# WHU-NERCMS AT TRECVID2022: DEEP VIDEO UNDERSTANDING TASK

**Ankang Lu**[†]**, Jiahao Guo**[†]**, Kuangyi Zhao**[†]**, Yingfei Sun**[†]**, Ruizhe Li, Chao Liang**[*]
Hubei Key Laboratory of Multimedia and Network Communication Engineering
National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University
cliang@whu.edu.cn

## ABSTRACT

We will make a brief introduction of the experimental methods and results of the WHU-NERCMS in the TRECVID2022 [1] in paper. This year we participate in the task of Deep Video Understanding (DVU). The DVU task supports two tracks, movie-level track and scene-level track, and we take part in both of them. We adopt a four-stage method including video structuralization, instance retrieval and the generation of the knowledge graph. After that, the four kinds of queries in both movie-level track and scene-level track are given answers by modified Cypher language for the specific questions. In addition, we make attempts to use refined methods to further improve search performance. We submit 2 runs for movie-level track and scene-level track respectively. The result of each run is shown in Table 1. The official evaluations show that the proposed strategies rank $3^{rd}$ in scene-level and rank $2^{nd}$ in movie-level tracks.

Table 1: Result of each run

| **Type** | **Run_ID** | **Percentage** | **Strategy** |
|---|---|---|---|
| Movie-level | Run_1 | 28.9% | refined |
| | Run_2 | 9.6% | pretrained |
| Scene-level | Run_1 | 11.1% | refined |
| | Run_2 | 3.1% | pretrained |

## 1 Introduction

The task of TRECVID DVU in 2022 is given a wholly original movie (e.g 1.5 - 2hrs long), image snapshots of main entities (persons, locations, and concepts) per movie, and ontology of relationships, interactions, locations, and sentiments used to annotate each movie at the global movie-level (relationships between entities) as well as on

[†] These authors contributed equally to this work.
[*] Corresponding author.

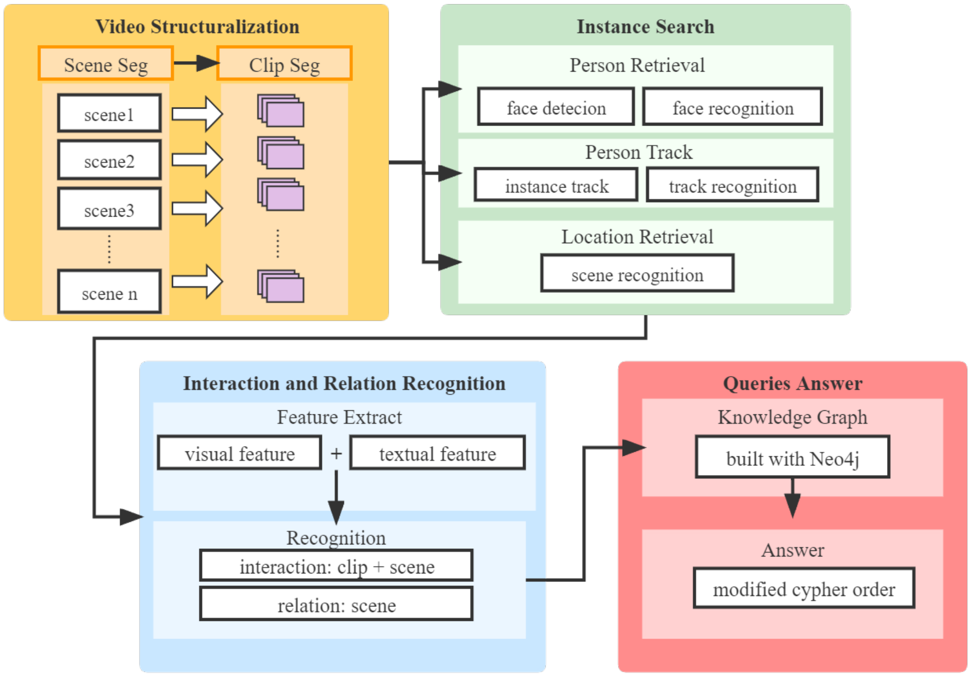


Figure 1: Our framework

fine-grained scene-level (scene sentiment, interactions between characters, and locations of scenes) [2, 3], systems are expected to generate a knowledge-base of the main actors and their relations (such as family, work, social, etc) over the whole movie, and of interactions between them over the scene level [4]. This representation can be used to answer a set of queries on the movie-level and/or scene-level per movie. Movie track where participants are asked queries on the whole movie level, and scene track where queries are targeted towards specific movie scenes.

## 2 Our Method

As shown in Fig. 1, the framework we proposed for DVU task consists of four parts. The first one is the video structuralization module, which includes auto speech recognition (ASR) using Youtube API and clip segmentation. The second part is the instance search module, which includes person recognition and track using SCRFD [5], ArcFace [6], faster RCNN [7] and Deepsort [8], as well as locations recognition using ResNet. The third module is the interaction and relation recognition module for recognizing the interaction within people and people and the interaction between people and location in scene-level. It also recognizes the relations between characters in movie-level. The fourth module is the knowledge graph module, which adapts the results from the interaction and relation recognition module to generate knowledge graphs for each movie to answer the queries.

### 2.1 Video Structuralization

The first step of our method is the structuralization of the movies. It contains two parts, the auto speech recognition and video segmentation. For each movie, we use Youtube ASR API to generate the subtitles and the corresponding time stamps, it can also locate the clip with speech of characters to filter out the clip with no sound. The Video Segmentation contains two kinds, scene segmentation and clip segmentation. For scene segmentation, we can segment the movie using officially provided time stamps by ffmpeg instruction or download the segmented video files directly from the official website. For clip segmentation, based on the generated subtitle file, the start and end time of each clip are obtained, and the corresponding scene to which clip belongs is located by combining the clip segmentation with the scene segmentation. For the clips which are between two segmented scenes, we add them to the scene which has longer a coincident video. Through this way can we generate the mapping table from scene to clip and the unit for the next process.

### 2.2 Instance Search

The second step of our method is the instance search module. It contains two parts, the recognition and track for the person and the recognition for the location. It is for finding the active and passive entities in a clip to recognize their relation or interaction.

#### 2.2.1 Person Recognition and Track

For characters in the movie, we run face detection using SCRFD frame by frame on the scene to get to bounding box of the face and we use ArcFace to extract the face features of the face library and the features of the detected face in frame-level, Then we calculate the similarity between features from face library and datected face to save the frame-level similarity matrix, it can confirm the identity of the detected face in the scene. The face library is generated from the entity images offered officially, we average the different image features of the same person for identifying. After face recognition, we run person track on the scene using faster RCNN and Deepsort and save the original tracking information. At last, we use the track sequence of characters and the face detection result on each frame to predict the identity of each tracking sequence, and concatenate different sequences with consistent identity into one sequence. We also use HOI to distinguish if the bounding-boxs of face and of body belong to the same identity. Based on the character track sequence on the scene and the start and end time of the clip, the track sequence on each clip is generated and saved as video for later using.

#### 2.2.2 Location Recognition

The location recognition is similar to person recognition, we extract the visual features of the location library and the feautures of location by ResNet clip by clip, then we calculate the similarity on clip-level to obtain the location identity in the clip. Considering there is interactions between person and location, for example, a person must cook in the kitchen, we define person and location as the same importance in interaction recognition. The location library is generated from the entity images offered officially, we choose the location images and we average the feature of different images for the same location to build our location library.

### 2.3 Interaction Relation Retrieval

The third step is the retrieval of interaction in scene-level and relation in movie-level. We adopt a similar structure of joint learning with Kukleva[9] for interaction recognition, and we adopt a single MLP for relation recognition.

#### 2.3.1 Multimodal Features

We use multi-modal features for the retrieval module, including visual features and text features as the scene feature to predict relationships and interactions between characters and locations.

**Visual features** We use TSM [10] to extract vectors of 2048 dimensions as visual features. For each clip, we need to extract features of the whole scene and for each combination of two track sequences. For example, if two persons and one location are recognized in the clip, there should be 6 different combinations of track features as there is active and passive relationships in relation or interaction.

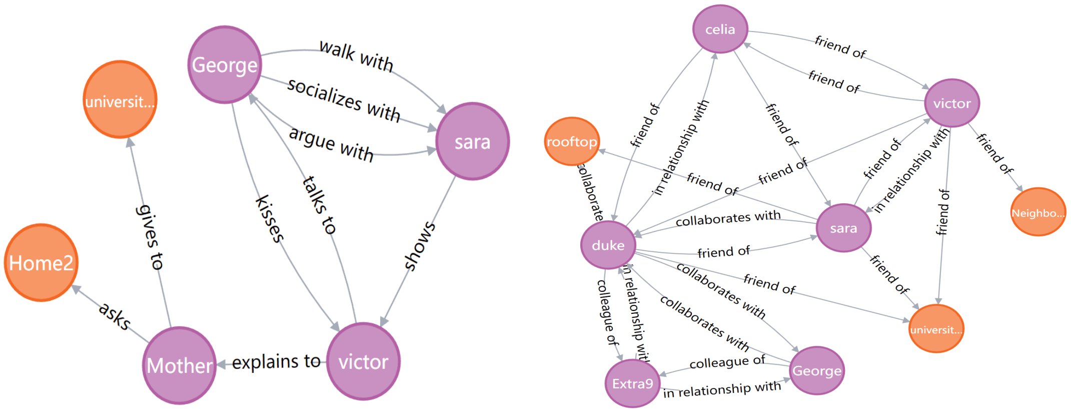
**Text features** For each clip, we use pre-trained BERT-base [11] to convert sentences to vectors of 768 dimensions as text features, then we gather all the clip text feature belongs to the same scene as the scene text feature. After we extract all the features in the scene, we concatenate the visual feature and text feature as the input of the module for interaction and relation recognition.

#### 2.3.2 Interaction and Relation Recognition

As mentioned in the task, the interaction is measured in terms of clip. Therefore, we unit the multi-model clip feature with the average of scene feature as the interaction in a scene has context-aware. We encode the feature and calculate the similarity to the target interaction to get the most possible interaction in the clip between the selected two persons or locations. For the recognition of relation, we average the features of each scene as the input to predict the relationship between two persons.

### 2.4 Knowledge Graph

For each movie, we can get a list of predicted relations and interactions respectively in a movie, the results also contain the possibilities for each relation and interaction. The we use the Neo4j as the graph data platform to generate the knowledge graph of a movie. The interactions and relations are saved in the same graph for each movie. The graph is shown in Fig. 2. There are two kinds of nodes, the person node and location node, which have the same status. In interaction part, the connecting line between nodes is the recognized interaction, the direction of arrow point to the passive entity from the active entity, it also contains the possibility of the predicted interaction. In relation part, the nodes are the same to the interaction part, the connecting line between nodes in the recognized relation with its possibility. In run 2 we just use the pretrained model of Kukleva [9], and in run 1 we train the model with our annotated data. For the interaction annotation, we look through the scene-level knowledge graph of the train set to locate where it happens and who does it. For the relation annotation, we look through the movie-level knowledge graph of the train set to see what relation is between the two entities from a scene. the result shows the validity of our training strategy.



(a) Interaction part

(b) Relation part

Figure 2: Knowledge graph of the movie

### 2.5 Queries Answer

There are two kinds of queries in both movie-level track and scene-level track, one is selecting an answer from given options, the other is submitting all the possible answers with their possibilities. For "Question Answering (QA)", we solve them by traversing all the relations in the relation part of the knowledge graph and finding the nodes that satisfy the right relation line in the graph and averaging the possibilities along the relation line. For "fill in the graph space", we solve them by finding the relation line between the character nodes mentioned in the queries. For "find next or previous interaction", we solve them in a similar way to "Question Answering", the difference is that we look through the interaction part of the knowledge graph instead of relation. For "find the unique scene", it is a little different from previous queries, we generate another knowledge graph for each movie that only contains nonredundant interactions. Then we traverse the unique knowledge graph to answer the queries.

## 3 Analysis

Fig. 3 and Fig. 4 show the movie-level track and scene-level track for all the test movies this year. The nodes on the line are accuracy percentages of our submitted runs and the best score of each submission from all entrants. Obviously, our trained model performs better than the pre-trained model which verifies that staying the same with the official namespace is good for relation understanding. But in scene-level track, our trained model is worse than the pre-trained model, it is understandable that we use a part of the action types in the official namespace, and the lack of annotated data also donates to the performance. By comparing our submission with the ground truth, we find that our model always regard the action "talks to" as the action "asks", we think there are two main reasons. For one, we lack information on audio, the two actions can be distinguished by the sound intensity. For the other, we don't have enough annotations for the two action types that our model cannot define them accurately. It is shown that in scene-level track, our submission is close to the best score but there is distance for our submission and the best score in movie-level track.

Fig. 5 and Fig. 6 show the count of recognized action and relation for each movie. As shown in the picture, the most recognized action is "socializes with" and the most recognized relation is "friend of". Both the two words have

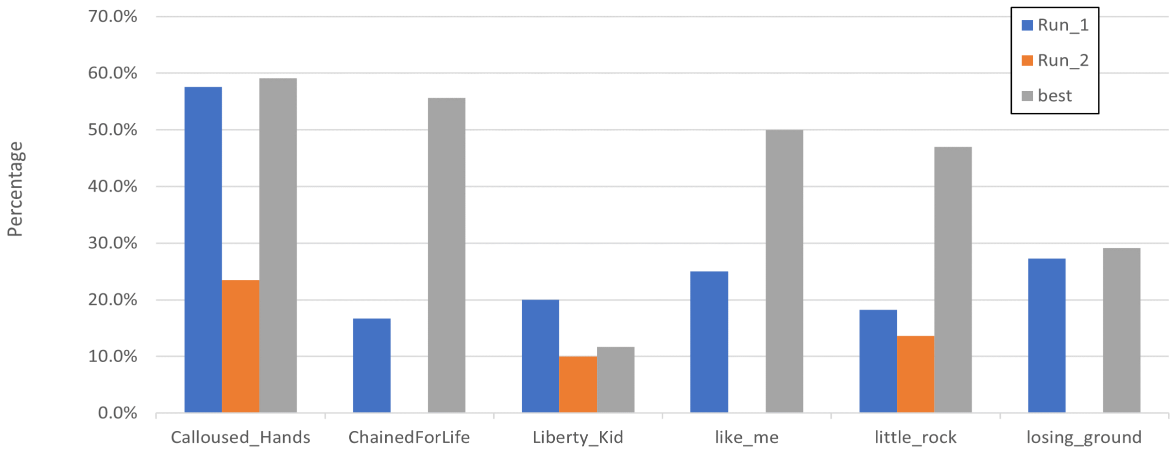


Figure 3: Movie-level Accuracy

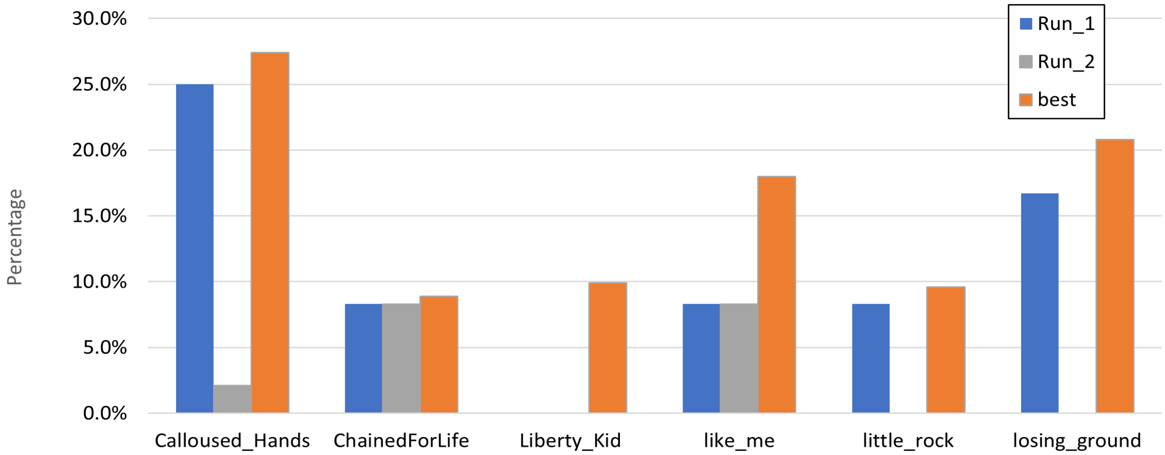


Figure 4: Scene-level Accuracy

fuzzy definitions that are hard to tell the difference from other similar words. The least recognized action is “asks” and the least recognized relation is “lives at”. For “asks”, the result shows that we often regard it as “talks to”, but the evidence shows that our annotation of “asks” is lacking and ambiguous. For “lives at”, the passive entity is restricted to locations, which means our annotation for the interaction between person and location is not identifiable enough.

## 4 Conclusion

Through the DVU task in TRECVID 2022, we conduct extensive experiments for our framework. By using advanced models and re-fined strategies, our submission achieves rank $3^{rd}$ in scene-level and rank $2^{nd}$ in movie-level tracks.

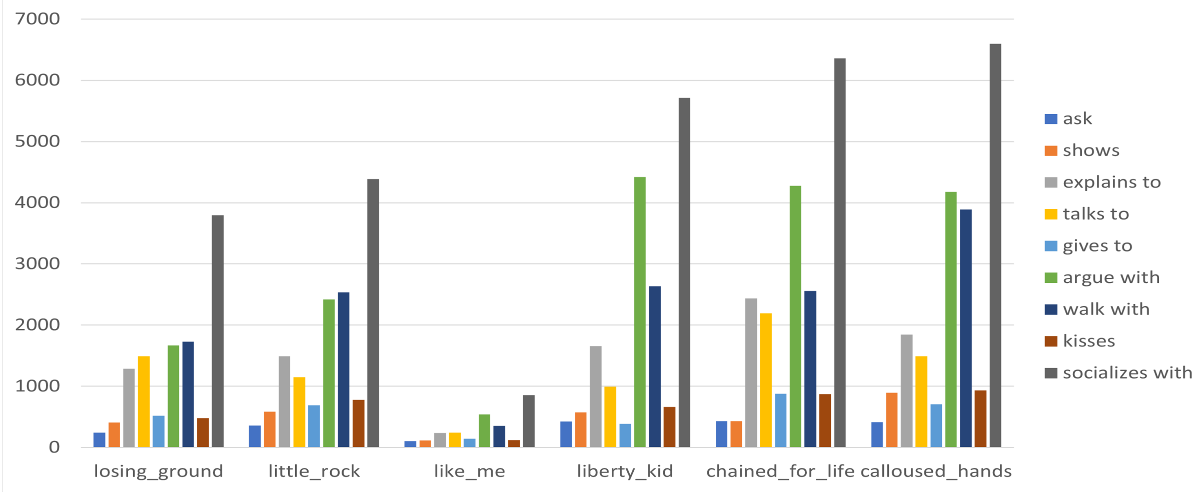


Figure 5: Action Recognition

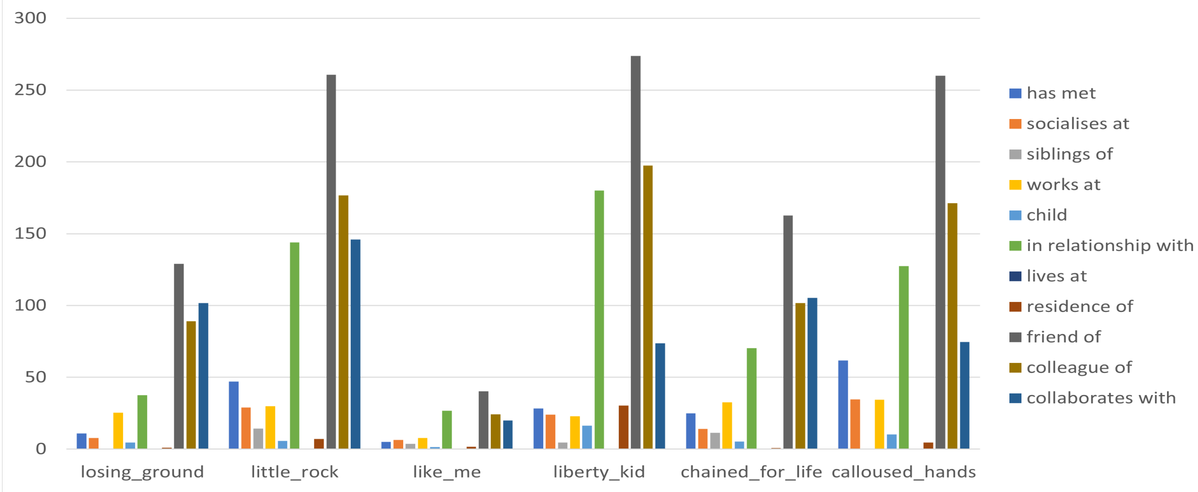


Figure 6: Relation Recognition

## Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. U1903214, 61876135, 62072026), Beijing Natural Science Foundation (No. JQ20022), the open research fund of the State Key Laboratory for Management and Control of Complex Systems (No. 20220103), and the open project of Anhui Provincial Key Laboratory of Multimodal Cognitive Computation, Anhui University (No. MMC202103). The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

## References

[1] George Awad, Keith Curtis, Asad A. Butt, Jonathan Fiscus, Afzal Godil, Yooyoung Lee, Andrew Delgado, Jesse Zhang, Eliot Godard, Baptiste Chocot, Lukas Diduch, Jeffrey Liu, Yvette Graham, , and Georges Quénot. An overview on the evaluated video retrieval tasks at trecvid 2022. In *Proceedings of TRECVID 2022*. NIST, USA,

2022.

[2] Erika Loc, Keith Curtis, George Awad, Shahzad Rajput, and Ian Soboroff. Development of a multimodal annotation framework and dataset for deep video understanding. In *Proceedings of the 2nd Workshop on People in Vision, Language, and the Mind*, pages 12–16, 2022.

[3] Keith Curtis, George Awad, Shahzad Rajput, and Ian Soboroff. Hlvu: A new challenge to test deep understanding of movies the way humans do. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pages 355–361, 2020.

[4] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and trecvid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[5] Ming Chen, Peng Du, and Jieyi Zhao. Scrfd: Spatial coherence based rib fracture detection. In *Proceedings of the 2018 5th International Conference on Biomedical and Bioinformatics Engineering*, pages 105–109, 2018.

[6] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

[7] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[8] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017.

[9] Anna Kukleva, Makarand Tapaswi, and Ivan Laptev. Learning interactions and relationships between movie characters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9849–9858, 2020.

[10] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7083–7093, 2019.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.