CapSmith: Contextual Refinement of Additive Text Captions for Video Storytelling

Saehui Hwang
Stanford University
Stanford, CA, USA
saehui@stanford.edu

Jean-Peïc Chou
Stanford University
Stanford, CA, USA
jeanpeic@stanford.edu

Anh Truong
Adobe Research
San Francisco, California, USA
truong@adobe.com

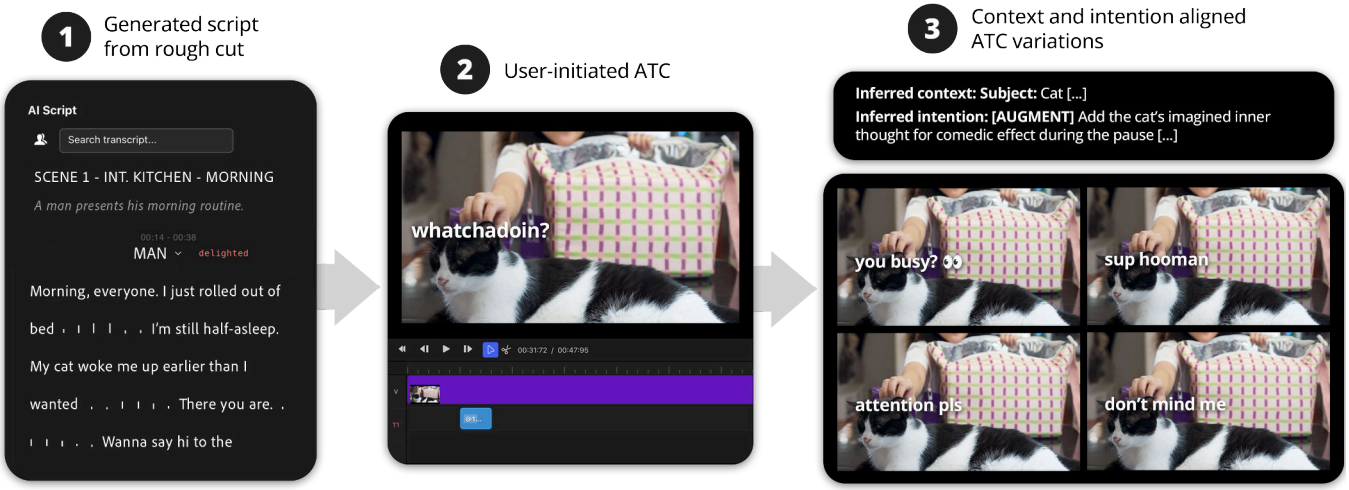

Figure 1: CapSmith is an ATC authoring tool that supports refinement from a user-initiated draft. Starting from a rough cut, it generates a rich formatted script that creators can use to anchor new ATCs. CapSmith then generates ATC variations grounded in inferred context and intention.

Abstract

Additive Text Captions (ATC) are powerful storytelling devices that extend video narratives by progressing story ("meanwhile..."), focusing attention ("HUGE!"), and providing unspoken context ("crashing"). ATC authoring is difficult because it requires balancing creative decisions (word choice, tone, etc.) against practical constraints (timing, readability, visual harmony). Current video-editing interfaces treat text as timeline-bound assets, prioritizing temporal manipulation over narrative function. While LLMs can support creative writing, our formative study (n=8) showed creators resisted LLMs that write ATCs from scratch, citing control and authenticity concerns, but wanted help refining drafts. We designed CapSmith, an ATC-authoring system that features (1) a script panel that externalizes narrative structure and (2) a variation-from-example workflow that proposes variants grounded in inferred intent and context. A comparative study (n=10) showed that the script panel contextualized ATCs against the narrative and that variation-from-example workflow supported refinement without requiring creators to articulate tacit intent while preserving authorship.

CCS Concepts

• **Human-centered computing** → **Interactive systems and tools**; • **Computing methodologies** → **Artificial intelligence**; • **Applied computing** → *Media arts.*

Keywords

Creativity Support Tool; Human-AI Interaction; Video Editing

ACM Reference Format:
Saehui Hwang, Jean-Peïc Chou, and Anh Truong. 2026. CapSmith: Contextual Refinement of Additive Text Captions for Video Storytelling. In *Creativity and Cognition (C&C '26), July 13–16, 2026, London, United Kingdom.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3803784.3807559

1 Introduction

Additive Text Captions as shown in Figure 2 are powerful storytelling devices for video as they allow creators to extend the existing story. We define Additive Text Captions (ATC) as on-screen text that *adds a narrative layer*, such as inner thoughts, clarifications, and jokes. Text overlays explored in prior works serve the purpose of reinforcing what is said in the video (e.g., closed captioning, subtitles, keyword emphasis, lyric videos) [13, 33, 38], and are documented

C&C '26, London, United Kingdom

ACM ISBN 979-8-4007-2583-8/26/07
https://doi.org/10.1145/3803784.3807559

Figure 2: Examples of Additive Text Captions (ATCs) in unscripted vlogs. ATCs respond to a range of contextual anchors—including visuals (e.g., a character's expression), audio events (e.g., laughter), narration (e.g., voiceover), and transitions. They serve various functions such as revealing inner thoughts, adding humorous commentary, or reinforcing a narrative beat. Frames from "3 days in hong kong ◦ a mini guide + vlog" and "summer in new york means good food & vibes" by Isa Sung, YouTube, licensed under CC BY.

to have accessibility [39] and retention benefits [38, 59]. In contrast, creators also deploy text in a more creative way to shape story beats, direct attention, and inject humor — hence the term *additive*. ATCs have roots in long-standing broadcast practices in Asian entertainment formats (e.g., telops[1] and impact captions [14, 28, 50]), and have become pervasive on platforms like YouTube and TikTok, providing clarity and humor particularly for unscripted moments [28]. They also sit within a larger lineage of multimodal media [36] such as memes [71], comics [51], and annotated visuals [29] where text reshapes how audiences interpret the underlying content [6, 7, 15, 24]. Yet despite their ubiquity, the authoring process of ATC remains largely understudied.

ATC authoring is demanding as it requires balancing multiple *creative* decisions (i.e., word choice, tone, humor, and trends) against *practical* constraints (i.e., timing, duration, readability, and visual harmony) [38]. Timeline interfaces inadequately support the ATC authoring process by forcing creators to focus on low-level frame manipulations instead of the ATC's meaning and narrative fit. Recent advances in multimodal large language models (MLLMs) [16] offer promise for assisting ATC ideation and refinement. These models can reason over video, audio, and text to summarize events, infer emotional beats, and propose multimodal suggestions [27, 64, 65]. Despite these capabilities, prior work has identified multiple human-AI interaction pitfalls [2, 53, 57, 72] that are amplified in creative contexts such as fixation [57, 63], style mismatch [70], and homogenization [4, 12].

Consistent with these concerns, our formative study with eight ATC creators surfaced three tensions that fundamentally limit end-to-end automation of ATC authoring. First, the intent of an ATC (what the ATC is trying to accomplish) is often tacit and cannot be reliably inferred from the video or audio alone, because ATC may add meaning that is not explicitly shown or spoken. Second, the context (what aspects of the video the ATC is grounded in) is variable in scope: each caption within a video may refer to different aspects of the story such as a character, an object, or a storyline. Third, because ATCs are closely tied to a creator's voice—often functioning as commentary or inner thought—systems that automatically write them can threaten authorship and authenticity.

Beyond these tensions, our formative study also revealed the diverse intentions that motivate ATC use. Creators employ ATCs to *Amplify, Disambiguate, Augment (with outside information), and Redirect* audience interpretation toward higher-level goals such as *creating interesting moments*, *crafting character*, and more. Participants expressed a strong desire to maintain creative autonomy over candidate moment selection (where to place) and producing initial ATC drafts (what to write). At the same time, because ATC authoring is a highly iterative refinement process, participants commonly used outside resources (e.g., social media, collaborator feedback, LLMs) to *improve* initial drafts.

Guided by these findings, we built CapSmith, which features a script representation of the narrative and a variation-from-example workflow: creators seed an ATC draft and the system proposes refinements grounded in the inferred context and intent. CapSmith uses our domain-specific intent taxonomy to surface and operationalize intent, giving creators controls to steer variations without heavy prompt specification. We explore scripts—a widely used textual representation in filmmaking—as a complement to the timeline for ATC authoring.

We use CapSmith to explore three research questions: (RQ1) How does having an explicit script representation affect the ATC creation process? (RQ2) How can ATC authoring be supported when intent is tacit and context is variable? and (RQ3) What are the tensions between variation-from-example and prompt-based chat workflows? We conducted a study involving 10 video creators who interacted with two design probes: CapSmith and a video-aware, chat-embedded timeline editor. Results show that CapSmith's script representation supported reflection, ideation, placement, and evaluation of ATCs. While the chat-embedded timeline interface required editors to articulate intent and context, which was burdensome and disengaging, our workflow supported refinement of ATCs while preserving their intent. Together, these results can inform design directions for AI-assisted creative tools where intent is tacit, context is variable, and authorship matters.

Our work makes the following contributions:

- **A formative study** (n=8) with ATC video creators, yielding (1) a taxonomy of intentions and goals in ATC creation and (2) an account of ATC workflows and challenges.
- **CapSmith**, an ATC-authoring system following a variation-from-example workflow that generates intent-aligned, contextually grounded variants from user-drafted ATCs.
- Findings from a **user study** (n=10) demonstrating the effectiveness of CapSmith and highlighting the importance of common-ground establishment through inferred and editable intent and context, allowing users to steer generation with minimal prompting friction and enhanced sense of authorship.

[1]Telops (short for "television opaque projector") refer to text or image superimposed in TV programs [50].

2 Related Work

Prior work has examined how text in media can steer audience interpretation [7, 15, 50]. We extend this line of work by further examining ATC authoring as a creative practice. We draw on prior work on creativity support tools and co-creativity to motivate new interfaces and workflows for ATC authoring.

2.1 Automatic Generation of Text Overlay

Automated tools for generating text overlays on video have largely focused on transcribing existing audio into on-screen text rather than crafting a new layer of meaning, as ATC does. Auto-captioning systems widely deployed in consumer editors—such as those in CapCut, TikTok, and Adobe Premiere—use speech recognition to place text on screen as open captions. Beyond speech, OnomaCap transcribes non-speech sounds into onomatopoeia [31], and other systems label non-speech sounds with text or graphics [3]. Typographic elements such as emojis [42, 55], colors [19, 55], and fonts [18, 19, 30, 34] have been added to videos for enhanced accessibility and empathy. Lyric video pipelines have been proposed to automatically tackle challenges regarding timing, readability, and visual harmony [38]. Other systems add structural text, such as converting document headings into on-screen overlays [13], or generating emphasis text effects [27]. Platforms overlay audience-authored comments in danmaku (e.g., Bilibili, Niconico) and live-stream chat (e.g., Twitch) [68, 69], but these serve social interaction rather than creator-driven storytelling. To our knowledge, prior work has not studied tools for creators to author overlays that intentionally add new meaning to a video.

2.2 Transcript-based Navigation and Segmentation

It is well documented that timeline-centric editing interfaces make it hard to locate moments quickly from video. Because video is time-based, editors often must scrub linearly while cross-referencing audio and video tracks to pinpoint events [11].

Prior works have explored treating video as a text document to enable precise navigation, segmentation, and search. Examples include time-aligned transcripts that highlight the currently spoken word for frame-accurate scrubbing [44]; auto-generated digests with chapter/section markers for rapid skimming [45]; and document-style views that combine shot actions, summaries, and dialogue for quick scene location [43]. Project Blink showcased a transcript that visualized sound effects and audio amplitudes as well as separated speakers [40].

Transcripts have been used not only for navigation but also for editing [8, 22]. Silver was an early transcript-based editor whereby users were able to edit the video through text [11]. Prior work showed users prefer editing voicemail through a transcript rather than audio waveforms, since text supports scanning and phrase-level edits [67]. Commercial software offers similar interactions: Adobe Premiere Pro's Text-Based Editing enables navigation, correction, and trimming from a transcript and generates caption tracks from speech [1]. Avid's ScriptSync aligns dialogue text to source video for rapid search and footage organization [5]. In contrast to systems that display only a raw transcript, CapSmith introduces a *script representation* that structures the video into scenes and adds scene descriptions. This design borrows from filmmaking practices, where scripts convey narrative structure and emotions, not just spoken dialogue [49].

In terms of adding new assets through transcript interaction, Rescribe supports the insertion of audio descriptions [46], B-Script enables B-roll addition [26] and QuickCut facilitates aligning raw footage with the narration transcript [61]. Descript, a commercially available software, supports the addition of new text overlays directly from the transcript [20], but provides limited tools for visualizing, refining, or manipulating them in the transcript after creation. Post-edit adjustments must occur in the timeline, creating a cognitive divide between the textual authoring of ATC and its temporal alignment.

Our work explores supporting within-modality authoring and manipulation of ATCs directly in a script interface, treating them as text rather than as a timeline element.

2.3 Intent Alignment in Co-Creativity

Challenges in aligning AI outputs with user intent have motivated a growing body of work on steering generative AI towards intended outcomes [32, 57]. Systems have supported intent communication through free-form prompts [37], elicited intent through clarification [52, 54, 62], and inferred intent from chat history [48, 54, 58]. While effective in some domains, these approaches have limitations that become acute in creative workflows where intent is often not clear upfront, difficult to articulate, and evolves throughout a project [56, 60, 62].

Prompt engineering tools for creative tasks have focused primarily on image generation, providing support for discovering effective prompts [9, 66] and extracting relevant contexts [21]. Other systems incorporate richer inputs such as sketches or multimodal contexts [17, 35, 47], but still require users to specify a prompt. To support intent alignment in video workflows where intent is difficult to articulate and context shifts rapidly, our variation-from-example workflow positions user-drafted ATCs as a starting point and systematically infers intent and context.

Surfacing intent as an explicit control layer has been shown to support alignment in generative tools. ToMigo [25] infers intent from prompt and reference images and exposes a design concept graph for steering generation. Intent Tagging [23] similarly surfaces intent as an explicit control layer for creative refinement. CapSmith builds on these approaches to ATC authoring by grounding intent in a domain-specific taxonomy derived from creator practice.

3 Formative Study

3.1 Participants and Procedure

We recruited eight participants with prior experience authoring ATCs: four professional video editors (P1–P4) producing client work and four solo creators (P5–P8) editing their own content. We recruited participants via a creator forum and word of mouth. All participants primarily produced long-form videos (>10 minutes), mostly from unscripted footage. Three participants (P1, P2, P5) had prior experience using commercially available LLMs to support their ATC creation process. Additional participant details can be found in Table 1.

Participant	Experience	Type of Content	Years of Experience with ATC
P1	Professional	YouTube variety shows	1-3
P2	Professional	YouTube variety shows, Informational	4-6
P3	Professional	YouTube variety shows	6+
P4	Professional	Educational, Documentaries	4-6
P5	Solo Creator	Hobbyist	6+
P6	Solo Creator	Hobbyist	1-3
P7	Solo Creator	Hobbyist	1-3
P8	Solo Creator	Hobbyist	<1

Table 1: Background information about the participants of the formative study.

All sessions were conducted remotely via video conferencing, lasted approximately 60 minutes, and participants received $40 USD compensation.

Each session began with an interview about participants' video genres, ATC goals, editing tools, and perceived challenges. Participants then completed an ATC walkthrough, screen-sharing one of their own videos edited with ATC while thinking aloud about intent, timing decisions, and difficulties. This was followed by an ATC critique of three short clips from multiple genres, where participants discussed ATC impact, effectiveness, and alternative designs. Next, participants completed an ATC authoring task using their preferred video editing tool on a work-in-progress video they had prepared. Sessions concluded with a semi-structured interview examining desired tool features and boundaries of automation. Two authors reviewed the transcripts in multiple passes through inductive thematic analysis [10].

3.2 Intentions and Goals in ATC Creation

An effective interface design ensures alignment with the creators' goal-based intentions [41]. Our formative study revealed consistent intention patterns during ATC creation.

3.2.1 Intentions. Creator intentions for employing ATCs fell into the following taxonomy: *Amplify*, *Disambiguate*, *Augment*, and *Redirect*. These categories reflect how ATCs are used to influence audience interpretation for a given moment. Figure 3 illustrates examples from this taxonomy.

Amplify. Creators use ATCs to *amplify* an existing interpretation. For example, P3 placed a question mark next to a surprised character to "heighten the moment and highlight his big surprise."

Disambiguate. ATCs also help *disambiguate* speech in cases when audio quality is bad (P4, P6), when the speaker has an accent (P6, P8), or when the wrong thing was said (P8). P3 described adding a "feeling bad" ATC to preempt conflicting viewer interpretations of a character's expression: "Depending on the person, some might not even think he is sorry… but adding [the ATC] makes everyone immediately understand."

Augment. Creators further use ATCs to *augment* viewer interpretation by introducing information that would otherwise not be available from the audiovisual signal—e.g., adding emotion to neutral characters (P4), details missing from the video (P5, P6), or afterthoughts and commentary (P8).

Redirect. Finally, ATCs enable creators to *redirect* audience interpretation by reframing a scene's meaning. For instance, P2 added the ATC "let's focus" to a scene where a character hits a sandbag, noting "through the text overlay I explain that this act is a means to focus more, not because he's angry."

3.2.2 Goals. Goals of ATCs also emerged as a result of our thematic analysis. Overall, participants described ATCs as especially valuable for unscripted videos such as vlogs. P4 added that films don't need ATCs "because the story is well built and well planned" while "vlogs need a lot of context ... and I need to create moments [with ATC]."

G1: Create interesting moments. Creators said ATCs often drove up engagement (P1) and retention (P4). Creators used ATCs to make otherwise ordinary footage engaging and to heighten anticipation (P3, P5). Participants described ATCs as "adding another layer of story" (P7) or even "making a skit out of the footage" in post-production (P5).

G2: Progress the story. ATCs establish structure by marking beginning, middle, and end (P3), create sections (P6) and transitions (P1, P5). ATCs also have a practical role of allowing editors to cut down by establishing context (P5) or sometimes explaining context that was not filmed (P7).

G3: Direct viewer attention. ATCs act as visual pointers to objects or on-screen details (P7). They also serve pragmatic editorial needs: P2 used ATCs to hide jumpcuts—"as the eyes are drawn to the text overlay, viewers don't notice the cut. I can add colors to draw the eye, or add animation."

G4: Craft character. Participants intentionally shaped persona through ATCs. P4 aimed to create a likeable, self-deprecating character with pop-culture references by using ATCs. P2 considered ATCs as a way to make characters more relatable.

G5: Cue audience reaction. Like laugh tracks in TV post-production, ATCs signal to the audience how to feel in the moment (P7). P3 joked that ATCs can "gaslight the audience... [I'm telling them they] should be excited."

3.3 Workflow

Participants consistently described ATC authoring as a post-production activity, typically beginning after rough cut completion. However, some leave notes or placeholder ATCs during editing, and return later to refine wording. Rather than following a universal recipe, creators rely on intuition (P3, P4) to identify moments that

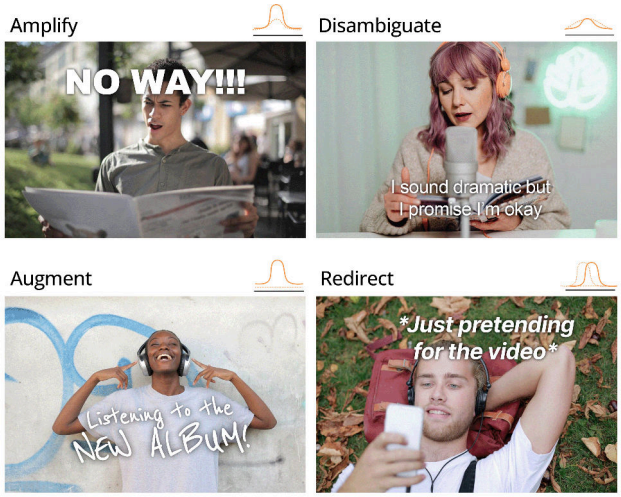

Figure 3: Intention Taxonomy of Additive Text Captions (ATC). ATCs may *Amplify* the tone or emotion of a moment (e.g., exaggerated disbelief), *Disambiguate* ambiguous speech or tone, *Augment* the narrative with additional context or information, or *Redirect* the viewer's interpretation through commentary. Images from Adobe Stock, licensed under Adobe Stock license.

"don't make sense" (P5), lack intensity (P4, P6), or need narrative reframing (P2). Workflows involve ideation, placement, refinement, and evaluation as follows.

W1: Anchored Ideation and Subject. When describing how they decide to add ATCs, participants repeatedly referenced two things: (1) the entity/event the caption was about and (2) the cue that prompted its addition. We respectively define these as *the subject* and *the anchor* of an ATC. For example, a loud thud sound prompted P4 to add the ATC "There goes the tripod." Here, the anchor was the thud (audio cue), while the subject was the tripod. Anchors ranged from speech segments to emotion, sound, and salient visual cues.

W2: ATC Placement. Participants emphasized that timing and frequency were highly dependent on the video's pacing, content, audience, and overall mood. For instance, P4, who produces educational content, favored more frequent ATCs to sustain engagement. In contrast, P1, who edits K-pop idol content, noted that aggressive ATCs tend to distract rather than help because fans frequently screenshot and remix frames, prioritizing the star's face and voice.

W3: Refinement. After drafting initial ATCs, editors iterate to ensure consistency and refine wording choices. Editors working in a team solicit feedback from coworkers (P2, P3, P4) while solo creators rely on re-watching their own edits. P8 explained, "Usually I would just write down what comes to mind. And then, after I watch it a few times, I'm like, oh, this doesn't quite feel right, or let me tweak it to match the tone I'm going for. I just let it sit."

W4: Evaluation. To properly evaluate an ATC, participants emphasized the need to audition them directly on the video. Judgments of an ATC's readability, timing, and pacing only made sense alongside the footage and audio. As P5 stated, "Seeing it in context, changes how good the text is." Editors often test multiple ATC options in batches, selecting the version that lands best (P4).

3.4 Challenges

C1: Timing. Participants repeatedly described timing decisions as one of the tedious parts of authoring ATC. Because ATCs must align tightly with both spoken words and the visual frame, editors often fine-tune the timing frame-by-frame, scrubbing through the video to inspect gestures, expressions, or cuts, while also relying on the audio waveform to ensure the ATC appears at the precise moment that complements the speech (P4).

Current timeline-based editors treat text as video clips, requiring users to examine text at the frame level. This design separates the tasks of authoring and temporal alignment. As a result, users often become overwhelmed by the mechanics of precise temporal placement and lose focus on what to write and how it contributes to the narrative.

C2: Deciding and Iterating Content. Users described crafting appropriate ATC wording as a time-consuming, iterative process. Editors emphasized several key criteria: avoiding offense (P5), striking the right tone ("not cheesy or corny," P3), and matching the intended audience. Much of their effort involves revising or replacing drafts. Professional editors often search online (e.g., YouTube) for phrasing ideas, but dislike that this process makes "editing time drag on" (P1, P4).

Some editors experimented with LLMs, especially for rephrasing or adjusting tone. P2 shared: "I struggle with deciding what sentence structure to use ... now I use AI sometimes to find a sentence that fits the situation ... I explain a bit of the overall context to the AI and get several suggested sentences." P2 even built a custom GPT for generating informational ATCs for healthcare videos, prompting extensively for tone adjustments such as "more formal." However, effective use of LLMs required extensive, time-consuming prompting to capture tone, situation, and nuance.

C3: Audience matters. Professional editors (P1, P2, P3, P4) routinely monitor trending videos, memes, and language on platforms such as X and GIPHY to keep ATC timely and resonant. Creators are highly cognizant of who the video is for, tailoring word choice and typography to audience expectations and platform norms.

C4: Limits of AI Assistance. Chat-based systems often had friction for editors who experimented with LLMs. These participants found AI use too slow due to the need to explain context and intent, making it impractical for frequent ATC authoring (P1, P8).

Moreover, editors repeatedly emphasized that they did not want a tool that writes ATCs for them: existing LLMs often failed to match their personal style. For instance, P7, whose ATCs mimic their texting style, remarked that "I wouldn't like getting suggestions (on the ATC) because it's just so personal." Several participants rejected LLM suggestions describing them as having an "unrealistic tone" (P3) or lacking the nuance required for humor (P5). Editors were typically more open to LLM suggestions when they already had a clear intention and sought refinement rather than generation.

4 Design Guidelines

Our formative study surfaced key challenges and opportunities, from which we derive four design guidelines to better support ATC authoring.

DG1 Provide contextual evaluation of outputs. As a multimodal overlay, an ATC's readability, emotional impact, and narrative flow depend on the surrounding audiovisual elements. Rather than judging text in isolation, a contextual evaluation environment enables creators to make informed decisions about ATC effectiveness (W4, C1).

DG2 Support context-aware co-creation. Crafting ATCs requires acute awareness of the video's audience, tone, and situational context (W1, C3). To reduce the need for creators to manually specify these elements (C4), systems should support context-aware co-creation by surfacing interpretable and editable contextual scaffolding. Informed by our formative study, we define **context** as a combination of:

- The subject: the entity ATC is about
- The anchor: the piece of content that ATC is bound to (i.e., a sound, dialogue, video frame)
- The audience and video style

Automatically identifying and presenting this context helps establish common ground between system and creator, reduces prompting overhead, and provides relevant reference points for better-aligned ATC generation (C4).

DG3 Support intent-based variation. Participants demonstrated resistance to automated ATC writing, emphasizing the personal nature of their creative voice (C4). At the same time, creators needed support refining specific wording once they had a rough idea (C2). Intent-based variation refers to generating ATC suggestions that align with inferred and user-refined intentions, supporting the kind of iterative exploration participants valued (W3). Rather than generating ATCs from scratch, this approach respects creator autonomy by surfacing meaningful alternatives that preserve the creator's creative intention.

DG4 Surface and operationalize the context and intent. In current chat-based LLMs, the system has no explicit representation of relevant context or understood intent (C4). Interfaces should externalize the system's current understanding of context and intent as metadata attached to each ATC. Making context and intent explicit creates shared common ground between creator and system, supports rapid diagnosis of misalignment, and enables controlled refinement.

5 CapSmith

Following these design guidelines, we present CapSmith (Figure 4), an ATC authoring tool designed to support refinement and evaluation of ATCs. CapSmith features a script panel for narrative representation (Section 5.1), a video player (Section 5.2), and timeline (Section 5.3) for navigation and playback, and a refinement panel for iterating on user-initiated ATC (Section 5.4). To illustrate how CapSmith supports ATC authoring, we walk through a scenario in which a creator, Tom, uses the tool to add ATCs to his vlog about his New York life with his cat. After creating a rough cut of his vlog, Tom realizes that some moments could benefit from additional context and narrative flair.

5.1 Script Panel

Tom first imports the rough cut of his vlog into CapSmith. The system constructs a script of the video, derived from the transcript and information extracted from the video. The script is segmented into scenes. Each scene is displayed (1 in Figure 4) with a heading specifying the scene location, time of day, the scene description followed by the dialogue. The dialogue is segmented according to the speaker. Users can rename characters and edit the segmentation if desired. Dialogue segments are annotated with sound events and analyzed emotions, which we call dialogue parentheticals, borrowing screenplay terminology. CapSmith also visualizes non-speech segments using descriptive tags and amplitude bars, with silences appearing as dots. Each bar or dot corresponds to 0.1 seconds of the video. Users can directly select parts of the script to anchor new ATCs. This script representation enables users to quickly navigate the video and create ATCs grounded not only in dialogue, but also the narrative beats and sound events. As he plays back the video, the corresponding region in the script is highlighted in real time.

In the first scene, Tom is interrupted mid-sentence by his cat, a moment he identifies as potentially comedic. Tom quickly locates the dialogue line in the transcript and highlights the cluster of silence dots that mark the interruption. He types "whatchadoin?" as an initial draft, framing it as an amusing internal monologue for the cat. The ATC is inserted as a block above the selected silence in the transcript. Tom can stretch, shorten, or reposition this new block within the script panel to adjust its timing and duration.

5.2 Video Player

To provide complementary anchoring to elements not captured in the script such as the visuals, CapSmith also allows the user to create ATCs directly in the video player.

While playing back the final scene of his vlog, Tom notices his cat strolling into the background, a visual cue absent from the script. In response, he selects the text tool in the upper-right corner of the video panel (2 in Figure 4), and places a playful ATC noting the cat's recurring appearances, *"guess who's back,"* at the spot where his cat appears.

5.3 Timeline

As Tom adds ATCs, visual boxes representing the duration of the text overlay appear in both the script panel and the timeline (3 in Figure 4). Tom can also use both the timeline and the script to make timing adjustments.

5.4 Refinement Panel

After drafting some initial ATCs, Tom switches to the refinement panel (4 in Figure 4) to polish them. When he selects an ATC, the system automatically infers and displays two types of information: the context and the intention **(DG4)**.

Context. The system provides the full context as defined in DG2. It identifies the ATC's subject (the cat) along with the anchor (selected silences from the script). If he had authored an ATC anchored to a verbal transcript segment (not silences as in this case),

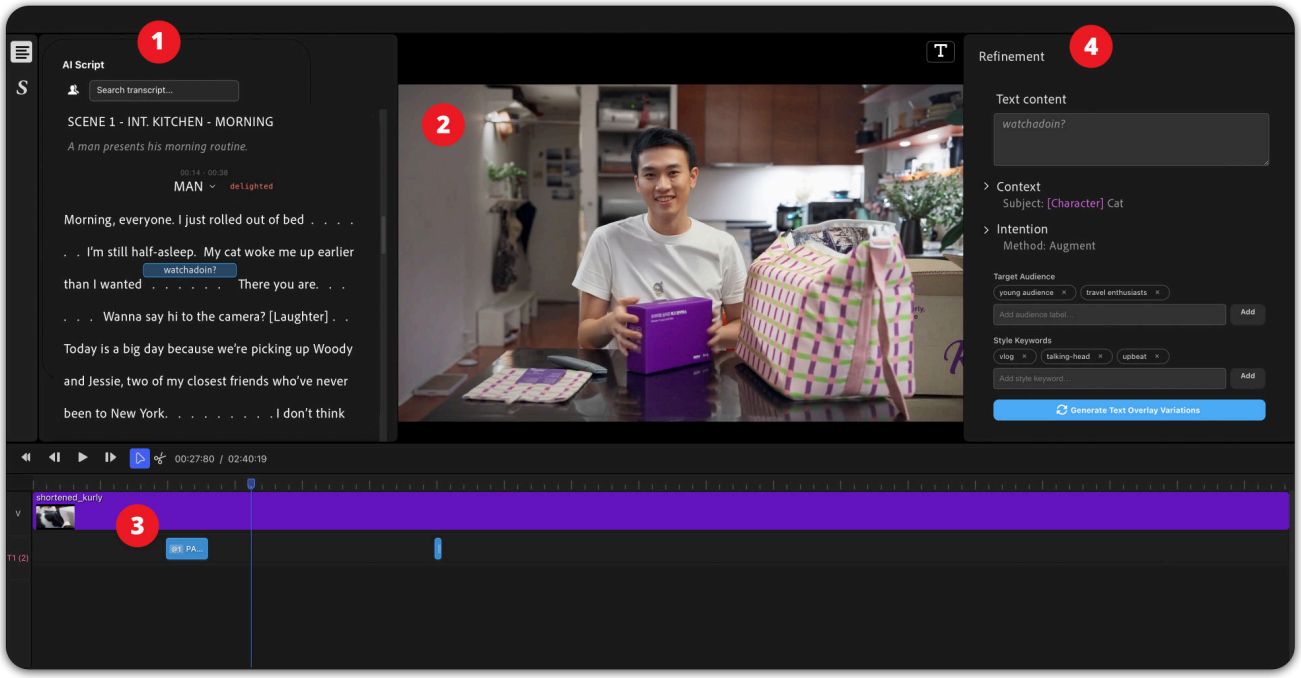

Figure 4: Overview of CapSmith interface. Users can both browse the Script Panel (1) or playback the video (2) to identify cues as anchors for adding ATCs. Authored ATCs are placed in both the script as text blocks and on the timeline as clips (3). The refinement panel (4) allows users to inspect and revise the inferred context and intent, and lets creators generate aligned variations.

the active speaker (Tom), and dialogue parentheticals (if any) at that timestamp would also be visualized. In addition, the system infers target audiences (young audience, travel enthusiasts), and video style keywords (vlog) for the video as a whole. Together, these elements situate the ATC within its narrative.

Intention. The system classifies what the ATC tries to accomplish using the intention taxonomy: *amplify, disambiguate, augment,* or *redirect*. For example, Tom's ATC *"whatchadoin"* is labeled as *Augment*, with the rationale: "adds the cat's imagined inner thought for comedic effect during the pause."

All of these fields make the system's assumptions explicit and transparent, and Tom can override or edit them if needed. For instance, the intent classification is editable, and Tom can also add free-form notes to refine the system's interpretation (e.g., "keep it playful, not snarky").

When Tom clicks *Generate Variations*, CapSmith produces ATC variations grounded in the displayed context **(DG2)** and aligned with the inferred intention **(DG3)**. Each suggestion can be previewed in the video player to immediately check timing, readability, and visual fit **(DG1)**. For Tom's "whatchadoin" ATC, CapSmith suggests variations such as "you busy? 👀" and "attention pls" that maintain his intent while catering to his young audience using emojis. Tom likes the idea of using emojis and selects the former, which refines his original ATC in place.

6 Implementation

CapSmith is a Node.js application that we developed using TypeScript and React. Figure 5 illustrates an instantiation of our system pipeline. After the user uploads a video, CapSmith computes a time-aligned transcript using AssemblyAI and uses Gemini 2.5 Pro to diarize the transcript, identify visible speakers, and segment the video into scenes with setting information, event descriptions, dialogue parentheticals and audio tags.

Users can initiate an ATC from either the script panel or video player. Each ATC is automatically bound to an anchor representing the referenced dialogue, sound, or visual element (Appendix A.2). Anchors created from the script panel capture transcript and audio metadata, while those created from the video player capture frame timing and placement coordinates. CapSmith uses Gemini 2.5 Pro to infer the ATC's referenced subject from the ATC text and anchor. The model classifies the subject (one of Character, Object, or Narrative), identifies relevant entities when applicable, and provides a rationale (Appendix A.3.2). CapSmith also prompts Gemini to classify the creative intent behind the ATC using our intent taxonomy (Appendix A.3.3). Finally, CapSmith generates 16 ATC variations conditioned on the inferred intention and context (Appendix A.3.4).

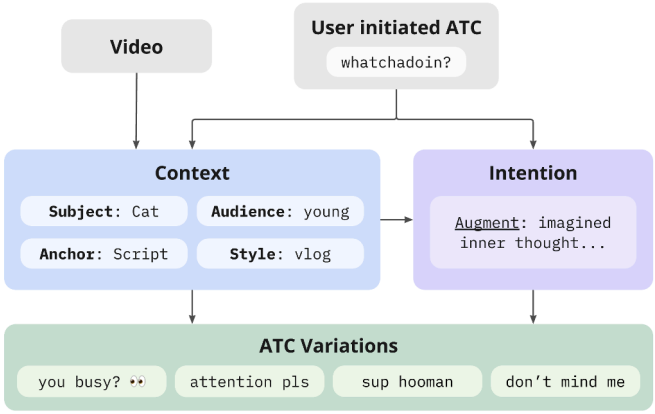

Figure 5: An instantiation of the CapSmith pipeline. Starting from a user-initiated ATC draft, the system constructs context by inferring the subject and identifying the relevant anchor. It then infers the user's intent based on both the draft and context. After surfacing these elements for user inspection and refinement, CapSmith generates caption variations that are both contextually grounded and aligned with the inferred intent.

7 User Study

We conducted a study inviting participants to add ATCs to their videos using two probes: CapSmith and a chat-embedded timeline editor (Figure 6). To our knowledge, confirmed by ATC creators in our formative study, no dedicated tools exist for ATC authoring, so we chose a chat-embedded video editor as a baseline. This baseline has full context of the video and can respond to free-form prompts pertaining to the video. This setup allowed us to compare our design with a typical AI-assisted workflow. Through this study, we aimed to answer:

- **RQ1:** How does having an explicit script representation affect the ATC creation process?
- **RQ2:** How can ATC authoring be supported when intent is tacit and context is variable?
- **RQ3:** What are the tensions between variation-from-example and prompt-based chat workflows?

7.1 Participants

We recruited ten video creators who employ ATCs in their video content via direct outreach and word-of-mouth. Their background information can be found in Table 2. Participants worked across both long-form and short-form platforms, and their content spanned a range of domains including travel vlogs, education, beauty, and marketing. This variation ensured diversity of workflows, narrative structures, and captioning styles.

7.2 Procedure

To counter learning effects between the two tasks, we asked creators to bring two videos of similar length and editing style. We began with a brief background interview on current ATC practices (5 min). For each interface, participants received a short tutorial using a provided practice clip (10 min), then performed authoring tasks on their own video using a think-aloud protocol (20 min). After each interface, participants answered a series of 7-point Likert-scale questions evaluating our design guidelines (Appendix A.1).

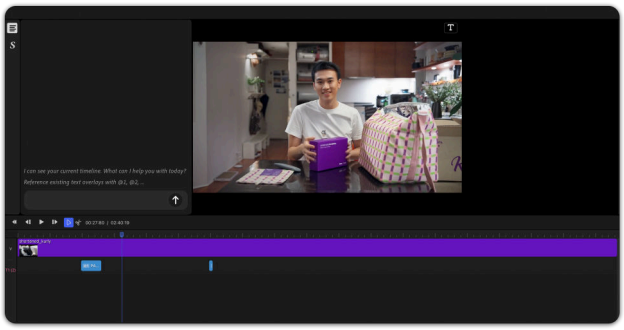

Figure 6: Overview of chat-embedded timeline editor. It integrated Gemini 2.5 Pro as a conversational agent with direct access to the user's video and a minimal system prompt ("You are a helpful AI assistant"). Participants interacted with the chat purely through free-form prompting, requesting caption ideas, revisions, or rephrasings. When users referenced existing captions by their displayed ID (e.g., `@1`), the chat received the caption text, timestamp, and duration. Participants then manually placed any accepted suggestions onto the timeline. Implementation details can be found in Appendix A.3.5.

We concluded with a semi-structured interview (20 min) probing experiences relative to our research questions. Sessions were conducted remotely and recorded. The interview was transcribed and we applied thematic analysis to analyze the interview transcripts. Total session time was approximately 90 minutes. Participants received a $70 Amazon gift card.

8 Findings

Eight out of the ten participants found satisfactory ATCs from CapSmith's suggestions without editing any fields and successfully

Participant	Experience	Type of Content
P1	Professional	Vlogs, Skits
P2	Hobbyist	Vlogs
P3	Hobbyist	Vlogs, Marketing
P4	Hobbyist	Vlogs
P5	Hobbyist	Vlogs
P6	Hobbyist	Podcast Videos
P7	Professional	Vlogs
P8	Hobbyist	Vlogs
P9	Hobbyist	Informational
P10	Hobbyist	Vlogs, Motivational

Table 2: Background information about the participants of the user study.

inserted the suggestions into their video, while the remaining two (P2 and P6) declined all AI outputs from both systems.

8.1 RQ1: Script Representation

CapSmith's script representation supported reflection, ideation, placement, and evaluation of ATCs in various ways.

8.1.1 Reflection: Provides narrative overview. Participants used the script to quickly get a high-level overview of their video without rewatching it in full. Some participants barely watched their video and relied almost entirely on reading the script. This was especially helpful in long videos as P6 explained, "If you're editing from, like, an hour or two of clips, it can be a lot to watch through and figure out when everything is happening." The script format aligned with how participants mentally structure their videos, especially matching their habit of chunking content into parts (P8, P9).

The script also helped identify moments of interest. Participants often had a strong sense of what types of moments they wanted to add ATCs to, such as funny mistakes or unexpected actions. Identifying these moments with the timeline interface required tedious scrubbing. CapSmith's script panel enabled participants to quickly locate moments of interest by skimming the script and searching for keywords (P1, P6, P8, P10).

8.1.2 Ideation: Supports ideation of ATCs. Participants found the script panel useful for ideation. P8 said, "seeing the words... [is] helping a lot to think of things [to add ATCs to]... Listening to me talk I would have missed it, but since it's visually there I can think of this idea," adding that seeing the text made it easier to connect the dialogue to "memes and pop culture."

Specifically, the script provided concrete anchors for ATCs and sparked ideas. P6 selected "this is my progress so far" and added "slow and steady"; P7 highlighted "smell" and added "ewwwww." P8 used a scene header marking a transition (falling asleep → walking to work) to add "good morning." Script cues also inspired multimodal editing ideas. For example, P8 considered adding the TikTok song that goes, "single mom who works two jobs" after seeing the word "work" in the transcript.

Finally, for informational ATCs, participants used the script to write effective ATCs without repeating the spoken words (P9, P10). P9 inserted a definition at the moment a concept was introduced, using the transcript to ensure the ATC complemented, rather than duplicated, the dialogue.

8.1.3 Placement: Facilitates ATC temporal alignment. Participants' experiences in the timeline interface echoed the temporal alignment challenges uncovered in the formative study (C1), which demanded frame-level scrubbing and inspection. The script panel let participants author ATCs directly alongside the transcript, reducing playback and memory load. P5 found it "much easier to just see the word and ... to snap [the ATC] to the word" while P8 remarked that the script panel "is a lot more helpful than the actual [audio] waveform."

8.1.4 Evaluation: Supports evaluation of overall narrative. Colocating ATCs in the script view helped participants quickly judge its coherence with the narrative. P5 described reading dialogue with ATCs as "as if I'm reading a paragraph ... I can see that it makes sense." Seeing the ATC in the script made it easier for participants to assess how the ATC fits the story (P6) and connect the ATC back to the narrative (P8). P2 used the script to check on direction and pacing, describing it as "a nice time to kind of step back and be like, what direction is this going?"

Seeing ATC variations in the video preview helped participants judge whether a variation worked. In the chat-embedded timeline interface, P5 emphasized that it was difficult to assess the viability of a suggested caption from a text-only output of the chat. In CapSmith, P1 noted that "having the ability of replaying those options very quickly and simply" was helpful for sensing whether "the text works" and for judging whether "the joke lands."

8.2 RQ2: Supporting refinement through surfacing and operationalization of tacit intent and context

For multiple participants, the prompt-based chat failed to align with their intent, context, tone, and style, leading to frustrating back-and-forth interactions. In contrast, CapSmith surfaced intent along with context to establish common ground and generate coherent variants (Figure 7), which participants consistently described as "closer to what I actually wanted" (P3) compared to the suggestions from the prompt-based chat.

8.2.1 ATC cannot be automatically generated from audiovisual signal alone. Participants emphasized that in general, much of what they want to convey through ATCs cannot be inferred from video alone (i.e., uncaptured moments, personal backstory, or inner thoughts), highlighting the limits of multimodal LLMs for automatic ATC generation. For example, P4 commented, "obviously [any AI system] won't know what happened in my trip like outside [the video]." Similarly, P9 explained that many of his ATCs draw on background context: "Context for that joke is that I was a finance major... A good joke is not always going to be evident from the video." The chat interface also substantially misinterpreted what should be emphasized within the video itself. P10 tried to add an ATC during a segment where she talks to the camera about how she focused on building relationships and briefly mentions that she had momentarily lost herself doing so. When P10 asked the prompt-based chat for suggestions for this moment, it generated the text

"I lost myself in that process," which she immediately rejected because "It was not something I would want the audience to focus on," explaining that she wanted to emphasize the relationship-building instead of the negative aspect of losing herself. This highlights the importance of incorporating creator intent in supporting ATC authoring.

8.2.2 Intent articulation is difficult. However, articulating intent explicitly through the prompt-based chat was a process creators found time-consuming and difficult. Because creators' goals are often abstract, translating them into words was challenging. As P10 admitted, "I'm not sure if I'm explaining my thought process just like, right to you [the researcher] what exactly I'm looking for." The effort was discouraging iteration and creation, as P1 reflected, "If I need to spend 10 minutes on each text overlay, I'd rather just keep it simple or have no text overlay." Even when the prompt-based chat eventually produced usable suggestions after multiple iterations, participants (P1, P3, P10) remarked that it "took longer than just writing it myself" (P3).

8.2.3 Variation-from-example preserves intent with low prompting burden. Conversely, users appreciated CapSmith's variation-from-example approach, which avoided the need for the user to articulate intent. P9 noted that "It did a very good job understanding the intention and it proves it, it shows it to you" while P10 enjoyed that CapSmith let her add "a little bit of extra spice" to her ATCs while still maintaining her original intent. This alignment arises from having a clear, shared vocabulary for intent: our domain-specific taxonomy turns otherwise tacit goals into legible, editable categories that help the system infer what the creator is trying to do, and gives creators a lightweight way to correct misalignment when needed.

8.2.4 Surfacing context establishes common ground. With the prompt-based chat, participants struggled to understand how much context to specify, leading to misinterpretation, and cognitively taxing back-and-forth prompting. For instance, P4 "thought [she] could be vague" since the prompt-based chat had access to the video and could answer questions about it. However, the prompt-based chat returned such irrelevant suggestions that she "[didn't] know at all where the context is coming from, is it the entire video? is it a certain moment?" This ambiguity around how context was used made steering outputs difficult. P3 remarked the cognitive load in "thinking how to give [the prompt-based chat] just enough information."

In contrast, by surfacing the inferred ATC context to the user, CapSmith gave participants tangible reference points. P9 valued that outputs clearly referenced "the words I'm saying," allowing him to specify "where and what the ATC is [about]." P4 saw the context editing capability as a direct way to improve suggestions when she felt dissatisfied with the output.

8.2.5 Provide explanations on-demand. In CapSmith, several participants preferred to skip the refinement panel and jump immediately to variations (P1, P2, P3, P8). P1 described, "I saw that the [suggestions] were great... if they weren't, I would go back to the intention and see if it was correct." Participants judged inference quality indirectly by auditioning variations in the video preview and found long explanations unnecessary or burdensome (P2, P8, P3). P2 explicitly did not want to see explanations because they felt too much like interacting with a chat agent. Similarly, P3 noted, "I already know what my intention is; I don't need AI to explain it to me." Although the explanations were designed to establish common ground, in practice they added cognitive load. To avoid this, reasoning explanations should be available on-demand to support control when needed, rather than restating the video understanding back to the user.

8.2.6 Support refinement through inferred tone and style. Tone and style preferences for ATC are also difficult to articulate through open-ended prompting because they are highly subjective and variable within the same video. For example, P8 wanted the early part of his video to adopt a "gen-z, funny-awkward" tone, while shifting to a "less funny, more honest" tone for a later segment. Creators (P8, P3, P4, P9) found that the ATC recommendations from the prompt-based chat lacked the right voice, describing the suggestions as "kind of cringe... like I gave this task to my mom" (P3) and "not really my voice" (P4). Participants were able to improve suggestions by prompting more explicitly for tone and style over several rounds, but found the process exhausting: "I found it extremely cringe until I specified exactly what I wanted. It got it eventually... but I had to prompt it extensively towards my style" (P8). Users also found it difficult to prompt from a blank textbox (P1) and struggled to envision what the "options are" (P9) for expressing tone. By comparison, participants appreciated CapSmith's pre-populated fields for audience and style, which encouraged reflection on otherwise overlooked dimensions (P5), and were "easier to edit than to write from scratch" (P9).

8.3 RQ3: Tensions between variation-from-example and prompt-based chat workflows

8.3.1 Authorship and Authenticity. Several participants felt that relying heavily on prompt-based chat would dilute their voice. They cautioned that if a system "suggests the same thing to everyone," (P8) then videos would converge. P4 noted that using the prompt-based chat "in a larger capacity" would reduce her sense of ownership. As P8 put it, "Creativity is a muscle to train. [ATCs] should come naturally to you, and having an AI suggest things from the very beginning stomps that [opportunity]."

In contrast, through CapSmith, participants felt a stronger sense of ownership from needing to seed an initial ATC and generating variants (Figure 7). P9 emphasized that transparency into CapSmith's reasoning also enhanced authorship: "the curtain is open a bit more. I can see the processes and feel more involved." As such, we observe that required user effort during ATC generation was correlated with perceived authorship.

However, some participants also expressed concerns about authenticity and authorship regarding AI-assisted ATC authoring that limited adoption. P2 emphasized that her text overlays are "super authentic to myself and my thoughts," and she would not accept suggestions "as is"; in fact, she did not accept any suggestions. For her, audience keywords were insufficient to capture tone. Even after verifying that the inferred audience, "Young Adult" was correct, she explained, "I'm very particular about the slang I use…

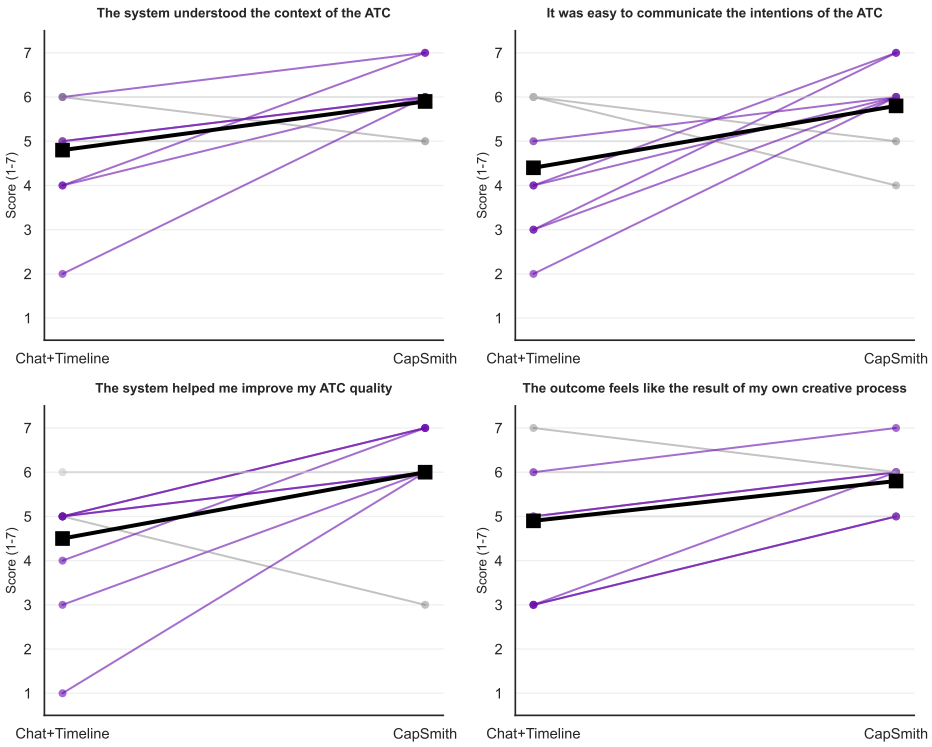

Figure 7: Quantitative results from post-study survey.

unless it's actually part of my own lexicon, I'm not going to say 'it's giving,' because I don't say that in real life." Other participants voiced similar reservations. P8 explained that if he wanted to push his creativity, he would not use any AI-assisted ATC tools. P10 anticipated a failure mode in creativity where she might "get lazy and not even try formulating my own ATC."

Interestingly, we observed a quality–agency paradox; higher-quality suggestions sometimes reduced participants' sense of authorship. P3 explained that because CapSmith's outputs were so close to what she would write herself, they were harder to reject. With the prompt-based chat, dismissing suggestions felt easier.

8.3.2 Generation Failure Burden. Chat-style prompting often leads users to internalize failures as their own [72]. When the prompt-based chat produced poor ATC suggestions, participants remarked, "I guess it's my fault, I needed to add more context" (P1) and "Part of it is user error… not knowing how to prompt… I can only blame the system so much if I'm feeding it garbage" (P9). This framing is especially problematic for creativity-related tasks, where intent is tacit and outcomes are open-ended. Rather than shifting responsibility to the user to "prompt better," chat interfaces should instead shift accountability toward the system to surface and operationalize the information that was used to generate variations and allow users to easily correct misinterpretations. As CapSmith explored, this may involve task-specific scaffolding to help the user uncover what the agent knows or doesn't know to establish common ground.

8.3.3 Prompt-based chat use cases. While the prompt-based chat interaction proved frustrating for ATC drafting and refinement, it remains useful in several other situations.

Speed-first, lower-stakes video editing. At times, participants were willing to trade control for speed and authenticity for virality. "If I truly wanted to rush something... and if my bar for quality was a lot lower, I can just prompt and drop it in" (P9). P3, who creates Instagram reels to market her business, explained, "I think it's making better [ATCs] for me so I don't see [loss of authorship] as a sacrifice. I think everything I write on Instagram or whatever is first run through GPT." For more intentional and personal edits, however, creators prefer variation tools that preserve control.

Information retrieval and factual augmentation. Automation workflows did not suit creators like P2 and P6, who avoided using LLMs to author ATCs to preserve authenticity. However, they still found the chat valuable for augmenting their work with factual information. P6 explained, "I would just use it as a video-intelligent search engine." In practice, she and other participants used the prompt-based chat to look up item names (P6, P7) and locations (P4, P6).

9 Limitations and Future Work

There are several opportunities for improving and extending the system to better support creators.

9.1 Transcript-based Asset Addition

CapSmith pushes the transcript into an authoring surface, but precise temporal synchronization remains an open challenge. The transcript inherently determines the granularity at which users can place or adjust ATCs: alignment is constrained to the resolution of diarization segments, words, or silence markers. While this was sufficient for many placements, participants still switched to the timeline for frame-level refinement. Future work could explore adaptive temporal scaling in the transcript panel to support finer-grained placement directly within the transcript while preserving the benefits of text-based video representations.

9.2 Stylization

Stylization also plays a significant role in how ATCs are interpreted. In our formative study, professional editors described predefined font families that encode speaker identity or function (e.g., comic relief vs. narration, anger vs. laughter). Such stylization is largely absent from social media ATCs, likely due to the additional effort required. Future work may explore how inferred intent and context, as developed here, could support automatic or assisted stylization of ATCs.

10 Conclusion

Additive Text Captions (ATCs) are a uniquely expressive yet understudied medium for enhancing video narratives. We introduce CapSmith, a system that supports ATC refinement by generating variations from a user-initiated draft, with low prompting burden. Central to this approach is a domain-specific taxonomy of intent derived from our formative study that positions intentions as a shared vocabulary between the creator and the system. CapSmith uses this vocabulary to surface and operationalize the inferred intent and context of the draft. Our findings also show that a script is an effective narrative representation for contextualizing and authoring text overlays. Together, these results show that variation-from-example workflows can support ATC refinement by systematically inferring and surfacing intent and context, reducing prompting burden while preserving authorship.

Acknowledgments

The authors thank the user study participants for their insights, and the reviewers for their feedback.

References

[1] Adobe. [n. d.]. Text-Based Editing in Premiere Pro. https://helpx.adobe.com/ie/premiere-pro/using/text-based-editing.html

[2] Maneesh Agrawala, Wilmot Li, and Floraine Berthouzoz. 2011. Design principles for visual communication. *Commun. ACM* 54, 4 (April 2011), 60–69. doi:10.1145/1924421.1924439

[3] Oliver Alonzo, Hijung Valentina Shin, and Dingzeyu Li. 2022. Beyond Subtitles: Captioning and Visualizing Non-speech Sounds to Improve Accessibility of User-Generated Videos. In *Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, Athens Greece, 1–12. doi:10.1145/3517428.3544808

[4] Barrett R Anderson, Jash Hemant Shah, and Max Kreminski. 2024. Homogenization Effects of Large Language Models on Human Creative Ideation. In *Proceedings of the 16th Conference on Creativity & Cognition (C&C '24)*. Association for Computing Machinery, New York, NY, USA, 413–425. doi:10.1145/3635636.3656204

[5] Avid. [n. d.]. Speed up script-based editing with AI. https://www.avid.com/products/media-composer-scriptsync-option

[6] Hillel Aviezer, Yaacov Trope, and Alexander Todorov. 2012. Body Cues, Not Facial Expressions, Discriminate Between Intense Positive and Negative Emotions. *Science* 338, 6111 (Nov. 2012), 1225–1229. doi:10.1126/science.1224313

[7] John Bateman. 2014. *Text and Image: A Critical Introduction to the Visual/Verbal Divide*. Routledge, London. doi:10.4324/9781315773971

[8] Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala. 2012. Tools for placing cuts and transitions in interview video. *ACM Transactions on Graphics* 31, 4 (Aug. 2012), 1–8. doi:10.1145/2185520.2185563 Publisher: Association for Computing Machinery (ACM).

[9] Stephen Brade, Bryan Wang, Mauricio Sousa, Sageev Oore, and Tovi Grossman. 2023. Promptify: Text-to-image generation through interactive prompt exploration with large language models. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–14.

[10] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 2 (2006), 77–101.

[11] Juan Casares, A Chris Long, Brad A Myers, Rishi Bhatnagar, Scott M Stevens, Laura Dabbish, Dan Yocum, and Albert Corbett. 2002. Simplifying video editing using metadata. In *Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques*. 157–166.

[12] Tuhin Chakrabarty, Philippe Laban, Divyansh Agarwal, Smaranda Muresan, and Chien-Sheng Wu. 2024. Art or artifice? large language models and the false promise of creativity. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–34.

[13] Peggy Chi, Tao Dong, Christian Frueh, Brian Colonna, Vivek Kwatra, and Irfan Essa. 2022. Synthesis-Assisted Video Prototyping From a Document. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*. ACM, Bend OR USA, 1–10. doi:10.1145/3526113.3545676

[14] Yean Fun Chow. 2024. Impact caption translation on a streaming media platform: the case of a Chinese reality show. *Perspectives* 32, 1 (2024), 154–173.

[15] Neil Cohn. 2013. Beyond speech balloons and thought bubbles: The integration of text and image. *Semiotica* 2013, 197 (Oct. 2013), 35–63. doi:10.1515/sem-2013-0079 Publisher: De Gruyter Mouton.

[16] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. doi:10.48550/arXiv.2507.06261 arXiv:2507.06261 [cs].

[17] Hai Dang, Frederik Brudy, George Fitzmaurice, and Fraser Anderson. 2023. Worldsmith: Iterative and expressive prompting for world building with a generative ai. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–17.

[18] Caluã De Lacerda Pataca, Saad Hassan, Nathan Tinker, Roshan Lalintha Peiris, and Matt Huenerfauth. 2024. Caption Royale: Exploring the Design Space of Affective Captions from the Perspective of Deaf and Hard-of-Hearing Individuals. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu HI USA, 1–17. doi:10.1145/3613904.3642258

[19] Caluã De Lacerda Pataca, Matthew Watkins, Roshan Peiris, Sooyeon Lee, and Matt Huenerfauth. 2023. Visualization of Speech Prosody and Emotion in Captions: Accessibility for Deaf and Hard-of-Hearing Users. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, Hamburg Germany, 1–15. doi:10.1145/3544548.3581511

[20] Descript Made Easy. 2024. Adding and Editing Text in Descript. https://www.youtube.com/watch?v=Wj4jqr-06Y0

[21] Xianzhe Fan, Zihan Wu, Chun Yu, Fenggui Rao, Weinan Shi, and Teng Tu. 2024. ContextCam: Bridging context awareness with creative human-AI image cocreation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–17.

[22] Ohad Fried, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, Christian Theobalt, and Maneesh Agrawala. 2019. Text-based editing of talking-head video. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–14.

[23] Frederic Gmeiner, Nicolai Marquardt, Michael Bentley, Hugo Romat, Michel Pahud, David Brown, Asta Roseway, Nikolas Martelaro, Kenneth Holstein, Ken Hinckley, et al. 2025. Intent tagging: Exploring micro-prompting interactions for supporting granular human-GenAI co-creation workflows. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–31.

[24] Srishti Goel, Julian Jara-Ettinger, Desmond C. Ong, and Maria Gendron. 2024. Face and context integration in emotion inference is limited and variable across categories and individuals. *Nature Communications* 15, 1 (March 2024), 2443. doi:10.1038/s41467-024-46670-5 Publisher: Nature Publishing Group.

[25] Lena Hegemann, Xinyi Wen, Michael A Hedderich, Tarmo Nurmi, and Hariharan Subramonyam. 2026. ToMigo: Interpretable Design Concept Graphs for Aligning Generative AI with Creative Intent. *arXiv preprint arXiv:2602.05825* (2026).

[26] Bernd Huber, Hijung Valentina Shin, Bryan Russell, Oliver Wang, and Gautham J. Mysore. 2019. B-Script: Transcript-based B-roll Video Editing with Recommendations. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland Uk, 1–11. doi:10.1145/3290605.3300311

[27] Mina Huh, Ding Li, Kim Pimmel, Hijung Valentina Shin, Amy Pavel, and Mira Dontcheva. 2025. VideoDiff: Human-AI Video Co-Creation with Alternatives. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, 1–19. doi:10.1145/3706598.3713417

[28] Moon Joon-hyun. 2025. Why Korean variety shows are so text-heavy – and why it works. https://www.koreaherald.com/article/10410946 Section: The Korea Herald.

[29] Dae Hyun Kim, Vidya Setlur, and Maneesh Agrawala. 2021. Towards Understanding How Readers Integrate Charts and Captions: A Case Study with Line Charts. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, Yokohama Japan, 1–11. doi:10.1145/3411764.3445443

[30] JooYeong Kim, SooYeon Ahn, and Jin-Hyuk Hong. 2023. Visible Nuances: A Caption System to Visualize Paralinguistic Speech Cues for Deaf and Hard-of-Hearing Individuals. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, Hamburg Germany, 1–15. doi:10.1145/3544548.3581130

[31] JooYeong Kim and Jin-Hyuk Hong. 2025. OnomaCap: Making Non-speech Sound Captions Accessible and Enjoyable through Onomatopoeic Sound Representation. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. ACM, Yokohama Japan, 1–22. doi:10.1145/3706598.3713911

[32] Michelle S Lam, Omar Shaikh, Hallie Xu, Alice Guo, Diyi Yang, Jeffrey Heer, James A Landay, and Michael S Bernstein. 2026. Just-In-Time Objectives: A General Approach for Specialized AI Interactions. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*. 1–26.

[33] Dawon Lee, Jongwoo Choi, and Junyong Noh. 2025. OptiSub: Optimizing Video Subtitle Presentation for Varied Display and Font Sizes via Speech Pause-Driven Chunking. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. ACM, Yokohama Japan, 1–12. doi:10.1145/3706598.3714199

[34] Daniel G. Lee, Deborah I. Fels, and John Patrick Udo. 2007. Emotive captioning. *Computers in Entertainment* 5, 2 (April 2007), 11. doi:10.1145/1279540.1279551

[35] David Chuan-En Lin, Hyeonsu B Kang, Nikolas Martelaro, Aniket Kittur, Yan-Ying Chen, and Matthew K Hong. 2025. Inkspire: Supporting design exploration with generative ai through analogical sketching. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–18.

[36] Scott D. Lipscomb and Roger A. Kendall. 1994. Perceptual judgement of the relationship between musical and visual components in film. *Psychomusicology: A Journal of Research in Music Cognition* 13, 1-2 (1994), 60–98. doi:10.1037/h0094101 Place: US Publisher: The Florida State University.

[37] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys* 55, 9 (2023), 1–35.

[38] Jiaju Ma, Anyi Rao, Li-Yi Wei, Rubaiat Habib Kazi, Hijung Valentina Shin, and Maneesh Agrawala. 2023. Automated Conversion of Music Videos into Lyric Videos. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–11. doi:10.1145/3586183.3606757 arXiv:2308.14922 [cs].

[39] Emma J McDonnell, Tessa Eagle, Pitch Sinlapanuntakul, Soo Hyun Moon, Kathryn E. Ringland, Jon E. Froehlich, and Leah Findlater. 2024. "Caption It in an Accessible Way That Is Also Enjoyable": Characterizing User-Driven Captioning Practices on TikTok. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu HI USA, 1–16. doi:10.1145/3613904.3642177

[40] Makoto Nakamura, Ross Buck, and David A Kenny. [n. d.]. Relative Contributions of Expressive Behavior and Contextual Information to the Judgment of the Emotional State of Another. ([n. d.]).

[41] Don Norman. 2013. *The design of everyday things: Revised and expanded edition*. Basic books.

[42] Kotaro Oomori, Akihisa Shitara, Tatsuya Minagawa, Sayan Sarcar, and Yoichi Ochiai. 2020. A Preliminary Study on Understanding Voice-only Online Meetings Using Emoji-based Captioning for Deaf or Hard of Hearing Users. In *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, Virtual Event Greece, 1–4. doi:10.1145/3373625.3418032

[43] Amy Pavel, Dan B. Goldman, Björn Hartmann, and Maneesh Agrawala. 2015. SceneSkim: Searching and Browsing Movies Using Synchronized Captions, Scripts and Plot Summaries. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, Charlotte NC USA, 181–190. doi:10.1145/2807442.2807502

[44] Amy Pavel, Dan B. Goldman, Björn Hartmann, and Maneesh Agrawala. 2016. VidCrit: Video-based Asynchronous Video Review. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, Tokyo Japan, 517–528. doi:10.1145/2984511.2984552

[45] Amy Pavel, Colorado Reed, Björn Hartmann, and Maneesh Agrawala. 2014. Video digests: a browsable, skimmable format for informational lecture videos. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, Honolulu Hawaii USA, 573–582. doi:10.1145/2642918.2647400

[46] Amy Pavel, Gabriel Reyes, and Jeffrey P. Bigham. 2020. Rescribe: Authoring and Automatically Editing Audio Descriptions. doi:10.48550/arXiv.2010.03667 arXiv:2010.03667 [cs].

[47] Xiaohan Peng, Janin Koch, and Wendy E Mackay. 2024. Designprompt: Using multimodal interaction for design exploration with generative ai. In *Proceedings of the 2024 ACM Designing Interactive Systems Conference*. 804–818.

[48] Chen Qu, Liu Yang, W. Bruce Croft, Yongfeng Zhang, Johanne R. Trippas, and Minghui Qiu. 2019. User Intent Prediction in Information-seeking Conversations. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval*. ACM, Glasgow Scotland UK, 25–33. doi:10.1145/3295750.3298924

[49] Anyi Rao, Jean-Peïc Chou, and Maneesh Agrawala. 2024. Scriptviz: A visualization tool to aid scriptwriting based on a large movie database. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–13.

[50] Ryoko Sasamoto. 2014. Impact caption as a highlighting device: Attempts at viewer manipulation on TV. *Discourse, Context & Media* 6 (2014), 1–10.

[51] Scott McCloud. 1993. *Understanding Comics: The Invisible Art*. http://archive.org/details/understanding-comics

[52] Vidya Setlur, Sarah E. Battersby, Melanie Tory, Rich Gossweiler, and Angel X. Chang. 2016. Eviza: A Natural Language Interface for Visual Analysis. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, Tokyo Japan, 365–377. doi:10.1145/2984511.2984588

[53] Omar Shaikh, Kristina Gligorić, Ashna Khetan, Matthias Gerstgrasser, Diyi Yang, and Dan Jurafsky. 2024. Grounding gaps in language model generations. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. 6279–6296.

[54] Omar Shaikh, Shardul Sapkota, Shan Rizvi, Eric Horvitz, Joon Sung Park, Diyi Yang, and Michael S. Bernstein. 2025. Creating General User Models from Computer Use. doi:10.48550/arXiv.2505.10831 arXiv:2505.10831 [cs].

[55] Jocelyn J Shen, Kathryn Jin, Ann Zhang, Cynthia Breazeal, and Hae Won Park. 2023. Affective Typography: The Effect of AI-Driven Font Design on Empathetic Story Reading. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, Hamburg Germany, 1–7. doi:10.1145/3544549.3585625

[56] Ben Shneiderman. 2007. Creativity support tools: accelerating discovery and innovation. *Commun. ACM* 50, 12 (Dec. 2007), 20–32. doi:10.1145/1323688.1323689

[57] Hari Subramonyam, Roy Pea, Christopher Pondoc, Maneesh Agrawala, and Colleen Seifert. 2024. Bridging the Gulf of Envisioning: Cognitive Challenges in Prompt Based Interactions with LLMs. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu HI USA, 1–19. doi:10.1145/3613904.3642754

[58] Yiwen Sun, Jason Leigh, Andrew Johnson, and Sangyoon Lee. 2010. Articulate: A Semi-automated Model for Translating Natural Language Queries into Meaningful Visualizations. In *Smart Graphics*, Robyn Taylor, Pierre Boulanger, Antonio Krüger, and Patrick Olivier (Eds.). Springer, Berlin, Heidelberg, 184–195. doi:10.1007/978-3-642-13544-6_18

[59] Adobe Communications Team. [n. d.]. Making your videos as accessible as possible | Adobe Video. https://blog.adobe.com/en/publish/2021/12/10/video-accessibility-guide-for-content-creators-and-viewers

[60] Michael Terry and Elizabeth D Mynatt. 2002. Recognizing creative needs in user interface design. In *Proceedings of the 4th Conference on Creativity & Cognition*. 38–44.

[61] Anh Truong, Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala. 2016. QuickCut: An Interactive Tool for Editing Narrated Video. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, Tokyo Japan, 497–507. doi:10.1145/2984511.2984569

[62] Priyan Vaithilingam, Ian Arawjo, and Elena L. Glassman. 2024. Imagining a Future of Designing with AI: Dynamic Grounding, Constructive Negotiation, and Sustainable Motivation. In *Designing Interactive Systems Conference*. ACM, Copenhagen Denmark, 289–300. doi:10.1145/3643834.3661525

[63] Samangi Wadinambiarachchi, Ryan M. Kelly, Saumya Pareek, Qiushi Zhou, and Eduardo Velloso. 2024. The Effects of Generative AI on Design Fixation and Divergent Thinking. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3613904.3642919

[64] Han Wang and Roy Ka-Wei Lee. 2024. MemeCraft: Contextual and Stance-Driven Multimodal Meme Generation. In *Proceedings of the ACM Web Conference 2024*. ACM, Singapore Singapore, 4642–4652. doi:10.1145/3589334.3648151

[65] Sitong Wang, Zheng Ning, Anh Truong, Mira Dontcheva, Dingzeyu Li, and Lydia B Chilton. 2024. PodReels: Human-AI Co-Creation of Video Podcast Teasers. In *Proceedings of the 2024 ACM Designing Interactive Systems Conference (DIS '24)*. Association for Computing Machinery, New York, NY, USA, 958–974. doi:10.1145/3643834.3661591

[66] Zhijie Wang, Yuheng Huang, Da Song, Lei Ma, and Tianyi Zhang. 2024. Promptcharm: Text-to-image generation through multi-modal prompting and refinement. In *Proceedings of the 2024 CHI conference on human factors in computing systems*. 1–21.

[67] Steve Whittaker and Brian Amento. 2004. Semantic speech editing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, Vienna

Austria, 527–534. doi:10.1145/985692.985759

[68] Qunfang Wu, Yisi Sang, and Yun Huang. 2019. Danmaku: A new paradigm of social interaction via online videos. *ACM Transactions on Social Computing* 2, 2 (2019), 1–24.

[69] Yaxing Yao, Jennifer Bort, and Yun Huang. 2017. Understanding Danmaku's potential in online video learning. In *Proceedings of the 2017 CHI conference extended abstracts on human factors in computing systems*. 3034–3040.

[70] Ann Yuan, Andy Coenen, Emily Reif, and Daphne Ippolito. 2022. Wordcraft: Story Writing With Large Language Models. In *27th International Conference on Intelligent User Interfaces*. ACM, Helsinki Finland, 841–852. doi:10.1145/3490099.3511105

[71] Francisco Yus. 2019. Multimodality in Memes: A Cyberpragmatic Approach. In *Analyzing Digital Discourse: New Insights and Future Directions*, Patricia Bou-Franch and Pilar Garcés-Conejos Blitvich (Eds.). Springer International Publishing, Cham, 105–131. doi:10.1007/978-3-319-92663-6_4

[72] J.D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny Can't Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, Hamburg Germany, 1–21. doi:10.1145/3544548.3581388

A Appendix

A.1 Post-Study Survey Questions

Design Goals Questions. On a scale from 1 (strongly disagree) to 7 (strongly agree):

- **Understanding Context:** The system understood the context of the text overlay
- **Understanding Intentions:** It was easy to communicate the intentions of the text overlay
- **Improvement of ATC:** The system helped me improve my text overlay quality
- **Authorship:** The outcome feels like the result of my own creative process

A.2 Anchor Generation Pipeline

The pipeline is demonstrated in Figure 8.

A.3 Prompts

A.3.1 Script Generation.

```
You are a professional script supervisor analyzing a video
together with its transcript. Create screenplay-style scene
segments that reflect natural breaks in story flow. For each
scene, provide a brief description, key visible actions, who
is present, and any useful parentheticals for line delivery.
Build a global character catalog (stable IDs, concise names,
one-sentence personality, short physical descriptors). Map
transcript lines to speaking characters. Detect notable
non-verbal sound events (e.g., laughs, gasps, door slams).
Provide a coarse shot breakdown (start/end, who is visible,
brief visual description, salient objects, character
actions). Also infer lightweight target_audience labels and
style_keywords.
```

Inputs:

- `VIDEO` (with audio)
- `TRANSCRIPT`: `$transcriptData`

Where:

- `{transcriptData}` = array of {*index, start_s, end_s, text, speaker*}

A.3.2 ATC Subject Inference.

```
Classify the anchor subject of a user-initiated caption in
its current frame. Using the composed context (anchor,
moment, story), decide whether the caption refers to a
Character, Object, or Narrative element. Justify the choice
with one concise sentence that cites both visual placement
and story context. If Character include the matched catalog
name; if Object, include a brief object label. Choose
exactly one label: Character | Object | Narrative. Return
the label and a one-sentence rationale grounded in the frame
and context.
```

Inputs:

- `FRAME`: `$frame`
- `CAPTION`: `$originalCaption`
- `COMPOSED CONTEXT STRING`: `$contextString` (anchor metadata, moment context, global story)

Where:

- `{originalCaption}` = the user's draft caption text for this moment.
- `{contextString}` = concatenation of:
  - anchor metadata: origin (script|visual). If origin is script, anchor metadata includes highlighted transcript indices (if any), current speaker (if any), dialogue parentheticals (if any) and non-verbal sound events (if any). If origin is visual, anchor metadata includes x,y coordinates of the placed ATC as well as a video frame captured at the ATC's start time.
  - moment context: scene ID/timecode, visible character IDs, nearby transcript lines.
  - story context: character catalog (stable IDs, names, one-line personalities, brief physical descriptors); brief scene overview.

A.3.3 Intention Inference.

```
Classify the creative intention behind a user-initiated
caption in its current moment. Use the composed context
(anchor, moment, story) to determine the intent behind the
caption's presence in this scene. Choose exactly one label:
Amplify | Disambiguate | Augment | Redirect. Return the
label and one concise reasoning sentence that includes the
keyword itself (e.g., "This caption helps clarify..."").
Amplify: heighten or intensify the existing
beat/emotion/action, Disambiguate: resolve potential misread
or ambiguity, Augment: surface a subtle/unseen detail the
audience likely misses, Redirect: reframe or counter the
default takeaway with editorial voice
```

Inputs:

- `CAPTION`: `$captionText`
- `TIMESTAMP/DURATION`: `$timestamps`, `$durations`
- `COMPOSED CONTEXT STRING`: `$contextString` (anchor metadata, moment context, global story)

Where:

- `{captionText}` = the user's draft caption text for this moment.
- `{timestamp}` = start time of the caption in seconds.
- `{duration}` = duration of the caption in seconds.
- `{contextString}` = concatenation of:
  - anchor metadata: origin (script|visual). If origin is script, anchor metadata includes highlighted transcript indices (if any), current speaker (if any), dialogue parentheticals (if any) and non-verbal sound events (if any). If origin is visual, anchor metadata includes x,y coordinates of the placed ATC as well as a video frame captured at the ATC's start time.
  - moment context: scene ID/timecode, visible character IDs, nearby transcript lines.
  - story context: character catalog (stable IDs, names, one-line personalities, brief physical descriptors); brief scene overview.

A.3.4 Variation Generation.

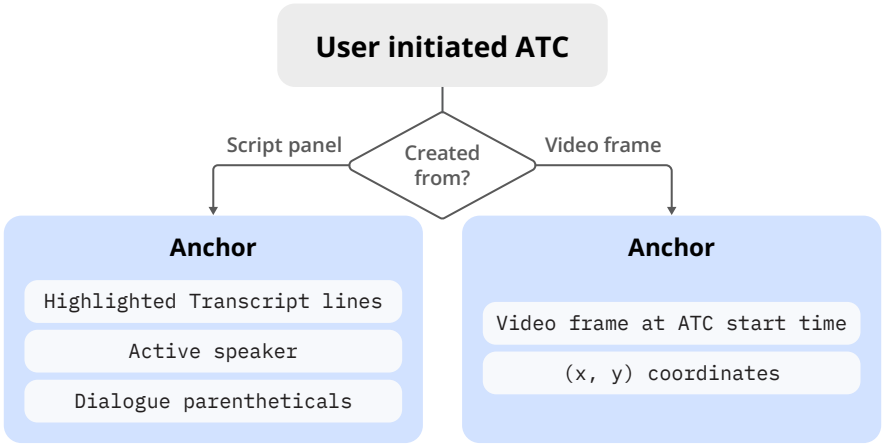

Figure 8: Overview of the anchor generation pipeline. An ATC created from either the script panel or timeline is bound to an anchor. Anchors created from the script panel capture transcript and audio metadata, while those created from the video player capture frame timing and x,y coordinates of the overlay with respect to the frame.

`You are a creative text overlay writer making an engaging video through text overlays. Your task is to provide alternatives for the given caption while considering the user's creative intention, and staying true to the subject of the ATC. Tailor phrasing to the TARGET AUDIENCE and STYLE KEYWORDS if present.`

Inputs:

- `ORIGINAL CAPTION: $originalCaption`
- `COMPOSED CONTEXT STRING: $contextString` (anchor metadata, moment context, global story)

Where:

- `{originalCaption}` = the user's draft caption text for this moment.
- `{contextString}` = concatenation of:
  - subject: inferred or user-specified subject
  - intention: inferred or user-specified intention
  - anchor metadata: origin (script|visual). If origin is script, anchor metadata includes highlighted transcript indices (if any), current speaker (if any), dialogue parentheticals (if any) and non-verbal sound events (if any). If origin is visual, anchor metadata includes x,y coordinates of the placed ATC as well as a video frame captured at the ATC's start time.
  - moment context: scene ID/timecode, visible character IDs, nearby transcript lines.
  - story context: character catalog (stable IDs, names, one-line personalities, brief physical descriptors); brief scene overview.
  - target audience (if any) and style keywords (if any).
  - optional creative intention provided by the user.

Generation Criteria:

- Generate exactly 16 unique alternatives.
- Adapt tone and phrasing to suit target audience and style keywords
- Respect emotional tone, dialogue parentheticals, and visual placement.

A.3.5 Prompt-based Chat.

`You are a helpful AI assistant. I have uploaded a video:`

Inputs:

- `VIDEO`: the clip currently on the user's timeline
- `USER MESSAGE`: free-form prompt (caption ideas, rewrites, etc.)
- `Any ATC reference through @id`: Caption text, Start timestamp (seconds), Duration (seconds)