Poet: Product-oriented Video Captioner for E-commerce

Shengyu Zhang[1*], Ziqi Tan[1*], Jin Yu[2], Zhou Zhao[1†], Kun Kuang[1†], Jie Liu[2], Jingren Zhou[2],

Hongxia Yang[2†], Fei Wu[1†]

[1] College of Computer Science and Technology, Zhejiang University

[2] Alibaba Group

{sy_zhang,tanziqi,zhaozhou,kunkuang,wufei}@zju.edu.cn

{kola.yu,sanshuai.lj,jingren.zhou,yang.yhx}@alibaba-inc.com

ABSTRACT

In e-commerce, a growing number of user-generated videos are used for product promotion. How to generate video descriptions that narrate the user-preferred product characteristics depicted in the video is vital for successful promoting. Traditional video captioning methods, which focus on routinely describing what exists and happens in a video, are not amenable for *product-oriented* video captioning. To address this problem, we propose a product-oriented video captioner framework, abbreviated as *Poet*. *Poet* firstly represents the videos as product-oriented spatial-temporal graphs. Then, based on the aspects of the video-associated product, we perform knowledge-enhanced spatial-temporal inference on those graphs for capturing the dynamic change of fine-grained product-part characteristics. The knowledge leveraging module in *Poet* differs from the traditional design by performing knowledge filtering and dynamic memory modeling. We show that *Poet* achieves consistent performance improvement over previous methods concerning generation quality, product aspects capturing, and lexical diversity. Experiments are performed on two product-oriented video captioning datasets, buyer-generated fashion video dataset (BFVD) and fan-generated fashion video dataset (FFVD), collected from Mobile Taobao. We will release the desensitized datasets to promote further investigations on both video captioning and general video analysis problems.

CCS CONCEPTS

• **Computing methodologies** → *Computer vision*; **Natural language generation**.

KEYWORDS

Video-to-Text generation; E-commerce; External knowledge; User-generated video analysis

ACM Reference Format:

Shengyu Zhang, Ziqi Tan, Jin Yu, Zhou Zhao, Kun Kuang, Jie Liu, Jingren Zhou, Hongxia Yang, Fei Wu. 2020. Poet: Product-oriented Video Captioner for E-commerce. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3394171.3413880

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00

https://doi.org/10.1145/3394171.3413880

1 INTRODUCTION

Nowadays, a growing number of short videos are generated and uploaded to Taobao. Among these videos, user-generated videos are massive in volume and share unique product experiences, such as the individual preference for the product usage scenario or usage strategy. When recommending these videos to customers for product promotion, accompanying a description that narrates the uploader-preferred highlights depicted in the product video is essential for successful promotion, *i.e.*, attracting potential customers with similar interests or preferences to buy the same product. Different from traditional video-to-text generation problems which mainly concern what exists or happens in the video, this problem cares about what the video uploader wants to highlight. We name this particular problem as *product-oriented* video captioning.

Product-oriented video captioning naturally requires a fine-grained analysis of prominent product characteristics depicted in the video. However, without some general understanding of the product, it can be hard even for a human to grasp what the uploader mainly concerns based on the isolated video. To this end, we view leveraging product-related knowledge as a fundamental ability for product-oriented video captioning. Concretely, we take the structured product aspects from the associated product as prior knowledge since they are easy to acquire (most user-generated videos in e-commerce have on-sell product associations.) and concise in meaning. The structured product aspects arranged by product sellers contain general and basic information necessary for fine-grained video understanding. Figure 1 reveals the task definition, the application scenario in Mobile Taobao, and how automatic tools contribute to product promotion [1].

Recent advances in deep neural networks [18, 33, 41], especially the RNNs, have convincingly demonstrated high capability in handling the general video captioning problem. Most of them [22, 37, 51] incorporate an RNN based encoder-decoder structure with/without attention mechanisms to perform sequential frames encoding and sequential words decoding. However, *product-oriented* video captioning poses some unique challenges. A first

[*] These authors contributed equally to this work.

[†] Corresponding Authors.

Work was performed when S. Zhang and Z. Tan were interns at Alibaba Group.

[1] For the original Chinese descriptions as well the desensitized datasets, please refer to https://github.com/shengyuzhang/Poet

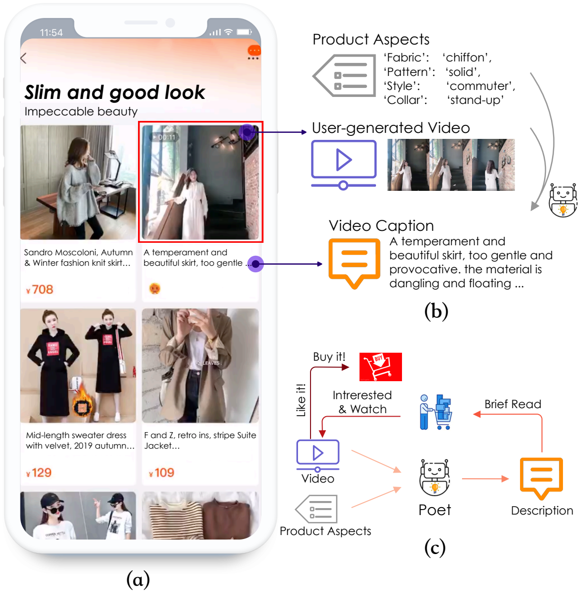

Figure 1: An illustration of *product-oriented* video captioning. (a) A real-world product video recommendation scenario. (b) A showcase of the product aspects, user-generated video, and a video caption. (c) An illustration of how *Poet* contribute to product promotion.

challenge comes from mining the product characteristics from the user-generated product videos and capturing the dynamic interactions of these characteristics within and across frames, as well as exploring the interactions between the product and the background scene. We name this fundamental requirement as product-oriented video modeling, which is essential for discovering what the uploader highlights. Another challenge relates to the joint modeling of discrete product aspects and the video. The cross-modal nature of these inputs makes the modeling even more challenging.

To address these challenges, we propose an approach, named **P**roduct-**o**riented Vid**e**o Cap**t**ioner and abbreviated as ***Poet***. For product-oriented video modeling, *Poet* first represents the product parts as graph nodes and build a spatial-temporal video graph. Then, *Poet* encapsulates the proposed spatial-temporal inference module to perform long-range temporal modeling of product parts across frames and spatial modeling of product parts within the same frame. We also mix a spatial frame node into the video graph to explore the interactions between the foreground (product) and the background (scene). For the second challenge, we propose the knowledge leveraging module, which comprises the knowledge filtering process and knowledge aggregation process.

To accommodate the *product-oriented* video captioning research, we build two large-scale datasets, named buyer-generated fashion video dataset (BFVD) and fan-generated fashion video dataset (FFVD), from Mobile Taobao. There are 43,166 and 32,763 <video, aspects, description> triplets in BFVD and FFVD, respectively. On the language side, descriptions in these two datasets have an extensive vocabulary and considerable product details. Compared with captions in existing video datasets that mostly describe the main objects and the overall activities, descriptions in BFVD and FFVD reflect the characteristics of product-parts, the overall appearance of the product, and interactions between products and backgrounds. Such new features present unique challenges for *product-oriented* video captioning and also for general video analysis.

To summarize, this paper makes the following key contributions:

- We propose to investigate a real-world video-to-text generation problem, *product-oriented* video captioning, to automatically narrate the user-preferred product characteristics inside user-generated videos.
- We propose a novel Poet framework for *product-oriented* video captioning via simultaneously capturing the user-preferred product characteristics and modeling the dynamic interactions between them with a product-oriented spatial-temporal graph.
- We introduce a novel knowledge leveraging module to incorporate the product aspects for product video analysis by performing knowledge filtering, dynamic memory writing, and knowledge attending. *Poet* yields consistent quantitative and qualitative improvements on two *product-oriented* video captioning datasets [1] that are collected in Mobile Taobao.

2 RELATED WORKS

2.1 Video to Text Generation

Most deep learning based video-to-text generation methods [2, 21, 22, 29, 37–39, 48, 50, 51, 56, 57] focus on sequence-to-sequence modeling and employ RNN based encoder-decoder structures. Typically, S2VT [37] firstly formulates video-to-text as a sequence to sequence process. SALSTM [51] improves the decoding process using the effective soft-attention mechanism. HRNE [22] builds a hierarchical RNN design for representing videos. RecNet [39] proposes to re-produce the input frames features after decoding.

More recently, there are works exploiting object-level features in representing the videos [11, 40, 49, 54] for video description generation. They mainly propose to detect salient objects and employ RNNs to model the temporal structure between them. However, they are not directly applicable to *product-oriented* video captioning for the following two reasons: 1) *product-oriented* video captioning requires even more fine-grained video analysis, *i.e.*, product-part characteristic recognition. 2) These methods neglect the spatial interactions between region-region and region-background within frames. *Poet* represents both the detected product-parts and the whole frames as spatial-temporal graphs and employs the graph neural network to model the interactions between product-parts and product-background.

2.2 Knowledge enhanced Video Analysis

Incorporating external in-domain knowledge is a promising research direction [23] for video analysis. There are mainly two kinds of external knowledge, *i.e.*, knowledge graph and topically related documents (*e.g.*, Wikipedia). Knowledge graph based methods [6, 7] typically retrieve the knowledge graph from off-the-shelf knowledge bases such as ConceptNet 5.5 [31] and employ the graph convolution network [12] to perform knowledge reasoning. These methods are not suitable for our task since there are no well-defined relationships among product aspects. For document-based approaches, Venugopalan *et al.*[36] uses the Wikipedia corpus to pre-train a language model (LM) and proposes the late/deep fusion strategies to enhance the decoding RNN with the LM. Whitehead *et al.*[46] first retrieves the relevant document and then use the pointer mechanism to directly borrow entities in the decoding stage. Different from these works, *Poet* performs knowledge leveraging in the product-oriented spatial-temporal inference stage.

3 METHODS

3.1 Overview

After data preprocessing (details in Section 4.1), we represent each product-oriented video as frame-level features $\{\mathbf{f}_i \in \mathbb{R}^{D_f}\}_{i=1,\ldots,N_f}$ where $\mathbf{f}_i$ is the D_v length feature vector for the ith frame f_i, and product-part features $\{\mathbf{p}_{i,j} \in \mathbb{R}^{D_p}\}_{j=1,\ldots,N_p}$ where $\mathbf{p}_{i,j}$ is the D_p length feature vector for the jth product-part in the ith frame. The video-associated product aspects are $\{a_k\}_{k=1,\ldots,N_a}$ and we use the embedding layer to learn an aspect embedding $\mathbf{a}_k \in \mathbf{R}^{D_a}$ of dimension D_a for the kth aspect a_k. We aim to generate a video description $\{w_m\}_{m=1,\ldots,N_w}$ that narrates the preferred product characteristics of e-commerce buyers/fans.

We firstly build a product-oriented spatial-temporal video graph (See Figure 2), which contains both frame nodes and product-part nodes. With the graph representation, the encoder of *Poet* mainly incorporates two sub-modules, *i.e.*, the spatial-temporal inference module for graph modeling, and the knowledge leveraging module for product aspects modeling. These sub-modules can be easily stacked to obtain a progressive knowledge retrieval and knowledge enhanced visual inference process. In the next several subsections, we will formally introduce the building blocks comprising *Poet*, including the graph building process, the spatial-temporal inference module, the knowledge leveraging module, and the attentional RNN-based decoder in detail.

3.2 Product-oriented Video Graph Modeling

3.2.1 *Graph Building.*
To better capture the highlights (*i.e.*, preferred product characteristics) inside the product videos, we propose to represent the videos as spatial-temporal graphs.

Nodes Different from previous works [11, 40, 49, 54] that represent the objects as graph nodes, we represent product parts as nodes to capture the dynamic change of these fine-grained details along the timeline. Product-part features are extracted by a pre-trained CNN-based detector (details in 4.1), and thus these features naturally contain spatial cues. However, since we do not model the product parts along the timeline using RNNs, there is no concept of frame order in the modeling process. To this end, we add an order-aware embedding $\mathbf{o}_i \in \mathbb{R}^{D_p}$ to each product-part feature, which is similar to the position embedding strategy employed in sequence learning [8, 20]. $\mathbf{o}_i$ stands for the embedding for the frame order i. Sharing similar spirit, we further obtain the type-aware product-part representation by adding the part-type embedding $\mathbf{t}_j \in \mathbb{R}^{D_p}$, which stands for the jth part for a particular product, such as waistline and hem. Then, the enhanced product-part feature $\mathbf{p}^e_{i,j}$ can be obtained by:

$$\mathbf{p}^e_{i,j} = \mathbf{p}_{i,j} + \mathbf{o}_i + \mathbf{t}_j, \tag{1}$$

Besides the product-part nodes, we further incorporate the frame node into each frame graph to capture the product as a whole and exploit the correlations between the products and the backgrounds. Similar to the product-part feature, we add the order-based embedding $\mathbf{o}_i$ and a special type embedding $\mathbf{t}_{[\text{frame}]}$ for obtaining the frame-order concept and the frame-type concept, respectively.

$$\mathbf{f}^e_i = \mathbf{f}_i + \mathbf{o}_i + \mathbf{t}_{[\text{frame}]}. \tag{2}$$

We then project the product-part features and the frame features into a common space by employing two linear transformations, *i.e.*, $\mathbf{W}_p \in \mathbb{R}^{D_{pf} \times D_p}$ and $\mathbf{W}_f \in \mathbb{R}^{D_{pf} \times D_f}$.

$$\mathbf{v}^p_{i,j} = \mathbf{W}_p \mathbf{p}^e_{i,j} + \mathbf{b}_p, \quad \mathbf{v}_{i,[\text{frame}]} = \mathbf{W}_f \mathbf{f}^e_i + \mathbf{b}_f. \tag{3}$$

$\mathbf{b}_p, \mathbf{b}_f \in \mathbb{R}^{D_{pf}}$ are the bias terms. Therefore, each frame graph contains N_p product-part nodes $\{v^p_{i,j}\}_{j=1,\ldots,N_p}$ with representations $\{\mathbf{v}^p_{i,j}\}_{i=1,\ldots,N_p}$ and one frame node $v_{i,[\text{frame}]}$ with representation $\mathbf{v}_{i,[\text{frame}]}$. There are totally N_f frame sub-graphs in the whole graph.

Edges To capture the correlations among product-parts within the same frame as well as the interactions between global frame context (*e.g.*, background, the product as a whole) and local part details, we propose a baseline method by fully connecting the product-part nodes and frame node within each frame graph. To obtain a comprehensive understanding and the dynamic change of product-parts across different viewpoints, we fully connect the nodes of the same type (including the frame type) from all frames. The edge weights are obtained using a fully-connected layer:

$$\mathbf{e}_{\iota,\kappa} = \mathbf{W}_e[\mathbf{v}_\iota, \mathbf{v}_\kappa] + \mathbf{b}_e, \tag{4}$$

where $\mathbf{e}_{\iota,\kappa} \in \mathbb{R}$ is the weight of edge between node v_ι and v_κ. We use a linear transformation $\mathbf{W}_e \in \mathbb{R}^{1 \times 2 * D_{pf}}$ with a bias term $\mathbf{b}_e \in \mathbb{R}$ to estimate the correlation of two nodes. [,] denotes the concatenation operation. For convenience, we use G_v to denote the initial video graph and intermediate video graphs since only nodes feature representations are updated.

3.2.2 *Spatial-temporal Inference.*
Although previous works [11, 40, 49] have proposed to capture the fine-grained region-of-interests such as objects and [54] proposes to represent these fine-grained cues as graphs, they all use RNN-based modeling which can be inefficient for its internally recurrent nature and can be less effective in modeling interactions of regions within a frame since these regions have no natural temporal dependencies. To this end, we employ the flexible graph neural networks for spatial-temporal inference. Existing works performing video graph modeling for video relation detection [25], temporal action localization [53], and video action classification [43] often leverage the off-the-shelf Graph Convolutional Networks [13] for information propagation. We propose a new modeling schema by separately modeling the root node and neighbor nodes when aggregation. For neighbor nodes information aggregation, we use an element-wise max function for its effectiveness in the experiment:

$$\bar{\mathbf{v}}^n_{\iota,\varsigma} = \max_{\kappa}\{\mathbf{e}_{\iota,\kappa} * \mathbf{v}_{\kappa,\varsigma}, v_\kappa \in \mathcal{N}(v_\iota)\}, \tag{5}$$

where $\mathcal{N}(v_\iota)$ denotes the neighbor nodes set of the root node v_ι and $\mathbf{v}_{\kappa,\varsigma}$ is ςth element in the feature vector of node v_κ. We note the edge weight $\mathbf{e}_{\iota,\kappa}$ will be re-computed for each information

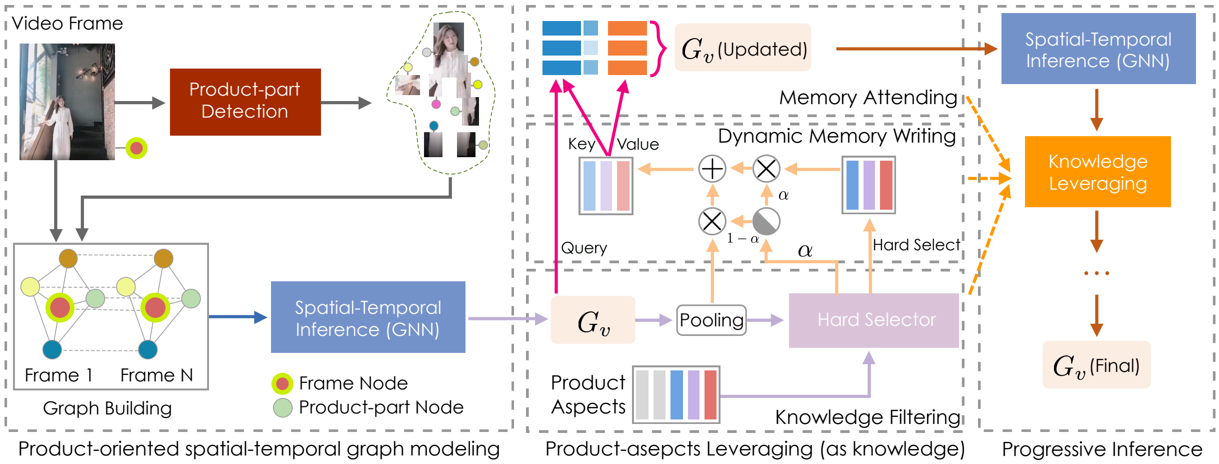

Figure 2: Schematic of the proposed *Poet* encoder. We represent the video as a product-oriented spatial-temporal graph with product parts and frames as nodes. The *Poet* encoder progressively performs spatial-temporal inference and knowledge leveraging to obtain a better understanding of the video with the product aspects. The knowledge leveraging module incorporates three sub-processes, *i.e.*, knowledge filtering, dynamic memory writing, and memory attending.

propagation process. We then perform separate modeling for the root node and the neighbor nodes:

$$\bar{\mathbf{v}}_\iota = \mathbf{W}_n \bar{\mathbf{v}}_\iota^n + \mathbf{b}_n + \mathbf{W}_r \mathbf{v}_\iota + \mathbf{b}_r, \tag{6}$$

where $\mathbf{W}_n, \mathbf{W}_r$ are linear transformations to project the root representation and the aggregated neighbors representation into a common space. $\mathbf{b}_n, \mathbf{b}_r$ are the bias terms. This schema further incorporates an element-wise function for re-weighting the importance of each position as well as a short-cut connection:

$$\tilde{\mathbf{v}}_\iota = \sigma(\mathbf{W}_n^a \bar{\mathbf{v}}_\iota^n + \mathbf{b}_n^a + \mathbf{W}_r^a \mathbf{v}_\iota + \mathbf{b}_r^a) * \bar{\mathbf{v}}_\iota + \mathbf{v}_\iota. \tag{7}$$

where $*$ denotes the Hadamard product. Matrices $\mathbf{W}_n^a, \mathbf{W}_r^a$ and the corresponding biases $\mathbf{b}_n^a, \mathbf{b}_r^a$ model the position-wise importance. σ denotes the element-wise sigmoid function.

3.3 Product-aspects Leveraging

3.3.1 Knowledge Filtering. Leveraging product aspects as knowledge is an essential part of obtaining the basic product information first and a better understanding of the user-generated product videos later. Different from other kinds of external knowledge (*e.g.*, Knowledge Base and Wikipedia), the aspects of the associated product contain noised values that may hurt the performance of product video understanding. For example, there can be both `black-white` and `red-blue` color choices for a certain t-shirt on sell while the buyer/fan may love the `black-white` one and wear it in the video. We therefore devise a knowledge filtering module based on the hard attention mechanism to filter noised values such as `red-blue` for each video. Formally, given the nodes features $\{\mathbf{v}_\tau\}_{\tau=1,\ldots,N_f*(N_p+1)}$ in the video graph G_v and product aspect embeddings $\{\mathbf{a}_k\}_{k=1,\ldots,N_a}$, we perform knowledge filtering by:

$$\mathbf{v}^\circ = \frac{1}{N_f * (N_p+1)} \sum_\tau \mathbf{v}_\tau, \tag{8}$$

$$\alpha_k = \sigma(\mathbf{W}_h[\mathbf{a}_k, \mathbf{v}^\circ] + \mathbf{b}_h), \tag{9}$$

$$\bar{A} = \{a_s | \frac{\exp(\alpha_s)}{\sum_k \exp(\alpha_k)} > \beta_s\}, \tag{10}$$

where $\bar{A}$ denotes the filtered aspect set, which includes aspect a_s with importance α_s over a certain threshold β_s. We empirically set the threshold to the uniform probability $1/N_a$. The importance indicator α_k is computed using a linear transformation $\mathbf{W}_h$ and a bias term $\mathbf{b}_h$. We use the global (or averaged-pooled) representation $\mathbf{v}^\circ$ of the video graph as the filtering context since we aim to remove aspects that are irrelevant to any part of the video. We add a sigmoid function σ to prevent large importance scores, which may lead to a small filtered aspect set after the scores being forward to the $softmax$ function.

3.3.2 Dynamic Memory Modeling. Previous works that incorporate external knowledge for video description generation [36, 46] often leverage the knowledge in the decoding stage, *i.e.*, using pointer mechanism to directly borrow the words/entities from the knowledge document or using attention mechanism to update the decoder hidden state. We propose to progressively retrieve relevant knowledge in the encoding stage, which enables a better understanding of the video for capturing user-preferred product highlights. Specifically, we employ a memory network [9, 32, 45] based approach and enhance it with dynamic memory writing:

$$\bar{\mathbf{a}}_s = \alpha_s * (\mathbf{W}_a \mathbf{a}_s + \mathbf{b}_a) + (1-\alpha_s) * (\mathbf{W}_{gl} \mathbf{v}^\circ + \mathbf{b}_{gl}), \tag{11}$$

$$\omega_{\tau,s} = \mathbf{W}_\omega \tanh(\mathbf{W}_m[\mathbf{v}_\tau, \bar{\mathbf{a}}_s] + \mathbf{b}_m), \tag{12}$$

$$\hat{\mathbf{v}}_\tau = \gamma * \mathbf{v}_\tau + \sum_s \bar{\omega}_{\tau,s} * \bar{\mathbf{a}}_s, \text{ where } \bar{\omega}_{\tau,s} = \frac{\exp(\omega_{\tau,s})}{\sum_o \exp(\omega_{\tau,o})}. \tag{13}$$

where the memory writing process (Equation 11) borrows the importance factor α_s from Equation 9 (Note that the importance factor α_s is in the range $(0, 1)$ after the σ function.) This process helps inhibit relatively irrelevant aspect information (with smaller α_s) and enliven the more relevant ones (with larger α_s). γ controls to what extend the final representation $\hat{\mathbf{v}}_\tau$ depends on the initial representation $\mathbf{v}_\tau$ and we empirically set it to 0.5.

3.4 Progressive Inference Encoder

Since the spatial-temporal inference module and the knowledge leveraging module updates the node representation without modifying the graph structure, we can easily stack multiple STI modules and multiple KL modules. Poet builds the inference encoder by progressively and alternatively performing STI and KL as depicted in Figure 2. In such a design, we aim to not only obtain higher-order graph reasoning (i.e., with access to remote neighbors) but also propagate the leveraged knowledge to the whole graph by the following STI modules. We denote the combination of one STI and one KL as one graph reasoning layer. We use two-layers graph reasoning in the experiment and we observe stacking more layers, which may make node representations over-smoothing and not distinct (i.e., all nodes contain similar information), will lead to a minor performance drop.

3.5 Decoder

Following many previous works [18, 57], we build the decoder with the RNN (here we use gated recurrent unit *GRU* [5]) and soft attention mechanism. We first initialize the hidden state of *GRU* as the global representation of the knowledge-aware video graph:

$$\mathbf{h}_0 = \mathbf{v}^\circ = \frac{1}{N_f * (N_p + 1)} \sum_\tau \mathbf{v}_\tau, \tag{14}$$

For each decoding step t, we attend to each node inside the video graph and aggregate the visual cues using the weighted sum:

$$\varrho_\tau = \mathbf{W}_\varrho \tanh(\mathbf{W}_{md}[\mathbf{v}_\tau, \mathbf{h}_t] + \mathbf{b}_{md}), \tag{15}$$

$$\hat{\mathbf{h}}_t = \sum_\tau \bar{\varrho}_\tau * \mathbf{v}_\tau, \text{where } \bar{\varrho}_\tau = \frac{\exp(\varrho_\tau)}{\sum_\tau \exp(\varrho_\tau)}, \tag{16}$$

where $\mathbf{W}_\varrho, \mathbf{W}_{md}$ are linear transformations and they together model the additive attention [5] with the bias term $\mathbf{b}_{md}$. $\bar{\varrho}$ is the attention weights. The tth decoding process can be formulated as:

$$\mathbf{h}_{t+1} = GRU(\mathbf{h}_t, [\hat{\mathbf{w}}_t, \hat{\mathbf{h}}_t]), \tag{17}$$

where $\hat{\mathbf{w}}_t$ denotes the embedding of the predicted word w_t at step t. For training objectives, we take the standard cross-entropy loss:

$$\mathcal{L} = -\sum_t \log p(w_t). \tag{18}$$

4 EXPERIMENTS

4.1 Product-oriented Video Datasets

Data Collection We collect two large-scale product-oriented video datasets, *i.e.*, buyer-generated fashion video dataset (BFVD) and fan-generated fashion video dataset (FFVD) for *product-oriented* video captioning research. Data samples from both datasets are collected from Mobile Taobao. We collect the videos, the descriptions, and the associated product aspects to form the datasets. Each recommended video has been labeled as buyer-generated or fan-generated by the platform. These two kinds of data are originally generated by users with different background knowledge and intentions. Buyers often focus on the general appearance, salient characteristics, and emotions while descriptions generated by fans often reflect deep insights and understandings about the products. Therefore, we regard these two kinds of videos as individual datasets since they may pose different challenges for modeling. Data collected from real-world scenario may contain noises and we select videos with PV (page views) over 100,000 and with CTR (click through rate) larger than 5%. Videos and descriptions of high quality are more likely to be recommended (more PVs) and clicked (bigger CTR).

Table 1: Comparing BFVD and FFVD with exiting video-to-text datasets (e-comm stands for e-commerce).

Dataset	Domain	#Videos	#Sentence	#Vocab	Dur(hrs)
MSVD [3]	open	1,970	70,028	13,010	5.3
TACos [26]	cooking	123	18,227	28,292	15.9
TACos M-L [27]	cooking	185	14,105	-	27.1
MPII-MD [28]	movie	94	68,375	24,549	73.6
M-VAD [34]	movie	92	55,905	18,269	84.6
VTW [52]	open	18,100	-	23,059	213.2
MSR-VTT [47]	open	7,180	200,000	29,316	41.2
Charades [30]	human	9,848	-	-	82.01
ActivityNet [14]	activity	20,000	10,000	-	849
DiDeMo [10]	open	10,464	40,543	-	-
YouCook2 [55]	cooking	2,000	-	2,600	175.6
VATEX [44]	open	41,300	826,000	82,654	-
BFVD	e-comm	43,166	43,166	30,846	140.4
FFVD	e-comm	32,763	32,763	34,046	252.2

Data Statistics As a result, we collect 43,166 <video, aspects, description> triplets in BFVD and 32,763 triplets in FFVD. The basic statistics and comparison results with other frequently used video captioning datasets are listed in Table 1. The distinguishing characteristics of BFVD and FFVD are 1) these datasets can be viewed as an early attempt to promote video-to-text generation for the domain of e-commerce. 2) Concerning the total number and total length of videos, BFVD and FFVD are among the largest. 3) As for language data, BFVD and FFVD contain a large number of unique words, and the ratio of $\frac{\#Vocab}{\#Setences}$ is among the largest. These statistics indicate that BFVD and FFVD contain abundant vocabulary and little repetitive information. 4) BFVD and FFVD associate corresponding product aspects, which permit not only the product-oriented video captioning research as we do but also a broad range of product-related research topics such as multi-label video classification for e-commerce.

Data Preprocessing For descriptions, we remove the stop words and use Jieba Chinese Tokenizer [2] for tokenization. To filter out those noised expressions such as brand terms and internet slang terms, we remove tokens with frequency less than 30. Descriptions with tokens more than 30 will be shortened, and the max length for product aspects is 12. Following the standard process, we add a `<sos>` at the beginning of each description and a `<eos>` at the last.

[2]https://github.com/fxsjy/jieba

For videos, we inspect 30 frames for each video and extract the product-part features for each frame using this detector [17][3] pre-trained on this dataset [19]. For each frame, the product-part detector will produce a prediction map with the same width/height as the input frame. Each pixel value at a particular part-channel (note that the number of channels is the same as the number of product parts) indicates how likely this pixel is belonging to this part. We then apply the prediction map on the activations obtained from an intermediate layer (by resizing the probability map first, apply softmax on this map to obtain probability weights and finally weighted sum over the intermediate activations per part channel) to obtain part features. Specifically, we use the activations from *pooled_5* and result in 8×64 features vectors for 8 product-parts. We mean-pool the activations from layer *conv4* as the representation for each frame.

For train/val/test split, we employ a random sampling strategy and adopt 65%/5%/30% for the training set, validation set, and testing set, respectively. Therefore, we have 21,554/1,658/9,948 data samples for FFVD and 28,058/2,158/12,950 data samples for BFVD.

4.2 Evaluation Measurements

Natural Language Generation Metrics Concretely, we adopt four numerical assessment approaches, BLEU-1 [24] for sanity check, METEOR [1] based on unigram precision and recall, ROUGE_L [16] based on the longest common subsequence co-occurrence, CIDEr [35] based on human-like consensus. With the user-written sentences as references, these measurements can help evaluate the generation fluency as well as whether the generation model captures the user-preferred highlights depicted in the video.

Product Aspects Prediction Since leverging product aspects as knowledge is essential for product video understanding, it is necessary to evaluate how well the generation model captures such information. We follow the evaluation protocol proposed in KOBE [4] and view the aspects prediction accuracy as the indicator.

Lexical Diversity The above measurements mainly concern the generation quality (fluency, highlight capturing, and aspects capturing), and they cannot explicitly evaluate the generation diversity. As we know, repetitive phrases and general descriptions can be less attractive for potential buyers. Therefore, as in KOBE [4], we view the number of unique n-grams of the sentences generated when testing as the indicator of the generation diversity. We empirically choose 4-grams and 5-grams, following KOBE.

4.3 Comparison Baselines

To evaluate the effectiveness of *Poet*, we compare it with various video description baselines. Since most baselines concern only video information, we re-implement these baselines and add separate encoders (with a similar structure to their video encoders) for product aspect modeling.

(1) *AA-MPLSTM.* Aspect-aware MPLSTM (originally [38]) employs two mean-pooling encoders and concatenate the encoded vectors as the decoder input.
(2) *AA-Seq2Seq.* Aspect-aware Seq2Seq (originally [37]) uses an additional RNN encoder for aspect modeling.
(3) *AA-SALSTM.* Aspect-aware SALSTM (originally [51]) equip the RNN decoder in AA-S2VT with soft attention.
(4) *AA-HRNE.* Aspect-aware HRNE (originally [22]) differs the AA-S2VT by employing hieararchical encoders.
(5) *AA-RecNet.* For Aspect-aware RecNet, we re-construct not only the frame features (as in RecNet [39]) but also the aspect features. Also, there is an additional aspect encoder.
(6) *Unified-Transformer.* We modify the Unified Transformer [21] by replacing the live comments encoder with an aspect encoder.
(7) *PointerNet.* We equip the Seq2Seq model with the entity pointer network, proposed by Whitehead *et al.*[46], for product aspect modeling.

[3] https://github.com/fdjingyuan/Deep-Fashion-Analysis-ECCV2018

4.4 Performance Analysis

In this section, we examine the empirical performance of *Poet* on two product-oriented video datasets, *i.e.*, BFVD and FFVD. We concern a couple of the following perspectives.

Overall generation quality . By generation quality, we mean both the generation fluency and whether the generated sentences capture user-preferred product characteristics. Referenced-based natural language metrics can help reflect the performance since they directly compare the generated sentences to what the video uploaders describe. In a nutshell, the clear improvement over various competitors and across four different metrics demonstrate the superiority of the proposed *Poet* (as shown in Table 2, NLG metrics). Specifically, on BFVD, we obtain +2.46 BLEU (relatively 20.3%) and +0.77 METEOR (relatively 12.1%) improvement over the best (PointerNet) among the competitors. For FFVD, we observe that fan-generated videos sometimes concern collections of clothes and present them for a specific theme (such as `dressing guide` or `clothes that make you look fit`) while only one of the clothes (in the same video) is associated with product aspects. Such a phenomenon may introduce noise and hurt the performance of models designed for single-product modeling. Nevertheless, *Poet* achieves the best performance across the most metrics. We attribute the clear advantage to **1) product-oriented video graph modeling.** *Poet* represents product parts across frames as spatial-temporal graphs, which can better capture the dynamic change of these characteristics along the timeline and find out those distinguishing highlights that are preferred by the user (*e.g.*, distinguishing characteristics of one product part can be highlighted in higher frequencies or with close-up views). In contrast, previous models (including the RNN-based and the transformer-based) are inferior in capturing such characteristics since they either model only the frame-level features without fine-grained analysis or only model the videos in a sequential way. **2) Knowledge enhanced video analysis.** *Poet* firstly perform hard attention to remove *noise aspects* that are of no use for video analysis and then perform dynamic memory writing/attending to progressively enhance the spatial-temporal inference process. This design can be superior over other designs like concatenation or the complex PointerNet design. We will further analyze different knowledge incorporation methods in *Aspect Capturing* and *Ablation studies.*

Aspect Capturing Knowledge incorporation methods can be categorized into three sub-groups: 1) **AA- methods.** We transform

Table 2: Qualitative results of the proposed *Poet* with diverse competitors. Comparisons concern generation quality (NLG metrics), product aspect capturing, and generation diversity. *Poet* achieves the best results on two *product-oriented* video captioning datasets.

Dataset	Methods	NLG Metrics BLEU-1	METEOR	ROUGE_L	CIDEr	Aspect Prediction	Lexical Diversity $n = 4(\times 10^5)$	$n = 5(\times 10^6)$
BFVD	AA-MPLSTM	11.31	6.02	10.08	9.76	54.31	2.94	3.20
	AA-Seq2Seq	11.96	6.14	11.05	11.67	54.85	4.74	4.52
	AA-HRNE	11.82	5.98	10.23	11.86	55.98	5.02	4.73
	AA-SALSTM	11.78	5.88	10.18	11.57	55.93	5.10	4.90
	AA-RecNet	11.17	6.01	11.05	11.67	54.94	5.06	4.92
	Unified-Transformer	11.28	6.32	10.43	12.66	55.12	3.35	2.91
	PointerNet	12.09	6.34	11.19	12.58	56.01	**5.36**	5.02
	Poet	**14.55**	**7.11**	**12.13**	**13.48**	**56.69**	5.16	**5.10**
FFVD	AA-MPLSTM	14.52	7.96	13.85	17.38	61.63	3.15	3.22
	AA-Seq2Seq	14.77	7.87	13.74	18.54	62.01	4.08	3.69
	AA-HRNE	13.58	6.75	12.06	20.10	60.39	4.32	3.88
	AA-SALSTM	**16.25**	7.72	14.63	19.46	62.17	4.58	4.20
	AA-RecNet	15.11	8.03	14.18	19.08	62.21	4.45	4.02
	Unified-Transformer	14.39	7.42	13.45	21.00	62.01	3.39	2.90
	PointerNet	15.28	7.77	14.02	18.85	61.30	4.40	3.99
	Poet	16.04	**8.06**	**14.82**	**21.71**	**62.70**	**4.60**	**4.25**

a video captioning method into a AA-method by adding a separate encoder, which has a similar structure to the existing visual encoder (such as the hierarchical RNN encoder in HRNE), for the product aspect modeling. We concatenate the encoded aspects feature and the encoded visual feature as the initial decoder input. 2) **Decoding-oriented methods.** Unified-transformer and PointerNet are decoding-oriented knowledge incorporation methods, which introduce the external knowledge in the decoding stage as an implicit or explicit reference. Unified-transformer implicitly utilizes the knowledge using the multi-head attention mechanism before prediction. PointerNet explicitly view the attended entities (product aspects) as candidate words for prediction and combine the attention weights and the vocabulary probability distribution before prediction. 3) **Poet**: the analysis-oriented method which aims to obtain a better understanding of the videos with the external knowledge as guidance. The aspect capturing scores are shown in Table 2 (Aspect Prediction). It can be seen that 1) the analysis based method (*Poet*) achieves the best performance on two datasets. As we illustrated earlier, the *product-oriented* video captioning requires a fine-grained analysis of distinguishing characteristics depicted in the video, and the product aspects can better serve as the prior background knowledge to obtain such kind of analysis. 2) The decoding-oriented methods cannot beat the AA- methods on FFVD. This is a reasonable result since the product aspect can be associated with only one cloth in the fan-generated video (as we stated in the *Overall Generation Quality*) and thus directly *borrowing* the aspect words in the decoding may introduce unnecessary local biases.

Generation Diversity The number of generated unique n-grams of different methods are listed in Table 2 (Lexical Diversity). Besides the generation quality, *Poet* can also generate relatively diversified sentences. The PointerNet achieves the best concerning 4-grams in BFVD, which is reasonable since they externally consider the aspects as candidate decoding words.

Table 3: Human judgements on the proposed *Poet* and two typical architectures concerning three task-oriented indicators.

Models	Fluency	Diversity	Overall Quality
AA-RecNet	2.73	3.49	3.04
AA-Transformer	2.66	3.37	2.95
Poet	**2.88**	**3.59**	**3.15**

Human Evaluation We agree with AREL [42] that human judgement is essential for stable evaluation especially when the captions are highly diverse and creative. Following the human evaluation protocol of KOBE [4] and Li *et al.*[15], we randomly select 1,000 instances from the testing set and distribute them to human annotators. The results are listed in Table 3. Compared to the typical Transformer-based design and RNN-based design (AA-RecNet), *Poet* generates more fluent (fluency +0.22/+0.15) and diversified (diversity +0.22/+0.10) descriptions. The indicator *Overall Quality* reflects whether the descriptions capture the product characteristics the video uploader hightlights in the video. In terms of this metric, *Poet* still demonstrates a clear performance improvement over AA-Transformer/AA-RecNet by +0.20/+0.11.

Case Study Figure 3 shows two generation cases on the FFVD testing set. In summary, *Poet* generates more fluent and complete sentences than the AA-Transformer and AA-Recnet, which are typical architectures of transformer-based and RNN-based models, respectively. For example, the phrases (such as "`to in`") generated by AA-Transformer in the first case are confusing, and the whole sentence is incomplete. Besides the generation fluency, *Poet* can generate sentences that better capture the product aspects. In the second case, the phrases "`soft`" and "`young fashionistas`" are derived from the aspects "`soft elastic`", "`youth`", and "`fashion`".

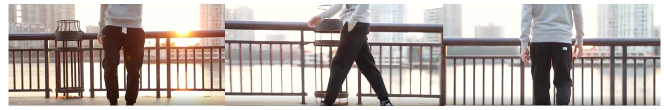

Figure 3: Generation samples of *Poet*, AA-Transformer and AA-Recnet on the FFVD testing set. We present the filtered aspects and the corresponding scores in the proposed knowledge leveraging module.

Table 4: Ablation study on the generation quality of Knowledge Leveraging module and the pointer mechanism.

Dataset	Methods	BLEU-1	METEOR	ROUGE_L	CIDEr
BFVD	*Poet*	**14.55**	**7.11**	**12.13**	**13.48**
	+ pointer	13.26	6.60	11.53	13.18
	- KL	12.43	6.48	10.86	12.25
FFVD	*Poet*	16.04	**8.06**	**14.82**	**21.71**
	+ pointer	**16.13**	7.79	14.50	20.57
	- KL	15.53	7.89	14.18	19.73

To further demonstrate the effectiveness of the proposed knowledge leveraging module, we extract and present the filtered aspects with the corresponding scores. We observe that the proposed KL module successfully filter those aspects of no use, such as product sizes (XL) and the release year *2019* in the first case, and the scores of remaining aspects are consistent to the contribution to the final description. Also, Poet can generate creative while accurate words (*e.g.*, `casual`) beyond the input aspects set based on its understanding of the video.

4.5 Ablation Studies

We conduct ablation studies to verify the effectiveness of proposed modules within *Poet*. We mainly concern the following two issues:

When modeling the video as a graph, does the knowledge leveraging module outperform the pointer mechanism? To answer this question, we construct two models, *i.e.*, the Poet+pointer model, which directly adds the pointer mechanism to *Poet*, and the Poet+pointer-KL model, which remove the proposed knowledge leveraging module from Poet+pointer model. The experiment results on two datasets are listed in Table 4. It can be seen that 1) adding pointer mechanism may hurt the performance in most cases. This further demonstrates the superiority of the analysis-oriented knowledge incorporation method for *product-oriented* video captioning. 2) removing the knowledge leveraging module leads to a clear performance drop (-0.93 CIDEr and relatively 7% in BFVD), which shows the effectiveness of the proposed module.

Do all the proposed modules improve the generation quality? we surgically remove the proposed modules and individually test the performance on BFVD. Table 5 shows the numeric results.

Table 5: Ablation study of the proposed *Poet* by surgically removing controlling components.

Models	BLEU-1	METEOR	ROUGE_L	CIDEr
Poet	**14.55**	**7.11**	**12.13**	**13.48**
- KL	11.85	6.34	10.58	12.09
- STI	11.90	6.09	10.31	9.52

We note that removing the KL (knowledge leveraging) module means ignoring the external product aspects totally. The result indicates the merit of leveraging the KL module to enhance the fine-grained video analysis. By "-SPI", we replace the SPI (spatial-temporal inference) module by the popular Graph Convolutional Networks [12]. The improvement over the GCN verifies the effectiveness of the spatial-temporal inference module.

5 CONCLUSION

In this paper, we propose to narrate the user-preferred product characteristics depicted in user-generated product videos, in natural language. Automating the video description generation process helps video recommendation systems in e-commerce to leverage the massive user-generated videos for product promotion. We propose a novel framework named *Poet* to perform knowledge-enhanced spatial-temporal inference on product-oriented video graphs. We conduct extensive experiments including qualitative analysis, ablation studies, and numerical measurements concerning generation quality/diversity. Experiment results show the merit of video graph modeling, the proposed spatial-temporal inference module, and the knowledge leveraging module for the *product-oriented* video captioning problem. We collect two user-generated fashion video datasets associated with product aspects to promote not only the *product-oriented* video captioning research, but also various product-oriented research topics such as product video tagging.

6 ACKNOWLEDGMENTS

The work is supported by the NSFC (61625107, 61751209, 61836002), National Key R&D Program of China (No. 2018AAA0101900, No. 2018AAA0100603), Zhejiang Natural Science Foundation (LR19F020006), Fundamental Research Funds for the Central Universities (2020QNA5024), and a research fund supported by Alibaba.

REFERENCES

[1] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 65–72.

[2] Elaheh Barati and Xuewen Chen. 2019. Critic-based Attention Network for Event-based Video Captioning.. In *Proceedings of the ACM International Conference on Multimedia*.

[3] David L. Chen and William B. Dolan. 2011. Collecting Highly Parallel Data for Paraphrase Evaluation.. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*.

[4] Qibin Chen, Junyang Lin, Yichang Zhang, Hongxia Yang, Jingren Zhou, and Jie Tang. 2019. Towards Knowledge-Based Personalized Product Description Generation in E-commerce.. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

[5] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches.. In *SSST@EMNLP*.

[6] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. 2018. Watch, Think and Attend: End-to-End Video Classification via Dynamic Knowledge Evolution Modeling.. In *Proceedings of the ACM International Conference on Multimedia*.

[7] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. 2019. I Know the Relationships: Zero-Shot Action Recognition via Two-Stream Graph Convolutional Networks and Knowledge Graphs.. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[8] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *ICML*.

[9] Çaglar Gülçehre, Sarath Chandar, Kyunghyun Cho, and Yoshua Bengio. 2018. Dynamic Neural Turing Machine with Continuous and Discrete Addressing Schemes. *Neural Computation* (2018).

[10] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. 2017. Localizing Moments in Video with Natural Language.. In *IEEE International Conference on Computer Vision, ICCV 2017*.

[11] Yaosi Hu, Zhenzhong Chen, Zheng-Jun Zha, and Feng Wu. 2019. Hierarchical Global-Local Temporal Modeling for Video Captioning.. In *Proceedings of the ACM International Conference on Multimedia*.

[12] Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

[13] Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

[14] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-Captioning Events in Videos.. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*.

[15] Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial Learning for Neural Dialogue Generation.. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[16] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 74–81.

[17] Jingyuan Liu and Hong Lu. 2018. Deep Fashion Analysis with Feature Map Upsampling and Landmark-Driven Attention. In *European Conference on Computer Vision*. Springer.

[18] Sheng Liu, Zhou Ren, and Junsong Yuan. 2018. SibNet: Sibling Convolutional Encoder for Video Captioning.. In *Proceedings of the ACM International Conference on Multimedia*.

[19] Ziwei Liu, Sijie Yan, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2016. Fashion Landmark Detection in the Wild.. In *European Conference on Computer Vision*.

[20] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*.

[21] Shuming Ma, Lei Cui, Damai Dai, Furu Wei, and Xu Sun. 2019. LiveBot: Generating Live Video Comments Based on Visual and Textual Contexts.. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[22] Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang. 2016. Hierarchical Recurrent Neural Encoder for Video Representation with Application to Captioning.. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[23] Yunhe Pan. 2019. Multiple Knowledge Representation of Artificial Intelligence. *Engineering* (2019).

[24] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. Association for Computational Linguistics, 311–318.

[25] Xufeng Qian, Yueting Zhuang, Yimeng Li, Shaoning Xiao, Shiliang Pu, and Jun Xiao. 2019. Video Relation Detection with Spatio-Temporal Graph.. In *MM*.

[26] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. 2013. Grounding Action Descriptions in Videos. *TProceedings of the Conference of the Association for Computational Linguistics* (2013).

[27] Anna Rohrbach, Marcus Rohrbach, Wei Qiu, Annemarie Friedrich, Manfred Pinkal, and Bernt Schiele. 2014. Coherent Multi-sentence Video Description with Variable Level of Detail.. In *GCPR*.

[28] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. 2015. A dataset for Movie Description.. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[29] Xiangxi Shi, Jianfei Cai, Shafiq R. Joty, and Jiuxiang Gu. 2019. Watch It Twice: Video Captioning with a Refocused Video Encoder.. In *Proceedings of the ACM International Conference on Multimedia*.

[30] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. 2016. Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding.. In *European Conference on Computer Vision*.

[31] Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge.. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[32] Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. 2015. End-To-End Memory Networks.. In *Advances in Neural Information Processing Systems*.

[33] Pengjie Tang, Hanli Wang, Hanzhang Wang, and Kaisheng Xu. 2017. Richer Semantic Visual and Language Representation for Video Captioning.. In *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017*.

[34] Atousa Torabi, Christopher J. Pal, Hugo Larochelle, and Aaron C. Courville. 2015. Using Descriptive Video Services to Create a Large Data Source for Video Annotation Research. *Arxiv* (2015).

[35] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4566–4575.

[36] Subhashini Venugopalan, Lisa Anne Hendricks, Raymond J. Mooney, and Kate Saenko. 2016. Improving LSTM-based Video Description with Linguistic Knowledge Mined from Text.. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[37] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond J. Mooney, Trevor Darrell, and Kate Saenko. 2015. Sequence to Sequence - Video to Text.. In *IEEE/CVF International Conference on Computer Vision*.

[38] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond J. Mooney, and Kate Saenko. 2015. Translating Videos to Natural Language Using Deep Recurrent Neural Networks.. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

[39] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. 2018. Reconstruction Network for Video Captioning.. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[40] Huiyun Wang, Youjiang Xu, and Yahong Han. 2018. Spotting and Aggregating Salient Regions for Video Captioning.. In *Proceedings of the ACM International Conference on Multimedia*.

[41] Junbo Wang, Wei Wang, Yan Huang, Liang Wang, and Tieniu Tan. 2018. Hierarchical Memory Modelling for Video Captioning.. In *Proceedings of the ACM International Conference on Multimedia*.

[42] Xin Wang, Wenhu Chen, Yuan-Fang Wang, and William Yang Wang. 2018. No Metrics Are Perfect: Adversarial Reward Learning for Visual Storytelling.. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*.

[43] Xiaolong Wang and Abhinav Gupta. 2018. Videos as Space-Time Region Graphs.. In *European Conference on Computer Vision*.

[44] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. 2019. VaTeX: A Large-Scale, High-Quality Multilingual Dataset for Video-and-Language Research.. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*.

[45] Jason Weston, Sumit Chopra, and Antoine Bordes. 2015. Memory Networks.. In *International Conference on Learning Representations*.

[46] Spencer Whitehead, Heng Ji, Mohit Bansal, Shih-Fu Chang, and Clare R. Voss. 2018. Incorporating Background Knowledge into Video Description Generation.. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[47] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language.. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[48] Jun Xu, Ting Yao, Yongdong Zhang, and Tao Mei. 2017. Learning Multimodal Attention LSTM Networks for Video Captioning.. In *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017*.

[49] Ziwei Yang, Yahong Han, and Zheng Wang. 2017. Catching the Temporal Regions-of-Interest for Video Captioning.. In *Proceedings of the ACM International Conference on Multimedia*.

[50] Ziwei Yang, Youjiang Xu, Huiyun Wang, Bo Wang, and Yahong Han. 2017. Multirate Multimodal Video Captioning.. In *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017*.

[51] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher J. Pal, Hugo Larochelle, and Aaron C. Courville. 2015. Describing Videos by Exploiting Temporal Structure.. In *IEEE/CVF International Conference on Computer Vision*.

[52] Kuo-Hao Zeng, Tseng-Hung Chen, Juan Carlos Niebles, and Min Sun. 2016. Title Generation for User Generated Videos.. In *European Conference on Computer Vision*.
[53] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. 2019. Graph convolutional networks for temporal action localization. In *IEEE/CVF International Conference on Computer Vision*.
[54] Junchao Zhang and Yuxin Peng. 2019. Object-Aware Aggregation With Bidirectional Temporal Graph for Video Captioning.. In *IEEE Conference on Computer Vision and Pattern Recognition*.
[55] Luowei Zhou, Chenliang Xu, and Jason J. Corso. 2018. Towards Automatic Learning of Procedures From Web Instructional Videos.. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*.
[56] Ming Zhou, Nan Duan, Shujie Liu, and Heung-Yeung Shum. 2020. Progress in neural NLP: modeling, learning, and reasoning. *Engineering* 6, 3 (2020), 275–290.
[57] Yongqing Zhu and Shuqiang Jiang. 2019. Attention-based Densely Connected LSTM for Video Captioning.. In *Proceedings of the ACM International Conference on Multimedia*.