# Video Captioning

## Datasets

### MSR-VTT Dataset

MSR-VTT \cite{xu2016msr} is a massive video captioning dataset consisting of 10,000 videos and 200,000 video clip-sentence pairs. The dataset is designed for video captioning tasks, which involves generating natural language descriptions of video content. Each video is around 20 seconds long on average, and has multiple human-generated descriptions or captions that describe the actions, objects, and concepts present in the video. The dataset is diverse in terms of content, containing videos from various domains such as sports, entertainment, and cooking.

MSR-VTT can be leveraged for a variety of downstream tasks in natural language processing (NLP) and computer vision, including video captioning, video summarization, video retrieval, and text-to-video synthesis. Video captioning models can be trained on this dataset to generate natural language descriptions for new videos. Video summarization models can use this dataset to identify key frames or moments in a video to create a shorter summary of the content. Video retrieval models can use captions as a query to retrieve videos that are semantically similar to the query. Text-to-video synthesis models can use this dataset to generate videos from natural language descriptions. The large size and diversity of the dataset make it a valuable resource for research in video understanding and natural language processing.

### MSVD Dataset

MSVD (Microsoft Research Video Description) \cite{ chen2011collecting } is a dataset of short video clips that are annotated with one or more natural language sentences that describes the content of the clip. It contains 1970 video clips, with an average duration of approximately 10 seconds, which are annotated with 41,408 sentences. The dataset covers a wide range of categories, such as sports, cooking, and music, and was designed to evaluate the performance of automatic video description systems.

The MSVD dataset can be used for several downstream tasks in video understanding, including:

1. Video captioning: given a video clip, generate a natural language sentence that describes its content.
2. Video retrieval: given a natural language query, retrieve relevant video clips that match the query.
3. Video summarization: create a shorter version of a longer video clip that captures its main content and conveys the same message.
4. Activity recognition: classify the activity being performed in a video clip, based on its visual content and associated natural language sentence.
5. Multimodal fusion: combine information from both the visual content and natural language sentence to improve performance on downstream tasks such as video captioning and retrieval.

The MSVD dataset has been widely used in research on video understanding and natural language processing, and has led to significant advances in video captioning and retrieval systems.

### YouCook2 Dataset

The YouCook2 dataset \cite{ zhou2018towards} is a huge video dataset of 2,000 cooking videos with more than 89,000 video segments. The dataset includes a variety of recipes from multiple cuisines, and the videos are captured from a first-person perspective, allowing for a natural and immersive experience. Each video is accompanied by a recipe text, which includes a list of ingredients and instructions for preparing the dish.

The dataset can be used for various downstream tasks, including video summarization, video captioning, and recipe generation. Video summarization aims to create a shorter version of the original video by selecting and stitching together the most important segments. Video captioning involves generating natural language descriptions of the videos. Recipe generation is a novel task that involves generating recipes from the video and text data.

### VATEX Dataset

The VATEX dataset \cite{ wang2019vatex} is a large-scale video-and-text dataset for language understanding in video. It consists of 41,250 video clips with a total duration of 155 hours, which are collected from YouTube. Each video is associated with multiple language descriptions, including English, Chinese, and Spanish, that are not translations of each other. In total, there are 825,124 human translations with an average of 20 translations per video.

The dataset is mainly designed for the task of video captioning, which aims to generate natural language descriptions of videos automatically. Besides video captioning, the dataset can also be used for other downstream tasks such as video summarization, machine translation, and video retrieval.

To facilitate research on video captioning, the dataset is further divided into training, validation, and test sets. The training set consists of 25,990 videos with 516,712 captions, the validation set consists of 3,000 videos with 60,000 captions, and the test set consists of 12,260 videos with 244,622 captions. The test set is also divided into two sub-sets: public test and private test. The public one is used for online evaluation, while the private test set is used for final evaluation and its results are not disclosed. The dataset is challenging due to its large scale, multi-lingual nature, and diverse video content.

### ActivityNet Captions Dataset

ActivityNet Captions \cite{ krishna2017dense} is a large-scale video captioning dataset, which contains over 20,000 videos and around 100,000 captions. The dataset is designed for the task of video captioning, where the goal is to generate a natural language description of a video.

Each video in the dataset is annotated with multiple captions that describe the activities and events happening in the video. The dataset is divided into three parts: training, validation, and testing. The training part has around 10,000 videos, while the validation and testing sets are comprised of around 4,500 videos each.

The videos in the dataset come from a diverse set of sources, including YouTube, Flickr, and Vimeo, and contain a wide range of activities, like cooking, dancing, sports, and more. The captions are written by human annotators and are designed to capture the salient aspects of the video.

The dataset can be used for various downstream tasks, including video captioning, video summarization, and activity recognition. The dataset has been used in several research studies to develop state-of-the-art methods for the task of video captioning, including methods that use attention mechanisms, multimodal embeddings, and language models.

### HowTo100M Dataset

The HowTo100M dataset \cite{ miech2019howto100m} is a massive video dataset with 1.22 million instructional videos, collected from YouTube, covering a wide range of daily activities such as cooking, repair, grooming, and crafting. The videos are annotated with timestamps, categories, and free-form textual descriptions of each action in the video.

The dataset can be used for various downstream tasks, such as action recognition, temporal localization, and instructional video retrieval. The large scale of the dataset allows for effective training of deep learning models, and the wide range of activities covered ensures a diverse set of examples for model training.

Action recognition models trained on the HowTo100M dataset have shown state-of-the-art performance on several benchmark datasets, including Kinetics and Something-Something V2. The dataset has also been used for instructional video retrieval, where models are trained to retrieve instructional videos given natural language queries. The dataset's annotations make it a valuable resource for developing and evaluating natural language processing and computer vision models.

### WebVid Dataset

The WebVid dataset \cite{ghadiyaram2020webvid} is a huge video dataset that has 1.5 million video clips and spans over 3,500 hours of video content. The dataset contains multiple topics, including news, sports, entertainment, and more. The videos were collected from a variety of sources, including YouTube, Vimeo, and Dailymotion.

The WebVid dataset is mainly intended for video retrieval and recommendation tasks. It can also be used for video classification and segmentation tasks.

One of the key features of the WebVid dataset is its diversity, which makes it suitable for training models that can generalize well to unseen data. The dataset also includes a large number of short video clips, which can be used for tasks that require quick processing, such as online video recommendations.

Overall, the WebVid dataset is a valuable resource for researchers working on video-related tasks, such as video retrieval, recommendation, classification, and segmentation.

Table below summarizes the datasets related information in terms of total videos, total hours of videos, average length per clip, average sentence length and words per clip, etc.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dataset Name** | **Video time (hours)** | **No. of videos** | **No. of clips** | **Avg. Clip length**  **(sec)** | **Sent. Per clip**  **(avg)** | **Total No. of sentences** | **Avg. sent. Length (unique words)** | **Benchmarks**  **(Papers that have**  **used this dataset)** |
| MSR-VTT \cite{ xu2016msr } | 41 | 1k | 1k | 4.1 | 20 | 20,000 | 290 | \cite{xu2017r},\cite{xu2023mplug},\cite{wang2022git},\cite{yan2022video}, \cite{ye2022hitea} |
| MSVD \cite{ chen2011collecting } | 4.13 | 1.97k | 1.97k | 10 | 1 | 1970 | // | \cite{ venugopalan  2015sequence},\cite{xu2023mplug},\cite{ye2022hitea},\cite{yang2023vid2seq},\cite{zhang2020object} |
| YouCook2 \cite{ zhou2018towards} | 31.5 | 2k | 2k | 88.3 | 8.3 | 16,600 | 13.2 | \cite{zhou2018towards}, cite{luo2020univl},\cite{xu2021vlm},\cite{yan2022video},\cite{huang2020multimodal} |
| VATEX \cite{ wang2019vatex} | 1300 | 40k  41,250 | 40k | 20 | 10 | 800k  826k | 12.4 English, 17.6 chineese | \cite{ Wang\_2020\_CVPR},\cite{ yan2022video },\cite{ zhang2020object },\cite{ singh2020nits } |
| ActivityNet Captions \cite{ krishna2017dense } | 849 | 20k | 20k | 200 | 3.65 | 100k | 13.48 | \cite{ gao2018video },\cite{yamazaki2022vltint},\cite{yamazaki2022vlcap},\cite{yan2022video},\cite{ging2020coot} |
| HowTo100M \cite{ miech2019howto100m } | 136,116 | 1174,541 | // | 7.5 | 0.22 | 262,500 | 16 | \cite{ zhukov2020cross },\cite{alayrac2020self},\cite{liu2021interactive},\cite{sun2020csn} |
| WebVid \cite{ghadiyaram2020webvid} | 3500 | 1.5M | 1.5M | 5.5 | 1.7 | 2,710,212 | 10.6 | \cite{ bain2021frozen} |

## Video Captioning benchmark scores analysis table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Dataset** | **Metric** | **Models score from highest to lowest** | | | | |
| **MSR-VTT** | CIDEr | 80.3 \cite{xu2023mplug} | 75.9 \cite{wang2022git} | 73.2 \cite{yan2022video} | 65.1 \cite{ye2022hitea} | 64.6 \cite{yang2023vid2seq} |
| BLEU-4 | 57.8 \cite{xu2023mplug} | 54.8 \cite{wang2022git} | 53.8 \cite{yan2022video} | 49.2 \cite{ye2022hitea} | 48.9\  cite{seo2022end} |
| METEOR | 38.7\cite{ seo2022end } | 34.9\cite{xu2023mplug} | 33.1 \cite{wang2022git} | 31.3 \cite{ yang2022clip } | 30.8 \cite{yang2023vid2seq} |
| ROUGEL | 70.1 \cite{xu2023mplug} | 68.2 \cite{wang2022git} | 68.0 \cite{yan2022video} | 65.0 \cite{ye2022hitea} | 64.8\cite{ yang2022clip } |
| **ActivityNet Captions** | CIDEr | 39.3 \cite{yan2022video} | 31.13 \cite{yamazaki2022vltint} | 31.29 \cite{yamazaki2022vlcap} | 28.19 \cite{ging2020coot} | 23.42\cite{lei2020mart} |
| BLEU-4 | 14.7 \cite{yan2022video} | 14.5 \cite{yamazaki2022vltint} | 13.38\cite{yamazaki2022vlcap} | 10.85 \cite{ging2020coot} | 10.33\cite{lei2020mart} |
| METEOR | 17.97\cite{yamazaki2022vltint} | 17.48\cite{yamazaki2022vlcap} | 15.99\cite{ging2020coot} | 10.68\cite{lei2020mart} | - |
| ROUGEL | 36.56\cite{yamazaki2022vltint} | 35.99 \cite{yamazaki2022vlcap} | 35.0 \cite{yan2022video} | 31.45\cite{ging2020coot} | - |
| **MSVD** | CIDEr | 165.8 \cite{xu2023mplug} | 146.9 \cite{ye2022hitea} | 146.2 \cite{yang2023vid2seq} | 139.2 \cite{fu2022empirical} | 95.2 \cite{zhang2020object} |
| BLEU-4 | 71.0 \cite{xu2023mplug} | 71.0 \cite{ye2022hitea} | 54.3 \cite{zhang2020object} | - | - |
| METEOR | 48.4 \cite{xu2023mplug} | 45.3\cite{ye2022hitea} | 45.3 \cite{yang2023vid2seq} | 36.4 \cite{zhang2020object} | - |
| ROUGEL | 85.3\cite{xu2023mplug} | 81.4 \cite{ye2022hitea} | 73.9 \cite{zhang2020object} | - | - |
| **HowTo100M** | CIDEr | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |
| BLEU-4 | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |
| METEOR | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |
| ROUGEL | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |
| **YouCook2** | CIDEr | 1.81\cite{luo2020univl} | 1.3869\cite{xu2021vlm} | 1.28\cite{yan2022video} | 1.22 \cite{huang2020multimodal} | 1.16 \cite{wang2022omnivl} |
| BLEU-4 | 17.35 \cite{ luo2020univl} | 14.2 \cite{ yan2022video } | 12.27\cite{ xu2021vlm } | 12.04 \cite{ huang2020multimodal } | 11.30 \cite{ ging2020coot } |
| METEOR | 22.35 \cite{ luo2020univl } | 19.85 \cite{ ging2020coot } | 18.32\cite{ huang2020multimodal } | 18.22\cite{ xu2021vlm } | 14.83\cite{ wang2022omnivl} |
| ROUGEL | 46.52 \cite{ luo2020univl } | 41.51 \cite{ xu2021vlm } | 39.03 \cite{ huang2020multimodal} | 37.94 \cite{ ging2020coot } | 37.7 \cite{ yan2022video } |
| **VATEX** | CIDEr | 77.8 \cite{ yan2022video } | 49.7 \cite{ zhang2020object } | 24.0 \cite{ singh2020nits } | - | - |
| BLEU-4 | 39.7 \cite{xu2023mplug} | 32.1 \cite{ zhang2020object } | 20.0 \cite{ singh2020nits } | - | - |
| METEOR | 22.2 \cite{ zhang2020object } | 18.0 \cite{ singh2020nits } | - | - | - |
| ROUGEL | 54.5\cite{ xu2023mplug } | 48.9 \cite{ zhang2020object } | 42.0 \cite{ singh2020nits } | - | - |
| **WebVid** | CIDEr | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |
| BLEU-4 | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |
| METEOR | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |
| ROUGEL | score \cite{} | score \cite{} | score \cite{} | score \cite{} | score \cite{} |

## Evaluation Metrices

### Dense Video Captioning

Dense Video Captioning (DVC) is the task of generating a sequence of textual descriptions that accurately summarize the activities and objects present in a given video segment. The evaluation of DVC models involves the use of several metrics to assess the quality of the generated captions. Commonly used metrics for DVC include the CIDEr, METEOR, ROUGE-L, and BLEU@1 , BLEU@2, BLEU@3, BLEU@4 scores.

#### CIDEr Metric

CIDEr (Consensus-based Image Description Evaluation) is a widely used evaluation metric for image and video captioning tasks that aims to measure or check the quality of the captions that were generated in comparison with the ground truth captions. It takes into account not only the accuracy of the individual words in the generated caption but also the diversity and uniqueness of the generated captions.

CIDEr computes the weighted sum of the cosine similarity of word embeddings between the generated and ground truth captions. The weights are computed based on the term frequency-inverse document frequency (TF-IDF) scores of each word. The TF-IDF score measures or finds the importance of a word in the entire caption corpus, with words that appear more frequently in many captions being weighted lower.

The CIDEr score is calculated as follows:

First, the cosine similarity between each generated caption and all ground truth captions is computed. The resulting similarity scores are then transformed using the term frequency-inverse document frequency (TF-IDF) weights of each word. Next, the average cosine similarity of the top k ground truth captions with the highest TF-IDF weights is calculated, where k is the number of ground truth captions. Finally, the average of the average cosine similarities for all generated captions is computed to obtain the final CIDEr score.

$CIDEr = \frac{1}{n}\sum\_{i=1}^{n}\frac{\min(\text{count}(w,g\_i),\text{maxn})}{\sum\_{w'\in{I}}\text{count}(w',g\_i)}\sum\_{j=1}^{m}\frac{\min(\text{count}(w\_j,c\_i),\text{maxn})}{|c\_i|^{p}}$

where:

* $n$ is the number of test instances
* $g\_i$ is the ground truth sentence for the $i$-th test instance
* $c\_i$ is the generated caption for the $i$-th test instance
* $m$ is the number of $n$-grams in the generated caption
* $\text{count}(w,g\_i)$ is the number of times the $n$-gram $w$ appears in the ground truth sentence $g\_i$
* $\text{count}(w\_j,c\_i)$ is the number of times the $n$-gram $w\_j$ appears in the generated caption $c\_i$
* $I$ is the set of unique $n$-grams in the ground truth sentences
* $\text{maxn}$ is a threshold on the maximum frequency of an $n$-gram in the ground truth sentences to avoid bias towards common words
* $p$ is a parameter that controls the length normalization

CIDEr is a popular metric for video captioning tasks, particularly for dense video captioning, as it accounts for both the diversity and accuracy of the generated captions.

#### METEOR Metric

METEOR (Metric for Evaluation of Translation with Explicit ORdering) is one of the frequently used evaluation metric or measure for natural language generation and machine translation tasks. It is a metric that measures the similarity between two sentences by computing their alignment using a combination of lexical, syntactic, and semantic information. METEOR takes into account unigram matching, word order, and synonyms to compute a score that shows the generated sentences quality.

The METEOR score is calculated based on the harmonic mean of three components: precision, recall, and alignment. Precision measures the proportion of words in the generated sentence that match with the reference sentence, while recall measures the proportion of words in the reference sentence that are captured by the generated sentence. Alignment is used to penalize incorrect word order and to account for word sense mismatches.

The METEOR score can be calculated using the following equations:

Precision = (number of matching unigrams) / (number of generated unigrams)

Recall = (number of matching unigrams) / (number of reference unigrams)

Fmean = ((α \* Precision) + ((1 - α) \* Recall)) ^ β

where α and β are the parameters that control the weight of precision and recall, and Fmean is the harmonic mean of precision and recall.

To account for word order, METEOR uses the concept of alignment. Alignment measures the amount or degree of correspondence between words in the reference and generated sentences. An alignment score is calculated for each pair of corresponding words, and then aggregated to obtain a single alignment score for the entire sentence.

The final METEOR score is obtained by combining the Fmean and alignment scores. The score is normalized to a scale of 0 to 1, with higher scores indicating better performance. METEOR is widely used in video captioning tasks, including Dense Video Captioning, to evaluate the quality of the generated captions.

#### ROUGE-L Metric

ROUGE-L is a widely used evaluation metric for measuring the quality of automatic summaries or captions by comparing them against human-written references. It measures the largest or longest common subsequence (LCS) between the reference summaries and generated summary, and then normalizes the score by the total number of words in the reference summary. The intuition behind ROUGE-L is that the longer the LCS, the better the generated summary is.

ROUGE-L can be mathematically represented by the following equation:

ROUGE-L = LCS / |R|

where LCS is the length of the longest common subsequence between the generated summary and reference summaries, and |R| is the length of the reference summary.

For example, let's consider the following generated and reference summaries for a video captioning task:

Generated summary: "A man is playing guitar on stage."

Reference summary: "A man is playing a guitar and singing on stage."

The LCS between the two summaries is "A man is playing guitar on stage", which has a length of 6. The reference summary has a total of 10 words, so the ROUGE-L score would be 6/10 or 0.6.

ROUGE-L is often used in combination with other metrics such as BLEU and METEOR to get a more comprehensive evaluation of the quality of generated summaries or captions.

#### BLEU Metric

BLEU (Bilingual Evaluation Understudy) is a popular evaluation metric used in NLP tasks such as text generation and machine translation. It is widely used to evaluate the quality of machine-generated text by comparing it with human-generated text. BLEU metric measures the similarity between a machine-generated sentence and one or more reference sentences, and returns a score between 0 and 1. A higher BLEU score indicates a better quality of machine-generated text.

BLEU score is based on n-gram precision, which measures the number or amount of overlapping n-grams (contiguous sequences of words) between the machine-generated sentence and the reference sentence. The BLEU score considers different n-gram precisions up to a certain limit, denoted as n. The score is computed as follows:

$$BLEU = BP \times \exp \left( \sum\_{n=1}^{N} w\_n \log P\_n \right)$$

where $BP$ is the brevity penalty that penalizes the machine-generated sentence length if it is shorter than the average length of the reference sentences, $P\_n$ is the n-gram precision, and $w\_n$ is the weight given to the n-gram precision. The weights are often set to $w\_n = 1/N$, where $N$ is the maximum n-gram order.

The brevity penalty ($BP$) is calculated as:

$$BP = \begin{cases} 1 & \text{if } c > r \ \exp(1-r/c) & \text{if } c \leq r \end{cases}$$

where $c$ is the length of the machine-generated sentence, and $r$ is the length of the reference sentence with the minimum difference to the machine-generated sentence.

BLEU score is widely used to evaluate the quality of machine-generated text in various NLP tasks, including video captioning. In Dense Video Captioning, BLEU score is used to evaluate the accuracy of the generated captions in comparison to the ground truth captions.

### Temporal Event Localization

#### Average Precion (AP)

This is a widely used metric for evaluating the performance of temporal event localization methods. It measures the precision of the detected events at different recall levels. AP is calculated by taking the area under the precision-recall curve. It is a plot of precision versus recall for different thresholds of the detection score. AUC gives an overall measure of the quality of the predictions.

#### Intersection over Union (IoU)

This is a measure of the overlap between the ground truth event and the predicted event. The IoU is calculated as the intersection of the two event intervals divided by their union. The IoU metric is frequently used for evaluating or finding the performance of object detection and segmentation methods and has been adapted for temporal event localization.

#### F1 Score

This is a measure of the harmonic mean of recall and precision. It is commonly used in classification tasks but has been adapted for temporal event localization. The score is calculated as 2 \* (precision \* recall) / (precision + recall).

#### Mean Average Precision (mAP)

This is the average of the AP values calculated for different classes of events. It is mostly used for evaluating and finding the performance of object detection and has been adapted for temporal event localization. mAP is calculated as the mean of the AP values for all classes.

#### Recall

This is the fraction of ground truth events that are correctly detected by the algorithm. The recall metric is used to evaluate the ability of the algorithm to detect events that occur in the video.

## For mAP% below link has all information I mentioned in table (fill all available information for each model and put a – in non available items)

## [Temporal Action Localization | Papers With Code](https://paperswithcode.com/task/action-recognition#benchmarks)

## This link may help you in finding righ way to fill the table [Charades-STA Benchmark (Moment Retrieval) | Papers With Code](https://paperswithcode.com/sota/moment-retrieval-on-charades-sta)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dataset** | **Model** | **Year of publication** | **R@1** | | | **R@5** | | | **mAP%** | | | | | **F1** | **Precision** | **Extra Training** |
| **IoU@.7** | **IoU@.5** | **IoU@.3** | **IoU@.7** | **IoU@.5** | **IoU@.3** | **.3** | **.4** | **.5** | **.6** | **.7** |  |  |  |
| ActivityNet -1.3 | PRN+BMN (ensemble) \cite{wang2021proposal} | 2021 |  |  |  |  |  |  |  |  | 59.7 |  |  |  |  | yes |
| PRN (CSN)\cite{wang2021proposal} | 2021 |  |  |  |  |  |  |  |  | 57.9 |  |  |  |  | yes |
| TCANet (slow fast R101) \cite{qing2021temporal} | 2021 |  |  |  |  |  |  |  |  | 54.33 |  | 39.13 |  |  | No |
| PRN (ViViT \cite{ wang2021proposal } | 2021 |  |  |  |  |  |  |  |  | 55.5 |  |  |  |  | No |
| AVFusion\cite{ bagchi2021hear } |  |  |  |  |  |  |  |  |  | 54.34 |  | 37.66 |  |  | No |
|  | Name \cite{} |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Name \cite{} |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Name \cite{} |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Name \cite{} |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Name \cite{} |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Name \cite{} |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Table

Description automatically generated