ILS-SUMM: Iterated Local Search for Unsupervised Video Summarization

Yair Shemer
Technion

Daniel Rotman
IBM Research

Nahum Shimkin
Technion
**Task Definition:**
Given an input video, automatically create a short video that summarizes the original video.

- In General, the goal is to enable faster consumption of video data.
Example of a video summary:

Original Video (38 Sec)

Summary Video (5 Sec)
Common video summarization algorithm scheme:
Problem Formulation

- Given an input video $v$, the video is divided temporally into a set of shots $S_v = \{s_1, s_2, \ldots s_N\}$
- We denote the duration in seconds of a shot $s$ as $t(s)$
- Each shot is represented by its middle frame feature vector $x(s)$

**Knapsack Median (KM) Optimization Problem**

$$S^*_{\text{summ}} = \arg\min_{S_{\text{summ}} \subseteq S_v} TD(S_{\text{summ}}|S_v, x(s)),$$

subject to:

$$\sum_{s \in S_{\text{summ}}} t(s) \leq T,$$

where:

$$TD(S_{\text{summ}}|S_v, x(s)) = \sum_{s' \in S_v} \min_{s \in S_{\text{summ}}} \{\text{dist} \, (x(s'), x(s))\}$$
Iterated Local Search (ILS) framework (Baxter, 1981)

1. Apply a local-search algorithm

2. Repeat:
   1. Perturb the current local minimum.
   2. Apply a local-search after starting from the modified solution.
ILS Perturbation Mechanism:
The perturbation we used: swap the $M$ costly medoids with the $M$ cheapest non-medoids.
The perturbation we used: swap the $M$ costly medoids with the $M$ cheapest non-medoids
Datasets:

- **SumMe** - 25 short user videos (Gygli et al., 2014)
- **TvSum** - 50 short user videos (Song et al., 2015)
- **Open Source Total Distance (OSTD)** - 18 movies of various lengths

For shot segmentation we use **KTS** (Potapov et al., 2014) and **FFprobe** Python tool (Python Software Foundation, 2019).
Total Distance optimality percentage, i.e., the ratio between the optimal value and the achieved value:

<table>
<thead>
<tr>
<th></th>
<th>SumMe</th>
<th>TVSum</th>
<th>OSTD</th>
</tr>
</thead>
<tbody>
<tr>
<td>DR-DSN</td>
<td>90.78%</td>
<td>82.50%</td>
<td>62.56%</td>
</tr>
<tr>
<td>Submodular</td>
<td>85.18%</td>
<td>94.14%</td>
<td>95.99%</td>
</tr>
<tr>
<td><strong>ILS-SUMM</strong></td>
<td><strong>98.48%</strong></td>
<td><strong>99.27%</strong></td>
<td><strong>98.38%</strong></td>
</tr>
</tbody>
</table>

**ILS-SUMM** exhibits a significant advantage
Experiments: Video Summarization

ILS-SUMM selections on Cosmos Laundromat movie:

<table>
<thead>
<tr>
<th>Frame Number</th>
<th>Frame Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><img src="image" alt="Frame 0" /></td>
</tr>
<tr>
<td>8</td>
<td><img src="image" alt="Frame 8" /></td>
</tr>
<tr>
<td>16</td>
<td><img src="image" alt="Frame 16" /></td>
</tr>
<tr>
<td>24</td>
<td><img src="image" alt="Frame 24" /></td>
</tr>
<tr>
<td>32</td>
<td><img src="image" alt="Frame 32" /></td>
</tr>
<tr>
<td>40</td>
<td><img src="image" alt="Frame 40" /></td>
</tr>
<tr>
<td>48</td>
<td><img src="image" alt="Frame 48" /></td>
</tr>
<tr>
<td>56</td>
<td><img src="image" alt="Frame 56" /></td>
</tr>
<tr>
<td>64</td>
<td><img src="image" alt="Frame 64" /></td>
</tr>
<tr>
<td>72</td>
<td><img src="image" alt="Frame 72" /></td>
</tr>
<tr>
<td>80</td>
<td><img src="image" alt="Frame 80" /></td>
</tr>
<tr>
<td>88</td>
<td><img src="image" alt="Frame 88" /></td>
</tr>
<tr>
<td>96</td>
<td><img src="image" alt="Frame 96" /></td>
</tr>
<tr>
<td>104</td>
<td><img src="image" alt="Frame 104" /></td>
</tr>
</tbody>
</table>
Conclusion: 

1. Using deep learning is not always the right choice: for optimization problem, heuristic algorithms can work better. 
2. Utilizing ILS framework leads to great solutions thanks to the perturbation mechanism.

Future Directions: 

1. Find a better objective function. 
2. Relax the “hard” segmentation.
Conclusion and Future Directions

Conclusion:

1. Using deep learning is not always the right choice: for optimization problem, heuristic algorithms can work better.
2. Utilizing ILS framework leads to great solutions thanks to the perturbation mechanism.

Future Directions:

1. Find a better objective function.
2. Relax the “hard” segmentation.
Conclusion and Future Directions

Conclusion:

1. Using deep learning is not always the right choice: for optimization problem, heuristic algorithms can work better.
2. Utilizing ILS framework leads to great solutions thanks to the perturbation mechanism.

Future Directions:

1. Find a better objective function.
2. Relax the “hard” segmentation.
Conclusion and Future Directions

Conclusion:

1. Using deep learning is not always the right choice: for optimization problem, heuristic algorithms can work better.
2. Utilizing ILS framework leads to great solutions thanks to the perturbation mechanism.

Future Directions:

1. Find a better objective function.
2. Relax the “hard” segmentation.
Conclusion and Future Directions

Conclusion:

1. Using deep learning is not always the right choice: for optimization problem, heuristic algorithms can work better.
2. Utilizing ILS framework leads to great solutions thanks to the perturbation mechanism.

Future Directions:

1. Find a better objective function.
2. Relax the “hard” segmentation.
Thanks!