IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 May 2024, accepted 12 July 2024, date of publication 18 July 2024, date of current version 6 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3430375

RESEARCH ARTICLE

Extraction of Meta-Data for Recommendation Using Keyword Mapping

GEON-WOO KIM[1], WOO-HYEON KIM[1], KYUNGYONG CHUNG[1], AND JOO-CHANG KIM[2]
[1]Department of AI Computer Science and Engineering, Kyonggi University, Suwon 16227, South Korea
[2]Contents Convergence Software Research Institute, Kyonggi University, Suwon 16227, South Korea

Corresponding author: Joo-Chang Kim (kjc2232@naver.com)

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2020R1A6A1A03040583). Additionally this work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No: RS-2023-00248899).

ABSTRACT Expanding traditional video metadata and recommendation systems encompasses challenges that are difficult to address with conventional methodologies. Limitations in utilizing diverse information when extracting video metadata, along with persistent issues like bias, cold start problems, and the filter bubble effect in recommendation systems, are primary causes of performance degradation. Therefore, a new recommendation system that integrates high-quality video metadata extraction with existing recommendation systems is necessary. This research proposes the ''Extraction of Meta-Data for Recommendation using keyword mapping,'' which involves constructing contextualized data through object detection models and STT (Speech-to-Text) models, extracting keywords, mapping with the public dataset MovieLens, and applying a Hybrid recommendation system. The process of building contextualized data utilizes YOLO and Google's Speech-to-Text API. Following this, keywords are extracted using the TextRank algorithm and mapped to the MovieLens dataset. Finally, it is applied to a Hybrid Recommendation System. This paper validates the superiority of this approach by comparing it with the performance of the MovieLens recommendation system that does not expand metadata. Additionally, the effectiveness of metadata expansion is demonstrated through performance comparisons with existing deep learning-based keyword extraction models. Ultimately, this research resolves the cold start and long-tail problems of existing recommendation systems through the construction of video metadata and keyword extraction.

INDEX TERMS Object detection, speech-to-text, recommendation system, contextual data, keyword extraction, textRank.

I. INTRODUCTION

Recent research on extracting video metadata and its application to recommendation systems is currently very active [1]. Recommendation systems that utilize various metadata extractions are critical research topics for managers of internet service platforms such as Over-The-Top (OTT) media services and YouTube [2]. Rich video metadata offers the advantage of providing users with more accurate recommendations while conveying approximate information about the video without the need for viewers to watch it. Traditionally, video metadata is manually created by those who distribute or edit the video [3]. However, because this process is manual, it can result in subjective metadata, potentially leading to inaccurate recommendations. Additionally, malicious metadata from editors or distributors can result in the recommendation of harmful videos, and important features of a video may be lost depending on its runtime [4]. Therefore, traditional methods of video metadata creation can undermine the reliability of the metadata and degrade the performance of recommendation systems.

Subsequent developments have been made in using deep learning models to extract video metadata and incorporate it into recommendation systems. The most common method involves using a video summarization system to extract the plot of a video [5]. The video is divided into fixed frames,

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo.

and a summary is generated through analyzing the behavior patterns and classifying the objects within these frames. This summary, along with the video's genre, is then mapped into the recommendation system. However, deep learning-based video summarization systems have limitations, such as the inability to summarize longer videos effectively. These systems face the cold start problem, where recommendations cannot be made without user information [6].

Additionally, methods that do not use deep learning models are also being developed and applied in real life. Content-based recommendation systems are employed, mapping user viewing logs or web cookie metadata with metadata from other videos. On internet services such as OTT platforms and YouTube, while the plot is included in the recommendation process, major content that could be a spoiler is excluded. This method, though recommending content with similar titles, genres, actors, and plots, may lead to differing perceptions of similarity among users. Additionally, relying on user viewing logs or web cookies can result in inaccurate recommendations if security settings restrict access to necessary data. Conversely, relying solely on user information can lead to a bubble effect, continuously recommending biased content [7].

Therefore, this paper proposes a model that addresses the creation of objective and accurate video metadata and the cold start problem, subsequently improving the performance of recommendation systems. The proposed model uses YOLO for object detection and STT for extracting voice information to create contextualized data. It further employs a TextRank-based keyword extraction algorithm to generate expanded video metadata [8]. By utilizing both User and Item information, this approach not only resolves the cold start problem but also enhances the accuracy of the recommendation system.

The contributions of the proposed method in this paper are as follows:

- The proposed approach suggests converting video data into text data to extract keywords, merging contextual data construction with the TextRank keyword extraction technique to derive more accurate keywords from videos. This new methodology provides a fresh perspective on video keyword extraction and explores the potential applications in various metadata expansion techniques. By leveraging this approach, the aim is to enhance the precision and utility of metadata, thereby improving the effectiveness of recommendation systems across different media platforms.
- This research aims to generate contextualized data by utilizing both image and voice information from videos, addressing a significant limitation of traditional video summarization systems, which primarily analyze video frames for image captioning and neglect audio content. By integrating both visual and auditory data, this approach more effectively constructs a comprehensive metadata set. This enriched metadata enhances the understanding and indexing of video content, which can significantly improve the accuracy and relevance of recommendations in video streaming platforms.
- The enhancement of recommendation system performance based on metadata expansion is a key component of this paper. By comparing recommendation systems on datasets with and without metadata expansion, the research objectively assesses the effectiveness of the proposed methodology. This comparison aims to demonstrate the superiority of the proposed approach and its contribution to improving recommendation system performance through metadata expansion. This approach not only validates the method but also highlights the critical role of detailed and comprehensive metadata in enhancing the accuracy and relevancy of recommendations.

The composition of this paper is as follows. In Section II, the existing video metadata expansion method and the currently used recommendation system are explained. In Section III, video is applied to a deep learning model to extract image information and voice information to make one contextualized data, and metadata is expanded by extracting keywords. After that, it describes how to apply it to the recommendation system. In Section IV, the results and performance evaluations are described. Finally, Section V concludes the paper, summarizing the findings and suggesting directions for future research.

II. RELATED WORK

Recommendation systems based on video metadata expansion often face challenges due to inaccurate metadata, the cold start problem, and the bubble effect, which can degrade system performance. Consequently, there is a push towards developing more advanced metadata expansion methods. This section reviews the latest approaches in video metadata expansion and the principles behind recommendation systems, discussing how these integrate with and enhance the current paper. The discussion aims to explore how recent advancements can mitigate issues associated with earlier systems and how they can be adapted or improved to provide more reliable and effective recommendations.

A. BASIC EXTEND VIDEO METADATA

Video metadata expansion has established itself as a significant research topic in streaming services like OTT platforms and YouTube, where keyword extraction and plot summarization are actively used in recommendation systems. This section provides an overview of basic video metadata expansion, introducing key theories and deep learning models that are widely adopted in the field. By discussing these elements, the section aims to illustrate how enhanced metadata can significantly improve the functionality and accuracy of recommendation systems, offering viewers content that is more aligned with their preferences and viewing history.

Video metadata expansion goes beyond extracting basic information from a video to include data derived from within the video itself. It's crucial to extract information from the video as accurately and comprehensively as possible. Current methods often involve processing a video input through multimodal techniques to handle image and voice data simultaneously. Tong et al. [9] introduced a method called videoMAE, which applies the Masked AutoEncoder technique to videos. This method exploits the static nature of adjacent frames by masking certain parts of a video clip and reconstructing the original video clip to derive results. Instead of using many frames, this model is trained with limited data to foster high-level understanding capabilities. While videoMAE optimizes the MAE for video frames, it has the limitation of not recognizing audio information, thus not fully utilizing all available video data. Sun et al. [10] proposed videoBERT, a model for summarizing videos that uses video as input. videoBERT processes video by extracting frames every 1.5 seconds and pre-processing them into image data. The image data then undergoes classification through the S3D embedding method, while the voice data uses captions obtained from the YouTube ASR (Automatic Speech Recognition) API. Consequently, videoBERT performs clustering using S3D embedding without a specific object detection model, allowing for the classification of frame classes. videoBERT is effective in summarizing long-term videos over one minute in length. However, without an object detection model, it can only classify one class per frame, and captions are generated through scripts rather than directly extracting voice from the YouTube ASR, presenting limitations in capturing direct speech nuances.

B. OPERATION OF THE EXISTING RECOMMENDED SYSTEM

Recommendation systems have become a significant research topic across various industries and are extensively utilized in areas such as product recommendations for businesses, OTT platforms, and video streaming services. This section provides an overview of the basic types of recommendation systems and introduces two widely adopted algorithms in the field: content-based filtering and collaborative filtering [11].

Recommendation systems are defined as systems that predict items preferred by users, and understanding the correlations between user and item metadata is essential. The content-based filtering algorithm recommends items that have a high correlation with the items a user has experienced in the past. Javed et al. [12] proposed a system that recommends different content to users based on the content's metadata. A content-based recommendation system suggests content similar to what the user has consumed, using metadata such as content titles, plot summaries, and image data as the basis for recommendations. The operation of content-based filtering recommendation systems is straightforward; however, because they focus only on content the user has already consumed, they often perform less effectively compared to collaborative filtering systems. Bogers and den Bosch [13] suggested using Word2Vec to extract metadata from each content and apply it in a content-based recommendation system. Unlike the traditional TF-IDF approach, this method uses Word2Vec to create semantic vectors from movie plots and utilizes the Jaccard coefficient index to convert movie features into vectors. This approach provides accurate recommendations under cold start conditions. However, while this method can recommend content between items effectively, it has limitations in recommending content directly to users due to its focus on content similarity rather than user behavior or preferences.

Collaborative filtering is a system that recommends items based on the consumption patterns of two users and the correlation between these users. Fu et al. [14] proposed a deep learning model for collaborative filtering that models the interactions between the characteristics of users and items to make recommendations to other users. This approach leverages deep learning rather than traditional cosine similarity-based algorithms, enhancing versatility and improving the representation and generalization of matrix factorization. However, it has limitations in addressing the cold start problem when there is no information available about new items. F. Wang et al. [15] suggested an enhanced collaborative filtering recommendation system that uses basic rating predictions along with user and item information, and trust data. They introduced a method that incorporates user trustworthiness into traditional collaborative filtering to address the cold start problem by supplementing the rating information. This approach effectively mitigates the issue by relying on trust assessments; however, it also faces limitations as the agent assessing user trustworthiness is not fully optimized for each item. This could potentially lead to inaccuracies in recommendations where the user trust metrics are not perfectly aligned with item characteristics.

In this paper, we build upon existing metadata expansion methods and recommendation systems by fully utilizing both image and voice information to construct video metadata. We then propose a new approach that maps this metadata to user and item information for recommendation purposes. This method is expected to extract more diverse and objective information compared to traditional metadata expansion techniques, exploring the potential for more effective recommendations. Thus, this research aims to provide technical advancements and a fresh perspective in the field of video metadata expansion, enhancing the sophistication and utility of recommendation systems. This could lead to significant improvements in how content is personalized and presented to users across various platforms.

III. EXTRACTION OF META-DATA FOR RECOMMENDATION USING KEYWORD MAPPING

Metadata is structured data about data, which is applied to content according to specific rules to effectively find the desired information from a vast amount of data. Metadata for content typically includes information such as location,

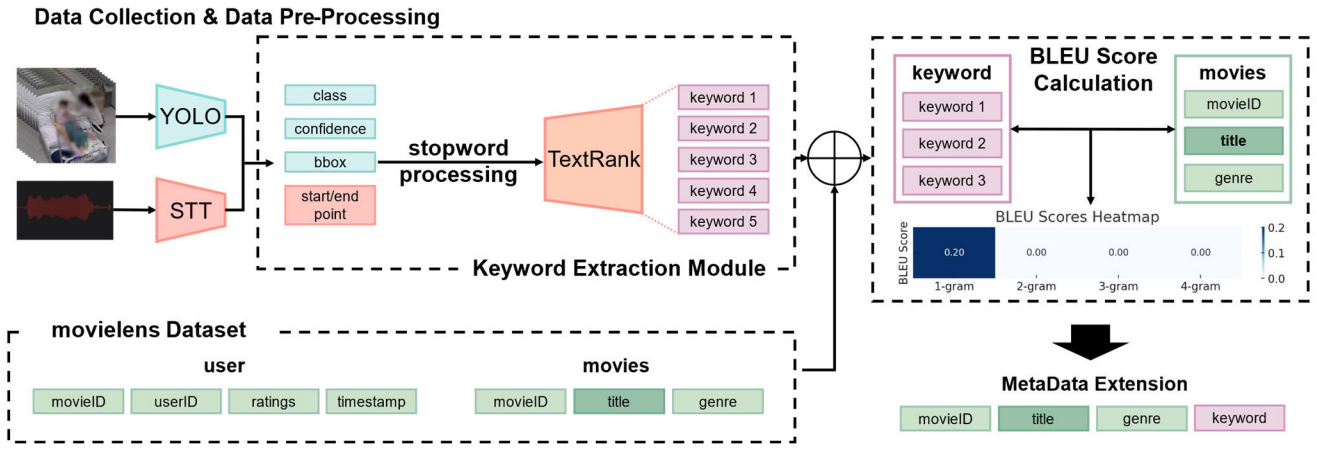

FIGURE 1. Process of extraction of meta-data for recommendation using keyword mapping.

content details, and authorship. However, traditional metadata often lacks sufficient information about the content, resulting in low recall for recommendations. Therefore, there is a need to expand metadata information to enhance recommendation performance. This research progresses in four stages to expand metadata information. The first stage involves constructing the dataset that will be used in the experiments. The second stage is data pre-processing, where constructed data is processed to remove stopwords and extract keywords. In the third stage, the extracted keywords are mapped to the MovieLens dataset. Finally, the fourth stage evaluates the performance of the recommendation system model to verify the effectiveness of the metadata expansion. Figure 1 shows the entire process of the methodology proposed in this paper.

The first stage involves constructing contextualized data by extracting image and voice information from videos using the YOLO model [16] and STT models. Image information is extracted by splitting the video into frames at 30 frames per second and using YOLO to detect the location and class of objects within these frames. Voice information is extracted using Google's Cloud Speech-to-Text API [17] to construct contextualized data. Next, stopwords including formatting words and special symbols are removed. Following this, the TextRank algorithm is used to calculate the frequency of words and extract keywords. TextRank is an algorithm that evaluates the importance of words or sentences within the text by utilizing their relationships. It constructs a graph based on the co-occurrence of words, where the strength of the connections is determined by how frequently the words appear together. Subsequently, keywords are extracted and mapped to expand the metadata based on the 1-gram BLEU Score between movie titles and keywords.

This research validates the superiority of TextRank-based metadata expansion by comparing the performance of recommendation systems using both traditional movie metadata and expanded metadata. Additionally, the paper analyzes to determine if the TextRank approach demonstrates higher performance compared to existing video keyword extraction models like CLIP [18] and deep learning-based models such as videoMAE. This comparative analysis aims to highlight the effectiveness and potential advantages of utilizing advanced text analysis methods over conventional deep learning techniques in the context of enhancing recommendation systems.

A. DATA COLLECTION

The data used in this paper consists of YouTube videos and the MovieLens dataset [19]. The videos are 2 minutes and 22 seconds long and feature multiple individuals. Using the YOLO model, class information of objects, bounding box coordinates [20], and confidence values are extracted. Additionally, Google Cloud Speech-to-Text API is utilized to indicate the frame positions where objects appear. It records the frame where a word is first mentioned as the start_frame and the frame where it is last mentioned as the end_frame. Table 1 displays the contextualized data obtained using YOLO and the Cloud Speech-to-Text API. This contextualized data represents information extracted in chronological order using object detection and voice recognition technologies for each video frame.

The metadata describing the basic settings and characteristics of a video includes key elements such as Rescale, width,

TABLE 1. YOLO, STT based contextual data extraction.

Contextual data
{"METADATA": {"KEY_EXAMPLE": "SOME PROPERTY","OBJ_TO_DETECT": ["PERSON","TV","CELL PHONE", "MOUSE","SPORTS BALL","BED"],"RESCALE": 1.0,"WIDTH": 1920,"HEIGHT": 1080,"FRAME": 1783,"FPS": 30.0},"STT_RESULT": {{"WORD": "내","START_FRAME": 0,"END_FRAME": 264}, "FRAME_3": {"ID": 3,"DETECTEDOBJS": { "PERSON": 1,"TV": 1},"DETAIL": {"OBJECT_1": {"CLASS": "PERSON","LOCATION": [1647.0,794.0,1856.0,1080.0], "CONFIDENCE": 0.7898675203323364}, "OBJECT_2": {"CLASS": "TV","LOCATION": [40.0,1.0, 1767.0,1074.0],"CONFIDENCE": 0.4210226535797119}}}, ...

TABLE 2. Movielens dataset.

movie			ratings			
movieId	title	genre	userId	movieId	ratings	timestamp
1	Toy Story	Adventure|Animation	1	1	4.0	887431973
2	Jumanji	Adventure|Children|Fantasy	1	110	4.0	878542960
3	Grumpier Old Man	Comedy|Romance	2	158	3.0	875071713
…	…	…	…	…	…	…
288971	Oujia Japan	Action|Horror	330975	7340	1.0	883601277
288977	Skinford: Death Sentence	Crime|Thiller	330975	8783	2.5	883599431

height, Frame, and fps. ''Rescale'' indicates the ratio at which data has been resized; a value of 1.0 means that the original size has not been altered, providing information when the size of input data has been adjusted for analyses like object detection. ''Width'' and ''height'' denote the video's resolution, where width refers to the video's width and height to its height, respectively. ''Fps'' (frames per second) represents the playback speed of the video. ''Obj_to_detect'' specifies the types of objects to be analyzed. ''Stt_result'' shows the results of speech-to-text conversion, detailing the start and end frames of spoken words and storing the transcribed words under ''word''. ''Frame_n'' indicates the object detection results for specific frames. ''ID'' denotes the frame ID, and ''DetectedObjs'' lists the types and counts of objects detected in a frame. ''Detail'' includes detailed information about the objects: ''object_1'' and ''object_2'' represent the detected objects, ''class'' specifies the type of object, ''location'' indicates the position within the frame, and ''confidence'' reflects the confidence level in the detection. These metadata elements are crucial for understanding the video content, facilitating more precise analysis and application in fields such as video indexing and recommendation systems.

The MovieLens dataset represents a collection of data where users rate movies. It is available in two versions: a small version and a full version. For the experiment, the full version is used, which includes data on 280,000 users and 58,000 movies. Since metadata from YouTube videos was constructed in the first stage, this expanded information is now integrated with the MovieLens dataset, which contains extensive movie information. Table 2 displays the MovieLens dataset.

The MovieLens dataset primarily consists of two subsets: movies and ratings. The movies dataset includes columns such as 'movieId' which is a unique identifier for each movie, 'title' which is the name of the movie, and 'genre' which distinguishes the different genres of the movies. The ratings dataset contains 'userId', a unique identifier for each user; 'movieId', which links to the unique identifier of the rated movie; 'ratings', which are the scores given by users to movies; and 'timestamp', which records the time when the rating or tag was applied by the user.

B. DATA PRE-PROCESSING

In this research, the data pre-processing stage involves formatting the contextualized data into a suitable form for the TextRank algorithm and extracting keywords. The data pre-processing stage consists of two main steps: the removal of stopwords and the extraction of keywords based on the TextRank algorithm. This process not only cleanses the data but also highlights significant terms that are crucial for enhancing the metadata.

Algorithm 1 represents the keyword extraction process and is described in Section III-B. It shows the steps from word recognition to the extraction of keywords.

Stopword removal is a process that eliminates words of relatively low importance [21]. In this section, the nltk library [22] is utilized for stopword removal, and special symbols are also defined for preprocessing the contextualized data. Special symbols and formulaic language used in the contextual data often appear frequently in texts but do not contribute significantly to understanding the context's meaning. By removing such words, the data size can be reduced, and processing speed can be increased. Additionally, this helps to elevate the importance of key information, facilitating. smoother keyword extraction. Table 3 shows the results of stopword removal from the data presented in Table 2.

TABLE 3. The result of the non-verbal treatment.

RESULTS OF APPLYING PREPROCESSING
CAR
BROCOOLI
POTTED
PLANT
MOTORCYCLE
MOTORCYCLE
PERSON
BROCCOLI
POTTED
PLANT
MOTORCYCLE

TextRank algorithm is used to extract significant phrases or words within a document. In this research, the TextRank algorithm operates in four stages. The first stage involves inputting contextual data that has been processed to remove stopwords, calculating the frequency of each word, and assigning a unique index to each. The second stage involves creating a co-occurrence matrix [23], which displays how frequently words appear together. This involves calculating the co-occurrence count of each word with others within a specified window range around it. The window is set as a hyperparameter to 2, meaning that the relationship between

Algorithm 1 TextRank Keyword Extraction

Input: input_path // Paths to csv files
Output : output_path //

preocedure ScanVocabulary(*objects*, *min_count*)
Count frequency of each object in *objects*
Filter out objects with a count less than *min_count*
Sort objects by Decreasing frequency
Create mappings from objects to indices and vice versa
Print the top 5 frequencies
return *idx_to_vocab*, *vocab_to_idx*

preocedure Cooccurrence(*objects*, *vocab_to_idx*, *min_cooccurrence*)
Initialize a counter for co-occurrences
for each *object* in objects within given window **do**
Increment count for pairs (object, neighboring object)
end for
Filter co-occurrences less than *min_cooccurrence*
Convert the counter to a sparse matrix
Print the top 5 co-occurrence data
return the sparse matrix

Procedure PageRank (x, df, max_iter)
Normalize x matrix
Initialize ranking vector R
for each iteration up to *max_iter* **do**
Update R based on the normalized matrix and damping factor df
end for
Print the top 5 page rank scores
return R

Procedure TextRankKeyword (*objects*, *min_count*, *window*, *min_cooccurrence* *min_cooccurrence*, df, *max_iter*; *topk*)
Extract vocabulary and indices using ScanVocabulary (*vocab_to_idx*, *window*, *min_cooccurrence*)
Create co-occurrence matrix using CooCCURRENCE (*objects*, *vocab_to_idx*, *window*, *min_cooccurrence*)
Calculate page ranks using PageRANK(x, df, *max_iter*)
Extract topk indices from R
Collect keywords and their scores
Print the top 5 keywords
return keywords

each central word and four neighboring words (two on each side) is considered for building the co-occurrence matrix. In the third stage, the generated co-occurrence matrix is used to apply a PageRank function to evaluate the importance of each word. The PageRank algorithm [24] incorporates a damping factor [25] to reflect the probability of each word connecting to other words. This damping factor helps the PageRank to converge and maintains the connectivity among indices in the co-occurrence matrix by linking isolated words. The algorithm is repeatedly executed as specified by the user to compute the final importance scores for each word. The final stage involves selecting the highest-importance words in sequence to derive the final list of keywords. Figure 2 shows the results of the keyword extraction from the co-occurrence matrix, displayed as a heatmap after stopword processing.

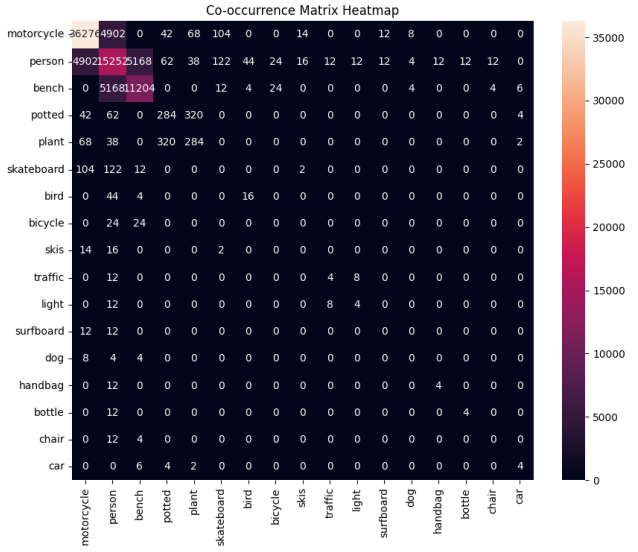

FIGURE 2. Co-occurrence matrix visualization results.

co-occurrence matrix is a matrix used to analyze the frequency with which words appear together within contextualized data, helping to understand their relationships. During the data pre-processing stage, the frequencies and associations of words, which have been assigned indices, are calculated to create the co-occurrence matrix. This matrix is utilized to discover relationships and patterns among items within the data, enhancing understanding of the data. Additionally, it offers the advantage of allowing analysis of interactions and significance among items within the data. Table 4 displays the data after keyword extraction has been completed.

TABLE 4. Keyword extraction results.

TEXTRANK BASED KEYWORD EXTRACTION
MOTORCYCLE
PERSON
BENCH
POTTED
...

Table 4 displays the results after applying the TextRank algorithm to the data processed for stopwords in Table 3, sorting the words by their frequency of occurrence. This is shown similarly to Figure 2, which shows the keywords.

C. META-DATA EXTRACTION USING DATA MAPPING

In this section, we utilize keywords extracted in Section III-B to expand the metadata through mapping with the MovieLens dataset. This process aims to enhance the metadata while

preserving the existing movie dataset. The mapping involves comparing the titles from the movie data with the keywords to expand the metadata. Movie titles, which encapsulate the essence of the entire plot in a single sentence, are suitable for metadata expansion using keywords. Therefore, we first compare the keywords extracted from the videos with the 'title' column in the movie data. If a keyword is found within a title, metadata expansion is carried out. For this metadata expansion, we use the BLEU SCORE [26] to base the similarity of the keywords. BLEU (Bilingual Evaluation Understudy) measures the performance of machine-translated text against human translation by comparing how similar they are. This comparison is based on n-grams [27], which are sequences of 'N' items that occur consecutively in text. For example, in the sentence ''The quick brown fox'', a 1-gram (Unigram) consists of individual items, thus being ''The'', ''quick'', ''brown'' and ''fox''. A 2-gram (Bigram) consists of sequences of two consecutive items, therefore ''The quick'', ''quick brown'' and ''brown fox''.

$$Precision = \frac{the\ number\ of\ Ca\ words\ which\ occur\ in\ any\ Ref}{the\ total\ number\ of\ words\ in\ the\ Ca} \tag{1}$$

Eq. (1) calculates the Precision of Unigram using the reference sentence *Ref* and the translated sentence *Ca*.

BLEU addresses the shortcomings of Unigram Precision by considering the order of words. Additionally, it mitigates the issue of shorter sentences receiving higher scores, as well as the problem of inflated scores when the same words are repetitively used in translation. BLEU achieves this by integrating various n-gram precisions, such as Unigram, Bigram, and Trigram. Furthermore, BLEU penalizes excessively concise translations by deducting points if the translated sentence is significantly shorter than the reference, thus preventing the loss of important information. Lastly, BLEU considers the maximum occurrences of n-grams in the reference to calculate Precision. Its advantages include objectivity, reproducibility, efficiency, and language independence. Since it is automatically computed, it remains objective without being influenced by subjective judgments. Moreover, it ensures reproducibility by consistently providing the same score for identical inputs. BLEU also demonstrates efficiency by facilitating quick and easy evaluations of large datasets, and it maintains language independence as it is not reliant on specific languages or language pairs.

$$BLEU = BP \times \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{2}$$

Eq. (2) represents the formula for calculating the BLEU Score. P_n denotes the n-gram precision, and W_n represents the weight for n-gram precision. Typically, W_n is set to $\frac{1}{N}$. N represents the maximum length of n-grams used. Generally, N is set to 4. Finally, BP denotes the Brevity Penalty, which deducts points if the candidate translation *Ca* is excessively shorter than the reference.

$$BP = \begin{cases} 1 & if\ c > r \\ e^{(1-r/c)} & if\ c \leq r \end{cases} \tag{3}$$

Eq. (3) expresses BPmathematically. Here, c represents the length of *Ca*, and r is the length of *Ref* . This condition prevents overly concise translations from receiving high scores.

Keywords are mapped based on the 1-gram BLEU Score. The reason for using a 1-gram BLEU Score to extract the score is that there are no matching n-grams of two or more words between the sentence-like title and the word-like keyword. The compatibility decreases in n-grams that consider sentence structure or context. For instance, ''personne,'' ''Twilight,'' and ''personnes,'' which were initially extracted from the video as ''person'' and ''light,'' achieve the highest 1-gram BLEU Scores, and therefore, they are mapped accordingly. Figure 3 displays a heatmap that represents the mapping results of movie titles and keywords based on the BLEU Score.

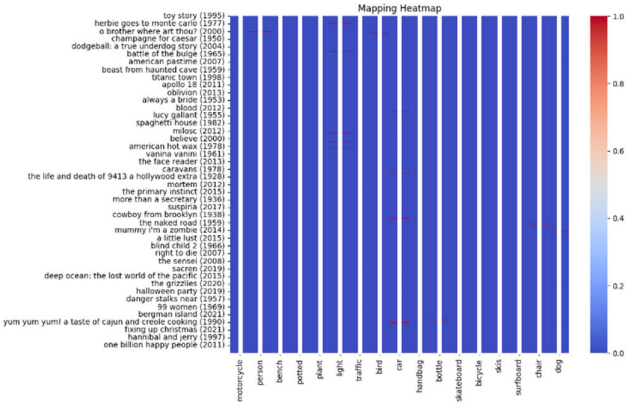

FIGURE 3. Metadata extension visualization results.

Figure 3 shows the results of mapping movie titles to keywords based on 1-gram BLEU Scores. In the heatmap, red bars indicate high similarity, suggesting a successful mapping to the title. Conversely, blue bars represent low BLEU Scores, indicating that the keywords do not map well to the titles. Table 5 displays a part of the movies data with an added 'keyword' column to show the expansion of metadata.

IV. EXPERIMENTAL RESULTS AND COMPARISONS

The metadata expansion technique proposed in this research targets videos. By utilizing both Object Detection on video frames and STT on audio tracks, the method leverages both image and voice information. This comprehensive approach provides a solid foundation for accurately understanding the content of videos.

The metadata expansion method used in this research is divided into two stages. In the first stage, Object Detection is applied to each frame to identify and classify objects within the frame. During this process, object detection models such as YOLO, DETR [28], and Faster R-CNN [29] are utilized to construct contextual data regarding the presence and location

TABLE 5. Keyword mapping results.

movieId	Title	genre	keyword
151030	A poem is a Naked Person	Documentary	person
1104	Street car Named Desire, A	Drama	car
131	Frankie Starlight	Drama|Romance	light
263	Ladybird Ladybird	Drama	bird
64923	Blackbird, The	Crime|Drama	bird
66304	Hotel for Dogs	Adventure|Children |Comedy	dog
277762	A Cat with a Dog	Comedy|Drama	dog
…	…	…	…

of various objects. This visual metadata quantifies the visual elements of video content, enhancing the accuracy of video content understanding and analysis in subsequent processing. Additionally, the suitability of using the YOLO model in the proposed metadata expansion technique is validated through performance evaluations with various state-of-the-art methods. In the second stage, STT conversion is performed on the audio track of the video to transform audio content into text metadata. High-performance STT models such as Google Cloud STT, Conformer [30], and Citrinet [31] are used to extract textual information from the audio tracks. The transformed text metadata is then utilized to aid in the understanding of the video's audio content. Performance evaluation is conducted using the WER (Word Error Rate) metric, which serves as a crucial benchmark to assess the practicality of Google Cloud STT in extracting voice information.

In this section, we conduct a performance evaluation of the experimental results. The experiments are carried out on a system equipped with an Intel(R) Core(TM) i7-9700 Processor, NVIDIA RTX A4000, 64GB RAM, and running Ubuntu 20.04.6 with Python 3.10.13. Additionally, the experiments utilize the torch library version 2.1.0. Table 6 shows the experimental and implementation environment.

TABLE 6. Metadata with keyword column added from title.

sortation	environment
CPU	Intel(R) Core(™) i7-9700 Processor
GPU	NVIDIA RTX A4000
RAM	64GB RAM
OS	Ubuntu 20.04.6
Language	Python 3.10.13
Develop Tool	Visual Studio Code
Library	torch 2.1.0 etc

A. OBJECT DETECTION MODEL FOR EXPERIMENTS

The proposed video metadata expansion is based on Object Detection and STT. Accordingly, a performance evaluation is conducted to determine the most suitable Object Detection model to use for extracting accurate information from videos.

The dataset used for evaluating the performance of Object Detection is the COCO 2017 Test. COCO (Common Objects in Context) [32] is one of the datasets widely used in the field of Object Detection within Computer Vision. COCO includes everyday objects in various scenes and provides extensive annotations for various vision tasks such as object recognition, segmentation, and captioning. Additionally, its high-quality annotations and hundreds of thousands of images make it useful for assessing the generalizability of models.

For evaluating the performance of Object Detection, the metrics used are Precision, Recall, and F1-Score [33]. Precision represents the ratio of correctly detected objects among the predicted objects. Recall indicates the number of actual objects that the model has correctly detected. The F1-Score, which is not mentioned in your input but is commonly used in conjunction with Precision and Recall, is the harmonic mean of Precision and Recall. This measure provides a balance between Precision and Recall, offering a more comprehensive evaluation of the overall accuracy of the model.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

Eq. (4) shows the process of calculating Precision using True Positives (TP) and False Positives (FP). TP refers to the count of cases where the model correctly predicts a positive result for actual positive instances. FP refers to the count of cases where the model incorrectly predicts a positive result for actual negative instances.

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

Eq. (5) shows the process of calculating Recall using True Positives (TP) and False Negatives (FN). FN refers to the count of instances where the actual condition is positive, but the prediction is negative.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{6}$$

Eq. (6) shows the process of calculating the F1-Score, which is derived from Precision and Recall. Since Precision and Recall often have a trade-off relationship, the F1-Score is used to provide a balance between the two. The F1-Score is the harmonic mean of Precision and Recall, representing a balance between these metrics and thus allowing for an assessment of the overall performance of a model. Table 7 displays the performance evaluation results of the Object Detection model.

The performance evaluation results in Table 7 show that the F1-Score for the YOLOv8-n model is 0.63, indicating that it outperforms other object detection models in terms of

TABLE 7. Object detection model experiments result.

	Precision	Recall	F1-Score
DETR	0.548	0.619	0.58
Faster R-CNN	0.457	0.559	0.5
YOLOv8-n	**0.535**	**0.768**	**0.63**

overall effectiveness. In Table 8, while the Precision of the DETR model is higher than that of YOLOv8-n, the Recall for YOLOv8-n is significantly higher than that for DETR. Therefore, the F1-Score of YOLOv8-n is measured to be higher than that of DETR. This suggests that while YOLOv8-n can detect more objects, it also has a higher incidence of false positives. Table 8 shows a comparison of the detection speeds between YOLOv8-n and other object detection models being considered.

TABLE 8. Object detection rate by object detection model.

	YOLOv8-n	DETR	FASTER R-CNN
Speed	**0.4ms**	0.6ms	0.7ms

The performance evaluation results in Table 8 show that YOLOv8-n has the fastest object detection speed among the models tested, with a detection time of 0.4ms. The speed of object detection varies depending on the method used. YOLOv8-n processes the entire image through a single neural network, focusing on speed and efficiency. DETR, a Transformer-based model, provides an End-to-End approach to object detection and is better at understanding the relationships between objects. Faster R-CNN uses a Region Proposal Network (RPN) to first identify Regions of Interest (ROI) and then perform object detection on each region. There are two main types of object detection models: One-Stage Detection Models and Two-Stage Detection Models [34]. Two-Stage Detection Models like Faster R-CNN first identify areas in the image where objects are likely to be found. It then classifies the object's class for each identified ROI and determines the precise bounding boxes. However, One-Stage Detection Models like YOLOv8-n perform object location identification and class classification in a single step. Since it processes the entire image at once, it is faster than Two-Stage Detection Models. DETR, based on the Transformer model, considers every location within an image to detect objects, which results in relatively higher computational complexity compared to other models. Consequently, YOLOv8-n has the fastest object detection speed.

In metadata expansion, object detection is used to extract image information from videos, thus the YOLOv8-n model is chosen for its highest F1-Score. Additionally, since it is more important for the recommendation system to match predictions with actual results, YOLOv8-n is also selected for its highest recall. For real-time recommendations, a fast model is necessary, making YOLOv8-n the preferred choice due to its speed.

B. SPEECH RECOGNITION MODEL FOR EXPERIMENTS

Voice recognition in videos is used alongside object detection as a crucial source of information. Videos consist not just of continuous image frames, but also of audio. If the results from object detection in image frames contradict the audio content, it can pose a problem. Therefore, to select an appropriate STT model, a performance evaluation of STT models is conducted at this stage. This ensures that the selected STT model accurately translates audio content, aligning it effectively with the visual data extracted through object detection.

For evaluating the performance of Speech-to-Text models, the LibriSpeech test-clean dataset is used. LibriSpeech is a large-scale, open-source speech recognition dataset designed for the training and evaluation of ASR systems [35]. The dataset consists of voice data extracted from audiobook recordings provided by LibriVox, a public domain book project where volunteers read books that are in the public domain and turn them into audiobooks. The dataset includes over 1000 hours of audio recordings, featuring a diverse range of voices across different ages (from 20s to 50s), genders, and accents. The test-clean subset of LibriSpeech includes only those recordings with clear enunciation, making it an ideal set for testing the clarity and accuracy of STT models.

For evaluating the performance of Speech Recognition systems, the metric used is WER, which stands for Word Error Rate [36]. WER measures the difference between the text generated by the Speech Recognition model and the actual text. It calculates the Error Rate to assess the accuracy of the system. This metric is essential in determining how effectively a speech recognition system transcribes spoken words into written text, considering errors made in the process.

$$Word\ Error\ Rate = (D + S + I)/N \tag{7}$$

Eq. (7) describes the calculation of the WER. D represents the number of words incorrectly deleted from the speech recognized text, S is the number of words incorrectly substituted, and I represents the number of words incorrectly inserted in the speech recognized text. N denotes the total number of words in the reference text. The closer the *WER* value is to 0, the higher the performance, indicating a more accurate speech recognition. A *WER* value of 1 means that every word was recognized incorrectly. Therefore, a lower *WER* indicates better recognition accuracy.

TABLE 9. Performance evaluation by speech recognition model.

	Google Cloud STT	Conformer	Citrinet
WER	**0.0766**	0.2473	0.3592

The performance evaluation results presented in Table 9 show that Google Cloud STT has a lower WER compared to other models. Google Cloud STT is a cloud-based voice recognition service offered by Google, which has been trained using massive datasets and robust machine learning models. Continuous optimization and updates have steadily improved its performance, and it supports a wide variety of languages and dialects, enhancing its recognition accuracy. Conformer, a modern speech recognition model, combines the advantages of CNN and Transformer models. Although it is highly accurate, it differs from Google Cloud STT in terms of the volume and diversity of training data and ongoing

optimization efforts. Lastly, Citrinet is an End-to-End learning model that can be trained with comparatively less data and offers fast inference speeds. However, Citrinet also faces challenges regarding data volume and model optimization. Therefore, it is evident that Google Cloud STT performs the best among the models evaluated.

In metadata expansion, STT processing does not take as much time as object detection, so the focus should be on accuracy rather than recognition speed. Therefore, the Google Cloud STT model, which has the lowest Average WER, is used.

C. COMPARISON OF KEYWORD EXTRACTION MODELS FOR METADATA EXPANSION

To evaluate the generalization performance of the metadata expansion method proposed in this paper, a comparative analysis is conducted with deep learning-based keyword extraction models, CLIP and VideoMAE. This analysis demonstrates the relative superiority of the proposed method. Table 10 displays the results of this comparative analysis, using RMSE (Root Mean Square Error), MAE (Mean Absolute Error), and NDCG (Normalized Discounted Cumulative Gain) as the key performance indicators. For objective performance evaluation, the batch size is standardized at 200, and k is set to 5

RMSE is the square root of the average of the squares of the differences between actual and predicted values. It represents the magnitude of prediction errors. A smaller RMSE value indicates better model prediction performance, and the squaring operation within RMSE gives greater weight to larger errors.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{8}$$

Eq. (8) represents the calculation of RMSE. In this equation, y represents the actual values, and $\hat{y}$ represents the predicted values. n denotes the total number of evaluation items. MAE is the average of the absolute differences between the actual values and the predicted values. Like RMSE, MAE indicates the magnitude of prediction errors, with smaller values indicating better model prediction performance.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{9}$$

Eq. (9) represents the calculation of MAE. Unlike RMSE, MAE involves an absolute value operation, which means it assigns equal weight to all errors, regardless of their size. This makes MAE particularly useful in contexts where all errors are considered equally important, providing a simple and intuitive measure of average error magnitude.

NDCG@k is a ranking-based evaluation metric that assigns higher scores when more relevant items are placed at the top of the list. NDCG is calculated as the ratio of the actual ranking's Discounted Cumulative Gain (DCG) to the ideal ranking's DCG.

$$NDCG@k = \frac{DCG@k}{IDCG@k} \tag{10}$$

Eq. (10) represents the calculation of NDCG@k. DCG@k is the DCG value for the top k items, and IDCG@k is the DCG value in an ideal scenario for the top k items. An NDCG value closer to 1 indicates a higher performance of the model.

TABLE 10. Performance evaluation results based on the presence or absence of keywords.

	RMSE	MAE	NDCG@k
CLIP	0.9147	0.7220	0.9048
VideoMAE	0.9140	0.7210	0.9056
ours	**0.8626**	**0.6816**	**0.9057**

Table 10, the proposed method consistently shows higher performance across all evaluation metrics compared to the CLIP and VideoMAE models. The proposed method's RMSE is 0.8626, slightly better than CLIP's 0.9147 and VideoMAE's 0.9140. It also records the lowest MAE scores at 0.7220, 0.7210, and 0.6816 respectively. Moreover, NDCG@k is the highest at 0.9057, indicating that the proposed method is more effective at reducing error rates and enhancing relevance in recommendation systems compared to existing models.

These results validate that the proposed metadata expansion approach is a significant method for enhancing the performance of recommendation systems. While advanced models like CLIP and VideoMAE also deliver outstanding performance, the proposed method specifically focuses on using metadata to better understand user preferences and improve the accuracy of item recommendations. This can be a crucial factor in maximizing the relevance and satisfaction of content delivered to users in recommendation systems, thereby enhancing the overall user experience.

D. COMPARISON OF RECOMMENDED SYSTEM PERFORMANCE WITH OR WITHOUT METADATA EXPANSION

The central aim of this research is to improve the accuracy and efficiency of recommendation systems through metadata expansion after extracting keywords from contextualized data. Therefore, we compare the performance of the recommendation system with and without the application of expanded metadata. In this paper, we employ the GLocal-K [37] and GHRS [38] models to actively utilize rating information and content metadata. GLocal-K is a recommendation system model with an AutoEncoder structure, capable of reducing a sparse and high-dimensional rating matrix to a lower dimension. GHRS utilizes graph-based features for making recommendations. Figure 4 shows the operational process of the recommendation systems used for performance evaluation. Table 11 shows the performance evaluation of the method proposed in this paper and the existing method.

According to the performance evaluation results in Table 11, when metadata expansion using keywords is

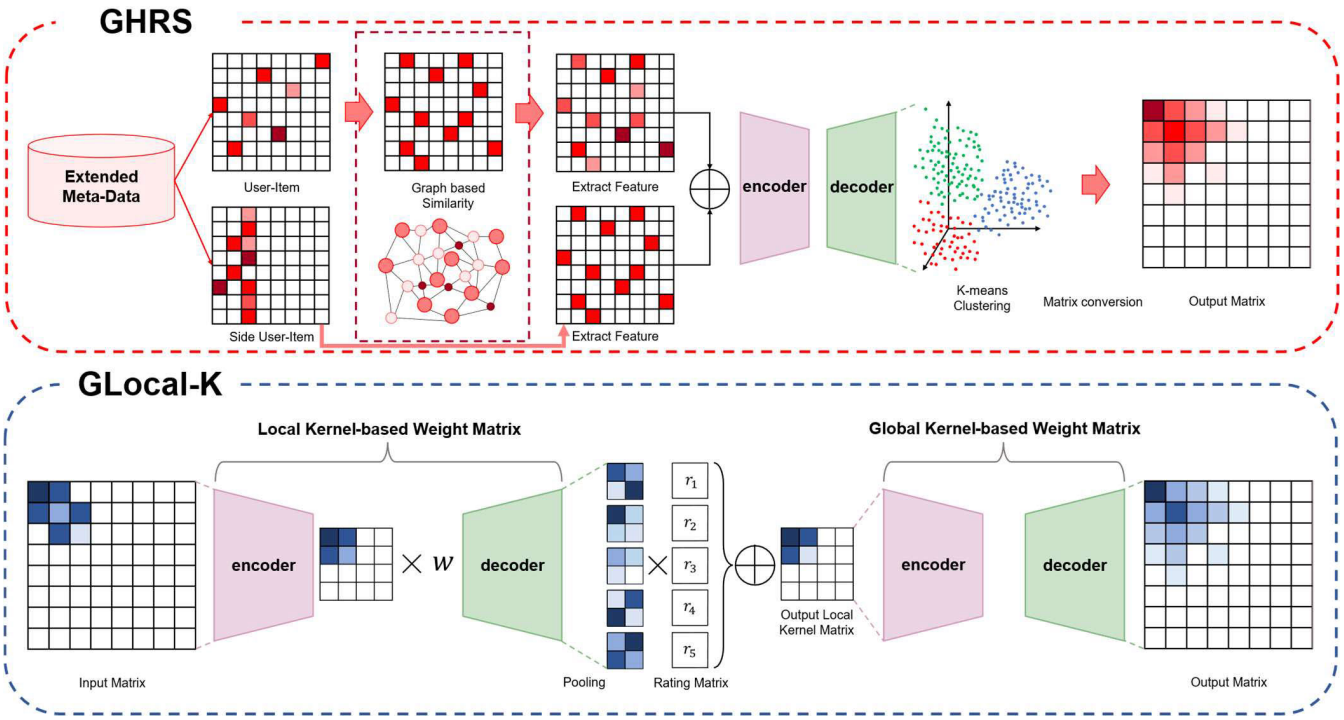

FIGURE 4. Performance evaluation for another recommendation system model.

TABLE 11. Performance evaluation results of the recommended model.

	Expanded Meta-data	RMSE	MAE	NDCG
GHRS	Non-Expanded	1.5704	1.3074	0.9955
	Expanded(ours)	**0.9544**	**0.7499**	**0.9945**
GLocal-K	Non-Expanded	0.9710	0.7721	0.8950
	Expanded(ours)	**0.9072**	**0.7135**	**0.9011**

TABLE 12. Recommendation results of the recommendation system with the proposed expanded metadata.

Input	Output		
keyword	genre	title	movieId
Person	Documentary	A Poem is a Naked Person	151030
Calr	Drama	Street car Named Desire, A	1104
Light	Drama|Romance	Frankie Starlight	131
Bird	Drama	Ladybird Ladybird	263
	Crime|Drama	Blackbird, The	64923
Dog	Adventure|Children|Comedy	Hotel for Dogs	66304
	Comedy|Drama	A Cat with a Dog	277762
…	…	…	…

applied, RMSE decreases from 0.9710 to 0.9072 and MAE from 0.7721 to 0.7135, while NDCG increases from 0.8950 to 0.9011. This clearly demonstrates the positive impact of metadata expansion including keywords on the performance of recommendation systems. It is also evident from the GHRS model, like the GLocal-K model, that metadata expansion has a positive effect. These results suggest that metadata expansion through additional information like keywords can play a crucial role in enhancing the accuracy of recommendation systems. By incorporating metadata expansion that includes keywords, the model can more precisely understand users' preferences and interests, and based on this, recommend more relevant items. Therefore, to improve the performance of recommendation systems and increase user satisfaction, active application and research into metadata expansion methods are necessary. Table 12 shows the recommendation results of the Recommendation System with the proposed expanded metadata.

The recommendation results in Table 12 are like the mapping results in Table 5. The n-gram BLEU Score between the title and keyword is highest when it is a 1-gram. For example, in Table 12, if a user inputs the keyword "dog" and shows interest in the Adventure genre, the system recommends the title "Hotel for Dogs" or the movieId 66304.

V. CONCLUSION

This research explored an approach of building contextualized data through deep learning models from video data and using extracted keywords for recommending other videos through a Hybrid Recommendation System. The goal was to achieve higher performance compared to existing deep learning-based keyword extraction methods. The methodology developed in this paper has proven effective in efficiently analyzing video data and extracting information. This signifies that the dynamic characteristics of videos can be successfully utilized to structure image and audio information over time. Using the proposed methodology, performance exceeded that of traditional methods using the Movielens dataset and deep learning-based keyword extraction. Additionally, the methodology presented in this paper showed better performance than when only applying the Hybrid Recommendation System with the existing Movielens dataset, thereby addressing the cold start problem. This suggests that expanded metadata based on keywords effectively enhances recommendations in a Hybrid Recommendation System.

This paper introduced a new perspective and methodology for video metadata extraction by using YOLO and Google's Cloud Speech-to-Text API models to create contextualized data and proposing keyword extraction based on TextRank. Additionally, it applied a recommendation system based on the interaction between movie data and rating data, addressing issues with existing recommendation systems. The superiority and effectiveness of the methodology adopted in this research were proven through performance comparisons based on keyword extraction methods. This demonstrates that the approach can be used to efficiently extract substantial information from videos and generate keywords effectively. This research presents a novel approach in the field of video metadata expansion and will serve as a foundation for future research and applications in this area.

Future research will proceed as follows: First, instead of using object detection models to construct contextual data, we plan to explore the use of pose estimation models that can infer the specific actions being performed by objects. This approach aims not merely to extract information about objects from videos but to build contextual data based on the behavior of objects, thereby enhancing its quality. Secondly, we plan to develop a methodology for mapping keywords to the Movielens dataset based on genres rather than titles. This would improve the performance of the Recommendation System by mapping based on the plot of the movie, rather than simply evaluating similarity based on titles. Lastly, we intend to build a search engine suitable for this methodology and conduct experiments with it. This will explore new possibilities to enhance the versatility of this methodology, potentially broadening its application across different domains.

REFERENCES

[1] I. Safder, S.-U. Hassan, A. Visvizi, T. Noraset, R. Nawaz, and S. Tuarob, "Deep learning-based extraction of algorithmic metadata in full-text scholarly documents," *Inf. Process. Manag.*, vol. 57, no. 6, Nov. 2020, Art. no. 102269, doi: 10.1016/j.ipm.2020.102269.

[2] S. Kumar, K. De, and P. P. Roy, "Movie recommendation system using sentiment analysis from microblogging data," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 4, pp. 915–923, Aug. 2020, doi: 10.1109/TCSS.2020.2993585.

[3] J.-C. Kim and K.-Y. Chung, "Knowledge expansion of metadata using script mining analysis in multimedia recommendation," *Multimedia Tools Appl.*, vol. 80, nos. 26–27, pp. 34679–34695, Nov. 2021, doi: 10.1007/s11042-020-08774-0.

[4] W. Han and M. Ansingkar, "Discovery of elsagate: Detection of sparse inappropriate content from kids videos," in *Proc. Zooming Innov. Consum. Technol. Conf. (ZINC)*, Novi Sad, Serbia, May 2020, pp. 46–47, doi: 10.1109/ZINC50678.2020.9161808.

[5] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Patras, "Video summarization using deep neural networks: A survey," *Proc. IEEE*, vol. 109, no. 11, pp. 1838–1863, Nov. 2021, doi: 10.1109/JPROC.2021.3117472.

[6] Y. Wei, X. Wang, Q. Li, L. Nie, Y. Li, X. Li, and T.-S. Chua, "Contrastive learning for cold-start recommendation," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 5382–5390, doi: 10.1145/3474085.3475665.

[7] J. Chen, H. Dong, X. Wang, F. Feng, M. Wang, and X. He, "Bias and debias in recommender system: A survey and future directions," *ACM Trans. Inf. Syst.*, vol. 41, no. 3, pp. 1–39, Feb. 2023, doi: 10.1145/3564284.

[8] M. Zhang, X. Li, S. Yue, and L. Yang, "An empirical study of TextRank for keyword extraction," *IEEE Access*, vol. 8, pp. 178849–178858, 2020, doi: 10.1109/ACCESS.2020.3027567.

[9] Z. Tong, Y. Song, J. Wang, and L. Wang, "VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training," in *Proc. NIPS*, 2022, pp. 10078–10093.>.

[10] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "VideoBERT: A joint model for video and language representation learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7463–7472, doi: 10.1109/ICCV.2019.00756.

[11] L. V. Nguyen, T.-H. Nguyen, and J. J. Jung, "Content-based collaborative filtering using word embedding: A case study on movie recommendation," in *Proc. Int. Conf. Res. Adapt. Convergent Syst.*, Oct. 2020, pp. 96–100, doi: 10.1145/3400286.3418253.

[12] U. Javed, K. Shaukat, I. A. Hameed, F. Iqbal, T. M. Alam, and S. Luo, "A review of content-based and context-based recommendation systems," *Int. J. Emerg. Technol. Learn. (iJET)*, vol. 16, no. 3, pp. 274–306, Feb. 2021, doi: 10.3991/ijet.v16i03.18851.

[13] T. Bogers and A. van den Bosch, "Recommending scientific articles using citeulike," in *Proc. ACM Conf. Recommender Syst.*, Oct. 2008, pp. 287–290, doi: 10.1145/1454008.1454053.

[14] M. Fu, H. Qu, Z. Yi, L. Lu, and Y. Liu, "A novel deep learning-based collaborative filtering model for recommendation system," *IEEE Trans. Cybern.*, vol. 49, no. 3, pp. 1084–1096, Mar. 2019, doi: 10.1109/TCYB.2018.2795041.

[15] F. Wang, H. Zhu, G. Srivastava, S. Li, M. R. Khosravi, and L. Qi, "Robust collaborative filtering recommendation with user-item-trust records," *IEEE Trans. Computat. Social Syst.*, vol. 9, no. 4, pp. 986–996, Aug. 2022, doi: 10.1109/TCSS.2021.3064213.

[16] C. H. Kang and S. Y. Kim, "Real-time object detection and segmentation technology: An analysis of the YOLO algorithm," *JMST Adv.*, vol. 5, nos. 2–3, pp. 69–76, Sep. 2023, doi: 10.1007/s42791-023-00049-7.

[17] A. B. Wahyutama and M. Hwang, "Performance comparison of open speech-to-text engines using sentence transformer similarity check with the Korean language by foreigners," in *Proc. IEEE Int. Conf. Ind. 4.0, Artif. Intell., Commun. Technol. (IAICT)*, Bali, Indonesia, Jul. 2022, pp. 97–101, doi: 10.1109/IAICT55358.2022.9887500.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, and S. Agarwal, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2021, pp. 8748–8763.

[19] F. M. Harper and J. A. Konstan, "The MovieLens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 1–19, Dec. 2015, doi: 10.1145/2827872.

[20] X.-T. Vo and K.-H. Jo, "Accurate bounding box prediction for single-shot object detection," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 5961–5971, Sep. 2022, doi: 10.1109/TII.2021.3138336.

[21] D. J. Ladani and N. P. Desai, "Stopword identification and removal techniques on TC and IR applications: A survey," in *Proc. 6th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, Coimbatore, India, Mar. 2020, pp. 466–472, doi: 10.1109/ICACCS48705.2020.9074166.

[22] M. Wang and F. Hu, "The application of NLTK library for Python natural language processing in corpus research," in *Proc. Theory Pract. Lang. Stud.*, vol. 11, no. 9, pp. 1041–1049, Sep. 2021, doi: 10.17507/TPLS.1109.09.
[23] H. Wu, Q. Zhou, R. Nie, and J. Cao, "Effective metric learning with co-occurrence embedding for collaborative recommendations," *Neural Netw.*, vol. 124, pp. 308–318, Apr. 2020, doi: 10.1016/j.neunet.2020.01.021.
[24] J. Cheriyan and G. P. Sajeev, "An improved PageRank algorithm for multilayer networks," in *Proc. IEEE Int. Conf. Electron., Comput. Commun. Technol. (CONECCT)*, Bangalore, India, Jul. 2020, pp. 1–6, doi: 10.1109/CONECCT50063.2020.9198566.
[25] Z. HaoLin, H. Jin, and L. WeiKai, "PageRank algorithm based on dynamic damping factor," in *Proc. Int. Conf. Cyber-Physical Social Intell. (ICCSI)*, Xi'an, China, Oct. 2023, pp. 381–386, doi: 10.1109/iccsi58851.2023.10303849.
[26] S. Chauhan, P. Daniel, A. Mishra, and A. Kumar, "AdaBLEU: A modified BLEU score for morphologically rich languages," *IETE J. Res.*, vol. 69, no. 8, pp. 5112–5123, Sep. 2023, doi: 10.1080/03772063.2021.1962745.
[27] H. Choi, J. Lee, and J. Yang, "N-gram in Swin transformers for efficient lightweight image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, Jun. 2023, pp. 2071–2081, doi: 10.1109/cvpr52729.2023.00206.
[28] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229, doi: 10.1007/978-3-030-58452-8_13.
[29] P. Bharati and A. Pramanik, "Deep learning techniques-R-CNN to mask R-CNN: A survey," in *Proc. CIPR*, Aug. 2019, pp. 657–668, doi: 10.1007/978-981-13-9042-5_56.
[30] P. Guo, F. Boyer, X. Chang, T. Hayashi, Y. Higuchi, H. Inaguma, N. Kamo, C. Li, D. Garcia-Romero, J. Shi, J. Shi, S. Watanabe, K. Wei, W. Zhang, and Y. Zhang, "Recent developments on espnet toolkit boosted by conformer," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Toronto, ON, Canada, Jun. 2021, pp. 5874–5878, doi: 10.1109/ICASSP39728.2021.9414858.
[31] S. J. Chun, J. B. Park, H. Ryu, and B. S. Jang, "Development and benchmarking of a Korean audio speech recognition model for clinician-patient conversations in radiation oncology clinics," *Int. J. Med. Inform.*, vol. 176, Aug. 2023, Art. no. 105112, doi: 10.1016/j.ijmedinf.2023.105112.
[32] M. Carranza-García, J. Torres-Mateo, P. Lara-Benítez, and J. García-Gutiérrez, "On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data," *Remote Sens.*, vol. 13, no. 1, p. 89, Dec. 2020, doi: 10.3390/rs13010089.
[33] G. Kim and K. Chung, "ViT-based multi-scale classification using digital signal processing and image transformation," *IEEE Access*, vol. 12, pp. 58625–58638, Apr. 2024, doi: 10.1109/ACCESS.2024.3389808.
[34] D. K. Sharma, "Information measure computation and its impact in MI COCO dataset," in *Proc. 7th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, vol. 1, Coimbatore, India, Mar. 2021, pp. 1964–1969, doi: 10.1109/ICACCS51430.2021.9441788.
[35] S. Alharbi, M. Alrazgan, A. Alrashed, T. Alnomasi, R. Almojel, R. Alharbi, S. Alharbi, S. Alturki, F. Alshehri, and M. Almojil, "Automatic speech recognition: Systematic literature review," *IEEE Access*, vol. 9, pp. 131858–131876, 2021, doi: 10.1109/ACCESS.2021.3112535.
[36] T. von Neumann, C. Boeddeker, K. Kinoshita, M. Delcroix, and R. Haeb-Umbach, "On word error rate definitions and their efficient computation for multi-speaker speech recognition systems," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Rhodes Isalnd, Greece, Jun. 2023, pp. 1–5, doi: 10.1109/ICASSP49357.2023.10094784.
[37] S. C. Han, T. Lim, S. Long, B. Burgstaller, and J. Poon, "GLocal-K: Global and local kernels for recommender systems," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manag.*, Oct. 2021, pp. 3063–3067, doi: 10.1145/3459637.3482112.
[38] Z. Z. Darban and M. H. Valipour, "GHRS: Graph-based hybrid recommendation system with application to movie recommendation," *Exp. Syst. Appl.*, vol. 200, Aug. 2022, Art. no. 116850, doi: 10.1016/j.eswa.2022.116850.

GEON-WOO KIM is currently pursuing the bachelor's degree with the Division of Computer Science and Engineering, Kyonggi University, South Korea. He was a Researcher with the Data Mining Laboratory, Kyonggi University. His research interests include data mining, big data, anomaly detection, and deep learning.

WOO-HYEON KIM is currently pursuing the bachelor's degree with the Division of AI Computer Science and Engineering. Kyonggi University, South Korea, since 2022. He was a Researcher with the Data Mining Laboratory, Kyonggi University. His research interests include data mining, big data, and anomaly detection.

KYUNGYONG CHUNG received the B.S., M.S., and Ph.D. degrees from the Department of Computer Information Engineering, Inha University, South Korea, in 2000, 2002, and 2005, respectively. He was with the Software Technology Leading Department, Korea IT Industry Promotion Agency (KIPA). From 2006 to 2016, he was a Professor with the School of Computer Information Engineering, Sangji University, South Korea. Since 2017, he has been a Professor with the Division of AI Computer Science and Engineering, Kyonggi University, South Korea. He was named, in 2017, as a Highly Cited Researcher by Clarivate Analytics. His research interests include data mining, artificial intelligence, healthcare, knowledge systems, HCI, and recommendation systems.

JOO-CHANG KIM received the B.S. and M.S. degrees from the School of Computer Information Engineering, Sangji University, South Korea, in 2014 and 2016, respectively, and the Ph.D. degree from the Department of Computer Science, Kyonggi University, South Korea, in 2021. Since 2021, he has been a Research Professor with the Contents Convergence Software Research Institute, Kyonggi University. His research interests include data mining, data management, knowledge systems, machine learning, deep learning, big data, healthcare, and recommendation systems.