AI Query On Personal Computer Data

Ong Yuan Xing1, a), Ian Chai1, 2, b), Aik-Siong Koh3, c)

1Faculty of Computing and Informatics, Multimedia University, 63100 Cyberjaya, Malaysia.

1Centre for Digital Innovations, CoE of Immersive Experience. ultimedia University, 63100 Cyberjaya, Malaysia

3ASKOH.COM LLC, Los Alamos NM 87544, USA.

*b) Corresponding author: ianchai@mmu.edu.my  
c) askoh@askoh.com*

*a) bpzong@gmail.com*

**Abstract.** Artificial Intelligence (AI) transforms how users handle and communicate with their digital personal data. This research project creates an artificial intelligence system that works with local machine data to perform insightful data retrieval, followed by analysis and summary generation. An integrated model featuring Graphics Processing Unit (GPU)-optimized image captioning and text summarization functions enables precise imaging metadata generation and text document insight extraction. All processed data goes into an Excel file for improved graphical presentation and unified access. Users can use the system interface provided to perform two different types of queries. Users can access free responses through a locally installed reasoning model, which operates from structured Excel data or opt for premium access through the DeepSeek 671B model accessible through an Application Programming Interface (API). Using the API, users gain better reasoning capabilities and benefit from a restricted free usage tier that can be extended with paid upgrades. The system provides users with two interrogation choices, which achieve accessibility while allowing for performance scalability. This paper analyzes the system structure together with the model choice, execution specifics, and assessment which, shows its ability to deliver efficient intelligent summaries and data extraction.

# **INTRODUCTION**

As the volume of digital content stored on personal computers continues to grow, the challenge of retrieving meaningful information from this data becomes increasingly complex. People commonly store documents, images, and various forms of structured and unstructured data on their storage disks, which serve as a digital extension of their memory. While human recollection can fade or become unreliable, storage disks retain everything—capturing significant life events, crucial records, and countless digital moments. In this sense, personal storage devices can be seen as repositories of one’s digital life. However, finding specific information within this growing volume of data often requires tedious manual searching, especially when traditional file systems rely heavily on file names and folder structures that may not be consistently maintained.

This research project proposes an AI-powered system that enables users to retrieve and comprehend personal data through natural language queries. The system integrates multiple AI models to handle a variety of tasks: high-performance image captioning models are used to generate metadata for image files, and advanced text summarization models condense the key content of long documents—all operable on a single GPU for practical deployment. Extracted data is structured and stored in an Excel file for improved accessibility and integration. The system also features two query options: a local, free reasoning model that interprets user questions based on the Excel dataset, and a higher-performing option utilizing the DeepSeek Reasoning model (671B parameters) via an API, which offers a free usage tier before requiring payment. This dual-mode design enhances flexibility and performance, allowing users to engage interactively with their digital memories using natural language. By transforming passive storage into an intelligent assistant, this system aims to redefine the way personal data is accessed and understood.

# **LITERATURE** **REVIEW**

Modern innovation in Artificial Intelligence has transformed how data professionals process and manage computer information. This paper examines five essential domains: The review examines AI-driven personal data query systems and three additional use cases that include image caption models and text summary applications, and Large Language Model (LLM) integration with databases and DeepSeek model performance outcomes. These domains demonstrate AI’s ability to transform information management operations.

AI-Driven Query Systems for Personal Data: The implementation of AI-powered query systems has brought about substantial improvements in both data accessibility and better efficiency. [1]. Journalistic “who what when why how where” questions enable automatic query generation for data corpus analysis and are also described. Through human feedback processes this methodology both detects information gaps and blocks nonsensical queries to optimize retrieval system adaptability.

The synergy between databases and AI is categorized into AI for databases (AI4DB) and databases for AI (DB4AI) [2]. AI4DB optimizes query performance and security, whereas DB4AI manages large-scale datasets for AI training. This bidirectional integration improves scalability and accessibility in applications like natural language to Structured Query Language (SQL) translation. [3] evaluated frameworks such as Structured Query Language Network (SQL-Net) and Relation-Aware Transformer for SQL (RAT-SQL), noting challenges in cross-domain query handling despite advances in semantic parsing.

The AI-enabled SQL optimization process brings technical users together with non-technical users. Juopperi [4] illustrated how AI technology performs automated optimization and accelerates SQL processing, yet it still encounters constraints during the processing of complex logical business contexts. Juopperi [4] shows that artificial intelligence serves as a valuable tool that supports human experts in their data retrieval tasks.

The automated process of image captioning uses Convolutional Neural Networks (CNNs) in combination with Recurrent Neural Networks (RNNs) to produce textual explanations. [5] A combined approach of CNNs for feature extraction alongside RNNs for sequence generation improved diagnostic accuracy in medical and assistive technology fields. The insufficient diversity of objects within Microsoft Common Objects in Context (MS COCO’s) dataset becomes a barrier to generalization for modeling systems [6].

The process of automated metadata annotation shares comparable obstacles. Wu [7] demonstrates how domain-focused training data quality combined with human-AI methods can solve time-disparities within heritage information databases. Wu demonstrates why captioning systems require contextual understanding capabilities.

Text summarization demonstrates superior performance when using Deep learning architectures like Long Short-Term Memory (LSTM) networks. [8] The use of sequence-to-sequence models delivered outstanding performance boosted through Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics confirmation. The combination of statistical methods with semantic technology and swarm intelligence techniques yields better summary results [9]. This hybrid strategy demonstrates how AI works well for both journalism and legal document analysis.

Structural data management receives benefits from LLMs through personal query modification combined with SQL query integration capabilities. [10] The application of gradient-boosted ranking models led to enhanced precision in conversational AI systems. The Galois system has connected unstructured text to database systems. [11] Two framework developments including Struct-Generative Pretrained Transformer (StructGPT) [12] and Database Generative Pretrained Transformer (DB-GPT) [13] enhance logical reasoning capabilities and query optimization with privacy protection while demonstrating Table processing scalability [14].

According to research DeepSeek delivers powerful classification outcomes with affordable capabilities at a reduced speed level.[15] Gao [15] and Bi [16] also stated that the system exhibited better results than Generative Pre-trained Transformer 3.5 (GPT-3.5) and Large Language Model Meta AI – Version 2 (LLaMA-2) when used for citation evaluation and that it allowed for optimized code scanning and mathematical computation. Gao and Bi demonstrate that DeepSeek represents an affordable solution to expensive models such as Claude.

Conversational systems have been applied in academic settings to improve access to information. Goh et al. [17] presented a campus-based chatbot for the Faculty of Information Science & Technology, Multimedia University, which used natural language processing and a multilayer perceptron model to interpret student queries and provide updated responses.

# **RESEARCH** **METHODOLOGY**

This research project uses a system approach to develop an AI-driven query system for the management of personal computer content. The system uses Python as the main programming language to collect file efficiency data, content analysis, and intelligent retrieval using AI models for image captioning, text summarization, and question answering. The methodology is divided into four key phases: file collection and indexing, image captioning, text summarization, and question answering.

Phase one looks for relevant files by searching through the user-favored folders, such as Desktop, Downloads, Documents and Pictures, systematically. Only .jpg, .jpeg, .png images and documents in .doc, .docx, and .pdf formats are eligible for selection. These are all compiled together using Excel files in a structured format as the precedent for further analysis (see Figure 1).

Phase two begins the procedure of analyzing image files that have a .jpg, .jpeg or .png extension by using deep learning-based image captioning models to produce detailed metadata. The captions produced enhance the richness of the data index and enable more accurate and relevant search results. To accommodate different hardware capabilities, the system integrates multiple image captioning models with varying performance levels. Users can select the model that best fits their system configuration as depicted in Table 1.

A diagram of a computer model

AI-generated content may be incorrect.

**FIGURE 1.** Sample photos from the directory

**TABLE 1.** Computer vision model details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model Name** | **Unit** | **Source** | **Runtime** | **Notes** |
| Bootstrapped Language Image Pretraining (BLIP) | CPU | Hugging Face | 10–30 seconds/image | Lightweight and CPU-friendly; suitable for systems without GPU support |
| Mini ChatGPT Pre-trained Multimodal Vision (MiniCPM-V) | GPU | Ollama | 20–40 seconds/image | Fast processing; balanced performance on single-GPU setups |
| Google’s Efficient Multimodal Model Architecture (Gemma3:12B) | GPU | Ollama | 30–50 seconds/image | Higher-quality captions; requires more GPU resources |
| Google’s Efficient Multimodal Model Architecture (Gemma3:27B) | GPU | Ollama | 150–210 seconds/image | Most accurate; suitable for high-end GPUs due to large model size |

Model integration is enabled by either pre-trained checkpoints on the Hugging Face Hub, or Ollama’s serving platform. Text summaries are stored alongside their original file locations in a formatted Excel index, which facilitates convenient processing at subsequent stages. In its third phase, the system creates summaries from Microsoft Word documents (doc, .docx), plain text files (txt) and Portable Document Format (PDF) files. Before extraction, the system identifies if a PDF only contains text or images; it uses OCR on image-based PDFs to retrieve text. Consequently, all crucial information is presented to the summarization stage regardless of the source format.

After this, the models apply Transformer architectures to generate concise and informative summaries that are structured for convenient question-answering. The pipeline can be integrated with a CPU or GPU; the user is able to choose models correctly adjusted for his or her hardware to balance a good compromise between speed and resource consumption. Bidirectional and Auto-regressive Transformers (BART) are less resource-demanding and can be performed on CPUs, unless confronted with long or involved documents. When working with hardware that is configured to utilize GPUs, the available models for usage are quite advanced and include Mistral-7B, DeepSeek-V3, Alibaba Qwen Language Model (Qwen) and QWQ LLM (QWQ) which support speedier and more reliable summaries for large and complex documents (see Table 2).

The following table presents an overview of the available models used for text summarization in the system, along with their runtime performance, compatibility with CPU or GPU environments, and their respective sources:

The system is designed to support hardware from basic CPUs to advanced GPUs to offer wide compatibility and flexible performance boosts. With the aid of the CPU-compatible BLIP model from Hugging Face, both image captioning and text summarization run effectively for results completed in 10–30 seconds for each image. For better performance, users can use models like MiniCPM-V and Gemma3, which support larger processing. There are various models that can be accessed, like DeepSeek and Fireworks AI, that serve various application scenarios. The modular design is readily customizable in features and models, and API-based execution is provided to guarantee painless integration with other platforms.

**TABLE 2.** Natural language model details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model Name** | **Unit** | **Source** | **Runtime** | **Notes** |
| BART | CPU | Hugging Face | 30–120 seconds | May fail on long/complex documents |
| Mistral-7B | GPU | Ollama | 10–30 seconds | Fast and efficient with good summarization |
| DeepSeek-V2 (16b) | GPU | Ollama | 20–60 seconds | Balanced between speed and accuracy |
| Qwen (7B/14B) | GPU | Ollama | 20–50 seconds | High-quality generation; memory intensive |
| QWQ | GPU | Ollama | 30–70 seconds | Performance may vary; suitable for experimentation |

The system uses Transformer-based Natural Language Processing (NLP) models to analyse queries and locate the relevant items in the structured Microsoft Excel dataset. It supports questions like “What is in cat3.jpg?” or “When does my passport expire?” by identifying metadata tied to each file. The system adapts model selection based on hardware (see Table 3): TAPAS functions as a lightweight option for CPU users, while more advanced models such as QWQ, Qwen 2.5, and DeepSeek-V2 16B serve GPU users. For state-of-the-art accuracy, DeepSeek-V3 671B is available via API. This adaptive system optimizes performance based on computational capacity and user precision requirements.

**TABLE 3.** Query model details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model Name** | **Unit** | **Source** | **Runtime** | **Notes** |
| TAPAS | CPU | Hugging Face | 10–30 seconds | Lightweight model suitable for CPU-only systems; limited accuracy |
| QWQ | GPU | Ollama | 30–50 seconds | Moderate performance; supports table-based QA with improved accuracy |
| Qwen 2.5 | GPU | Ollama | 30–60 seconds | Efficient model for structured data QA; balanced speed and accuracy |
| DeepSeek-V2 16B | GPU | Ollama | 60–120 seconds | High performance; good for comprehensive question answering tasks |
| DeepSeek-V3 671B (API) | API | DeepSeek.ai | Varies by network | State-of-the-art accuracy; requires API key and internet connectivity |

# **RESULTS AND DISCUSSION**

The results show the superiority of the AI-powered personal storage query system over ordinary retrieval methods, which normally use manual navigating or high-keyword parsing — methods that are often unreliable when the actual file names are unknown or not remembered accurately. Searching for files manually is sometimes slow and can result in errors when keywords are misremembered or mislabeled. In contrast, our system supports user requests to query files for content, paths, sizes, even correcting or refining queries for improved accuracy and processing time.

Figures 2 to 4 show the performance of the AI in caption generation. In this example, an image of foggy railway tracks, models such as BLIP, MiniCPM-V, and Gemma3 have provided their captions with different levels of detail. BLIP constructed simple summaries whereas bigger models analyzed in-depth interpretations of lighting, atmosphere, and composition. This multi-step strategy supports users’ understanding and control over visual files without going to individual ones. As shown in Figure 3, the system illustrates how it can group images, arranging them into a directory, using semantic analogy to suggest relevant content, and therefore increase the relevance of search.

The illustration shown in Figure 5 demonstrates how the system recognizes natural language commands for both understandable as well as unorthodox formulations. Near-real-time detection of query typos and ambiguous verbiage enables the system to generate possible correction options for improved user clarity. Effective user input management through the system improves its usability function which enables operation by both technical and non-technical users.

|  |  |
| --- | --- |
| **A train tracks in a foggy forest  AI-generated content may be incorrect.**  **FIGURE 2.** Sample Photo | A collage of two birds  AI-generated content may be incorrect.  **FIGURE 3.** Sample photos from the directory |

A screenshot of a white page

AI-generated content may be incorrect.

**FIGURE 4.** Sample of metadata generated by models in excel file



**FIGURE 5.** Answer generated by query models

The system operates efficiently with both visual data types and textual data sources. The system displays its summarization results in Figure 6. A complete text with more than 200 words underwent compression maintaining essential information about AI industrial applications and advantages together with ethical considerations. The summarizer maintained vital content elements such as automation and process optimization and data privacy throughout the text and created condensed versions which help users understand complex documentation. The system provides exceptional value when processing collections of textual documents including PDFs and reports and notes.

Figure 7 shows the system’s intelligent handling of user input errors, specifically typographical mistakes in query terms. In the example, the user requested information about a file named “brid3.jpg”, which does not exist in the system’s metadata. Instead of returning an error or empty result, the model inferred that the user likely intended to query “bird3.jpg” based on filename similarity and existing entries. The system responded by displaying the correct file details, including the filename, path, size, timestamps, and an AI-generated image caption (“a photo of a yellow and black bird is sitting on a twig”). This highlights a major improvement over traditional search methods, which typically require exact matches. The AI’s ability to recognize and correct probable typos ensures smoother interaction, minimizes user frustration, and improves search reliability—even when the user input is imprecise.

A screenshot of a computer

AI-generated content may be incorrect.

**FIGURE 6.** Sample of before VS after text summarizing task performed by AI

A screen shot of a computer

AI-generated content may be incorrect.

**FIGURE 7.** Sample of typo in user query and getting the correct response

To quantitatively validate the effectiveness of the system (see Table 4), the image captioning module was tested on 100 image-based search tasks, with both CPU and GPU based models. The BLIP model, which is suitable for CPU, demonstrated a 75% task match rate, meaning that 75 out of 100 captions generated by the model were relevant and content-wise met the author’s expectations. The MiniCPM-V model, which is suitable for GPU, demonstrated a better match rate of 90% on the same dataset. These results show the modularity of the system which presents the user with flexibility of trading off between processing and captioning quality depending on the capability of their hardware and demands.

**TABLE 4.** Performance metrics for CPU VS GPU model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Hardware** | **Match Rate** | **Pros** | **Cons** |
| BLIP | CPU | 70% | Fast, lightweight, CPU-friendly | Less detail, short/brief captions |
| MiniCPM-V | GPU | 90% | More accurate, rich descriptions | Slow, requires GPU hardware support |

To show the advantages of the system, the comparative analysis was performed with current system, such as Windows Search, ChatGPT, Everything, AnythingLLM and Msty (see Table 5). The system is typo-tolerant and interprets natural language, unlike Windows Search and Everything, which employ exact matching of keywords and do not accommodate user error to boost accessibility. ChatGPT and AnythingLLM are powerful tools that offer language comprehension capabilities, yet they still require one to manually upload files and cannot be executed directly on the data stored locally. It implements image captioning, text summarization and question answering features offline. It is therefore more conducive to data privacy and convenience to users. It also embeds image captioning, text summarization, and question answering functionalities in an offline setting, which helps facilitate data privacy and convenience of use. The system also does not need the opening of individual files manually since it can give a direct answer to the query of the user.

**TABLE 5.** Comparison table of system versus existing solution

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Feature** | **Windows Search** | **ChatGPT** | **Everything** | **AnythingLLM / Msty** | **System** |
| Accesses local files directly | Yes | No | Yes | No | Yes |
| Requires manual upload | No | Yes | No | Yes | No |
| Analyzes image data | No | Limited | No | Yes | Yes |
| Analyzes text data | No | Yes | Yes | Yes | Yes |
| Tolerates typos in search queries | No | Yes | No | Yes | Yes |
| Works offline | Yes | No | Yes | Yes | Yes |
| Supports natural language queries | No | Yes | No | Yes | Yes |
| Requires opening files manually | Yes | Yes | Yes | Yes | No |
| Processing time | Short | Short | Long | Long | Long |

# **CONCLUSION**

This research project demonstrates the feasibility and practicality of an AI-powered system for efficient data retrieval on personal computers. By integrating advanced image captioning, text summarization, and structured data querying, the system allows users to intuitively retrieve relevant information without manually browsing through files. Unlike traditional methods that require users to search file by file or remember exact names and locations, this AI-enhanced approach enables natural language interaction for faster and smarter access to personal data.

The system is designed with flexibility in mind—users can run it on machines with CPU, GPU, or even access it through an API, with the ability to select AI models best-suited to their device’s performance capabilities. This inclusivity ensures broader usability across different hardware setups.

Future work will focus on several key areas: enabling the system to directly open files upon user request, enhancing metadata accuracy—especially for sensitive or complex tasks—and implementing photo clustering to group images by individuals. Another planned enhancement includes predicting which photos might contain the user, further improving personal content management. These upgrades aim to boost the system’s intelligence, efficiency, and personalization for real-world deployment.

# **References**

1. Altman, E. (2018). Understanding AI data repositories with automatic query generation. *arXiv preprint arXiv:1804.07819*. Retrieved from https://arxiv.org/abs/1804.07819
2. X. Zhou, C. Chai, G. Li, and J. Sun, “Database Meets Artificial Intelligence: A Survey,” IEEE Trans. Knowl. Data Eng. **34**(3), 1096–1116 (2022).
3. M.S. Baig, A. Imran, A. Yasin, A.H. Butt, and M.I. Khan, “Natural Language to SQL Queries: A Review,” IJIST **4**(1), 147–162 (2022).
4. Juopperi, T. (2024). AI-driven SQL query optimization techniques.
5. Y.P. Singh, S.A.L. Ezaz Ahmed, P. Singh, N. Kumar, and M. Diwakar, “Image Captioning using Artificial Intelligence,” J. Phys.: Conf. Ser. **1854**(1), 012048 (2021).
6. R. Staniūtė, and D. Šešok, “A Systematic Literature Review on Image Captioning,” Applied Sciences **9**(10), 2024 (2019).
7. M. Wu, H. Brandhorst, M.-C. Marinescu, J.M. Lopez, M. Hlava, and J. Busch, “Automated metadata annotation: What is and is not possible with machine learning,” Data Intelligence **5**(1), 122–138 (2023).
8. M.-Y. Day, and C.-Y. Chen, “Artificial Intelligence for Automatic Text Summarization,” in *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, (IEEE, Salt Lake City, UT, 2018), pp. 478–484.
9. Nazari, N., & Mahdavi, M. A. (2019). A survey on automatic text summarization. Journal of AI and Data Mining, 7(1), 121-135.
10. E. Cho, Z. Jiang, J. Hao, Z. Chen, S. Gupta, X. Fan, and C. Guo, “Personalized Search-based Query Rewrite System for Conversational AI,” in *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI*, (Association for Computational Linguistics, Online, 2021), pp. 179–188.
11. M. Saeed, N.D. Cao, and P. Papotti, “Querying Large Language Models with SQL,” (2023).
12. J. Jiang, K. Zhou, Z. Dong, K. Ye, X. Zhao, and J.-R. Wen, “StructGPT: A General Framework for Large Language Model to Reason over Structured Data,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, (Association for Computational Linguistics, Singapore, 2023), pp. 9237–9251.
13. X. Zhou, Z. Sun, and G. Li, “DB-GPT: Large Language Model Meets Database,” Data Sci. Eng. **9**(1), 102–111 (2024).
14. W. Lu, J. Zhang, J. Fan, Z. Fu, Y. Chen, and X. Du, “Large language model for table processing: a survey,” Front. Comput. Sci. **19**(2), 192350 (2025).
15. T. Gao, J. Jin, Z.T. Ke, and G. Moryoussef, “A Comparison of DeepSeek and Other LLMs,” (2025).
16. X. Bi et al., “DeepSeek LLM: Scaling Open-Source Language Models with Longtermism,” (2024).
17. T.-J. Goh, L.-Y. Chong, S.-C. Chong, and P.-Y. Goh, “A Campus-based Chatbot System using Natural Language Processing and Neural Network,” *Journal of Informatics and Web Engineering* **3**(1), 96–116 (2024)