Toward Automation in Text-based Video Retrieval with LLM Assistance

Supplementary Material

6. Specification of Prompts Used

6.1. Temporal-Assisted Module

This prompt is designed to support the temporal analysis of video descriptions by leveraging LLM capabilities. It consists of two main components: 1. Temporal Classification, and 2. Temporal Segmentation.

6.1.1. Temporal Classification

This component determines whether a given input describes a single continuous shot or multiple distinct shots in a video. If it is identified as *"Single shot"*, no further action is taken. If classified as *"Multiple shots"*, the description proceeds to the segmentation stage.

```
Prompt = """
Given the following input, determine whether it describes a single shot or multiple shots in a video.
If it describes multiple shots, identify the transition words or phrases that indicate the scene change.

Input: "{description}"

Output format:
- "Single shot" if it only describes one scene.
- "Multiple shots" if it describes more than one scene, along with the transition indicators.

Output:
"""
```

6.1.2. Temporal Segmentation

Activated only when the input is classified as *"Multiple shots"*, this component decomposes the description into exactly *two* coherent shot segments. It ensures that shared context is preserved across both segments while removing redundant transition phrases or structural indicators (e.g., "a series of scenes"). The goal is to produce standalone and meaningful descriptions for each segment to facilitate downstream analysis.

```
Prompt = """
Given the following video description, separate it into exactly TWO meaningful shots.
- Preserve shared context at the beginning of both shots while removing shot numbering, transitional phrases, or any introductory phrases indicating multiple shots (e.g., 'A sequence of shots')
- Ensure that each shot reads naturally as a standalone description.
- If the description contains more than two shots, merge them logically into two coherent segments without losing any critical contextual information.

### Input:
"{description}"

### Output format:
{
    "shot_1": "First shot description",
    "shot_2": "Second shot description"
}
Ensure the response is in strict JSON format without additional text.
"""
```

6.2. Query Refinements Module

This prompt is designed to restructure input queries to optimize them for search engines. The goal is to produce a version that is highly searchable, concise, and rich in keywords, while remaining clear and informative.

```
Prompt = """
You are an AI specialized in optimizing video descriptions for top search rankings. Your goal is to generate a *highly searchable, concise, and keyword-rich* version of the given video description.

### Optimization Rules:
1. *Start with the most relevant keywords* (e.g., key actions, objects, and locations).
2. *Remove unnecessary details* while keeping the core meaning.
3. *Structure in a way that fits common search patterns* (e.g., noun + action + location).
4. *Ensure readability and search relevance * without sacrificing clarity.
5. *Use active voice* and keep the sentence *concise yet informative*.

### Example:
*Input Description:*
"A man with two boys entering a comic book store, they are greeted by the owner.
```

```
    People inside and outside the store
    cheering. A group of boys wearing blue
    Dodgers and Royals shirts. The store own
     wears black and has grey hair."

*Optimized Search-Friendly Output:*
"Man and two boys enter comic book store,
    greeted by grey-haired owner in black.
    Cheering crowd inside and outside store.
     Boys in blue Dodgers and Royals shirts
    present."

Now, generate an optimized search-friendly
    description for the following video:
{description}
"""
```

6.3. Results Reranking Module

This module uses a prompt-based approach to rerank retrieved images based on their relevance to the input query. A language model is prompted to assess each image on a scale from 0 to 10, considering content similarity, visual clarity, and contextual alignment. The model outputs a single numeric relevance score per image.

```
Prompt = """
On a scale of 0 to 10, where 10 means '
    highly relevant' and 0 means 'not
    relevant at all,' how relevant is this
    image to the query? Consider factors
    such as content similarity, visual
    clarity, and contextual alignment.
    Respond with a single numeric score only
    .
<image>
Query: {input_query}
"""
```

6.4. QA-Description Separation

The prompt separates the description and the question from a query, with the description containing details about actions and scenes, and the question including hints and the question itself. The result is output in JSON format for easy processing.

```
Prompt = """
Identify and separate the description and
    the question from the following passage:

{text}

- *Description*: Only includes details
    describing the video without any
    questions.
- *Question*: Includes both the hint in the
     video leading to the question and the
    actual question. This part usually
    starts with words like "Who", "Whom", "
    Which", "What", "When", "Where", "Why",
    "How" or contains a "?".

Please provide the response in JSON format:
{
    "description": "<extracted description
        >",
    "question": "<extracted question along
        with any relevant hint>"
}
"""
```