with explanations should be further differentiated, as simply copying code (passive recording) and explaining it (active summarization) are two distinct behaviors. In other words, Code-Note with explanations exhibit characteristics of Summary-Note.

Additionally, some notes ventured beyond the immediate scope of the lecture content, introducing other relevant themes and personal reflections. These notes are rich with the student's thoughts and show a deep level of engagement with the material, indicating an interactive or constructive approach to learning.

We refined the original categorization and developed a second tier based on the cognitive processes and initiatives reflected in the content of the notes. Table 1 presents a new analytical framework, which draws inspiration from the ICAP framework proposed by Chi et al. [37]. The ICAP framework categorizes cognitive engagement into four modes: interactive, constructive, active, and passive. We have adapted this framework to enable a systematic analysis and explanation of the cognitive processes involved in students' note-taking behaviors and the characteristics of these processes manifested in the content of their notes.

TABLE I. ANALYSIS FRAMEWORK FOR VPB-NOTES IN PROGRAMMING LEARNING

Analysis Framework for VPB-Notes in Programming Learning	
Note Type	Code-Note, Textual-Note, Simple-Note, Question
Initiative	Passive, Active, Constructive, Inquiry
Cognitive Processes	Storing, Applying, Inferring, Creating
Feature	Citation, Explanatory, Extended, Abstract
Hypothesis	Abstract > Extended > Explanatory > Citation

a. The "Initiative" here refers to the comparison at the level of note characteristics.

The framework comprises four key dimensions: Note Type, Initiative, Cognitive Processes, and Feature. Each note with different patterns or categories can be interpreted by these four dimensions (Each note corresponds to a single note type, characteristic, and initiative; however, a note may involve two or more cognitive processes). We hypothesize that notes demonstrating abstract Feature exhibit higher levels of learning than those showing extended Feature, followed by citation and interactive outcomes, Abstract > Extended > Explanatory > Citation. Here, a higher level of learning implies a deeper understanding and more advanced cognitive processing.

The Note Type dimension is an optimized version of previous research classifications, categorizing notes based on their content and format. Table 2 presents four categories and corresponding note examples. Code-Note includes notes with code snippets (including code format and single numeric names), such as C1, C2, and C3. Notes described in natural language are classified as Textual-Note, like T1, T2, and T3. Simple-Note contains more concise content, often including only one or two words, as exemplified by S1 and S2. Finally, the Question category encompasses notes that pose inquiries and provide learning feedback, such as Q1, Q2, andQ3.

TABLE II. NOTE TYPE AND SAMPLES

Type	No.	Samples
Code-Note	*C1*	Lists can contain various types, including other lists, e.g., L = [1, 2, [3, 4]] print(L[2]) output [3, 4]], only three elements here.
	C2	a= [1, 2, [3, 4]]

	C3	>>> a = [1, 2, 3]
		>>> b = [1, 'abc', 2.0, ['a', 'b']]
		>>> c = a+b
		>>> print(len(b),c)
		4 [1, 2, 3, 1, 'abc', 2.0, ['a', 'b']]

Textual-Note	*T1*	Tuple is an immutable Python object.
	T2	Conclusion: Elements in a tuple can't be changed. Tuple access creation is faster than lists, not as flexible as lists.
	T3	Tuple can't change. But could use a list to mess around with the data before switching if need to tweak things temporarily.
Simple-Note	*S1*	index order
	S2	Mutability
Question	*Q1*	What does this line of code mean?
	Q2	a = [1, 2, 3] why is a[0] equal to 1?
	Q3	Can this write particle systems

b. For comparison, experimental Samples limited to the python basics course were filtered.

In the Code-Note category, C1, C2, and C3 demonstrate learners recording a specific example from the video (which we verified against the video content) to illustrate the concept of nested lists in Python. C1 includes an explanation and inference about the code snippet ("Explanatory" feature), while C2 only includes the code ("Citation" feature) without further elaboration. C3 adds more data types to the list ("Extended" feature) and attempts to build upon the previous content (list addition, len() function). In the Initiative dimension, we qualitatively categorize C1 as "Active," C2 as "Passive," and C3 as "Constructive." C2 merely copies the code from the video (Storing), whereas C1 provides a more specific explanation of its application (Cognitive Processes dimension). C3 engages in Inferring and attempts to apply the learned knowledge to more complex structures, demonstrating cognitive processes of Inferring and Creating, even though no explicit explanation is provided. Thus, we can describe them as follows (qualitative description):

- C1 is an explanatory Code-Note with active initiative, reflecting the learner's cognitive process of Storing knowledge.
- C2 is a citation Code-Note with passive initiative, reflecting the learner's cognitive process of Applying knowledge.
- C3 is an explanatory Code-Note with Constructive initiative, reflecting the learner's cognitive processes of Inferring and Creating from known content.

Similarly, in the Textual-Note category, T1, T2, and T3 provide textual descriptions of the same topic (tuple characteristics). By cross-referencing the video content, T1 merely paraphrases the instructor's words ("Citation" feature), and the note is purely informative (Storing) with low cognitive engagement (Passive). In contrast, T2 presents a clear "Conclusion" ("Abstract" feature), demonstrating knowledge organization and constructive engagement by inferring tuple characteristics and comparing them with lists. The comparison requires learners to synthesize and apply information to understand the efficiency and limitations of different data structures. Thus, in the cognitive process dimension, we classify T2 as "Storing" and "Applying." T3 provides a practical application scenario (Extended) and critically considers the strategy in real-world applications (Inquiry). The investigation shows that learners may possess relevant knowledge, reflecting a higher level of cognitive processing in applying theoretical knowledge to practical problem-solving situations (Applying and Inferring). According to our framework, we can describe T1, T2 and T3 as follows (qualitative description):

- T1 is a citation Textual-Note with passive initiative, reflecting the learner's cognitive process of Storing knowledge.
- T2 is an abstract Textual-Note with constructive initiative, reflecting the learner's cognitive processes of Storing and Applying knowledge.
- T3 is an extended Textual-Note with Inquiry initiative, reflecting the learner's cognitive processes of Inferring and Applying.

The Simple-Note category includes concise notes without detailed elaboration. They contain little information but may be reminders or observations of important concepts captured during video viewing. S1 is an attempt to mark a pivotal moment in the learning process, likely emphasizing the importance or functionality of subscript ordering in data structures. It is a direct note-taking behavior without deeper engagement. On the other hand, S2 is a passive recording of a critical term, but the specific meaning and cognitive processes behind it may require more work to decipher fully. At the note content level, we consider both to be passive. We describe them as follows:

- S1 is an Abstract Simple-Note with passive initiative, reflecting the learner's cognitive process of Storing.
- S2 is a Citation Simple-Note with passive initiative, reflecting the learner's cognitive process of Storing.

In the "Question" category, we discovered different subcategories that illustrate learners' varying cognitive engagement and information needs, such as Definition Inquiry (What), Rationale Inquiry (Why), Procedure Inquiry (How), Capability Inquiry (Can), and Learning Feedback. However, not all subcategories belong to "Inquiry" in the Initiative dimension. For example, Definition Inquiry (Q1) mostly cites specific terms or concepts from the video but remains at the basic cognitive processing level of storing information. Moreover, simple web searches can quickly answer the "What" questions, so we consider it passive in the Initiative dimension.

In contrast, Rationale Inquiry (Q2) demonstrates learners' constructive engagement, seeking to understand the underlying principles behind programming practices. At the same time, Procedure Inquiry (How) indicates application-focused active engagement, requiring detailed step-by-step guidance for practical implementation. Capability Inquiry (Q3), which often explores the potential applications of concepts in new scenarios, shows an advanced level of cognitive engagement—creating, where learners understand and innovate with the learned material. The Q3 type of inquiry benefits greatly from project-based learning, in which students challenge themselves to apply their knowledge practically.

As discussed above, although the "Question" category presents different subcategories, our classification framework remains applicable. For example:

- Q1 (What) is a citation question with passive initiative, reflecting the learner's cognitive process of storing.
- Q2 (Why) is an explanatory question with an inquiry initiative that reflects the learner's cognitive processes of inferring and applying.
- Q3 (Can) is an extended question with an inquiry initiative that reflects the learner's cognitive processes of applying, inferring, and creating.

The qualitative analysis of VPB-Notes generated in on-demand video programming courses, provides a comprehensive examination of the notes, and introduces an analytical framework based on note characteristics and cognitive processes. The proposed framework offers a robust explanatory structure for understanding the various types of notes taken by learners in the courses. Through the qualitative analysis of case examples, the chapter elucidates the features of notes across different dimensions, providing a nuanced understanding of students' engagement and cognitive processes. This in-depth exploration contributes to a holistic understanding of learners' note-taking behaviors in video-based programming education.

V. Quantitative Analysis

In this section, we meticulously employ correlation and cluster analysis to investigate the relation between note-taking characteristics and learning outcomes. The dataset used for the analysis includes 359 notes taken by 26 students in an on-demand programming course (Python Basics) and the student's scores in the stage test. We manually annotated these notes according to the classification framework presented in Section 4. The data organization revealed the following information: In the Feature dimension, explanatory notes were the most prevalent (39%), followed by citations (33%), extended (17%), and abstract notes (11%). Regarding the Note Type dimension, Code-Note accounted for 38% of the entries, while Textual-Note comprised 37%. Questions and Simple-Note comprised 13% and 12% of the notes, respectively. The analysis of the Initiative dimension showed that the active were the most common (45%), followed by passive (24%), constructive (22%), and inquiry (9%). The mean quiz score across the cohort was an impressive 87.13 (SD = 3.84), indicating a high and consistent student performance and further highlighting our work's importance.

A. *Correlation Analysis at the Note*

We conducted a correlation analysis between note characteristics and quiz scores. By calculating the Pearson correlation coefficients, we found varying degrees of association between each note feature and quiz performance. In the Feature dimension, the results showed the following order of correlation strength: Abstract (0.29) > Extended (0.17) > Explanatory (0.12) > Citation (-0.44). The analysis also revealed a hierarchy of note type dimensions in terms of their association with academic performance: Textual Note (0.22) > Code Note (0.08) > Question (-0.15) > Simple Note (-0.30).In the Initiative dimension, the results suggest a descending order of correlation strength: Constructive (0.24) > Active (0.07) > Inquiry (0.05) > Passive (-0.35).

B. *Cluster Analysis at the Individual*

We investigate note-taking differences among students with varying academic performance by conducting a K-Means cluster analysis, grouping them into three clusters based on quiz scores. Students in Cluster A, the high-performing group, had an average score of 91.22, and their notes were primarily characterized by explanatory notes (43.1%), textual notes (43.1%), and active initiative (49.2%). Cluster B, the medium-performing group, with an average score of 86.74, exhibited a predominance of explanatory notes (44.7%), code notes (42.3%), and active initiative (43.1%). Cluster C, the lowest-scoring group with an average score of 82.56, showed a notable prevalence of citation notes (62.3%), textual notes (31.1%), and passive initiative (42.5%). The results from the K-Means clustering analysis of students grouped by their quiz scores provide nuanced insights but do not directly align with the initial hypothesis regarding correlation with academic performance. The observed patterns indicate that explanatory notes are consistently prevalent across all levels but do not diminish usage as scores decrease. Instead, high-performing students utilize a balanced approach incorporating abstract and extended notes more than their lower-scoring peers.

VI. Discussion

A. *Discussion on QualitativeAnalysis*

The qualitative analysis in Section 4 uncovers the unique characteristics of VPB-Notes. The proposed framework is valuable for assessing learning progress, identifying areas needing support, and encouraging active, constructive, and inquiry-based note-taking behaviors. Instructors can promote deeper learning by designing activities and prompts that encourage students to go beyond merely storing information (e.g., C2, S1, S2) and instead focus on applying concepts (e.g., T2), making inferences (e.g., C1, T1), drawing comparisons (e.g., T1), and generating questions (e.g., Q1, Q2).

By analyzing the initiative, cognitive processes, and feature outcomes reflected in students' VPB-Notes, instructors can gain insights into individual students' understanding and tailor their teaching approaches accordingly. Moreover, the findings have significant implications for programming education, informing the development of automated assessment and feedback systems, and providing insights for tailoring machine learning techniques to the environment. Developers can leverage these insights to create tools and platforms that support personalized feedback and recommendations based on the levels of cognitive engagement and learning depth reflected in students' note-taking behaviors.

B. Discussion on Quantitative Analysis

The quantitative analysis in Section 5 yields intriguing insights into the relationship between note-taking characteristics and learning outcomes in an asynchronous video-based programming course. The correlation analysis at the note (Section 5. A) confirms the hypothesis proposed in Chapter 4, revealing a hierarchy of note features, types, and initiative levels in terms of their association with academic performance. The results indicate that abstract and extended notes, textual notes, and constructive and active initiative levels positively correlate with quiz scores. In contrast, citation notes, simple notes, and passive initiative levels exhibit negative correlations. However, the cluster analysis at the individual (Section 5. B) based on students' quiz scores does not entirely align with the initial hypothesis. The high-performing students (Cluster A) tend to create more explanatory and textual notes and demonstrate higher levels of active initiative than their lower-performing peers (Cluster C). The unexpected finding suggests that several factors may contribute to the inconsistency between the note and individual analyses, warranting further investigation.

Complexity of Individual Note Portfolios. While the note analysis validates the hypothesis, the reality is that an individual's notes possess different features. The quantitative analysis relies on data from a single experiment with a small sample size of 26 participants. A student might have a mix of highly analytical (abstract) and more straightforward (citation) notes. The diverse features of aggregated notes might mask underlying patterns that otherwise indicate some relationship between note characteristics and learning outcomes.

Non-Linearity of Scores and Learning Depth. While valuable as quantitative outcomes, academic scores do NOT always linearly correlate with the depth of understanding or cognitive processes involved. Scores are influenced by myriad factors, including test design, student test-taking strategies, and external pressures, which might not reflect the actual depth of learning. Thus, these intervening variables might obscure the direct correlation between note types and scores.

Individual Differences in Learning Styles. The significant variance in how individuals absorb, and process information complicates the ability to draw broad conclusions from note-taking behaviors alone. Personal backgrounds, prior knowledge, and even emotional states can influence how a student interacts with content and what note-taking strategies they employ. The cluster analysis reveals that high-performing students adopt a balanced approach, incorporating abstract and extended notes more than their lower-scoring counterparts. It suggests that effective note-taking may involve a strategic combination of different note types and features tailored to individual preferences and learning styles.

The impact of video content. The effectiveness of a note-taking strategy varies depending on the content being learned. Optimal note-taking may not simply be a collection of notes with uniform features but rather a systematic distribution across multiple dimensions that adapt to the specific content and themes of the video. Engaging in different note-taking methods based on the nature of the video content may manifest in diverse note characteristics. A more sophisticated analysis accounting for the interaction between individual strategies and video content is necessary to unravel these complexities. The current study lays the theoretical groundwork for such future investigations.

Despite these limitations, the quantitative analysis yields valuable implications for programming education. The correlation analysis suggests that encouraging students to create notes involving higher levels of cognitive engagement, such as making inferences, drawing comparisons, and generating questions, can lead to better learning outcomes. The cluster analysis reveals that instructors can offer targeted feedback and recommendations tailored to individual needs by examining students' note-taking behaviors.

C. Research Limitations and Future Directions

One of the main limitations of this study is the need for more generalizability of the quantitative analysis results. Although the qualitative analysis drew data from programming courses in different contexts, the current research stage still relies on manual annotation of VPB notes. The process involves nuanced interpretations of content and context that still need to be fully automated, and developing fully automated classifiers to handle such tasks remains challenging. However, the present work lays the theoretical foundation for future research. Leveraging pre-trained models from related domains in combination with the proposed analytical framework and incorporating domain-specific knowledge (e.g., creating dedicated stop-word lists for programming notes) into the pre-trained models can improve their accuracy and interpretability.

Moreover, our study has highlighted the need to investigate the relationship between individual learning styles, note-taking patterns, and learning content. Effective note-taking is a collection of highly personalized characteristics that are context-dependent. Therefore, our future research should focus on determining which note-taking strategies are most effective for different types of content and learners. Understanding how individual and contextual factors influence note-taking can lead to developing personalized learning interventions that adapt to each student's unique needs. Additionally, future studies should examine the long-term impacts of these note-taking strategies on learning outcomes, which would provide deeper insights into how these practices influence student engagement, retention, and success over time, offering a more nuanced understanding of the pedagogical strategies that best support effective learning in asynchronous video-based environments.

VII. Conclusion

In conclusion, this study thoroughly examines Video Position-Based Notes (VPB-Notes) in asynchronous video-based programming courses, highlighting their characteristics, cognitive processes, and link to learning outcomes. Using a mixed-methods approach, the qualitative analysis identifies unique features and proposes a cognitive engagement framework, while the quantitative analysis uncovers relationships between note-taking and learning outcomes. Despite limitations like limited generalizability and challenges with automated classifiers, this research establishes a theoretical foundation for VPB-Notes, offering valuable insights for instructional design, personalized learning, and automated feedback systems.

REFERENCES

[1] J. Bishop and M. A. Verleger, "The flipped classroom: A survey of the research," in *2013 ASEE Annual Conference & Exposition*, 2013, pp. 23.1200. 1-23.1200. 18.

[2] I. E. Allen and J. Seaman, Learning on demand: Online education in the United States, 2009. ERIC, 2010.

[3] K. F. Hew and W. S. Cheung, "Students' and instructors' use of massive open online courses (MOOCs): Motivations and challenges," *Educational research review,* vol. 12, pp. 45-58, 2014.

[4] S. Schwan and R. Riempp, "The cognitive benefits of interactive videos: Learning to tie nautical knots," *Learning and instruction,* vol. 14, no. 3, pp. 293-305, 2004.

[5] R. Moreno and R. Mayer, "Interactive multimodal learning environments: Special issue on interactive learning environments: Contemporary issues and trends," *Educational psychology review,* vol. 19, pp. 309-326, 2007.

[6] F. Martin and D. U. Bolliger, "Engagement matters: Student perceptions on the importance of engagement strategies in the online learning environment," *Online learning,* vol. 22, no. 1, pp. 205-222, 2018.

[7] C. J. Brame, "Effective educational videos: Principles and guidelines for maximizing student learning from video content," *CBE—Life Sciences Education,* 2017.

[8] M. N. Giannakos, K. Chorianopoulos, and N. Chrisochoides, "Making sense of video analytics: Lessons learned from clickstream interactions, attitudes, and learning outcome in a video-assisted course," *International review of research in open and distributed learning,* vol. 16, no. 1, pp. 260-283, 2015.

[9] C. Shi, S. Fu, Q. Chen, and H. Qu, "VisMOOC: Visualizing video clickstream data from massive open online courses," in *2015 IEEE Pacific visualization symposium (PacificVis)*, 2015: IEEE, pp. 159-166.

[10] P. J. Guo, J. Kim, and R. Rubin, "How video production affects student engagement: An empirical study of MOOC videos," in *Proceedings of the first ACM conference on Learning@ scale conference*, 2014, pp. 41-50.

[11] M. Nakayama, K. Mutsuura, and H. Yamamoto, "Student's Reflections on Their Learning and Note-Taking Activities in a Blended Learning Course," *Electronic Journal of e-Learning,* vol. 14, no. 1, pp. 43-53, 2016.

[12] W. He, "Examining students' online interaction in a live video streaming environment using data mining and text mining," *Computers in Human Behavior,* vol. 29, no. 1, pp. 90-102, 2013.

[13] A. Piolat, T. Olive, and R. T. Kellogg, "Cognitive effort during note taking," *Applied cognitive psychology,* vol. 19, no. 3, pp. 291-312, 2005.

[14] H. W. Lee, K. Y. Lim, and B. L. Grabowski, "Improving self-regulation, learning strategy use, and achievement with metacognitive feedback," *Educational Technology Research and Development,* vol. 58, pp. 629-648, 2010.

[15] D. F. Kauffman, R. Zhao, and Y.-S. Yang, "Effects of online note taking formats and self-monitoring prompts on learning from online text: Using technology to enhance self-regulated learning," *Contemporary Educational Psychology,* vol. 36, no. 4, pp. 313-322, 2011.

[16] A. Mitrovic, V. Dimitrova, A. Weerasinghe, and L. Lau, "Reflective experiential learning: Using active video watching for soft skills training," in *Proceedings of the 24th international conference on computers in education*, 2016: Asia-Pacific Society for Computers in Education.

[17] V. Dimitrova, A. Mitrovic, A. Piotrkowicz, L. Lau, and A. Weerasinghe, "Using learning analytics to devise interactive personalised nudges for active video watching," in *Proceedings of the 25th conference on user modeling, adaptation and personalization*, 2017, pp. 22-31.

[18] A. Robins, J. Rountree, and N. Rountree, "Learning and teaching programming: A review and discussion," *Computer science education,* vol. 13, no. 2, pp. 137-172, 2003.

[19] X. Wang, Y. Sun, and Y. Su, "A study of interaction design for E-learning system based on distributed asynchronous communication," in *Proceedings of the Tenth International Symposium of Chinese CHI*, 2022, pp. 249-255.

[20] X. Wang, H. Kiyomitsu, Y. Su, K. Ohtsuki, Y. Sun, and K. Shigyo, "Visualization of potential differences in comprehension by distribution of notes and questions in online programming courses," in *2023 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE)*, 2023: IEEE, pp. 1-8.

[21] K. A. Kiewra, S. L. Benton, S.-I. Kim, N. Risch, and M. Christensen, "Effects of note-taking format and study technique on recall and relational performance," *Contemporary Educational Psychology,* vol. 20, no. 2, pp. 172-187, 1995.

[22] A. Dillon and R. Gabbard, "Hypermedia as an educational technology: A review of the quantitative research literature on learner comprehension, control, and style," *Review of educational research,* vol. 68, no. 3, pp. 322-349, 1998.

[23] G. Beauchamp and S. Kennewell, "Interactivity in the classroom and its impact on learning," *Computers & education,* vol. 54, no. 3, pp. 759-766, 2010.

[24] K. Seo, S. Dodson, N. M. Harandi, N. Roberson, S. Fels, and I. Roll, "Active learning with online video: The impact of learning context on engagement," *Computers & Education,* vol. 165, p. 104132, 2021.

[25] S. Dodson, I. Roll, M. Fong, D. Yoon, N. M. Harandi, and S. Fels, "An active viewing framework for video-based learning," in *Proceedings of the fifth annual ACM conference on learning at scale*, 2018, pp. 1-4.

[26] Y. Sidi, I. Blau, and T. Shamir-Inbal, "Mapping active and collaborative learning in higher education through annotations in hyper-video by learning analytics," *Journal of Computer Assisted Learning,* vol. 38, no. 6, pp. 1752-1764, 2022.

[27] A. Piolat, T. Olive, and R. T. Kellogg, "Cognitive effort during note taking," *Applied cognitive psychology,* vol. 19, no. 3, pp. 291-312, 2005.

[28] P. A. Mueller and D. M. Oppenheimer, "The pen is mightier than the keyboard: Advantages of longhand over laptop note taking," *Psychological science,* vol. 25, no. 6, pp. 1159-1168, 2014.

[29] A. Baddeley, *Working memory, thought, and action*. OuP Oxford, 2007.

[30] A. D. Katayama and D. H. Robinson, "Getting students "partially" involved in note-taking using graphic organizers," *The Journal of Experimental Education,* vol. 68, no. 2, pp. 119-133, 2000.

[31] J. Sweller, "Cognitive load theory, learning difficulty, and instructional design," *Learning and instruction,* vol. 4, no. 4, pp. 295-312, 1994.

[32] K. A. Kiewra, S. L. Benton, S.-I. Kim, N. Risch, and M. Christensen, "Effects of note-taking format and study technique on recall and relational performance," *Contemporary Educational Psychology,* vol. 20, no. 2, pp. 172-187, 1995.

[33] T. Makany, J. Kemp, and I. E. Dror, "Optimising the use of note-taking as an external cognitive aid for increasing learning," *British Journal of Educational Technology,* vol. 40, no. 4, pp. 619-635, 2009.

[34] K. A. Kiewra, S. L. Benton, S.-I. Kim, N. Risch, and M. Christensen, "Effects of note-taking format and study technique on recall and relational performance," *Contemporary Educational Psychology,* vol. 20, no. 2, pp. 172-187, 1995.

[35] C. Liu, C.-L. Yang, J. J. Williams, and H.-C. Wang, "Notestruct: Scaffolding note-taking while learning from online videos," in *Extended abstracts of the 2019 CHI conference on human factors in computing systems*, 2019, pp. 1-6.

[36] U. H. Graneheim and B. Lundman, "Qualitative content analysis in nursing research: concepts, procedures and measures to achieve trustworthiness," *Nurse education today,* vol. 24, no. 2, pp. 105-112, 2004.

[37] M. T. Chi and R. Wylie, "The ICAP framework: Linking cognitive engagement to active learning outcomes," *Educational psychologist,* vol. 49, no. 4, pp. 219-243, 2014.

[38] Y. Taskin, T. Hecking, H. U. Hoppe, V. Dimitrova, and A. Mitrovic, "Characterizing comment types and levels of engagement in video-based learning as a basis for adaptive nudging," in *Transforming Learning with Meaningful Technologies: 14th European Conference on Technology Enhanced Learning, EC-TEL 2019, Delft, The Netherlands, September 16–19, 2019, Proceedings 14*, 2019: Springer, pp. 362-376.

[39] T. Hecking, V. Dimitrova, A. Mitrovic, and U. Hoppe, "Using network-text analysis to characterise learner engagement in active video watching," 2017.