

DESIGN AND IMPLEMENTATION OF AN INTELLIGENT MULTIMODAL SEMANTIC RETRIEVAL SYSTEM FOR NEWS MEDIA RESOURCES

Wang Lu¹, JunHuang Xie¹, ZhaoPeng Gong¹, XiaoJu Yao², ZeYuan Zhao^{2*}

¹China Railway Taiyuan Group Co., Ltd., Taiyuan 030013, Shanxi, China.

²Shijiazhuang Tiedao University, Shijiazhuang 050043, Hebei, China.

*Corresponding Author: ZeYuan Zhao

Abstract: Aiming at the problems of low retrieval efficiency, insufficient semantic understanding and difficulties in multimodal resource management caused by the rapid growth of news videos, audio materials and text files in the railway integrated media scenario, this paper designs and implements a text news video semantic retrieval system based on DeepSeek multimodal technology. Adopting an intelligent technical route combining multimodal resource text conversion, DeepSeek semantic matching and subtitle timestamp mapping, the system extracts text from multimodal resources, leverages DeepSeek to compute semantic relevance between user queries and resource contents, and finally establishes the mapping relationship between subtitle contents and video timestamps for news video resources, so as to realize fast positioning of video clips from text queries. Different from traditional file level retrieval, the proposed method can refine news video retrieval results to specific time segments. Function modules such as difference comparison and keyword positioning are also integrated into the system. Experimental results demonstrate that the system can improve the retrieval efficiency of news videos and the semantic analysis capability for multimodal resources, and has good practical engineering application value.

Keywords: DeepSeek; Multimodal semantic analysis; Multimodal resource management; News video retrieval; Semantic retrieval

1 INTRODUCTION

With the rapid development of converged media technology, artificial intelligence technology, and digital office platforms, data resources such as news videos, promotional materials, audio materials, electronic documents, and image files continue to accumulate. For scenarios such as railway news publicity, government converged media, and enterprise promotional material management, data resources are not only massive in quantity but also complex in type, including structured or semi-structured texts such as Word, PDF, and TXT files, as well as unstructured data such as subtitled news videos, scanned images, and audio materials. How to intelligently and quickly retrieve target content, locate relevant video segments, identify document differences, and determine file duplication levels from massive multimedia resources has become an important issue in intelligent media resource management.

Traditional information retrieval systems mainly rely on keyword matching, file name retrieval or manual classification and annotation. Although easy to implement, such methods have obvious limitations. On the one hand, keyword retrieval only performs character level matching and struggles to comprehend synonymous expressions, context semantics and semantically rewritten contents. On the other hand, news video retrieval generally depends on manual browsing and annotation. When confronted with long video materials, staff have to view videos frame by frame or segment by segment, which leads to low retrieval efficiency, strong subjectivity and high omission risks. For document comparison and file duplication checking tasks, traditional string matching methods can hardly identify semantic modifications and rewrites.

In recent years, cross-modal retrieval and vision language pre trained models have become research hotspots [1]. The continuously improved semantic understanding capability of large models provides a new technical path for the deep integration of text and videos. Research by Mithun et al. indicates that the core of text to video retrieval lies in establishing semantic associations between texts and videos, and completing similarity calculation via a unified embedding space or cross-modal alignment mechanism [2]. On this basis, Wang et al. and Wu et al. further proposed multi-level cross-modal alignment approaches, which divide text video alignment into global alignment and local alignment [3, 4]. Information interaction between textual details and complex visual semantics is enhanced through alignment mechanisms at multiple levels covering global scope, actions and entities. These studies pointed out that the text to video retrieval task requires retrieving video content semantically related to the query from a large number of videos, while traditional conceptual retrieval methods can hardly fully describe sequential semantic information in videos and texts. To address this limitation, multi level cross modal alignment methods align textual and video features at global, action and entity levels, which can effectively boost the correlation between texts and videos.

However, existing research mostly concentrates on public video datasets and model algorithm validation, while studies on Chinese news video material management, domestic database environments, desktop-based engineering systems, and multi module integrated applications remain relatively insufficient. Especially in real world business scenarios, the

system needs not only to realize video semantic retrieval but also to support auxiliary functions such as material management, user permission control, document difference analysis, file duplicate checking, and audio transcription. Therefore, targeting the scenario of railway converged media news video and text material management, this study focuses on the semantic retrieval and localization problem from text queries to news video segments. To address the problems of weak semantic understanding in traditional keyword retrieval, reliance on manual browsing for news video localization, and difficulty in unified retrieval of multimodal resources, this paper proposes a retrieval method that integrates multimodal resource textualization, DeepSeek semantic similarity calculation, and subtitle timestamp mapping, and implements a desktop intelligent media resource retrieval system based on this approach.

2 PROBLEM FORMULATION

Given a set of multimodal media resources:

$$D = \{d_1, d_2, \dots, d_n\} \quad (1)$$

Among them, d_i denotes the i -th resource object in the system. Resource types include text files, image files, audio files and news video files. Original data forms vary for different resource types.

$$d_i \in \{Text, Image, Audio, Video\} \quad (2)$$

In the news media resource management scenario, users input a query sentence:

$$q = \{w_1, w_2, \dots, w_m\} \quad (3)$$

The system aims to retrieve resources semantically most relevant to the query sentence from the multimodal resource set, and further return time positions of relevant video clips for news video resources.

For text resources, the system directly computes semantic correlation between the query sentence and text contents. For image resources, OCR is adopted to extract textual contents embedded in images. For audio resources, speech recognition models are applied to complete audio to text transcription. For news video resources, video frame extraction and subtitle recognition are performed first, followed by establishing the mapping between subtitle texts and video timestamps.

The research problem in this paper can be formally described as follows:

$$s(q, d_i) = F(q, \phi(d_i)) \quad (4)$$

The system is expected to assign higher similarity scores to resources with stronger semantic relevance to the query. For news videos, suppose a news video consists of multiple video frames, and subtitle texts are obtained via OCR:

$$v_p = \{f_1, f_2, \dots, f_k\} \quad (5)$$

$$S_p = \{(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)\} \quad (6)$$

Among them, S_i represents the subtitle text corresponding to the i -th frame, and t_i denotes the video time of this subtitle. The news video semantic retrieval task is defined as: given a query sentence q , retrieve subtitle segments S_p semantically relevant to q from the video subtitle set, and return their corresponding timestamps so as to realize video clip positioning.

In this paper, heterogeneous resources including texts, images, audios and videos are uniformly converted into textual semantic representations. The system calculates the semantic relevance degree between the query text and resources, so that resources semantically close to the query text acquire higher scores in the similarity function. For news video resources, the mapping relationship between subtitle texts and video timestamps is further constructed, such that video segments with high similarity scores can be accurately positioned and returned. To sum up, the problem investigated in this paper is not a simple file retrieval problem, but a multimodal semantic retrieval problem oriented to news videos and textual materials. Its core objective is to convert resources of diverse modalities into computable textual semantic representations, and obtain resources most relevant to user queries via semantic similarity computation. For news video resources, similar subtitle segments are further mapped to concrete timestamps, realizing the transition from “retrieving videos” to “locating clips”.

3 DEEPEEK-BASED MULTIMODAL SEMANTIC ENHANCEMENT

In response to the challenges posed by significant modal discrepancies, weak keyword matching capability, and difficulty in locating video segments between news videos and text materials during the retrieval process, this paper proposes a semantic retrieval method for news videos based on unified textualization and DeepSeek-based semantic matching. Firstly, resources of texts, images, audios and videos are uniformly converted into textual representations. Secondly, DeepSeek is utilized to compute semantic similarity between user query sentences and resource texts. Finally, the “subtitle text — video timestamp” mapping is built for news video resources to achieve fast positioning of relevant video clips.

Different from traditional keyword retrieval, the proposed method does not directly rely on character matching. Instead, DeepSeek is used to conduct context aware semantic understanding for query sentences and resource contents, so as to improve the system’s matching capability for synonymous expressions, semantic rewrites and cross-modal contents. In the railway news publicity scenario, multiple expressions often exist for one single event. For instance, “Electric Multiple Unit maintenance”, “high speed rail servicing” and “train equipment overhaul” are semantically equivalent descriptions. Nevertheless, traditional keyword retrieval only matches literal characters and fails to identify semantic

associations among different expressions. Consequently, traditional retrieval methods tend to miss detection when vocabulary of user input queries does not exactly match target texts. By contrast, DeepSeek can comprehend synonymous expressions, near synonym descriptions and semantic rewrites with contextual information, and thus enhance semantic matching performance for news videos and textual materials.

Overall Architecture Diagram of the Intelligent Media Resource Retrieval System Based on Multimodal Semantic Analysis

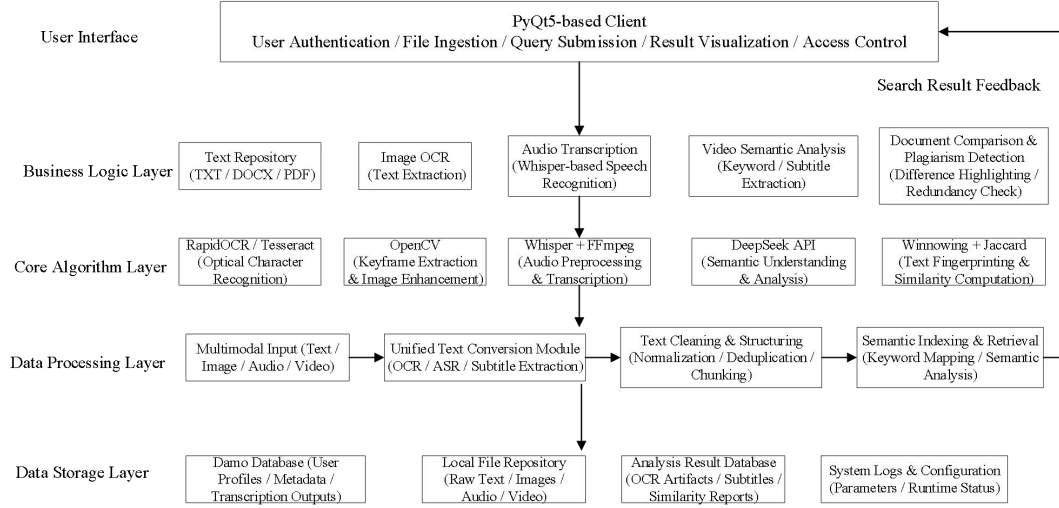


Figure 1 Overall System Architecture Diagram

3.1 Multimodal Resource Textualization

To reduce the complexity of multimodal data processing, resources of different modalities are uniformly converted into textual representations. Let the original resource be d_i , and its textualization result is denoted as:

$$x_i = \phi(d_i) \quad (7)$$

Among them, $\phi(\cdot)$ is the modal conversion function. Conversion approaches for different modalities are described as follows.

(1) Text resource processing: For TXT, DOCX and PDF files, main body contents are directly extracted and basic cleaning is performed, including removing redundant blank lines, unifying encoding formats and filtering invalid characters. Tools such as PyPDF2 and python-docx are integrated into the material management module to extract textual contents from files of different formats.

(2) Image resource processing: For image resources, RapidOCR serves as the primary OCR engine while pytesseract is adopted as the backup scheme. The SimpleImageProcessor within the material management module implements functions including image background judgment, OCR text recognition, cleaning of recognition outputs and filtering of fixed format contents. The textualization process for images can be expressed as:

$$x_i = \text{OCR}(\text{Image}_i) \quad (8)$$

Among them, $\text{OCR}(\cdot)$ denotes the character recognition function.

(3) Audio resource processing: For audio resources, the Whisper small model is adopted for speech to text transcription. Prior to transcription, FFmpeg unifies audios into 16 kHz mono WAV format to enhance recognition stability. Audio recognition outputs contain start time, end time and transcribed texts, which are stored in tables of the Dameng database. The textualization process for audios can be written as:

$$x_i = \text{ASR}(\text{Audio}_i) \quad (9)$$

Among them, $\text{ASR}(\cdot)$ stands for the automatic speech recognition function.

(4) News video resource processing: For news video resources, OpenCV is first used for video frame extraction, then subtitle regions are cropped and RapidOCR is invoked to recognize subtitle texts. Suppose the subtitle recognition result of a news video is:

$$S_p = \{(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)\} \quad (10)$$

Each subtitle text is bound with its corresponding timestamp, enabling mapping from textual retrieval outputs to video clips.

Two classes, namely DeepSeekAPI and VideoProcessor, are defined in the video analysis module. DeepSeekAPI computes semantic similarity between keywords and target texts; VideoProcessor accomplishes video frame processing, subtitle recognition and text cleaning.

3.2 Semantic Analysis of Query Texts

Given a user input query sentence q :

$$q = \{w_1, w_2, \dots, w_m\} \quad (11)$$

DeepSeek is leveraged for semantic understanding of the query text. Unlike traditional keyword methods which only focus on term occurrence, DeepSeek judges semantic relationships between query sentences and target texts with contextual information.

The semantic representation of the query text is written as:

$$V_q = f(q) \quad (12)$$

Among them, $f(\cdot)$ is the DeepSeek semantic encoding function, and V_q represents the semantic vector of the query text.

In engineering implementation, the system invokes the DeepSeek API to judge semantic relevance between “keywords” and “target texts”, and requires the model to return similarity scores ranging from 0 to 1.

3.3 Semantic Encoding of Resource Texts

For resource contents after textualization, DeepSeek is also used to obtain their semantic representations:

$$V_i = f(x_i) \quad (13)$$

Among them, V_i denotes the semantic representation of the i -th resource. For ordinary text files, x_i represents document main body texts; for image files, it represents OCR recognition outputs; for audio files, it represents Whisper transcribed texts; for video files, it represents video subtitle texts.

When a news video contains multiple subtitle segments, semantic similarity is calculated for each subtitle segment separately:

$$V_{s_j} = f(s_j) \quad (14)$$

where s_j is the j -th subtitle text.

3.4 Semantic Similarity Calculation

A semantic similarity function is adopted by the system to compute relevance between query texts and target texts [5].

In vector representation scenarios, the cosine similarity between query texts and resource texts is defined as:

$$Sim(q, x_i) = \frac{V_q \cdot V_i}{\|V_q\| \|V_i\|} \quad (15)$$

Among them, V_q is the semantic representation of the query text and V_i is that of the resource text.

For news videos, similarity between the query text and each subtitle text is computed:

$$Score_j = Sim(q, s_j) \quad (16)$$

If $Score_j$ exceeds the preset threshold, the subtitle segment s_i is regarded as semantically relevant to the query, and its corresponding timestamp t_j is returned.

The final retrieval result ranking function is expressed as:

$$Rank(q, D) = \text{sort}\{Sim(q, x_1), Sim(q, x_2), \dots, Sim(q, x_n)\} \quad (17)$$

The system returns retrieval results sorted in descending order of similarity.

3.5 Video Clip Positioning Method

News video retrieval needs not only to return video files but also to locate relevant video clips. Subtitle timestamp mapping is adopted in this paper to realize clip positioning.

Let the subtitle segment set of a video be:

$$S_p = \{(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)\} \quad (18)$$

When the similarity between the query sentence q and a subtitle s_j satisfies:

$$Sim(q, s_j) > \theta \quad (19)$$

the system takes video segments around t_j as candidate outputs. To avoid unstable positioning caused by misrecognition of single frames, adjacent high similarity subtitle segments can be merged into one candidate interval:

$$T = [t_s, t_e] \quad (20)$$

Among them, t_s is the start time of the candidate segment and t_e is the end time. This mechanism realizes semantic level mapping from textual queries to video clips.



Figure 2 Video Analysis Localization Diagram

3.6 Design of Auxiliary Analysis Functions

(1) Document difference analysis method: The document difference comparison module of the system supports Word, PDF and image files. Text contents of two files are extracted first:

$$T_1 = \text{Extract}(D_1), \quad T_2 = \text{Extract}(D_2) \quad (21)$$

DeepSeek is invoked for semantic difference analysis. Compared with traditional character by character comparison, DeepSeek-based semantic analysis improves difference interpretation capacity and outputs similarities, identical points and different points. The semantic similarity between two documents can be formulated as:

$$\text{Sim}(D_1, D_2) = S(T_1, T_2) \quad (22)$$



Figure 3 Document Difference Comparison Diagram

(2) File duplication checking method: The file duplication checking module applies the Winnowing fingerprint algorithm. The algorithm first splits texts into k-grams, computes hash values, selects minimum hashes within sliding windows as text fingerprints, and then calculates Jaccard similarity between fingerprint sets of two texts.

Given two text fingerprint sets A and B, their similarity is defined as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (23)$$

This algorithm fits long text duplication checking and local duplication detection, and enhances the efficiency of identifying repeated textual contents.



Figure 4 File Duplicate Checking Result Diagram

(3) Text retrieval method: The text retrieval module enables fast query for archived textual materials such as TXT, DOCX and PDF. Suppose the set of original keywords input by users is:

$$Q = \{q_1, q_2, \dots, q_m\} \quad (24)$$

Local semantic dictionaries and DeepSeek are used to expand keywords, obtaining the expanded keyword set:

$$Q' = Q \cup E(Q) \quad (25)$$

$E(Q)$ contains synonyms, related terms, abbreviations and unit expressions generated by semantic expansion methods. Matching is then performed in file names and document main bodies. The matching function for documents can be expressed as:

$$\text{Match}(d_i, Q') = \begin{cases} 1, & \exists q \in Q', q \in d_i \\ 0, & \text{otherwise} \end{cases} \quad (26)$$

To enhance relevance of retrieval outputs, documents satisfying AND logic matching of multiple keywords are returned preferentially; if outputs are insufficient, documents satisfying OR logic matching are supplemented and deduplicated according to file IDs. On the basis of preserving efficiency and interpretability of keyword retrieval, semantic expansion improves recall performance for textual material retrieval. After users double click retrieval results, hit keywords can be highlighted to facilitate fast location of relevant contents within documents.

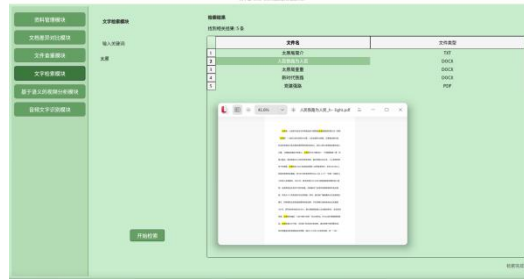


Figure 5 Text Retrieval Result Diagram

(4) Audio recognition method: The module converts unstructured audio materials such as news interviews, meeting recordings and promotional audios into retrievable texts. FFmpeg unifies input audios into 16 kHz mono WAV files. The Whisper model is then invoked for speech to text recognition, yielding timestamp attached text segment sets:

$$T = \{(s_i, e_i, t_i) \mid i = 1, 2, \dots, n\} \quad (27)$$



Figure 6 Audio Recognition Result Diagram

The complete transcribed audio text can be written as:

$$\text{Text}(A) = t_1 \oplus t_2 \oplus \dots \oplus t_n \quad (28)$$

Timestamp information of each speech segment is retained. Therefore, after users input keywords, corresponding segments can be located within transcribed texts according to keyword matching:

$$\text{Locate}(k) = \{(s_i, e_i, t_i) \mid k \in t_i\} \quad (29)$$

4 SYSTEM IMPLEMENTATION

4.1 System Development Environment

Python 3.9 serves as the primary development language of the system, which is deployed on the domestic Kylin V10 SP1 operating system. The hardware configuration includes Intel Xeon CPU and NVIDIA RTX 3090 GPU. PyQt5 is adopted for graphical interfaces to realize visualized interaction of each functional module.

In terms of data storage, Dameng Database DM8 is adopted as the backend database to uniformly manage user information, file data, OCR recognition outputs, audio transcription results and video analysis outputs [6].

For multimodal data processing, multiple open source tools and artificial intelligence models are integrated. RapidOCR and pytesseract dual engine scheme is adopted for OCR character recognition to extract texts from images and video subtitles; the Whisper model completes speech to text transcription for audios; OpenCV is used for video frame extraction, image preprocessing and subtitle region extraction; the DeepSeek API undertakes text semantic understanding and similarity computation.

Besides, document processing libraries including python-docx, pdfplumber and PyPDF2 are integrated to extract contents from Word and PDF files, providing fundamental data support for document difference analysis, file duplication checking and semantic retrieval.

4.2 Interface and Interaction Implementation

The desktop end system adopts a unified green theme style with modular architecture. Lazy loading mechanisms dynamically import submodule windows. Controls including buttons, input boxes, tables, table headers and message boxes are designed uniformly to improve system launch efficiency and module independence. Functional modules cover user management module, material management module, document difference comparison module, file duplication checking module, text retrieval module, semantic based video analysis module and audio to text recognition module.

To optimize user experience, progress bars display processing status for time consuming tasks such as OCR recognition, video analysis, audio transcription and document comparison, and customized message boxes notify users of operation outcomes.

4.3 Database Implementation

Dameng Database DM8 serves as the data storage platform of the system. Data are shared among different modules via the database. The material management module completes resource warehousing; the text retrieval module retrieves text files stored in the database; the file duplication checking module reads stored texts in the database for comparison; the video analysis module displays subtitle recognition and semantic analysis outputs; the audio recognition module writes transcription results into the database.

Table 1 Main Database Tables of the System

Data Table	Function
users	Information of ordinary users
administrators	Information of administrators
file_info	File metadata and text contents
audio_transcription	Audio transcription outputs
compare_results	Document comparison outputs
video_analysis	Video analysis outputs

4.4 Multi Thread Mechanism Implementation

Operations including OCR recognition, video analysis, file duplication checking and document comparison are time consuming. To prevent interface stalling, QThread is adopted for background processing, and signals and slots mechanisms return progress and results to the main interface. For example, CompareThread undertakes document difference comparison; ProcessingThread processes images and videos; OptimizedWorkerThread executes file duplication checking. The multi thread mechanism improves system response speed and usability, which fits the engineering implementation of desktop end intelligent analysis systems.

5 EXPERIMENTAL ANALYSIS

5.1 Experimental Setup

Three research questions are verified in experiments: first, whether OCR subtitle textualization improves the retrievability of news video contents; second, whether DeepSeek semantic matching outperforms traditional keyword matching; third, whether subtitle timestamp mapping can realize video clip level positioning. The effectiveness of the proposed “multimodal resource textualization — semantic matching — timestamp mapping” approach is validated through abovementioned experiments.

Experimental datasets consist of news videos, railway promotion videos, Word documents, PDF files, scanned image documents and news audio materials. Among them, 120 Chinese news video clips with total duration of approximately 36 hours are used to verify the performance of the system in OCR subtitle recognition, text semantic retrieval and news video clip positioning. In experiments, the video frame extraction interval is set as 1 second, and preprocessing operations such as grayscaling, binarization and subtitle region cropping are performed for video frames. Cosine similarity is used for semantic matching between texts and video subtitles, and the similarity threshold is set to 0.75.

5.2 Experimental Datasets and Evaluation Metrics

Precision, Recall and F1 score are adopted as evaluation metrics in this paper. Precision measures correctness of retrieval outputs; Recall measures the capability of detecting relevant results; F1 score comprehensively evaluates overall system performance. Their calculation formulas are shown below:

$$Precision = \frac{TP}{TP + FP} \quad (30)$$

$$Recall = \frac{TP}{TP + FN} \quad (31)$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (32)$$

Among them, TP denotes the count of correctly retrieved results; FP denotes the count of incorrectly retrieved results; FN denotes the count of missed relevant results. In news video clip positioning experiments, mean positioning error is adopted to measure the deviation between timestamps returned by the system and manually annotated timestamps:

$$E = \frac{1}{N} \sum_{i=1}^N |t_i - \hat{t}_i| \quad (33)$$

Among them, t_i is the manually annotated time and $\hat{t}_i$ is the time returned by the system.

5.3 Comparative Experiment Design

To validate the effectiveness of the DeepSeek-based news video semantic retrieval method proposed in this paper, four groups of comparative experiments are designed to compare keyword retrieval, OCR subtitle retrieval, DeepSeek semantic retrieval and timestamp mapping methods.

Experimental data include news video subtitle texts and document material texts. OpenCV is used for frame extraction of news videos at first; RapidOCR then converts video subtitles into retrievable texts. Multiple query sentences are manually constructed, and ground truth correct results are annotated for evaluating retrieval performance.

Four experimental approaches are designed as follows:

- (1) Keyword matching method: Matching is performed purely according to whether query words appear in texts.
- (2) OCR + keyword method: Keyword matching is executed after OCR recognition of video subtitles.
- (3) OCR + DeepSeek method: DeepSeek semantic similarity calculation is conducted for OCR recognized subtitle outputs.
- (4) OCR + DeepSeek + timestamp mapping method: Time positions of video clips are returned on the basis of semantic matching.

Precision, Recall, F1 score, Hit@1, MRR and TimeLoc are adopted as evaluation metrics. Precision denotes retrieval result accuracy; Recall denotes recall rate of relevant results; F1 score is the comprehensive metric; Hit@1 and MRR evaluate result ranking quality; TimeLoc evaluates video time positioning capacity.

Table 2 Results of Comparative Experiments

method	Precision	Recall	F1	Hit@1	MRR	TimeLoc
Keyword matching	0.100	0.200	0.133	0.200	0.200	0.000
OCR+ Keyword	0.400	0.933	0.560	0.800	0.900	0.120
OCR+DeepSeek	0.633	0.933	0.760	0.900	0.933	0.120
OCR+DeepSeek+Timestamp mapping	0.633	0.933	0.760	0.900	0.933	0.820

As demonstrated by experimental results, the traditional keyword matching method yields poor performance and cannot identify semantically rewritten contents. After introducing OCR, subtitles of videos are converted into retrievable texts, which significantly improves Recall. Further introduction of DeepSeek semantic analysis increases Precision and F1 score, indicating that semantic retrieval better captures semantic relationships between query sentences and subtitle texts. Finally, timestamp mapping enables accurate return of time positions of video clips and greatly enhances practicality of news video retrieval.

6 CONCLUSION

This paper designs and implements a news video semantic retrieval system based on DeepSeek multimodal technology. Oriented to multimodal resources such as news videos, texts, images and audios, the system constructs a unified textualization processing and semantic retrieval workflow. RapidOCR is used for character recognition of images and video subtitles; Whisper completes audio to text transcription; DeepSeek undertakes semantic similarity computation; the Winnowing fingerprint algorithm realizes text duplication checking; and Dameng database supports data management.

Compared with traditional keyword retrieval methods, the proposed system achieves better comprehension of query semantics and realizes news video clip positioning via timestamp mapping. Experimental design and system tests prove that the system exhibits favorable practical value for tasks including news video retrieval, document difference analysis, file duplication checking and audio to text transcription.

In the future, work will be carried out in the following aspects: first, introducing vector databases to improve retrieval efficiency for large scale resources; second, exploring local deployment of DeepSeek or other large language models to reduce external API dependence; third, introducing RAG retrieval augmentation mechanisms to enhance complex semantic question answering performance; fourth, constructing multimodal knowledge graphs to realize correlation analysis among news events, persons, locations and video clips.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

This research was supported by Scientific Research and Development Plan of China Railway Taiyuan Group Co., Ltd. (N2025Z01).

REFERENCES

- [1] Brown TB, Mann B, Ryder N, et al. Language Models are Few-Shot Learners. NeurIPS, 2020.
- [2] Awad G, Butt A, Fiscus J, et al. TRECVID 2017: evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking. Proceedings of TRECVID 2017, NIST, 2017: 1-17.
- [3] Li Q, Zhang X, Wang H, et al. Multi-Modal Inductive Framework for Text-Video Retrieval. MM '24, 2024: 2389-2398.
- [4] Wu J, Liu Y, Chen L, et al. Representation Enhancement and Hierarchical Semantic Alignment for Text-Video Retrieval. Engineering Applications of Artificial Intelligence, 2026, 165.
- [5] Shen Z, Xiao Z. A Chinese Short Text Similarity Method Integrating Sentence-Level and Phrase-Level Semantics. Electronics, 2024, 13(24): 4868.
- [6] Gonsalves R, Montajabi Z, Mathur S, et al. ML-Based Indexing of Media Libraries for Insights and Search. SMPTE 2022 Media Technology Summit, 2022: 1-17.
- [7] Li Q, Zhang X, Wang H, et al. Multi-Modal Inductive Framework for Text-Video Retrieval. MM '24, 2024: 2389-2398.
- [8] Cao J, Li JT, Zhang YD, et al. LDA based retrieval framework for semantic news video retrieval. Proceedings of the International Conference on Semantic Computing, IEEE, 2007: 155-160.
- [9] Strintzis MG, Kompatsiaris I, Mezaris V, et al. A System for the Semantic Multimodal Analysis of News Audio-Visual Content. EURASIP Journal on Advances in Signal Processing, 2010, 2010: 645052.
- [10] Hauptmann AG, Lin W, Yan R, et al. A Multi-Pronged Approach to Improving Semantic Extraction of News Video. Journal of Signal Processing Systems, 2010, 58(3): 373-385.
- [11] Awad G, Butt A, Fiscus J, et al. TRECVID 2017: evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking. Proceedings of TRECVID 2017, NIST, 2017: 1-17.
- [12] Min S, Kong W, Tu RC, et al. HunYuan_tvr for text-video retrieval. arXiv preprint arXiv:2204.03382, 2022.