Event and Entity Extraction from Generated Video Captions

Johannes Scherer[1], Deepayan Bhowmik[2][0000−0003−1762−1578], and Ansgar Scherp[1][0000−0002−2653−9245]

[1] Universität Ulm, Germany firstname.lastname@uni-ulm.de
[2] Newcastle University, UK deepayan.bhowmik@newcastle.ac.uk

Abstract. Annotation of multimedia data by humans is time-consuming and costly, while reliable automatic generation of semantic metadata is a major challenge. We propose a framework to extract semantic metadata solely from automatically generated video captions. As metadata, we consider entities, the entities' properties, relations between entities, and the video category. Our framework combines automatic video captioning models with natural language processing (NLP) methods. We use state-of-the-art dense video captioning models with masked transformer (MT) and parallel decoding (PVDC) to generate captions for videos of the ActivityNet Captions dataset. We analyze the output of the video captioning models using NLP methods. We evaluate the performance of our framework for each metadata type, while varying the amount of information the video captioning model provides. Our experiments show that it is possible to extract high-quality entities, their properties, and relations between entities. In terms of categorizing a video based on generated captions, the results can be improved. We observe that the quality of the extracted information is mainly influenced by the dense video captioning model's capability to locate events in the video and to generate the event captions.
An earlier version of this paper has been published on arXiv [20]. We provide the source code here:
https://github.com/josch14/semantic-metadata-extraction-from-videos

Keywords: metadata extraction · vision models · natural language processing

1 Introduction

The annotation of multimedia with semantic metadata by humans is time-consuming and costly. Automatic extraction methods exist for different types of high-level metadata, but these methods usually have high error rates and therefore manual correction of the user is still required [19]. Thus, in contrast to the value of semantic metadata, especially when it can be generated automatically, the reliable automatic generation of semantic metadata is still a major challenge. For each semantic metadata type, one could use a different computer vision method to generate the data. For example, video object detection could be used to detect entities in a video, while video visual relation tagging methods find instances of relations between depicted entities. However, when using multiple methods, they need to be trained separately and the training is, especially for videos, computationally expensive.

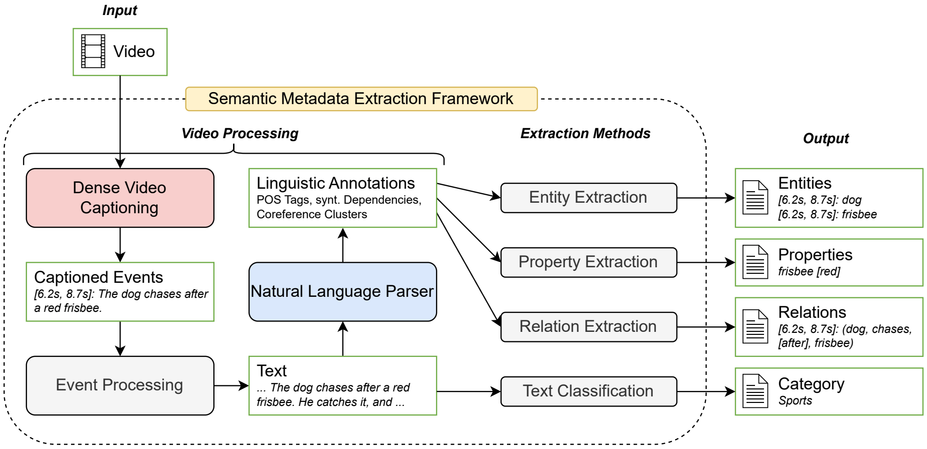


Fig. 1: Semantic metadata extraction and its key components: a dense video captioning model and a natural language parser

From this motivation, we propose a framework that generates semantic metadata from videos of not only one, but multiple types. Depending on the video application, there are various semantic metadata types of interest. We focus on four different of those types, namely the depicted *entities* and their *properties*, the observable *relations* between entities and the video *category*. Additionally, we consider semantic metadata on different levels, namely event-level, where temporal information is relevant, and video-level. Our framework combines several methods from the fields of computer vision and natural language processing (NLP) (see Figure 1). For an input video, a dense video captioning (DVC) model generates a set of natural language sentences for multiple temporally localized video events, thus providing a richly annotated description of video semantics. We process the captioned events into text to make them accessible for different NLP methods. Text classification determines the category of a video, while the extraction methods for entities, properties, and relations rely on linguistic annotations of a language parser. In summary, our contributions are:

- A framework for extracting semantic metadata combining an automatic video captioning model with several NLP methods for entity detection, extraction of entity properties, relation extraction, and categorical text classification.
- We evaluate the capabilities of our framework using the ActivityNet Captions [10] dataset. We compare two state-of-the-art dense video captioning models with masked transformer (MT) [27] and parallel decoding (PVDC) [25].

- The quality of the extracted metadata mainly depends on the event localization in the video and the performance of the event caption generation.

Below, we discuss the related work. Section 3 introduces the methods used to extract semantic metadata in the form of entities, properties, relations, and categories. Section 4 describes our experimental apparatus. The results of our experiments are reported in Section 5 and discussed in Section 6.

2 Related Work

2.1 Dense Video Captioning

For each of the semantic metadata types entities, their properties, relations, and the video categories, one could think of a computer vision method to extract only a certain type. For example, video object detection involves object recognition, that means, identifying objects of different classes, and object tracking, i. e., determining the position and size of an object in subsequent frames [8]. Therefore, an object detection model could be used to determine the entities of a video and the information about when these are visible. Shang et al. [22] propose video visual relation tagging to detect relations between objects in videos. Here, relations are annotated to the whole video without the requirement of object localization. A relation is denoted by a triplet *(subject, predicate, object)*, where the predicate may be a transitive or intransitive verb, comparative, or spatial predicate. Further methods that could be used for the extraction of video semantics include video classification for determining the category of a video [14], and emotion recognition, which aims to classify videos into basic emotions [26]. However, it is not efficient to use one computer vision method at a time for the extraction of only one semantic metadata type.

Automatic video description involves understanding and detection of different types of information like background scene, humans, objects, human actions, and events like human-object interactions [1]. In such a way, automatic video description can be seen as a task that unites the mentioned computer vision tasks like object detection, visual relation tagging, and emotion recognition. Dense video captioning (DVC), as first introduced by Krishna et al. [10], generate *captioned events*, which not only involves the localization of multiple, potentially overlapping events in time, but also the generation of a natural language sentence description for each event. Because of the rich information DVC models provide, we utilize such model in our framework. We present two DVC models with masked transformer (MT) [27] and parallel decoding (PVDC) [25], which we use in our experiments, in detail in Section 3.1 (together with our framework).

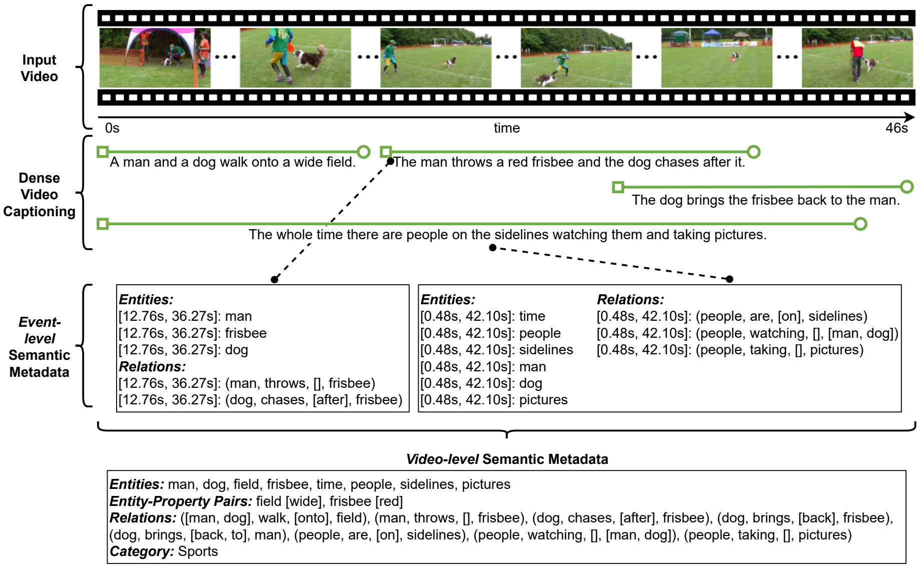
2.2 Text Information Extraction and Classification

In our framework, semantic metadata is extracted from the captioned events generated by a DVC model. This includes the analysis of the events' textual descriptions, for which methods from Open Information Extraction (Open IE) can be employed [16]. Open IE is the task of generating a structured representation of the information extracted from a natural language text in the form of relational triples. A triple *(arg1, rel, arg2)* consists of a set of argument phrases and a phrase denoting a semantic relation between them [16]. Existing Open IE approaches make use of a set of patterns, which are either hand-crafted rules or automatically learned from labeled training data. Furthermore, both methodologies can be divided into two subcategories: approaches that use shallow syntactic analysis and approaches that utilize dependency parsing [18]. Fader et al. [6] proposed REVERB, which makes use of hand-crafted extraction rules. They restrict syntactic analysis to part-of-speech tagging and noun phrase chunking, resulting in an efficient extraction for high-confidence propositions. Relations are extracted in two major steps: first, relation phrases are identified that meet syntactic and lexical constraints. Then, for each relation phrase, a pair of noun phrase arguments is identified. Contrary to REVERB, ClausIE (clause-based Open IE) uses hand-crafted extraction rules based on a typed dependency structure [4]. It does not make use of any training data and does not require any postprocessing like filtering out low-precision extractions. First, a dependency parse of the sentence is computed. Then, using the dependency parse, a set of clauses is determined. The authors define seven clause types, where each clause consists of one subject, one verb and optionally of an indirect object, a direct object, a complement, and one or more adverbials. Finally, for each clause, one or more propositions are generated. Since dependency parsing is used, ClausIE is computationally more expensive compared to REVERB.

Algur et al. [2] argue that the proper category identification of a video is essential for efficient query-based video retrieval. This task is traditionally posed as a supervised classification of the features derived from a video. The features used for video classification can be of visual nature only, but if user-provided textual metadata (i. e., title, description, tags) is available, it can be used in a profitable way [3]. However, in our proposed framework the video category is predicted only with the textual information the DVC model provides. So although we address the video classification problem, we do this by utilizing existing work in the text classification area. Text classification models can be roughly divided into two categories [12]. First, traditional statistics-based models such as k-Nearest Neighbors and Support Vector Machines require manual feature engineering. Second, deep learning models consist of artificial neural networks to automatically learn high-level features for better results in text understanding. For example, TextCNN [9] is a text classification method using text-induced word-document cooccurence graph and graph learning. We use the pre-trained BERT (Bidirectional Encoder Representations from Transformers) [5] model, which set the state-of-the-art for text classification [7].

3 Semantic Metadata Extraction from Videos

In our framework, the extraction of semantic metadata is based only on the captioned events generated by the DVC model. As a result, certain metadata types such as emotions are difficult to extract. This depends mainly on how detailed the DVC model is able to describe video semantics. Considering the capabilities of current video description models, we focus on four semantic metadata types that we aim to extract from a video: the depicted **entities** (i. e., persons, objects, locations) and their **properties**, observable visual **relations** between entities, and the video **category**, see Figure 2. We distinguish between *event-level* and *video-level* semantic metadata depending on whether semantic metadata is assigned to a specific time interval or not. For example, assume that at some point in a video there is a *cat* visible. On video-level, the corresponding entity item only stores the information that there is a cat occurring in the video. On event-level, the metadata item does not only store the name of the entity, but also a time interval in which the entity is visible.


Input Video
0s
time
46s
Dense Video Captioning
Event-level Semantic Metadata
Video-level Semantic Metadata

A man and a dog walk onto a wide field.

The man throws a red frisbee and the dog chases after it.

The dog brings the frisbee back to the man.

The whole time there are people on the sidelines watching them and taking pictures.

Entities:
[12.76s, 36.27s]: man
[12.76s, 36.27s]: frisbee
[12.76s, 36.27s]: dog
Relations:
[12.76s, 36.27s]: (man, throws, [], frisbee)
[12.76s, 36.27s]: (dog, chases, [after], frisbee)

Entities:
[0.48s, 42.10s]: time
[0.48s, 42.10s]: people
[0.48s, 42.10s]: sidelines
[0.48s, 42.10s]: man
[0.48s, 42.10s]: dog
[0.48s, 42.10s]: pictures

Relations:
[0.48s, 42.10s]: (people, are, [on], sidelines)
[0.48s, 42.10s]: (people, watching, [], [man, dog])
[0.48s, 42.10s]: (people, taking, [], pictures)

Entities: man, dog, field, frisbee, time, people, sidelines, pictures
Entity-Property Pairs: field [wide], frisbee [red]
Relations: ([man, dog], walk, [onto], field), (man, throws, [], frisbee), (dog, chases, [after], frisbee), (dog, brings, [back], frisbee), (dog, brings, [back, to], man), (people, are, [on], sidelines), (people, watching, [], [man, dog]), (people, taking, [], pictures)
Category: Sports

Fig. 2: Our framework extracts semantic metadata in the form of entities, properties of entities, relations between entities, and the video category from automatically generated captioned events. We distinguish event-level and video-level semantic metadata, depending on whether semantic metadata is assigned to a specific time interval or not. Image adapted from [13].

Revisiting Figure 1, it can be seen that our framework consists of several methods. For an input video, a DVC model generates captioned events (Sec. 3.1), which are then processed into text to make them accessible for different methods (Sec. 3.2). The natural language parser produces linguistic annotations, namely part-of-speech (POS) tags, a dependency parse, and coreference clusters (Sec. 3.3). The extraction methods for the semantic metadata types entities (Sec. 3.4), properties (Sec. 3.5), and relations (Sec. 3.6) use these linguistic annotations, while the text classification method determines the category of a video using only the generated captioned events (Sec. 3.7). The lexical database WordNet [15] is used at various points to ensure that extracted semantic metadata consists of linguistically correct English nouns, verbs, adjectives, and adverbs.

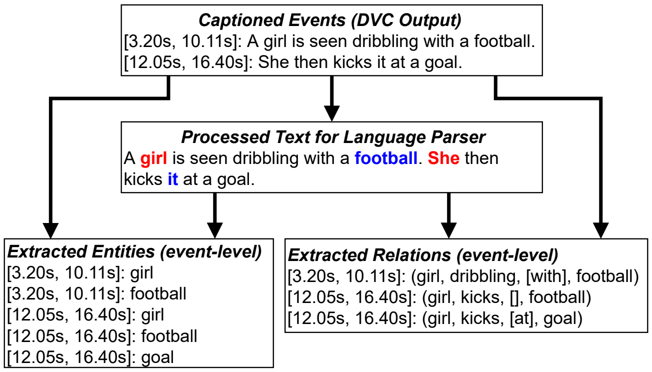
3.1 Dense Video Captioning (DVC)

From an input video, DVC models generate a set of captioned events. Each captioned event consists of the event itself, a temporal segment which potentially overlaps with segments of other captioned events, and a natural language sentence that captions the event. While introducing the task of DVC, Krishna et al. [10] proposed a model which consists of a proposal module for event localization, and a separate captioning module, an attention-based Long Short-Term Memory network for context-aware caption generation. Zhou et al. [27] argue that the model of Krishna et al. is not able to take advantage of language to benefit the event proposal module. To this end, they proposed an end-to-end DVC model with masked transformer (MT) that is able to simultaneously produce event proposals and event descriptions. Like many methods that tackle the DVC task, Zhou et al.'s model consists of three components. The video encoder, composed of multiple self-attention layers, extracts visual features from video frames. The proposal decoder takes the features from the encoder and produces event proposals, i. e., temporal segments. The captioning decoder takes input from the visual encoder and the proposal decoder to caption each event.

Wang et al. [25] state that methods like the model of Zhou et al. follow a two-stage "localize-then-describe" scheme, which heavily relies on hand-crafted components. In contrast to the usual structure of DVC models, they proposed a simpler framework for end-to-end DVC with parallel decoding (PDVC). Their model directly decodes extracted frame features into a captioned event set by applying two parallel prediction heads: localization head and captioning head. They propose an event counter, which is stacked on top of the decoder to predict the number of final events. The authors claim that PVDC is able to precisely segment the video into a number of events, avoiding to miss semantic information as well as avoiding replicated caption generation.

3.2 Event Processing

The event processing module processes the captioned events generated by the preceding DVC model into text in order to make semantic information accessible to the natural language parser and the text classification method. In detail, the sentences of the captioned events are sorted in ascending order of the start times of the corresponding events. Afterwards, the sentences are concatenated, resulting in a single text per video, and forwarded to the language parser and text classification method, respectively. By not passing the sentences separately to the language parser, this enables it to use coreference resolution (see Section 3.3). When extracting entities and relations on event-level, we annotate each entity and relation with the temporal segment of the captioned event whose sentence contains the name of the entity or the words of the relation, resp. (see Figure 3).



Captioned Events (DVC Output)
[3.20s, 10.11s]: A girl is seen dribbling with a football.
[12.05s, 16.40s]: She then kicks it at a goal.

Processed Text for Language Parser
A **girl** is seen dribbling with a **football**. **She** then kicks **it** at a goal.

Extracted Entities (event-level)
[3.20s, 10.11s]: girl
[3.20s, 10.11s]: football
[12.05s, 16.40s]: girl
[12.05s, 16.40s]: football
[12.05s, 16.40s]: goal

Extracted Relations (event-level)
[3.20s, 10.11s]: (girl, dribbling, [with], football)
[12.05s, 16.40s]: (girl, kicks, [], football)
[12.05s, 16.40s]: (girl, kicks, [at], goal)

Fig. 3: Captioned events are processed into text and made accessible to the language parser and text classifier.

3.3 Language Processing

The extraction of entities, the entities' properties, and relations is done through syntactic analysis based on the linguistic annotations generated by the natural language parser. It is required to provide POS tags of tokens, a dependency parse, and coreference clusters. The POS tag is a label assigned to the token to indicate its part of speech. The dependency parse consists of a set of directed syntactic relations between the tokens of the sentence. Coreference clusters aim to find all language expressions that refer to the same entity in a text.

We use a CNN-based model from spaCy[3]. The model generates the desired linguistic annotations in a pipe-lined manner: First, the tokenizer segments the input text into tokens. Afterwards, the tagger and dependency parser assign POS tags and dependency labels to tokens, respectively. Finally, coreference clusters are determined using spaCy's NeuralCoref extension. For an input sentence, the dependency parse produced by spaCy models is a tree where the head of a sentence, which is usually a verb, has no dependency. Every other token of the dependency tree has a dependency label that indicates its syntactic relation to its *parent*. The *children* of a token are all its immediate syntactic dependents, i. e., the tokens of the dependency tree for which it is the parent.

3.4 Entity Extraction

The entity extraction method determines which (video-level) and when (event-level) entities like persons, objects, and locations are visible in the video. A

[3] spaCy is available at: https://github.com/explosion/spaCy.

video-level **entity** only consists of a *name* describing the entity. On event-level, the entity additionally consists of a *temporal segment* containing the information about when the entity is seen. Entities are extracted by determining (compound) nouns with the POS tags and dependency labels of tokens, and coreference clusters, which are all provided by the language parser.

On event-level, before assigning a temporal segment to each name in order to obtain the entities, each pronoun is analysed whether it refers to a noun or not (see Figure 3). First, the tokens of the text are filtered for pronouns. Using the computed coreference clusters, for each pronoun it is checked whether it refers to a noun that has been determined in the previous step or not. If this is the case, then the pronoun is replaced with the corresponding noun, i. e., the name of a new entity. Finally, the event-level entities are built by assigning temporal segments to the names of the entities. Here, for an entity, the temporal segment is the segment of the corresponding sentence in which its name (or the pronoun that was previously replaced) occurs.

3.5 Property Extraction

The property extraction method determines properties of entities such as their color, size, and shape. The method is only used to extract video-level information in order to collect information for an entity from different captioned events. An **entity-property pair** is a tuple consisting of an *entity* (i. e., its name) and a *property*, which further describes the entity. For each (video-level) entity, extracted in the previous section, properties are determined as follows. The candidate properties for an entity are the children of the corresponding token (or tokens for compound nouns) that are marked with the dependency labels. We use WordNet to recognize the token or its lemma as an adjective. Such candidate property is considered as a property of the entity. An entity-property pair is formed with the name of the entity and the detected property. In such way, the method results in a set of video-level entity-property pairs. There is no restriction on the tag a property token needs to have. Therefore, properties may be marked by the language parser as `ADJ`ective (e. g., *round* ball), `VERB` (e. g., *provoking* film), or other tags.

3.6 Relation Extraction

As seen for visual relation tagging (Section 2.1) or Open IE (Section 2.2), relations are usually formulated as triples. In contrast, we define relations as follows. A video-level **relation** is a 4-tuple of the form *(subjects, verb, modifiers, objects)*. With a *temporal segment*, an event-level relation has an additional element containing the information about when the relation is observed. The first element *subjects* is a list containing the names of the relation's acting entities. *Objects*, also a list of names of entities, contains the entities that are the receiver of the action. Note that usually subjects and objects contain only a single entity. In some cases, such as in the sentence *”A boy and a girl are seen playing football.”*, multiple entities are the actors in a relation: *([boy, girl], play,*

[], football). With the above definition, we are able to capture relations with different types of verbs: single-word verbs and multi-word verbs, i. e., prepositional verbs (verb+preposition), phrasal verbs (verb+particle), and phrasal-prepositional verbs. In both cases, *verb* contains the verb. *Modifiers* is an empty list for single-word verbs. For multi-word verbs, however, *modifiers* contains the verb's particles and prepositions. Both, particles and prepositions, provide information about how the verb and the object are related to each other. For example, the relation *(girl, catches, [up, with], kids)* for the sentence *"The girl catches up with the other kids."* is better understood than the relation *(girl, catches, kids)*.

As for entity extraction, the relation extraction method utilizes POS tags and dependency labels of tokens and coreference clusters. In fact, the entity extraction method is used here in order to determine valid subjects and objects for relations. The relation extraction method proceeds in three steps: search for candidate verbs, search for candidate tuples consisting of a subject and a verb, and search for corresponding objects and modifiers for each verb in a candidate tuple. In brief, these steps are conducted by analyzing the dependency tree, exploiting the POS tags, and exploiting WordNet.

The video-level relation extraction is finished here. Event-level relations are built by assigning temporal segments to the extracted relations. Here, for a determined relation, the event is the temporal segment of the corresponding sentence from which the relation was extracted.

3.7 Text Classification

Although our proposed framework performs video classification, it predicts the category of a video only with the textual information its DVC model provides. The motivation is to see how far a text classifier on generated video captions can correctly classify a video. As text classifier, we adopt BERT [5].

4 Experimental Apparatus

4.1 Datasets

We introduce the datasets to evaluate the DVC models and the different tasks of our metadata extraction framework.

Dense Video Captioning We use the large-scale benchmark dataset ActivityNet Captions [10] to train and evaluate the dense video captioning models. It consists of 20k YouTube videos of various human activities split up into train/-val/test sets of 0.5/0.25/0.25. Each video is annotated with captioned events, each consisting of a descriptive sentence and a specific temporal segment to which the description refers. On average, each video is annotated with 3.65 temporally-localized sentences. Each captioned event on average covers 36 seconds and is composed of 13.5 words. Temporal segments in the same video can overlap in time, which enables DVC models to learn complex events and relations.

Entity, Property, and Relation Extraction The information that ActivityNet Captions provides for each video is limited to captioned events, i. e., pairs of temporal segments and sentences. To be able to evaluate the entity, property, and relation extraction methods, we need information about depicted entities, properties of entities, and relations of videos. For this purpose, we utilize the gold standard captioned events of ActivityNet Captions' validation videos. We extract entities, their properties, and relations from the captioned events, and treat the results as gold standard for semantic metadata extraction. We generate five different datasets for videos in the ActivityNet Captions validation set: each one for event-level entities, video-level entities, entity-property pairs, event-level relations, and video-level relations. Using these datasets, we evaluate the framework's ability to extract the semantic metadata and compare it to metadata extracted from the captioned events generated by the DVC models.

Text Classification We build a new dataset to train and evaluate our text classification method. Here, we take advantage of the fact that ActivityNet Captions consists of YouTube videos. For each video of the ActivityNet Captions' train and validation sets, we query the corresponding category from the YouTube Data API. We were able to query the category of $12,579$ ActivityNet Captions videos (on March 20, 2022). We split the videos in train/val/test of 0.6/0.2/0.2, while ensuring that the category distribution is the same for all splits.

4.2 Procedure

Dense Video Captioning We train MT and PDVC using the ActivityNet Captions dataset. For action recognition, both models adopt the same action recognition network, a pre-trained temporal segment network [24], to extract frame-level features. To ensure consistency, we evaluate both models on the ActivityNet Captions validation set (using both annotation files) and compare the performances with those reported in the corresponding works.

For our proposed semantic metadata extraction methods, the number of captioned events forwarded to each method is an important parameter. With increasing number of captioned events forwarded to an extraction method, it can extract more semantic information. In this work, we denote the number of captioned events that are generated by a DVC model and forwarded to a specific method as $|E|$. MT and PDVC, the dense video captioning models of our choices, internally calculate confidence scores for their generated captioned events. If the number of considered captioned events is limited, then the generated captioned events with the highest confident scores are used. For high values of $|E|$, DVC models tend to produce identical sentences for different temporal segments. However, our property extraction, and video-level entity and relation extraction methods rely mainly on the textual information that the captioned events provide. Therefore, captioned events with duplicate sentences do not provide further semantic information. Because of that, we only forward events with distinct captions to these methods. We denote the number of distinct captioned events that are forwarded to a specific method as $|dist(E)|$. If two events share the exact same caption, then the event with the higher confidence score is forwarded. We finally evaluate MT and PDVC with $|E|$ set to 10, 25, 50, and 100, and $|dist(E)|$ set to 1, 3, 10, and 25.

Entity, Property, and Relation Extraction For both trained dense video captioning models, MT and PDVC, we extract event-level entities and relations from captioned events that they generate for ActivityNet Captions validation videos with $|E|$ set to 10, 25, 50, and 100. Video-level entities and relations, and entity-property pairs are extracted with $|dist(E)|$ set to 1, 3, 10, and 25. We then evaluate the entity, property, and relation extraction methods by comparing the extracted semantic metadata with our generated gold standards for entities, entity-property pairs, and relations.

For the evaluation of video-level entity extraction, we introduce the entity frequency threshold f. We extract video-level entities from ActivityNet Captions' train and validation videos. The frequency of a video-level entity is the number of different videos in which it occurs. We evaluate the generated video-level entities with f set to 0, 10, 25, and 50, meaning that the gold standard only contains those entities which have a frequency higher than the frequency threshold. Thus, a larger f means that the DVC models are given a higher chance to learn and reproduce the entities in the videos.

Text Classification We train and evaluate the text classification model of our framework in three different settings: using the captioned events generated by the MT and PDVC model, respectively, and using the gold standard captioned events provided by ActivityNet Captions. This allows us to analyze how useful the automatically generated captioned events are for text classification compared to gold standard captioned events. For each data input, we set $|dist(E)|$ to 10.

4.3 Hyperparameter Optimization

For the training of the DVC models, we use largely the same parameters that were used to train the models in their original works (refer to MT [27] and PDVC [25] or their latest codebases). For MT, we have to switch to a batch size of 84 and a learning rate of 0.06 to make sure that the model training converges. In the default configuration, PDVC generates a maximum of 10 events per video. We change the corresponding parameter such that 100 captioned events are generated to ensure that there is enough information for our metadata extraction methods to work on. For both, MT and PDVC, we use the models' best states w.r.t. their METEOR score performances (see Section 4.4). For MT, this was achieved after 34 epochs, and for PDVC after 13 epochs.

For the entity, property, and relation extraction methods, no hyperparameter optimization is necessary. In our text classification model, we use an uncased BERT model. Each input text is truncated to 128 tokens. For training, we use cross-entropy loss and the Adam optimizer with 10% warmup steps. We use class weights to help the model learn on the imbalanced data of our classification dataset. While using captioned events from ActivityNet Captions as training data, we perform grid search over dropout rates in $\{0.0, 0.1, \ldots, 0.5\}$, maximum number of training epochs in $\{1, 2, \ldots, 5\}$, learning rates between $5e$-6 and $1e$-4, and a batch sizes in $\{1, 2, 4, 8\}$. The lowest validation loss was achieved after epoch 2 while using dropout of 0.1, 2 maximum training epochs, learning rate of $2e$-5, and batch size of 4. Using this parameter configuration, we train the model with each data input separately. We repeat the training three times and select the model that achieved the lowest final validation loss.

4.4 Measures and Metrics

Dense Video Captioning To evaluate the dense video captioning models MT and PDVC, we use the official evaluation toolkit provided by ActivityNet Captions Challenge 2018[4] that measures the capability to localize and describe events in videos. For evaluation of dense video captioning, we report METEOR [11] and BLEU [17] scores. Using temporal Intersection over Union (tIoU) thresholds at {0.3, 0.5, 0.7, 0.9} for captioned events, we report recall and precision of their temporal segments and METEOR and BLEU scores of their sentences. Given a tIoU threshold, if the event proposal has a segment overlapping larger than the threshold with any gold standard segment, the metric score is computed for the generated sentence and the corresponding gold standard sentence. Otherwise, the metric score is set to 0. The scores are then averaged across all event proposals and finally averaged across all tIoU thresholds.

Entity, Property, and Relation Extraction For property extraction and video-level entity and relation extraction, we measure micro-averaged precision, recall, and their harmonic mean, the F1 score. Similarly to the evaluation of DVC models, when evaluating entity and relation extraction on event-level, we also take the quality of the predictions' temporal segments into account. Here, we compute micro-averaged precision and recall across all videos using tIoU thresholds at {0.3, 0.5, 0.7, 0.9}, and then average the results across all thresholds. Additionally, we report the F1 score of averaged precision and recall values.

For the evaluation of entities, entity-property pairs, and relations, we use WordNet to compare words in terms of whether they are synonyms of each other or not. This is fair as the variety of extracted semantic metadata is large. For example, *"stand up"* is considered a correct prediction for the verb *"get up"*. However, this means that for a video the number of predictions that are treated as correct (the set TP_p) and the number of gold standard targets (denoted as TP_g) is not necessarily the same. With other words, the gold standard is enriched with synonyms. For example, $|TP_p|= 2$ and $|TP_g|= 1$ when accepting *person* and *individual* as synonymously correct predictions for $TP_g = \{person\}$. To ensure the validity of precision (proportion of correct predictions) and recall (proportion

[4] See: https://github.com/ranjaykrishna/densevid_eval/

of correctly predicted targets), we use TP_p for calculating precision and TP_g for recall, respectively.

All entities, properties, and relations are compared using their word lemmas. On video-level, this means that potential duplicates of entities and relations are removed, i. e., they are unique. For example, $\{man, men\}$ results in $\{man\}$.

Text Classification We report weighted and macro-averages of precision, recall, and F1 score across all categories.

5 Results

5.1 Dense Video Captioning

Event Localization & Dense Video Captioning						
DVC		avg.	avg.	2018 eval. toolkit		
Model	$\|E\|$	Recall	Precision	B@3	B@4	M
MT	10	43.61	48.96	2.18	1.02	5.89
	25	56.55	46.15	2.33	1.14	5.74
	50	67.70	41.60	2.31	1.13	5.35
	100	76.33	34.78	2.12	1.04	4.68
PDVC	10	61.88	45.41	3.10	1.59	6.34
	25	73.81	36.89	2.54	1.23	5.17
	50	78.76	25.75	1.92	0.90	3.88
	100	82.24	15.63	1.36	0.63	2.63

(a) Results for varying numbers of generated captioned events $|(E)|$.

Event Localization & Dense Video Captioning						
DVC		avg.	avg.	2018 eval. toolkit		
Model	$\|dist(E)\|$	Recall	Precision	B@3	B@4	M
MT	1	22.04	50.99	1.55	0.68	5.37
	3	32.72	49.65	1.97	0.90	5.76
	10	50.08	44.75	2.22	1.05	5.58
	25	63.67	36.23	2.05	0.97	4.80
PDVC	1	20.05	49.63	2.52	1.30	5.86
	3	40.28	48.26	2.98	1.54	6.46
	10	62.38	44.10	2.85	1.41	6.01
	25	71.62	31.69	2.14	1.00	4.54

(b) Results for varying numbers of generated distinct captioned events $|dist(E)|$.

Table 1: Event localization and dense video captioning results of MT and PDVC for different numbers of generated captioned events on the ActivityNet Captions validation set. We report recall and precision of temporal segments, METEOR (M), and BLUE@N (B@N) of generated captioned events.

We introduced the parameters $|E|$ and $|dist(E)|$, which are primarily used in our framework to control the amount of information that is forwarded from the DVC model to the semantic metadata extraction methods. Using these parameters, we are also able to compare the DVC models in a fair way, meaning that their performances are evaluated while generating an equal number of captured events. Table 1 shows the event localization and dense video captioning performances of MT and PDVC with respect to $|E|$ and $|dist(E)|$, the number of (distinct) captioned events that are generated by each DVC model and used for the evaluation. For event localization, PDVC constantly achieves better recall than MT, while MT constantly outperforms PDVC in terms of precision. For both models, higher values of $|E|$ and $|dist(E)|$ results in better recall and worse precision performance in event localization. Looking at the results for dense video captioning, we can state that the METEOR performances of both models degrade with increasing number of captioned events, both distinct and non-distinct, except for when increasing $|dist(E)|$ from 1 to 3. When increasing $|E|$ from 10 to 100, the METEOR performance of PDVC drops by 3.71 points in total, while the METEOR performance of MT drops by 1.21 points. This is a consequence of the declining precision for event localization for higher $|E|$, which affects the PDVC model more heavily than the MT model.

5.2 Entity Extraction

Table 2 shows precision, recall, and F1 score performances of our framework for **video-level entity** extraction. We evaluate the extracted video-level entities with entity frequency threshold f set to 0, 10, 25, and 50. One observation is that, with increasing $|dist(E)|$, precision decreases and recall improves for both models and all thresholds. For both DVC models and all thresholds, the best F1 score performance is achieved with $|dist(E)|= 10$. This indicates a limited level of semantic information that any further generated captioned events provide. For higher entity frequency thresholds, for both DVC models and all $|dist(E)|$, our framework is able to predict video-level entities with improved recall, at the cost of only slightly lower precision. When our framework is using the PDVC model, it achieves better precision and F1 scores for all $|dist(E)|$. The framework achieves better recall performances when using the MT model for $|dist(E)|$ set to 10 and 25. Overall, for video-level entity extraction, our framework achieves its highest F1 scores when using the PDVC model with $|dist(E)|$ set to 10. Here, the achieved F1 scores range from 31.27 for $f = 0$ to 34.21 for $f = 50$.

Table 3 shows the results of **event-level entity** extraction of our framework. Depending on the framework's used DVC model and the number of generated captioned events $|E|$, we report precision and recall for different temporal Intersection over Union (tIoU) thresholds. F1 score is calculated using the averages of precision and recall. Note that for a prediction to be correct for an event-level entity, a condition is that its temporal segment overlaps with the gold standard entity's temporal segment larger than the tIoU threshold. Therefore, precision and recall decreases at higher tIoU thresholds. Regardless of the DVC model used, the framework's precision decreases for higher $|E|$, while at the same time it benefits in recall performance. When using the PDVC model, the framework's precision drops more (1.13 on average) when increasing $|E|$ from 10 to 100 as

Video-level Entity *Extraction*

DVC Model	$\lvert dist(E)\rvert$	Precision(@f) 0	10	25	50	Recall(@f) 0	10	25	50	F1(@f) 0	10	25	50
MT	1	39.94	39.88	39.66	38.91	15.38	16.49	17.57	18.99	22.21	23.33	24.35	25.52
	3	33.91	33.83	33.61	32.78	23.44	25.12	26.72	28.70	27.72	28.83	29.77	30.60
	10	26.44	26.37	26.16	25.43	32.59	34.90	37.06	39.67	29.20	30.04	30.67	30.99
	25	20.76	20.69	20.49	19.82	40.69	43.55	46.16	49.15	27.49	28.05	28.38	28.25
PDVC	1	45.13	45.05	44.91	44.43	15.44	16.56	17.68	19.28	23.01	24.22	25.37	26.89
	3	39.01	38.95	38.83	38.28	23.33	25.03	26.72	29.04	29.20	30.48	31.66	33.03
	10	31.77	31.71	31.60	31.03	30.78	33.01	35.21	38.11	31.27	32.35	33.30	34.21
	25	26.30	26.25	26.14	25.61	36.18	38.79	41.33	44.66	30.46	31.31	32.02	32.55

Table 2: Video-level entity extraction using dense video captioning models MT and PDVC. We report precision, recall, and F1 score for different numbers of generated distinct captioned events $|dist(E)|$ and entity frequency thresholds f.

Event-level Entity *Extraction*

DVC Model	$\lvert E\rvert$	Precision(@tIoU) 0.3	0.5	0.7	0.9	Avg	Recall(@tIoU) 0.3	0.5	0.7	0.9	Avg	F1
MT	10	28.97	20.08	8.55	1.26	14.72	21.11	15.31	9.59	2.94	12.24	13.37
	25	28.12	18.44	7.40	1.03	13.75	26.48	20.66	14.29	5.58	16.75	15.10
	50	26.30	15.93	6.10	0.83	12.29	31.15	25.32	18.69	8.47	20.91	15.48
	100	23.31	12.70	4.52	0.59	10.28	36.35	30.05	22.89	11.04	25.08	14.58
PDVC	10	30.47	19.79	9.11	2.53	15.47	26.36	21.13	14.52	5.87	16.97	16.19
	25	27.16	15.83	6.41	1.53	12.73	31.14	26.34	20.01	8.21	21.42	15.97
	50	20.75	10.49	3.84	0.86	8.98	33.66	28.89	22.40	8.75	23.42	12.98
	100	13.76	6.36	2.22	0.48	5.71	35.35	30.65	24.00	9.02	24.75	9.28

Table 3: Event-level entity extraction results. Reported are precision and recall for different numbers of generated distinct captioned events $|E|$ and tIoU thresholds. The F1 scores are the averages of precision and recall across all thresholds.

compared to when it is using the MT model (0.57 on average). Note that we made the same observation when we evaluated the event localization and dense video captioning performances of the DVC models. For event localization, the PDVC model could not convert the large drops in precision for higher $|E|$ into better recall performance. Thus, on the one hand, this results in worse dense video captioning performance. On the other hand, when our framework is using the PDVC model for event-level entity extraction, this results in degrading F1 score performance for higher $|E|$. Still, our framework achieves its highest F1 score with 16.19 when it uses the PDVC model with $|E|= 10$. On the other hand, when using the MT model, the highest achieved F1 score is 15.48 for $|E|= 50$.

5.3 Property Extraction

The results of the extraction of entity-property pairs with our framework are shown in Table 4. In general, increasing $|dist(E)|$ leads to drops in precision, however, our framework benefits from much improved recall. Consequently, our framework achieves its highest F1 scores for $|dist(E)|= 25$. In contrast to video-level entity extraction where highest F1 scores were achieved for $|dist(E)|= 10$, we can observe that increasing $|dist(E)|$ from 10 to 25 leads to even better property extraction performance with respect to the F1 score, indicating that the DVC models still provide meaningful semantic information about the properties of entities when many captioned events are generated. For all $|dist(E)|$, our framework achieves better recall and F1 scores when using the MT model for captioned events generation and better precision when using the PDVC model. The highest achieved precision is 9.08 when using PDVC with $|dist(E)|= 1$. Using the MT model with $|dist(E)|= 25$ results in the framework's highest achieved recall (8.86) and F1 score (4.94).

Property *Extraction*				
DVC Model	$\|dist(E)\|$	Prec.	Rec.	F1
MT	1	6.48	0.82	1.45
	3	6.64	1.85	2.90
	10	5.66	3.54	4.36
	25	4.53	5.43	4.94
PDVC	1	9.08	0.72	1.34
	3	8.76	1.61	2.72
	10	6.96	2.68	3.87
	25	4.79	3.76	4.21

Table 4: Results for the extraction of entity-property pairs.

5.4 Relation Extraction

Table 5 shows the results of video-level relation extraction with our framework. In general, using the PDVC model leads to better precision performance except for $|dist(E)|= 1$, while using the MT model leads to better recall performance except for $|dist(E)|= 3$. For both DVC models, our framework achieves its highest F1 scores for $|dist(E)|= 10$. This suggests that any larger number of generated captioned events can provide more semantic information for relations only at a higher cost of precision, the same observation as we made for video-level entity extraction. However, for property extraction, the highest F1 scores were achieved for $|dist(E)|= 25$. The highest achieved F1 score of our framework for video-level relation extraction is 5.02 while using the PDVC model with $|dist(E)|= 10$.

The results for event-level relation extraction are shown in Table 6. Very similar observations can be made as for event-level entity extraction. The framework's precision decreases for higher $|E|$ while benefiting in recall performance. When using the PDVC model, the framework's precision performance suffers more for higher $|E|$ compared to when it is using the MT model. As observed before for event-level entity extraction, this is not converted into much higher recall, resulting in a degradation of the framework's F1 score performance. Therefore, for $|E|$ set to 50 and 100, i. e., high numbers of generated captioned events, our framework achieves its highest F1 scores when using the MT model, while for $|E|$ set to 10 and 25 the highest F1 scores are achieved when using the PDVC model. The event-level relation extraction achieves its best performance with respect to F1 score when it uses the PDVC model with $|E|$= 10.

***Video-level Relation** Extraction*				
DVC Model	$\|dist(E)\|$	Prec.	Rec.	F1
MT	1	5.76	2.07	3.04
	3	4.88	4.07	4.44
	10	3.64	6.61	4.70
	25	2.89	8.86	4.35
PDVC	1	5.64	2.06	3.02
	3	5.02	4.18	4.56
	10	4.09	6.50	5.02
	25	3.47	8.36	4.91

Table 5: Results for video-level relation extraction.

***Event-level Relation** Extraction*												
		Precision(@tIoU)					Recall(@tIoU)					
DVC Model	$\|E\|$	0.3	0.5	0.7	0.9	Avg	0.3	0.5	0.7	0.9	Avg	F1
MT	10	3.76	2.52	1.08	0.14	1.87	3.81	2.81	1.73	0.42	2.20	2.02
	25	3.61	2.28	0.93	0.12	1.74	4.89	3.90	2.56	0.87	3.05	2.22
	50	3.35	1.99	0.77	0.10	1.55	5.87	4.78	3.29	1.32	3.81	2.20
	100	2.96	1.59	0.56	0.07	1.30	7.06	5.60	3.91	1.63	4.55	2.02
PDVC	10	3.64	2.33	1.14	0.32	1.86	4.87	3.87	2.63	1.06	3.11	2.33
	25	3.33	1.92	0.80	0.20	1.56	6.01	4.95	3.49	1.37	3.95	2.24
	50	2.66	1.33	0.50	0.11	1.15	6.53	5.37	3.81	1.41	4.28	1.81
	100	1.75	0.82	0.30	0.07	0.73	6.75	5.53	3.91	1.41	4.40	1.25

Table 6: Experimental results for event-level relation extraction.

5.5 Text Classification

Finally, Table 7 shows the weighted and macro-averages of precision, recall, and F1 score performances of the framework's text classification method, while trained and evaluated in three different settings: using the captioned events generated by the MT or PDVC model, respectively, and using the captioned events provided by ActivityNet Captions. The most noteworthy observation is that the classification performance of our framework, when using captioned events generated by MT and PDVC, is not far from the classification performance that is achieved when using gold standard captioned events of ActivityNet Captions. Therefore, we can state that the DVC models generate specific semantic information for videos of different categories at a similar level as the captioned events of the gold standard provide. This is important for the text classifier to categorize videos successfully.

When using automatically generated captioned events for video classification, our framework achieves its best performances for all metrics, both weighted and macro-averaged, when using the PDVC model. Here, for weighted precision, recall, and F1 score, our framework performs around 2 points better as compared to when using the MT model. When using PDVC, our framework achieves an overall accuracy (i. e., weighted recall) of 50.22, which is only 0.59 points lower than the accuracy achieved when classifying videos using the gold standard captioned events of ActivityNet Captions. Taking the category imbalance in our dataset into account, we observe that the achieved macro-averages of precision, recall and F1 scores are much lower as compared to their weighted-averages.

Text Classification				
Averaging	Captioned events input	Prec.	Rec.	F1
Weighted	MT	43.72	48.24	44.80
	PDVC	45.75	50.22	46.96
	ActivityNet Captions	46.97	50.81	48.23
Macro	MT	27.51	30.45	28.08
	PDVC	34.31	32.30	30.88
	ActivityNet Captions	33.42	34.74	33.19

Table 7: Results for classification of video captions in the settings: (i) captioned events generated by MT, (ii) PDVC, and (iii) gold standard captioned events from ActivityNet Captions. For MT and PDVC, $|dist(E)|= 10$ is used.

6 Discussion

6.1 Key Results

The experiments show that our proposed framework is able to automatically generate multiple types of semantic metadata in a meaningful way. We have to keep in mind that in our framework semantic metadata is extracted only from captioned events that are automatically generated by its DVC model, i. e., a model that is designed and trained for a different task. To extract higher quality semantic information for each semantic metadata type, a dedicated computer vision method could be used, such as video object detection and video visual relation tagging. Here, however, our framework trades quality for effectiveness as it only requires training for one computer vision model, the DVC model. We must also bear in mind that the variety of entities, entity-property pairs, and relations that occur in our gold standards, and that are extracted by our framework, is large, in particular when compared to related computer vision tasks. For example, the VidOR (Video Object Relation) dataset [21,23], which is used to train models for visual relation detection from videos, contains annotations of 80 categories of objects and 50 categories of relation predicates. In our gold standard for relations, however, 2,493 different entities acted either as subject (905 entities) or object (2,299 entities), while 905 different verbs occur in the relations. Also, for video-level entity extraction, we observed that for higher entity frequency thresholds f, the framework's precision decreases only slightly, while recall performance improves greatly. This observation is not surprising, as the number of different entities in our gold standard for entities, which is based on ActivityNet Captions, is large, and many entities occur only a few times. This leads to the conclusion that the DVC models are only able to learn entities when there is enough training data, i. e., the models see them sufficiently enough during training. Regarding the video classification results (based on the generated captions), our framework could not reliably predict the video category. This is because the captioned events generated by the DVC models do not contain sufficiently specific semantic information for videos of different categories. Here, a visual-based video classifier is preferred over a purely text-based approach.

6.2 Threats to Validity and Future Work

We generated a gold standard using the captioned events of ActivityNet Captions validation videos. As described in Section 4.1, we use our proposed entity, property, and relation extraction methods on the processed captioned events, and treat the results as gold standards for semantic metadata. This requires that the extraction methods work sufficiently well, i. e., they are able to extract semantic metadata using the linguistic annotations provided by the framework's language parser. In order to validate this hypothesis, we annotated a subset of 25 ActivityNet Captions videos with the video-level entities, entity-property pairs, and video-level relations that we expect the methods to extract from captioned events. In total, we made annotations of 110 captioned events in these 25 videos. Table 8 shows the precision and recall performances of our entity, property, and relation extraction methods on these manually annotated videos. The results show that our entity and property extraction methods are certainly reliable. In some cases, however, our entity extraction method wrongly determines entities. For example, for the sentence *"the camera pans around the field"*, spaCy wrongly classifies *camera*, *pans* and *field* as nouns, while *pans* actually acts as verb in the sentence. Since *pan* is listed in WordNet as a noun, *pans* is still determined as an entity. The relation extraction method is able to determine only around 70% of the relations. This is due to the complexity of relation extraction.

7 Conclusion

We presented a framework for metadata extraction of various types from generated video captions. The metadata quality mainly depends on two factors: The event localization and video captioning performance of the dense video captioning model, and the number of captioned events forwarded from the dense video

Semantic Metadata Type	Prec.	Rec.
Entities (video-level)	94.21	98.39
Properties	92.98	91.38
Relations (video-level)	78.36	70.62

Table 8: Evaluation of the entity, property, and relation extraction methods on captioned events from 25 manually annotated videos.

captioning model to the semantic metadata extraction methods. This opens the path for future research on integrated models for semantic metadata extraction.

References

1. Aafaq, N., Mian, A., Liu, W., Gilani, S.Z., Shah, M.: Video description: A survey of methods, datasets, and evaluation metrics. ACM Comput. Surv. **52**(6) (2019)
2. Algur, S., Bhat, P.: Metadata construction model for web videos: A domain specific approach. IJECS **3** (2014)
3. Algur, S.P., Bhat, P.: Web video mining: metadata predictive analysis using classification techniques. IJ Information Technology and Computer Science **2** (2016)
4. Del Corro, L., Gemulla, R.: Clausie: Clause-based open information extraction. In: World Wide Web. ACM (2013)
5. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR **abs/1810.04805** (2018)
6. Fader, A., Soderland, S., Etzioni, O.: Identifying relations for open information extraction. In: EMNLP. ACL (2011)
7. Galke, L., Scherp, A.: Bag-of-words vs. graph vs. sequence in text classification: Questioning the necessity of text-graphs and the surprising strength of a wide MLP. In: Assoc. for Comp. Linguistics. ACL (2022)
8. Jiao, L., Zhang, R., Liu, F., Yang, S., Hou, B., Li, L., Tang, X.: New generation deep learning for video object detection: A survey. IEEE Transactions on Neural Networks and Learning Systems (2021). https://doi.org/10.1109/TNNLS.2021.3053249
9. Kim, Y.: Convolutional neural networks for sentence classification. In: EMNLP. ACL (2014)
10. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Niebles, J.C.: Dense-captioning events in videos. In: ICCV (2017). https://doi.org/10.1109/ICCV.2017.83
11. Lavie, A., Agarwal, A.: METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In: Workshop on Statistical Machine Translation. ACL (2007)
12. Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P.S., He, L.: A survey on text classification: From shallow to deep learning. CoRR **abs/2008.00364** (2020)
13. Li, Y., Yao, T., Pan, Y., Chao, H., Mei, T.: Jointly localizing and describing events for dense video captioning. In: 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (2018). https://doi.org/10.1109/CVPR.2018.00782
14. Long, X., Gan, C., de Melo, G., Wu, J., Liu, X., Wen, S.: Attention clusters: Purely attention based local feature integration for video classification. In: CVPR (2018). https://doi.org/10.1109/CVPR.2018.00817
15. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11) (1995)
16. Niklaus, C., Cetto, M., Freitas, A., Handschuh, S.: A survey on open information extraction. In: Int. Conf. on Comp. Linguistics. ACL (2018)
17. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: ACL. ACL (2002)
18. Sarhan, I., Spruit, M.: Uncovering algorithmic approaches in open information extraction: A literature review. In: Benelux Conf. on AI. Springer (2018)
19. Sarvas, R., Herrarte, E., Wilhelm, A., Davis, M.: Metadata creation system for mobile images. In: MobiSys. ACM (2004)
20. Scherer, J., Scherp, A., Bhowmik, D.: Semantic metadata extraction from dense video captioning. CoRR **abs/2211.02982** (2022), https://doi.org/10.48550/arXiv.2211.02982

21. Shang, X., Di, D., Xiao, J., Cao, Y., Yang, X., Chua, T.S.: Annotating objects and relations in user-generated videos. In: Multimedia Retrieval. ACM (2019)
22. Shang, X., Ren, T., Guo, J., Zhang, H., Chua, T.S.: Video visual relation detection. In: Multimedia. ACM (2017)
23. Thomee, B., Shamma, D.A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., Li, L.J.: YFCC100M: The new data in multimedia research. Communications of the ACM **59**(2) (2016)
24. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: ECCV. Springer (2016)
25. Wang, T., Zhang, R., Lu, Z., Zheng, F., Cheng, R., Luo, P.: End-to-end dense video captioning with parallel decoding. ICCV (2021)
26. Zhou, H., Meng, D., Zhang, Y., Peng, X., Du, J., Wang, K., Qiao, Y.: Exploring emotion features and fusion strategies for audio-video emotion recognition. 2019 Int. Conf. on Multimodal Interaction (2019). https://doi.org/10.1145/3340555.3355713
27. Zhou, L., Zhou, Y., Corso, J.J., Socher, R., Xiong, C.: End-to-end dense video captioning with masked transformer. In: CVPR (2018). https://doi.org/10.1109/CVPR.2018.00911