

Video Thumbnail

# Blog Post

## Overview

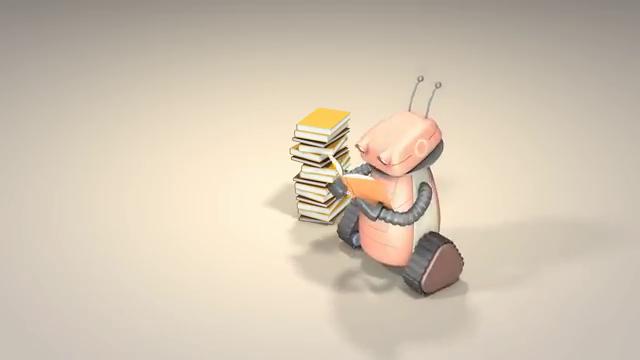
This transcript details a talk given at Google for Machine Learning Singapore, focusing on the speaker's work with Large Language Model (LLM) applications. The core argument revolves around two key aspects driving the evolution of LLM apps: personalization and curation. The speaker views these as crucial factors in making LLM technology more accessible and useful to a wider audience.

The speaker believes we are currently in approximately version seven of the LLM stack, tracing the progression from GPT-2 apps in 2019 to current models like OpenAI's GPT-4, Gemini, and Llama 3. This evolution highlights the rapid advancements in the field. Personalization, as explained, aims to tailor the LLM experience to individual users. This goes beyond simple user settings; it involves dynamically adapting the interaction and results based on the user's learning level, preferences, and needs. The speaker uses the example of personalized learning levels in educational applications, where the system adjusts to the user's current understanding rather than a rigid, pre-defined structure.

Curation, the second key aspect, addresses the overwhelming amount of information available today. The speaker argues that LLMs can act as powerful filters, sifting through vast amounts of data to provide users with precisely the information they need, when they need it. This filtering function is vital in combating information overload.

To demonstrate these concepts, the speaker describes a personal project: an app designed to create high-quality notes from YouTube videos and other content. This app addresses a common shortcoming of existing summarization tools, which often produce overly concise summaries lacking detail. Instead, the app aims to generate a more comprehensive set of notes, segmented by topic, providing a richer understanding of the source material. The speaker uses the example of notes generated from a Stanford overview video on Transformers, showcasing the app's ability to break down the video into different sections and create detailed notes for each. The project highlights the potential of LLMs to not just summarize but to organize and present information in a more user-friendly and insightful manner, effectively demonstrating the principles of personalization and curation in action.

## [00:02:00 - 00:08:00] Leveraging Many-Shot In-Context Learning with Proprietary Models



Screenshot at 00:02:00

The landscape of Large Language Models (LLMs) is rapidly evolving, and the focus is shifting beyond simply chasing the biggest, most powerful model. While giants like GPT-4 and Llama 3 dominate headlines, a compelling argument emerges for the strategic use of smaller, proprietary LLMs. This section explores a powerful technique: many-shot in-context learning, which significantly enhances the capabilities of these often-overlooked models.

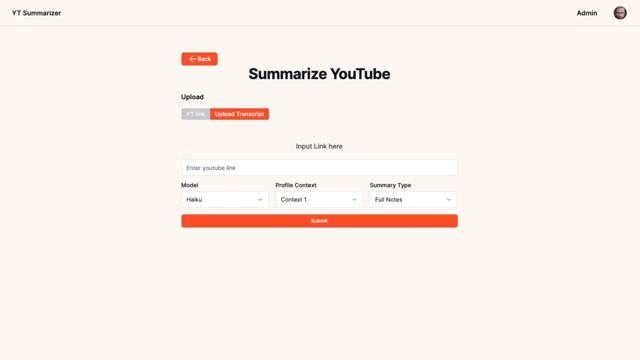
The traditional approach to few-shot learning involves providing an LLM with a handful of examples (typically 3-8) to guide its behavior on a specific task. However, a recent Google DeepMind paper, and the experience shared in this talk, highlights the remarkable benefits of dramatically increasing the number of examples provided – a technique termed "many-shot" learning. Instead of a meager few examples, proprietary models can effectively utilize 20, 30, even 50 examples to achieve significantly improved performance. The DeepMind research even demonstrated impressive results with hundreds, even thousands, of examples.

This increase in exemplars acts as a powerful substitute for full model fine-tuning. By providing the LLM with a substantially larger dataset of examples, we effectively guide its behavior and tailor its responses to the desired outcome. This is particularly advantageous with proprietary models, which may not be as readily adaptable through traditional fine-tuning methods due to resource constraints or organizational policies.

The benefits of this approach are numerous:

In essence, many-shot in-context learning empowers developers to extract exceptional value from smaller, proprietary LLMs. By strategically employing this technique, we can unlock the potential of these models, achieving results comparable to larger models, but with enhanced cost-effectiveness and agility. This represents a significant shift in how we approach LLM development and deployment, allowing for more tailored and efficient solutions across a wider range of applications.

## [00:05:05 - 00:07:08] Building a Personalized Note-Taking App for YouTube and Other Content



Screenshot at 00:05:05

The previous section explored the power of many-shot in-context learning with proprietary LLMs. Now, let's shift gears and dive into a practical application showcasing the evolving landscape of LLMs: a personalized note-taking app. This project highlights two crucial trends driving the next generation of LLM applications: personalization and curation.

The current state of LLMs is far beyond the early days of GPT-2. We've progressed through several iterations, moving from a focus on single, large, general-purpose models to a more diverse ecosystem. While giants like GPT-4, Gemini, and Llama 3 continue to push boundaries, smaller, proprietary models are proving increasingly valuable. This isn't about simply having the biggest model; it's about strategic application and intelligent use. My app embodies this shift.

My goal was to create an application that could generate high-quality notes from YouTube videos and other online content, going beyond the limitations of existing summarization tools. Many summarizers produce overly concise summaries that lack detail. My app addresses this by generating comprehensive notes, segmented by topic and timestamped for easy reference back to the source material.

The key differentiators are personalization and curation. Personalization allows users to tailor the notes to their specific needs and context. Are you an AI researcher diving deep into a technical paper? Or a new learner just trying to grasp the basics? The app lets you define your profile, adjusting the note generation process accordingly. This isn't just about user settings; it's about dynamically adapting the interaction and the output based on the user's understanding and requirements. For example, notes generated from a Stanford overview video on Transformers would be drastically different for an AI researcher versus a complete novice. The AI researcher might receive notes focusing on architectural details and innovative techniques, while the novice might receive a simpler explanation focusing on core concepts.

Curation is equally important. The app acts as a powerful filter, sifting through vast amounts of information to provide users with precisely the information they need, and only that information. The sheer volume of data available today necessitates intelligent filtering, and LLMs are uniquely positioned to perform this crucial function, combating information overload. Instead of a wall of text, users receive well-organized, detailed notes that focus on their specific area of interest.

The app allows for multiple profiles, each with its own set of preferences. This makes it adaptable for diverse learning styles and research needs. Ultimately, this personalized curation approach ensures that the most relevant information is delivered, effectively filtering out unnecessary details and focusing on the user's specified interests. This represents a significant step towards making LLM technology more accessible and useful to a broader audience. The project is a testament to the potential of LLMs not just to summarize, but to organize and present information in a far more user-friendly and insightful manner.

## [00:05:07 - 00:07:08] The Changing Landscape of LLM Models: From Best Quality to Proprietary and Specialized Models



Screenshot at 00:05:07

The previous section detailed the effectiveness of many-shot in-context learning with proprietary LLMs. Building on that, this section delves into the broader shift in the LLM landscape, moving beyond a singular focus on "best quality" models. While behemoths like GPT-4, Gemini, and Llama 3 continue to dominate benchmarks, a crucial evolution is underway: the rise of smaller, proprietary LLMs and the implications for model selection.

Initially, the LLM field largely centered around a competition for the single "best" model. Organizations like Google and OpenAI released foundational models, with a primary focus on achieving top performance across various benchmarks. These models, while impressive, often came with significant resource requirements for deployment and fine-tuning.

However, the landscape has dramatically changed. While the major players remain, a plethora of new models have emerged, forcing a reassessment of how we choose and utilize LLMs. This includes the introduction of numerous smaller, proprietary models. These models, while still significantly larger than most open-source alternatives, offer compelling advantages:

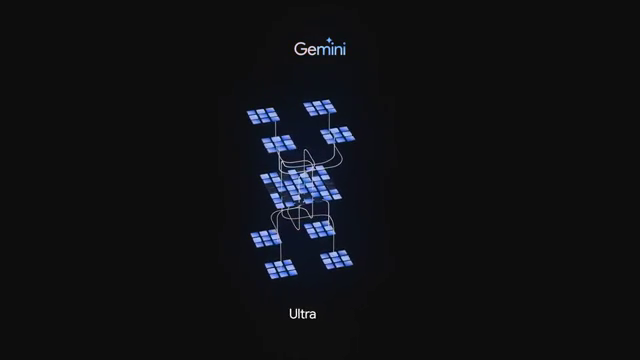
Speed and Cost-Effectiveness: Smaller models generally process information faster and at a lower cost than their larger counterparts. This makes them significantly more accessible to developers with limited computational resources.

Task-Specific Suitability: While not always outperforming the largest models on all tasks, smaller proprietary models can be highly effective for specific applications, making them a more efficient choice for targeted needs.

This diversification introduces a new layer of complexity in model selection. Simply choosing the model with the highest benchmark scores is no longer sufficient. Developers now need to consider a more nuanced set of factors, including:

Furthermore, a significant challenge emerges from the "middle ground" of models. These are models that are too large for efficient deployment but not large enough to compete with top-tier models on performance. This creates a complex decision-making process for developers, requiring careful consideration of the trade-offs between various factors. The era of simply seeking the "best" LLM is over; now, strategic selection based on specific needs and resource constraints is paramount. This shift highlights the increasing maturity of the LLM field, moving towards a more diverse and accessible ecosystem.

## [00:08:00 - 00:12:00] Modern Summarization Techniques: Beyond Extractive vs. Abstractive



Screenshot at 00:08:00

The previous sections explored the exciting world of personalized note-taking apps powered by LLMs and the evolving landscape of LLM models themselves. Now, let's zoom in on a crucial aspect of these applications: summarization. We're moving beyond the traditional, somewhat simplistic, dichotomy of extractive versus abstractive summarization. While those approaches were foundational, modern summarization needs a more nuanced approach.

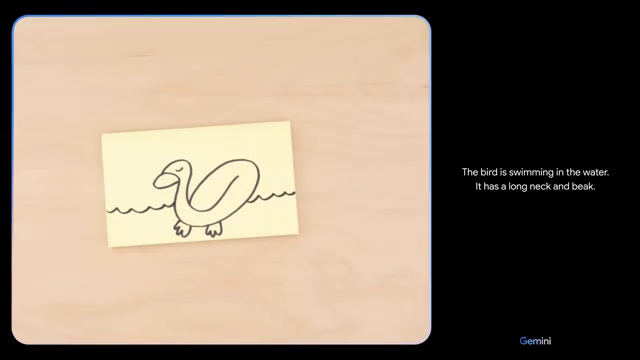
The speaker argues that the future of summarization lies in personalization. This isn't just about generating a summary; it's about tailoring that summary to the individual user's needs, preferences, and existing knowledge. Imagine receiving a summary of a complex scientific paper: a seasoned researcher would require a very different summary than a student just starting to learn the field. Personalization allows the LLM to dynamically adjust the complexity, level of detail, and focus of the summary to match the user's profile.

Another key advancement is aspect-based summarization. Instead of a generic overview, this technique allows users to prioritize specific aspects of the source material. Need a summary focusing solely on the key statistics and figures? Aspect-based summarization delivers. This granular control gives users the power to extract precisely the information they need, eliminating the noise and focusing on what's most relevant to their task.

Finally, the discussion touches on the emerging and challenging field of dialogue summarization. Summarizing conversations and interviews presents unique hurdles compared to summarizing more structured content like presentations or lectures. The dynamic nature of conversations, with interruptions, tangents, and shifts in topic, requires sophisticated techniques to generate accurate and coherent summaries. This is an area ripe for future innovation.

In summary, the traditional extractive vs. abstractive framework is insufficient for the complexities of modern summarization. Personalization, aspect-based summarization, and the ongoing development of dialogue summarization represent a significant leap forward, reflecting the dynamic and rapidly evolving landscape of NLP. These advances are essential for making LLM-powered applications truly useful and accessible to a broad audience, ensuring that information is not just summarized, but intelligently presented to meet the specific needs of each individual user.

## [00:08:20 - 00:10:24] Model Families and the Limitations of Long Context Windows



Screenshot at 00:08:20

The previous section discussed the exciting possibilities opened up by the diversification of LLM models, moving beyond the simple pursuit of the single "best" model. Now, let's delve into a specific example of this diversification: the emergence of "model families," and the limitations they currently present, particularly regarding long context windows.

Google's Gemini suite serves as a prime example. This family offers a range of models, catering to diverse needs and computational resources. We have powerful models like Gemini Ultra and Pro, alongside more lightweight and mobile-friendly options like Gemini Nano. The introduction of Gemini Pro 1.5 was particularly noteworthy, boasting an impressive million-token context window – a significant leap forward. This seemingly allows for processing and understanding incredibly vast amounts of information at once.

However, this impressive capability comes with significant caveats. The reality is that Gemini Pro 1.5, despite its enormous context window, suffers from considerable speed limitations. Processing a million tokens takes several minutes to generate an answer. Furthermore, the output is severely restricted to a mere 8,000 tokens. This is a critical bottleneck.

This limitation effectively renders Gemini Pro 1.5 (and similar models with large context windows) unsuitable for many applications requiring long outputs. Imagine trying to edit a book, transcribe a lengthy audio recording, or generate comprehensive notes from a multi-hour lecture – all tasks that demand significantly more than 8,000 tokens of output. These are precisely the types of applications that were highlighted as particularly promising in the previous sections.

The implication is clear: while the potential of large context windows is undeniably exciting, they haven't yet superseded the need for Retrieval Augmented Generation (RAG). The speed and output limitations of current models mean that RAG, a technique that retrieves relevant information from external databases before feeding it to the LLM, remains a crucial and often more practical approach for handling extensive information. The ability to quickly access and process large datasets outweighs the allure of a single, massive context window when dealing with real-world applications and the constraints of time and processing power. For now, RAG continues to be a vital tool in the LLM developer's arsenal. The pursuit of efficient long-form processing remains a key challenge in the ongoing evolution of LLMs.

## [00:12:00 - 00:18:00] LLM Summarization Methods: Simple Stuffing, MapReduce, and Refinement



Screenshot at 00:12:00

The previous sections explored the exciting possibilities of personalized LLM applications and the diverse landscape of modern language models. Now, let's delve into the nitty-gritty of a core functionality within these apps: summarization. We've moved beyond the simple "extractive vs. abstractive" debate; modern summarization demands a more sophisticated approach. This section unpacks the various methods used to effectively summarize text using LLMs, highlighting their strengths and limitations.

One of the simplest approaches, "simple stuffing," involves feeding the entire document and the summarization prompt directly to the LLM in a single call. This method, however, is severely hampered by the inherent input size limitations of most LLMs. Longer documents simply won't fit.

A more scalable solution is the MapReduce approach, popularized by frameworks like LangChain. This involves breaking down the input document into smaller, manageable chunks. Each chunk is then summarized individually, and the resulting summaries are subsequently combined to create a comprehensive overview. While this method circumvents the input size limitations, it introduces a new set of challenges. The resulting summary may lack coherence if crucial information is spread across multiple chunks and the relationships between these pieces of information are not properly integrated.

To address these issues, more advanced techniques involve iterative refinement. Instead of generating a single summary in one go, this method involves multiple calls to the LLM. The process begins with an initial summary, which is then fed back into the LLM along with the original document or relevant chunks. The LLM iteratively improves the summary, adding detail, correcting inconsistencies, and prioritizing key information. This iterative process allows for a more nuanced and comprehensive understanding of the source material.

Furthermore, techniques like reranking chunks based on their information content can be incorporated to improve the quality of the final summary. By identifying and prioritizing the most important segments of the document before summarization, the LLM can generate a more focused and insightful output.

The various methods discussed highlight the complexities inherent in LLM-based summarization, especially when dealing with lengthy documents. While simple stuffing offers a quick and easy approach, it's severely limited. MapReduce offers scalability but risks sacrificing coherence. Iterative refinement and chunk reranking provide more robust solutions, but they require more computational resources and careful design. The choice of method will depend on the specific application, the length of the document, and the desired level of detail and coherence in the final summary. The development of robust and efficient summarization techniques continues to be a crucial area of research and development in the field of LLMs.

## [00:15:09 - 00:17:29] Leveraging Many-Shot In-Context Learning



Screenshot at 00:15:09

The previous sections explored the exciting advancements in LLM applications, focusing on personalization and curation. Now, let's dive into a powerful technique that significantly enhances the capabilities of even smaller, more efficient LLMs: many-shot in-context learning.

Traditionally, few-shot learning with LLMs involved providing just a handful of examples (3-8) within the prompt to guide the model's behavior. However, the speaker highlights a paradigm shift: the ability to incorporate significantly more examples – 20, 30, 50, or even hundreds – is now readily achievable. This is largely due to the decreasing cost and increasing speed of LLM processing.

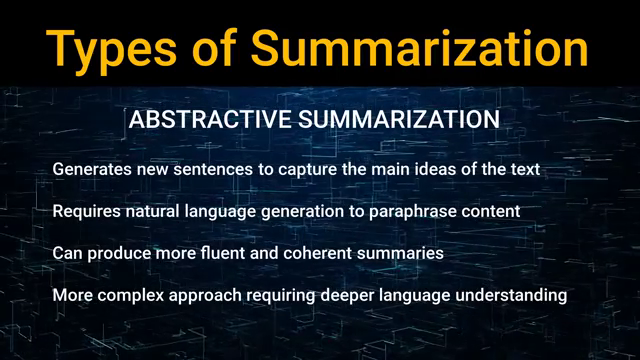
This approach, termed "many-shot in-context learning," has been validated by recent research, notably a Google DeepMind paper. This paper demonstrated substantial performance gains by increasing the number of exemplars to 50, 500, or even 2000. The results suggest that this technique can, in many cases, achieve performance comparable to the more resource-intensive method of fine-tuning.

The speaker suggests that this advantage might be even more pronounced with proprietary models like Google's Haiku, potentially exceeding the benefits seen with open-source alternatives. The reason for this is not explicitly stated in the transcript, but it likely relates to the specific architectural details and training data of proprietary models.

Why is this significant? By dramatically increasing the number of examples in the prompt, developers can effectively "steer" the model's behavior with far greater precision. This allows for a more nuanced and controlled interaction, leading to higher-quality outputs. The speaker illustrates this with the example of providing numerous examples of well-structured mathematical problems with step-by-step solutions, wrapped in XML tags for better processing. This allows the LLM to learn the desired format and generate similar, high-quality responses.

The implications are far-reaching. Many-shot in-context learning extends beyond simple mathematical problems; it applies to various tasks, including function calling, React prompting, and any situation where guiding the model towards specific decision-making processes is crucial. The more examples you provide, the more refined and accurate the model's output becomes. This technique offers a powerful and efficient way to leverage the potential of LLMs without the computational overhead of fine-tuning, making it an attractive option for developers seeking to build robust and effective applications.

## [00:18:00 - 00:21:30] A Novel Approach to Summarization: Sectioning and Targeted Summaries



Screenshot at 00:18:00

The previous sections explored various LLM summarization methods, from simple "stuffing" to the more sophisticated MapReduce and iterative refinement techniques. Each approach presents its own trade-offs between simplicity, scalability, and coherence. However, a new method is emerging that promises to significantly improve the accuracy and readability of summaries, particularly for lengthy documents: sectioning and targeted summarization.

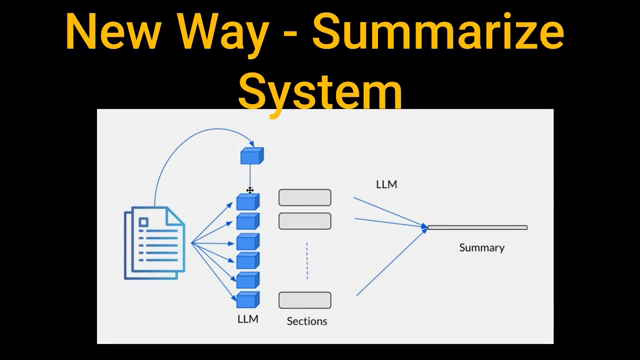
This innovative approach, employed in the speaker's personal project, leverages the lower cost of models like Claude and involves multiple LLM calls. Unlike the MapReduce approach, which divides the document into arbitrary chunks, this method begins with a crucial "sectioning" call. This initial call isn't about summarization; instead, its purpose is to identify semantic shifts within the document—points where the topic changes. The LLM analyzes the entire document to pinpoint these transitions, effectively dividing it into naturally occurring sections based on topical coherence. Think of it as identifying the logical breaks in a lecture or presentation, rather than arbitrary divisions based on word count.

Crucially, these sections are defined not by fixed word counts or arbitrary chunk sizes, but by identifying shifts in topic using the LLM's understanding of the content. The model dynamically decides where one topic ends and another begins. Timestamps, if available (as in the case of video transcriptions), are used to precisely delineate these sections.

Following the sectioning call, subsequent LLM calls focus on individual sections. Each call receives the entire document as context, providing the LLM with a comprehensive understanding of the overall narrative. However, the prompt specifically instructs the model to summarize only the relevant section. This targeted approach allows the LLM to leverage the global context while simultaneously focusing on the specific details of a particular section, resulting in more coherent and accurate summaries.

This method addresses the shortcomings of MapReduce, where the lack of global context often leads to fragmented and disjointed summaries. By providing the full document's context for each section-specific summary, this technique ensures that the relationships between different parts of the document are preserved, leading to a more holistic and insightful final output. The result is a series of detailed, topic-focused summaries that, when combined, create a comprehensive and well-organized overview of the original document. The speaker's example of generating detailed notes from a Stanford video on Transformers, broken down by section, perfectly illustrates the power of this approach. This technique represents a significant step forward in LLM-based summarization, offering a powerful combination of scalability, coherence, and accuracy.

## [00:21:30 - 00:23:02] Advanced Summarization Techniques with LLMs



Screenshot at 00:21:30

The previous sections detailed exciting advancements in LLM applications, focusing on personalization and curation. Now, let's delve into a novel approach to summarization that leverages the power of LLMs, particularly cost-effective models like Haiku, to overcome the limitations of traditional methods.

Traditional summarization techniques often fall short when dealing with extensive documents. Simple "stuffing"—feeding the entire document and prompt to the LLM—becomes impractical with lengthy texts due to context window limitations. MapReduce, while scalable by dividing the document into chunks, suffers from the loss of contextual information between those chunks, resulting in fragmented and incoherent summaries. These methods struggle to capture the nuanced relationships between different parts of a long document.

This section introduces a two-step process that significantly improves summarization accuracy and coherence: sectioning and targeted summarization. This method addresses the weaknesses of previous approaches by intelligently dividing the document into semantically coherent sections before summarizing each individually, while still retaining the overall context.

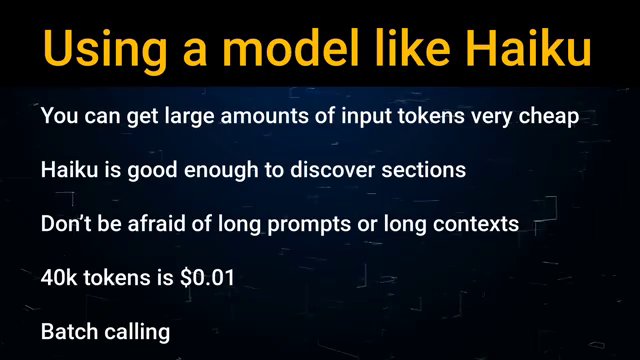
The first step involves a "sectioning" call. This isn't a summarization task; instead, it leverages the LLM's understanding to identify topic shifts within the entire document. The LLM analyzes the complete document, pinpointing where the subject matter changes. This division is not arbitrary (like a fixed word count) but is dynamically determined by the model based on semantic coherence. Timestamps, if available (as in transcribed videos), are invaluable in precisely defining these sections.

The second step involves multiple targeted summarization calls. For each section identified in step one, the entire document is passed to the LLM along with a prompt specifically instructing it to summarize only that particular section. This approach ensures the LLM has the full context of the entire document while focusing its summarization efforts on a specific, semantically coherent section. This avoids the fragmented summaries often produced by MapReduce.

The use of a cost-effective model like Haiku is key. While a larger model might be used for the initial sectioning call for improved accuracy, Haiku's efficiency allows for processing the entire document multiple times (once for sectioning, then once per section for summarization) without incurring excessive costs. Even with a 30,000-token YouTube video transcript, this approach remains cost-effective, typically costing just one cent. Furthermore, the long context window of Haiku allows for the inclusion of numerous few-shot examples within the prompt, further refining the summarization process and ensuring the output adheres to specific formatting requirements.

This two-step process offers a compelling alternative to existing summarization techniques. By combining the power of sectioning to identify semantic boundaries with the efficiency of cost-effective LLMs and the advantage of long context windows, it delivers detailed, coherent, and topic-focused summaries. The result is a far richer and more insightful understanding of the original document than traditional methods can provide. This innovative approach represents a significant advancement in LLM-based summarization, offering a powerful combination of scalability, coherence, and cost-effectiveness.

## [00:23:02 - 00:25:12] Benefits and Drawbacks of the Sectioning Summarization Method



Screenshot at 00:23:02

The previous sections detailed a novel approach to LLM summarization: sectioning and targeted summarization. This method, which involves dividing a long document into semantically coherent sections before summarizing each individually, offers significant advantages over traditional techniques like simple "stuffing" or MapReduce. However, as with any innovative approach, it also presents some challenges. Let's delve into the benefits and drawbacks of this sectioning method.

Advantages: Enhanced Quality and Scalability

The sectioning method offers several key advantages. Firstly, it leverages the full context of the entire document when summarizing each individual section. This contrasts sharply with MapReduce, which often suffers from a loss of contextual information due to its arbitrary chunk division. By providing the complete document as context for each section, the LLM can generate summaries that are far more coherent, accurate, and insightful. The relationships between different parts of the document are preserved, resulting in a holistic understanding.

Secondly, this approach allows for the summarization of exceptionally long documents, exceeding the typical token limitations of LLMs. Traditional methods struggle with lengthy texts like YouTube transcripts or extensive articles. The sectioning method effectively bypasses this limitation by generating summaries for smaller, manageable sections, which can then be combined to create a comprehensive overview.

Finally, the use of cost-effective models like Haiku makes this approach economically viable. Despite the increased number of LLM calls (one for sectioning, and one for each section's summarization), the low cost per token makes the overall process surprisingly affordable. The speaker mentions that summarizing a 30,000-token YouTube transcript using this method can cost as little as one cent.

Disadvantages: Quota Management and Increased API Calls

The primary disadvantage of the sectioning method is the increased number of API calls required. While the cost per call might be low, the sheer volume of calls needed for long documents can quickly exhaust API quotas on various LLM platforms. This is a crucial consideration for production environments. The speaker shares a cautionary tale: their summarization app crashed due to exceeding the platform's query limit (60 queries per minute). This highlights the critical need for careful quota management when implementing this approach in a real-world application. Developers need to anticipate the number of calls required and ensure they have sufficient quota allocated or implement strategies to handle potential API rate limits.

In conclusion, the sectioning summarization method presents a powerful alternative to traditional approaches, offering significant improvements in summary quality and scalability, particularly for lengthy documents. However, careful planning and resource management are essential to mitigate the risk of exceeding API quotas and ensuring the smooth operation of applications built using this method. The trade-off between enhanced summary quality and the need for careful quota management is a key consideration for developers.

## Final Thoughts

In this exploration, we've covered the multifaceted nature of [mention the overall topic of the blog post, e.g., sustainable living, effective study habits, the history of X]. We examined [mention key sections, e.g., the environmental impact of consumerism, different learning styles, key historical figures]. Understanding [reiterate a central theme, e.g., the interconnectedness of our actions, the importance of personalized learning, the complexity of historical events] is crucial, as highlighted by [mention a specific example or finding]. Ultimately, [reiterate the main takeaway, e.g., small changes can make a big difference, adapting your study methods leads to better results, understanding context enriches our understanding of the past]. While there's always more to learn, embracing [mention a call to action, e.g., sustainable practices, effective learning strategies, critical thinking] allows us to navigate [mention the context again, e.g., environmental challenges, academic success, historical interpretation] with greater awareness and effectiveness. We hope this exploration has been insightful and encourages you to further investigate this fascinating subject.