Guided Attention for Interpretable Motion Captioning

Karim Radouane, Andon Tchechmedjiev, Sylvie Ranwez, Julien Lagarde

GUIDED ATTENTION FOR INTERPRETABLE MOTION CAPTIONING

Karim Radouane[1], **Andon Tchechmedjiev**[1], **Sylvie Ranwez**[1], **Julien Lagarde**[2]
1. EuroMov Digital Health in Motion, Université de Montpellier, IMT Mines Alès, Alès, France
2. EuroMov Digital Health in Motion, Université de Montpellier, IMT Mines Alès, Montpellier, France

ABSTRACT

While much effort has been invested in generating human motion from text, relatively few studies have been dedicated to the reverse direction, that is, generating text from motion. Much of the research focuses on maximizing generation quality without any regard for the interpretability of the architectures, particularly regarding the influence of particular body parts in the generation and the temporal synchronization of words with specific movements and actions. This study explores the combination of movement encoders with spatio-temporal attention models and proposes strategies to guide the attention during training to highlight perceptually pertinent areas of the skeleton in time. We show that adding guided attention with adaptive gate leads to interpretable captioning while improving performance compared to higher parameter-count non-interpretable SOTA systems. On the KIT MLD dataset, we obtain a BLEU@4 of 24.4% (SOTA+6%), a ROUGE-L of 58.30% (SOTA +14.1%), a CIDEr of 112.10 (SOTA +32.6) and a Bertscore of 41.20% (SOTA +18.20%). On HumanML3D, we obtain a BLEU@4 of 25.00 (SOTA +2.7%), a ROUGE-L score of 55.4% (SOTA +6.1%), a CIDEr of 61.6 (SOTA -10.9%), a Bertscore of 40.3% (SOTA +2.5%). Our code implementation and reproduction details will be soon available at `https://github.com/rd20karim/M2T-Interpretable/tree/main`.

1 Introduction

Motion-to-language datasets such as KIT-ML Plappert et al. [2016] have garnered significant interest in motion-language applications. Most works, have focused on generating motion from language (e.g., Petrovich et al. [2022]), leaving the other direction less explored in recent years, despite a line of long-standing contributions Takano et al. [2006] in robotics. Yet, the motion to language direction is particularly important for a range of applications beyond robotics, particularly in elderly care, healthcare, or rehabilitation medicine to facilitate the indexing of long sequences of movement data stemming from clinical surveillance (e.g., for Parkinson's sufferers) or telesurveillance (elederly care) and the identification of segments indicative of pathologies or emergencies. A summary of a textual description can be an efficient way to concisely alert of concerning precursor signs of pathology or of accidents. Beyond directly captioning videos, pose data has many specificities relative to skeleton geometry, biomechanical constrains, multiscale invariants/fractals that would lead to very different learned representations better suited for tasks related to human movement Phinyomark et al. [2020], Gilfriche et al. [2019].

Moreover, such applications require the ability of precisely aligning generated text with the specific moments movements are performed by adding a temporal dimension, contrarily to all the existing literature, thus transforming motion-to-language generation into motion captioning, which shows many synergies with the significant body of work on captioning in computer vision.

Deep neural networks have been shown to be effective in captioning tasks, due to their ability to learn high-level representations from large amounts of data. The use of convolutional neural networks (CNNs) for feature extraction, combined with recurrent neural networks (RNNs) for sequence modeling, has become a popular approach for captioning in machine learning, particularly within the Encoder-Decoder framework. Strides have also been made in interpretable captioning identifying zones in images or videos that most contribute to the captions, particularly with the introduction of adaptive and guided attention approaches. In this paper, taking inspiration from captioning approaches in vision, we devise an interpretable motion captioning system using guided spatio-temporal attention and an adaptive gate, the first such system for captioning in general, let alone motion captioning in particular. We train and evaluate our architecture

on an updated KIT Motion-Language Dataset Plappert et al. [2016] and HumanML3D Guo et al. [2022a], using the BLEU-4, METEOR-L and Cider metrics, in alignment with current best practices for this task. We show that adding guided attention with adaptive gate leads to interpretable captioning while improving performance compared to higher parameter-count non-interpretable SOTA systems.

Main contributions

1. Architecture interpretable by design, mimicking human-like perception and analysis, allowing to overcome motion captioning bias, thus ensuring correct predictions are made for the right reasons.
2. In the context of motion to text mapping, we propose a first formulation of adaptive gating mechanism, along with spatio-temporal attention.
3. Proposition of methodologies for spatial and adaptive attention supervision.
4. Novel human skeleton partitioning with adequate motion representation to enhance the model interpretability.
5. Action localization, body part identification and motion word distinguishing as side product of interpretability.
6. Tools for qualitative and quantitative evaluations of interpretability, through specific histograms, attention maps and density distributions.

2 Related Work

In spite of the extensive research dedicated to captioning, and although some methodologies overlap with motion-to-language generation, the two have hardly intersected. Meaning there are no motion-captioning systems from the captioning community, and conversely motion-to-language systems could benefit from developments in captioning over other modalities. Particularly, there's been a lot of work on interpretable captioning using attention on images and video that have never been exploited for motion captioning.

There are two mechanisms we take inspiration from for our model, the gating mechanism for information selection and guided attention.

2.1 Adaptive and Guided Attention

Adaptive attention was first introduced by Lu et al. [2017]. Using a gate variable and a visual sentinel vector, adaptive attention implicitly learns when to attend (which words) and where to attend in the image, with the option of focusing attention on the visual sentinel for words that do not refer to meaningful parts of the image.

The process was further refined to hierarchically classify meaningful words by type. For movement captioning this is particularly relevant as only specific words or phrases directly describe the movement and pertain to different levels of description: type of movement, manner, direction, etc.

Guided attention Liu et al. [2017] introduces the supervision of attention and has been shown to improve the performance and accuracy of image captioning models, particularly where the attention mechanism is not able to effectively capture the relevant image content on its own.

Guided attention (spatial) has also been shown to improve captioning performance for video captioning Yu et al. [2017]. This is relevant, as video captioning is closer to motion captioning (over a sequence of poses) due to the presence of the time dimension. The drawback of guided attention is that it needs a correspondence between the text and specific areas in the frames of the sequence. This has been tackled by introducing various levels of semantic analysis of the text to automatically find correspondences.

In our work, we show that for human motion captioning in particular, since the poses have identifiable body parts and since particular actions are performed with specific body parts, guiding attention becomes easier as the alignment process can be automated using semantic parsers focused on action verbs and the propositional structures around them. This approach goes much further than the simple part-of-speech based analyses of Liu et al. [2017] or Yu et al. [2017], or subsequent works.

2.2 Motion Captioning

There are two public datasets mapping human motion sequences to linguistic descriptions: the KIT Motion Language dataset (KIT-ML) Plappert et al. [2016] and the HumanML3D (HML3D) Guo et al. [2022a]. The former is build by crowdsourcing annotations on the KIT Whole-Body Human Motion Database, and the latter by expert annotation of a

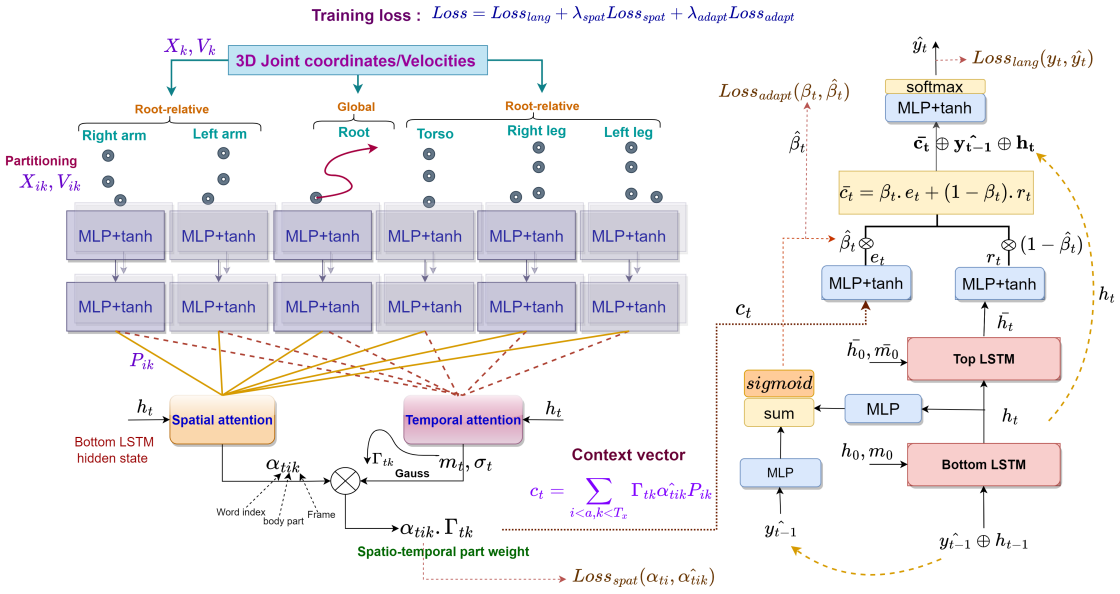


Figure 1: Architecture design with an MLP part-based encoder, and a two-stage LSTM decoder.

subset of the AMASS dataset. Guo et al. [2022a] also introduce an augmentation of KIT-ML using the same process as for HumanML3D. We use both augmented datasets for our experiments, with the same splits as used in Guo et al. [2022a].

Table 1: Training splits, for KIT and Human ML3D after augmentation (aug) Guo et al. [2022a].

Subset	Number	Train	Test	Val.
KIT-ML-aug	motions	4886	830	300
	samples	10408	1660	636
HML3D-aug	motions	22068	4160	1386
	samples	66734	12558	4186

The first deep learning approach on the KIT-ML dataset was introduced by Plappert et al. [2017]. They created a system capable of generating movement from textual descriptions and generating text that describes movement using a bidirectional LSTM encoder-decoder architecture. Later systems mainly focused on language to motion generation [Lin et al., 2018, Ghosh et al., 2021, Petrovich et al., 2022], but motion to language generation has seen a resurgence with the introduction of HumanML3D.

Goutsu and Inamura [2021] explore adversarial seq2seq architectures for motion-to-language generation, achieving SOTA on KIT-ML. Guo et al. [2022b] take a bidirectional motion quantization based transformer approach, and are the first to use HumanML3D for motion-to-language generation. Although they achieve SOTA for motion generation on HumanML3D (the first system evaluated on the dataset). The transformer based method have not led to higher result on reverse direction (motion2text) specifically on KIT-ML (BLEU4 18.4%).

3 Methods

We first present the general model architecture for our captioning approach, followed by a more in-depth presentations of the variants of guided spatial and adaptive attention and finally a presentation of our training hyperparameters.

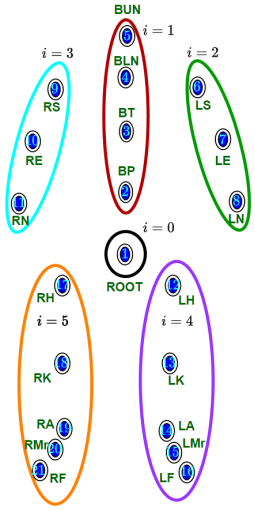


Figure 2: Partitioning of KIT skeleton of 21 joints into 6 body parts: Left Arm, Right Arm, Trunk, Left Leg, Right Leg, Root. A functionally identical segmentation is done for the HML3D skeleton (22 joints).

3.1 Model design for skeleton-based captioning

Here, we discuss our architecture, first giving a general overview, followed by a detailed formal definition. Our model, summarized in Figure 1, is composed of an encoder block, a spatio-temporal attention block and a text generation/decoder block.

Let $\mathbf{X} \in \mathbb{R}^{T_x \times J \times D}$ be the input sequence of motion features of T_x time steps, where J is the number of joints in the skeleton and D is a number of spatial dimensions. We note X_k the 3D joints positions and V_k their corresponding velocities at frame time k.

$$\mathbf{X}_k = [\mathbf{x}_{k,1}, \mathbf{x}_{k,2}, \cdots, \mathbf{x}_{k,J}]$$
$$\mathbf{V}_k = [\mathbf{x}_{k+1,1} - \mathbf{x}_{k,1}, \cdots, \mathbf{x}_{k+1,J} - \mathbf{x}_{k,J}]$$

Body-part partitioning. In this work, we propose another skeleton partitioning, grouping the joints in six parts (cf. Figure 2) rather than commonly used five parts. The motivation is to reduce the global information at the root joint while retaining complementary local information in other parts. We convert the global coordinates to coordinates relative to the root, except for the root itself, which describes the global trajectory of the motion. X_{ik} denotes the group of joints of part i for every frame k as described in Figure 1.

Encoder. Each of the six body parts is embedded by two-layered MLPs with $tanh$ activation, as illustrated in Figure 1. The MLPs encode positions X_{ik} and velocities V_{ik} separately. The final embedding P_{ik} for a given part i and frame k is the concatenation of the position and velocity embeddings. We note by P the frame-level motion features of all human body parts. $P \in \mathbb{R}^{T_x \times a \times h_{enc}}$ where h_{enc} the dimension of the final output encoder and $a = 6$ is the number of body parts.

$$\mathbf{P} = \mathbf{Enc}(\mathbf{X})$$

Decoder. We adopt a two-LSTM decoder configuration, a *Top LSTM* for learning attention weights and language context and a *Bottom LSTM* for final word generation based on the relevant information extracted from language and motion Song et al. [2017]. We note by $\mathbf{y} = (y_1, \ldots, y_{T_y}),\ y_i \in \mathbb{R}^{K_y}$ the sequence of words describing the motion. Let $h_t \in \mathbb{R}^{h_{dec}}$ be the decoder hidden state of the bottom LSTM for a word w_t in the sequence and $\bar{h}_t$ for the Top LSTM. We note by K_y the size of the target vocabulary, and T_x and T_y are respectively the length of the motion sequence and the length of its description. The decoder Dec is used to predict the next word y_t given the adaptive context vector described by $\bar{c}_t$ and the previous word y_{t-1} and the bottom hidden state h_t.

$$p(y_t \mid \{y_1, \cdots, y_{t-1}\}, \bar{c}_t) = Dec(y_{t-1}, \bar{h}_t, \bar{c}_t) \tag{1}$$

The context vector c_t is computed by a spatial-temporal attention mechanism, where temporal attention determines when to focus attention, and spatial attention determines where to focus in the body part graph.

For both *Top LSTM* and *Bottom LSTM,* we initialize the hidden and memory states by *zeros vectors* instead of an average motion feature information. This forces the network to focus and get motion information only through the adaptive context vector, which is important for learning a correct attention map.

3.1.1 Attention mechanisms

Here, we detail the *Spatial* and *Temporal* attention blocs (see Figure 1), and then formally define each successive step of the architecture.

Temporal attention formulation. Temporal attention weights are computed from extracted motion features P and the current decoder hidden state h_t.

$$\boldsymbol{z}_t = \boldsymbol{w}_h^T \tanh(\boldsymbol{W}_p \boldsymbol{P} + ep(\boldsymbol{W}_h \boldsymbol{h}_t)) \tag{2}$$

$$\boldsymbol{\gamma}_t = \text{softmax}(\boldsymbol{z}_t) \tag{3}$$

Here $\boldsymbol{W}_p \in \mathbb{R}^{h_{enc} \times d}$,$\boldsymbol{W}_h \in \mathbb{R}^{h_{dec} \times d}$ and $\boldsymbol{w}_h \in \mathbb{R}^{d \times 1}$ are learnable parameter, ep is an expansion operator mapping to $T_x \times a \times d$, and a the number of body parts.

Moreover, γ_t is the vector of temporal attention weights, such that $\gamma_t = [\gamma_{t,1}, \gamma_{t,2}, \cdots, \gamma_{t,T_x}]$, with $\gamma_{t,k}$ the attention score for the frame k for the word generated at time t.

With the above formulation, we often have discontinuities in the attention maps, yet such discontinuities are undesired, as the action happens continuously in a given frame range. The distribution of attention weights for a *particular motion word* can be modelled as a Gaussian distribution with a learnable mean and standard deviation. The mean m_t and standard deviation σ_t are computed from the previous temporal attention weights γ_{tk}, which are replaced by Γ_{tk} during training in this case (See Figure 1). Intuitively, the mean m_t will approximately represent the center time of action duration described by a motion word w_t, and the spread of the distribution approximately corresponds to the duration of the action.

$$\Gamma_{tk} = \exp\left(-\frac{(k - m_t)^2}{2{\sigma_t}^2}\right) \tag{4}$$

Spatial attention formulation. Then, attention weights are computed for each body part (e.g., Torso, left/right arm, left/right leg) as follows:

$$\boldsymbol{s}_t = \boldsymbol{w}_{\boldsymbol{s}}^{\boldsymbol{T}} \tanh\left(\boldsymbol{W}_{\boldsymbol{p_s}} \boldsymbol{P} + ep(\boldsymbol{W}_{\boldsymbol{h_s}} \boldsymbol{h}_t)\right) \tag{5}$$

$$\boldsymbol{\alpha}_t = \text{softmax}\left(\boldsymbol{s}_t\right) \tag{6}$$

Here $s_t \in \mathbb{R}^a$. The learnable parameters are $\boldsymbol{W}_{p_s} \in \mathbb{R}^{h_{enc} \times d}$,$\boldsymbol{W}_{h_s} \in \mathbb{R}^{h_{dec} \times d}$ and $\boldsymbol{w}_s \in \mathbb{R}^{d \times 1}$. $\alpha_t \in \mathbb{R}^a$ is the spatial attention weight. We have $\alpha_{\mathbf{t}} = [\alpha_{\mathbf{t},\mathbf{1},\mathbf{1}}, \alpha_{\mathbf{t},\mathbf{1},\mathbf{2}}, \cdots, \alpha_{\mathbf{t},\mathbf{a},\mathbf{T_x}}]$ where $\alpha_{t,m,k}$ represents the attention score for part m of the skeleton graph at frame k for the word generated at time t.

Adaptive attention. Non-action words, particularly grammatical words, do not carry any information about the movement. Thus, based on the literature, we propose to learn a gating variable $\hat{\beta}_t$ to decide the proportion to which to use language context over motion features.

$$\hat{\beta}_t = sigmoid(W_b^h.h_t + W_e.(Ey_{t-1})) \tag{7}$$

Where $W_b \in \mathbb{R}^{h_{dec} \times 1}, W_e \in \mathbb{R}^{d_{emb} \times 1}$ are learnable matrices. The gating variable depends on the hidden state, which encodes residual information about generated words up to the time step t, as well as on the embedding of the previous word, as detailed in equation 7.

Context vector. The context vector is derived by weighting the motion features with spatial and temporal attention weights, and averaging across the frame-time dimension 8.

$$c_t = \sum_{k=1}^{T_x} \sum_{i=1}^{a} \Gamma_{tk} \alpha_{tik} P_{ik} \tag{8}$$

Dataset	λ_{spat}	λ_{adapt}	BLEU@1	BLEU@4	CIDEr	ROUGE_L	BERTScore
KIT-ML	0	0	57.3	23.6	109.9	57.8	41.1
	0	3	56.3	22.5	108.4	56.5	39.8
	2	**3**	**58.4**	**24.4**	**112.1**	**58.3**	**41.2**
HML3D	0	0	69.3	24.0	58.8	54.8	38.7
	0	**3**	**69.9**	**25.0**	61.6	**55.3**	**40.3**
	2	3	69.2	24.4	**61.7**	55.0	**40.3**

Table 2: BLEU-4 score result for different supervision modes.

Here, $\bar{h_t}$, the hidden state of the *Top LSTM*, plays the role of *motion sentinel vector* (cf.Figure 1), analogous to the visual sentinel vector in adaptive attention for image captioning. The motion c_t and language information $\bar{h_t}$ are embedded into the same space through an MLP layer with $tanh$ activation (values in [-1,1]), giving e_t and r_t respectively.

Adaptive context vector. The adaptive context vector is given by equation 9. When $\beta_t = 1$ the model uses full motion information and when β_t is close to, 0 the model relies more on language structure.

$$\bar{c_t} = \beta_t.e_t + (1 - \beta_t).r_t \tag{9}$$

Finally, the probability outputs are computed as in Eq. 10, similarly to previous work on video captioning Song et al. [2017], except we include the bottom hidden state. This ensures that the language information of previously generated words is always present, which is important for correct syntax, even for motion words (e.g. jogs, jogging...).

$$p(y_t \mid y_{1:t-1}, \hat{c_t}) = softmax(\tanh\left(W_f.concat([\hat{c_t}; y_{t-1}, h_t])\right)) \tag{10}$$

3.2 Spatial and adaptive attention supervision

To our knowledge, supervision of attention mechanisms with an adaptive gate and spatial attention have never been applied to captioning tasks, let alone motion captioning. In the following, we give a formal definition of how the losses for attention supervision are formulated.

Global loss definition. The standard loss for motion-to-text generation is $loss_{lang}$ (Eq.11). To define the global loss, we add the loss terms for spatial attention $loss_{spat}$, adaptive attention gate $loss_{adapt}$, respectively weighted by $\lambda_{spat}, \lambda_{adapt}$, to control their contributions.

$$Loss = loss_{lang} + \lambda_{spat}.loss_{spat} + \lambda_{adapt}.loss_{adapt} \tag{11}$$

Please note that all loss terms are formulated sample-wise (sample x with source length T_x), we omit batch averaging in the notations for the sake of brevity.

Language loss. The language loss (Eq.12) is defined as the cross entropy between the target and predicted words.

$$Loss_{lang} = -\sum_{j=1}^{T_y} y_j \log(\hat{y}_j) \tag{12}$$

Adaptive loss. To build a ground truth for adaptive attention, we define mapping rules to distinguish between motion words (action verbs and qualifying adjectives, e.g., walk, circle, slowly ...) from non-motion words (e.g., a, person,$\cdots$). We assign $\beta_{\mathbf{t}} = \mathbf{1}$ for motion words and $\beta_{\mathbf{t}} = \mathbf{0}$ for non-motion words.

$$Loss_{adapt} = -\sum_{y} \frac{1}{T_y} \sum_{t=0}^{T_y - 1} \beta_t \log(\hat{\beta_t}) + (1 - \beta_t) \log(1 - \hat{\beta_t}) \tag{13}$$

Spatial loss. The predicted attention score is $\hat{\alpha_{itk}}$ for a given word w_t and part i of the source motion at the frame k. The loss is formulated in Eq. 14, where N_y is a normalization factor that count the number of supervised words for a given target description y.

Dataset	Model	BLEU@1	BLEU@4	ROUGE-L	CIDEr	Bertscore
KIT-ML	RAEs Yamada et al. [2018]	30.6	0.10	25.7	8.00	0.40
	Seq2Seq(Att)	34.3	9.30	36.3	37.3	5.30
	SeqGAN Goutsu and Inamura [2021]	3.12	5.20	32.4	29.5	2.20
	TM2T w/o MT	42.8	14.7	39.9	60.1	18.9
	TM2TGuo et al. [2022b]	46.7	18.4	44.2	79.5	23.0
	Ours-[Spat+adapt] (2,3)	**58.4**	**24.4**	**58.3**	**112.1**	**41.2**
HML3D	RAEs Yamada et al. [2018]	33.3	10.2	37.5	22.1	10.7
	Seq2Seq(Att)	51.8	17.9	46.4	58.4	29.1
	SeqGAN Goutsu and Inamura [2021]	47.8	13.5	39.2	50.2	23.4
	TM2T w/o MT	59.5	21.2	47.8	68.3	34.9
	TM2TGuo et al. [2022b]	61.7	22.3	49.2	**72.5**	37.8
	Ours-[adapt] (0,3)	**69.9**	**25.0**	**55.3**	61.6	**40.3**

Table 3: Text generation performance compared with state-of-the-art approaches and baselines.

$$Loss_{spat} = -\frac{1}{N_y}\sum_{i,t,k} \alpha_{ti}\log(\hat{\alpha_{ti}}) + (1-\alpha_{tik})\log(1-\hat{\alpha_{tik}}) \tag{14}$$

Spatial Guidance. Spatial attention is guided by focusing attention on the body parts through the ground-truth scores α_{itk}. These values are constructed by first, categorizing motion words into those describing local motion (e.g., waves, stretch arms, pick…) and those of trajectory words (circle, forward, clockwise…). Then, motion words describing trajectory characteristics are associated to the root, and local actions associated with relevant parts, e.g., *waving* associated to arms part, *kicking* to legs part). These associations are based on a predefined dictionary (See more details in Appendix A.2).

For global motion words such as *jump*, we let the model freely decide what body parts are relevant, as it depends on the context (*jumping jack or in place or forward* …). Formally, we define the target word as w_t and the corresponding ground truth α_{ti} for each part $i \in [0, a]$. Given a word $w_t = waves$, the corresponding α_{ti} is set to 1 for arms and 0 for other parts. Similarly, for a movement with global trajectory, the attentions α_{ti} are pushed to be maximal on the root joint, which contains the global trajectory information.

For motion words not matching a specific body part we do not guide attention but let the scores be learned freely and inferred from synonym words. By these mechanisms, put it together, we push the model to determine the relevant word to output based on the most attended body part and generalize to uncategorized words. Since the exact time of the action is unknown, we keep the ground truth weight α_{itk} frame-independent (see.15). The temporal attention block focuses on learning action time and performing temporal filtering of spatial weights through element-wise multiplication (cf. Figure 1).

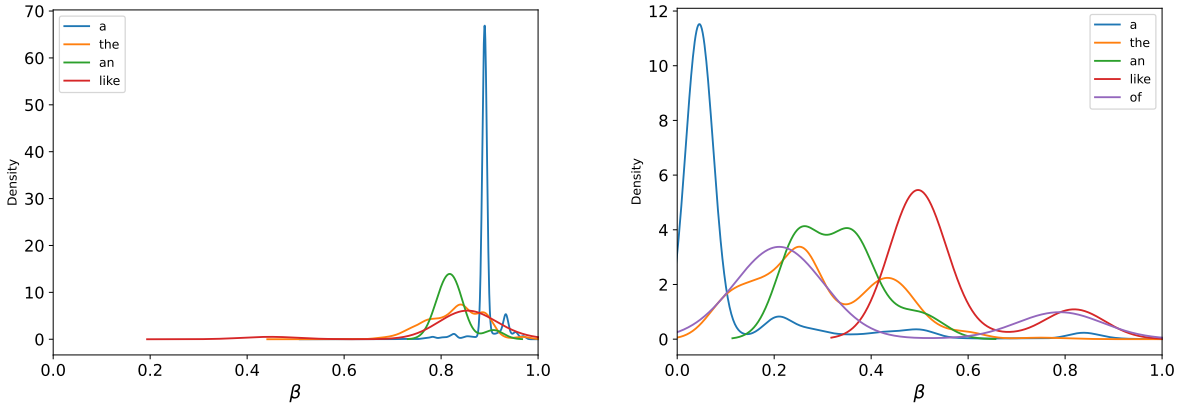
$$\forall k \in [0, T_x - 1] : \alpha_{tik} = \alpha_{ti} \tag{15}$$

4 Experimental validation and discussion

In this section, we first present the overall hyperparameters for the evaluation, then perform an ablation study to determine the impact of adaptive and guided attention, followed by a comparative evaluation against SOTA baselines, to finish with a qualitative evaluation of interpretability.

4.1 Evaluation protocol

All evaluations are performed on the same splits of the dataset as the other SOTA systems to ensure reproducibility, and use standard text generation metrics widely used for captioning tasks or motion-to-language models. While all reproduction details will be made available at the code repository, we give some main hyperparameters considered. For KIT-ML, the word embedding size is set to $d_{emb} = 64$, the decoder hidden size to $h_{dec} = 128$, the dimension of each output of MLP_i for layer 1 is 128 and 64 for layer 2, for joint positions and for velocities. After concatenation, we obtain 128 joint-velocity features per frame. For HumanML3D, the word embedding size is set to $d_{emb} = 128$, the



(a) Without gate supervision.

(b) With gate supervision.

Figure 3: β density distribution over test set for some non-motion words (stemmed) on HumanML3D.

decoder hidden size to $h_{dec} = 256$, the dimension output of MLP_i for layer 1 to 256 and to 128 for layer 2, which is the final output dimension for joint positions and for velocities. After concatenation, we obtain 256 joint-velocity features per frame.

We use the AdamW Loshchilov and Hutter [2017] optimizer with a weight decay of $1e-4$ and $1e-5$ respectively for KIT-ML and HumanML3D, both with a teacher forcing strategy of ratio 0.5. For loss supervision, we configure a search space for $(\lambda_{spat}, \lambda_{adapt})$ and run the search using WandB [Biewald, 2020] (See appendix A.3).

4.2 Quantitative evaluation

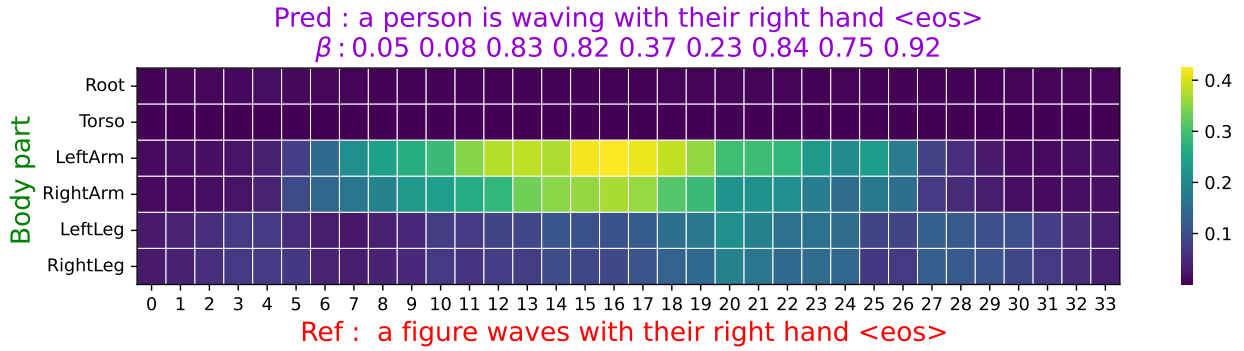
4.2.1 Impact of adaptive and spatial supervision

We train the model for both datasets with different combination of (λ_{spat} and λ_{adapt}), where zero corresponds to the case without any supervision. Table 2 presents the compared performance of our system for the combinations considered, using the text generation metrics defined above. The gate (adapt) and spatial (spat) supervision, perform well when used together on KIT-ML (small), as we obtain a BLEU@4 of 24.4 (+0.8%) compared to 23.6% with no gate and attention guidance. For HumanML3D adaptive attention was always beneficial (+1% on BLEU@4, + 0.3% on ROUGE), but guided spatial attention slightly degraded exact matching scores (BLEU@4, ROUGE) compared to just adaptive attention. We can hypothesize that guiding spatial attention effectively leads to semantically equivalent sentences that use a slightly more diverse vocabulary, leading to a dip in the exact n-gram matching metrics.

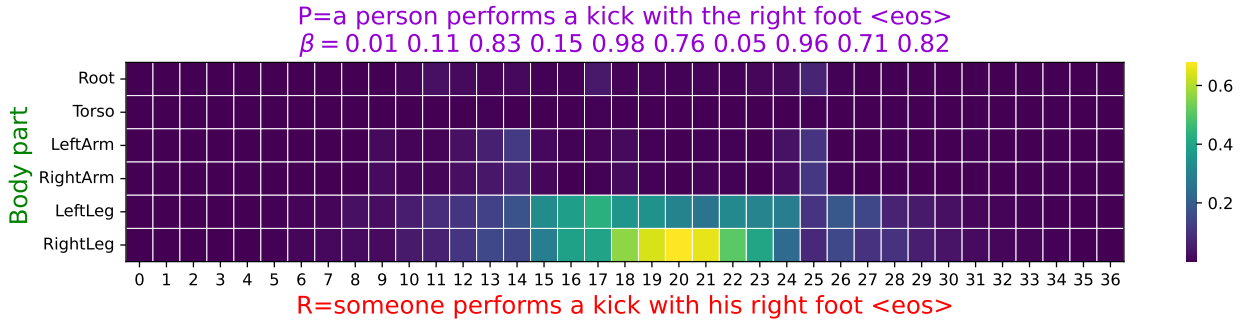
Evaluation against SOTA baselines Table 3 presents the comparison to SOTA systems for KIT-ML and HumanML3D. Only Guo et al. [2022b] use the updated version of KIT-ML augmented by language adaptation, but they replicate several SOTA architectures on the same split of the dataset, which we also report. For HumanML3D, we report systems released after the dataset, and some of their older baselines replicated on HumanML3D.

Our interpretable architecture design, was sufficient to surpass SOTA results even without attention supervision (Table (2 $\lambda_{spat} = \lambda_{adapt} = 0$) vs. Table 3 (model `TM2T`)). When adding attention supervision, our approach performs very significantly better than other SOTA approaches across all metrics on KIT-ML (+6% BLEU@4, +14.1% on ROUGE-L, +32.6% CIDEr, +18.20% Bertscore) and significantly better on HumanML3D (+2.7% BLEU@4, +6.1% ROUGE-L, +2.5% Bertscore, except for CIDEr at -10.9%), including the transformer-based TM2T. Our system performs more consistently across datasets, with an order of magnitude fewer parameters, and with an interpretable output.

We hypothesize that taking into account skeletons-structure is one of the advantages we have, this was shown to be an effective improvement in motion encoders for action recognition architectures and translated well to motion-to-language generation. In addition, we have shown that facilitating interpretability improves performance, while being essential for many potential applications to motion captioning (e.g., indexing long movement sequences to detect specific pathology markers on specific parts of the body). We have also observed that adding adaptiveness and attention guidance tends to generate more diverse sentences that are semantically equivalent (lower exact matching metrics, higher semantic matching metrics), although the effect is small. More analysis are provided in appendix A.4.



(a) HML3D-(2,3), word *waving*



(b) KIT-(2,3), word *kick*

Figure 4: Spatial temporal attention map for different motion words. The Y axis represents body parts, while the X axis represents frames. The color scale represents the intensity of the attention score for the given body part at each frame.

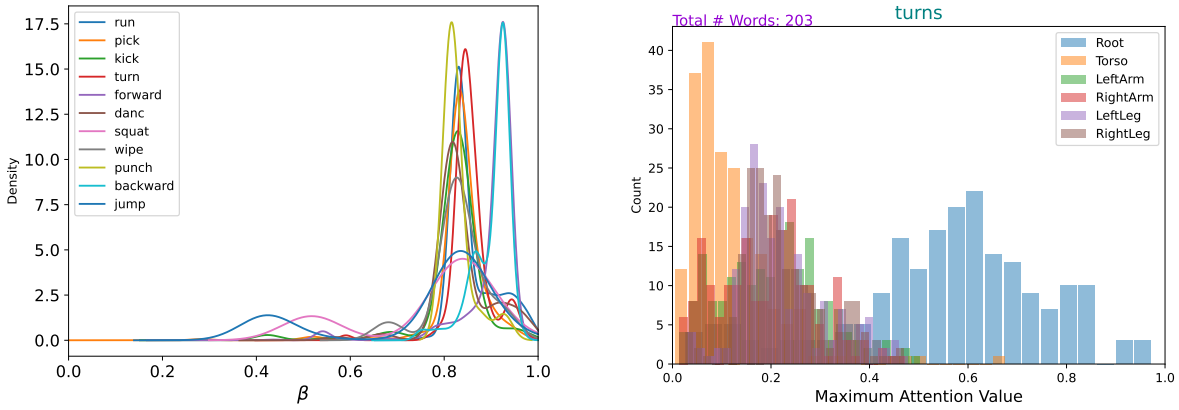
β (gate)	**Prediction**
1	***kicking*** kicks with right leg <eos>
adaptive	a person kicks with their right leg <eos>
1	***jumping jacks*** and jumping jacks <eos>
adaptive	a person does jumping jacks <eos>
1	***walking forward*** in a diagonal line <eos>
adaptive	a person walks forward in a straight line <eos>
1	***punching*** boxing and *moving hands around* <eos>
adaptive	a person is boxing with *both hands* <eos>
1	**jogs** in in place <eos>
adaptive	the person is jogging in place <eos>
1	***throws*** throws with right hand then then throws <eos>
adaptive	a person throws something ball with their right hand then then catches something with both hands <eos>

Table 4: Comparison of the prediction when setting $\beta = 1$ and adaptive on HML3D (adapt $(0, 3)$).

4.3 Qualitative analysis of generation quality and interpretability

In this section, we propose to conduct several analyses on the interoperability of the proposed architecture through a discussion on the effect of attention guiding and architecture design.

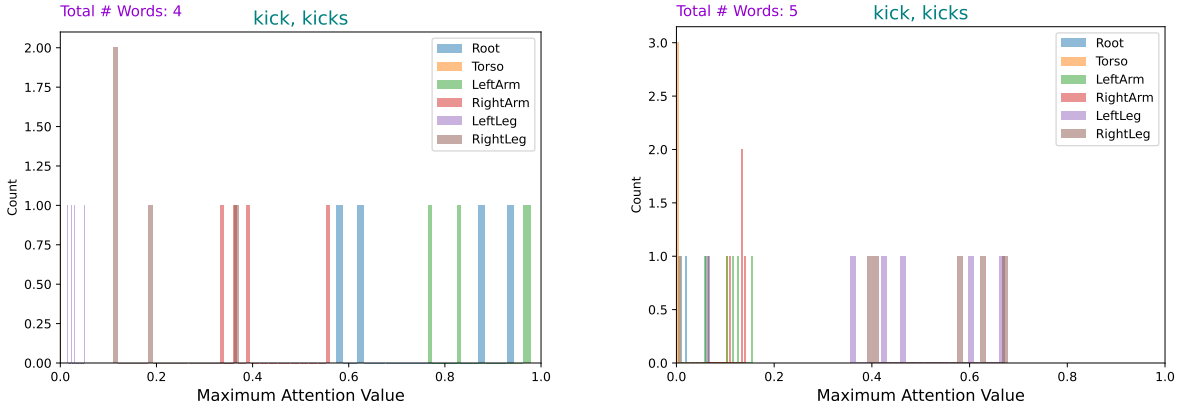
Adaptive attention When training a model without guiding spatial attention, we observe that the β gate frequently takes higher values for non-motion words (a:0.9, the:0.8) as illustrated in Figure 3a.



(a) With gate supervision.

(b) Word *turn*.

Figure 5: β test set density distribution for a few motion words stems on HumanML3D and part distribution for word "turn" (global trajectory).



(a) Without spatial supervision.

(b) With spatial supervision.

Figure 6: Effect of spatial supervision on KIT-ML.

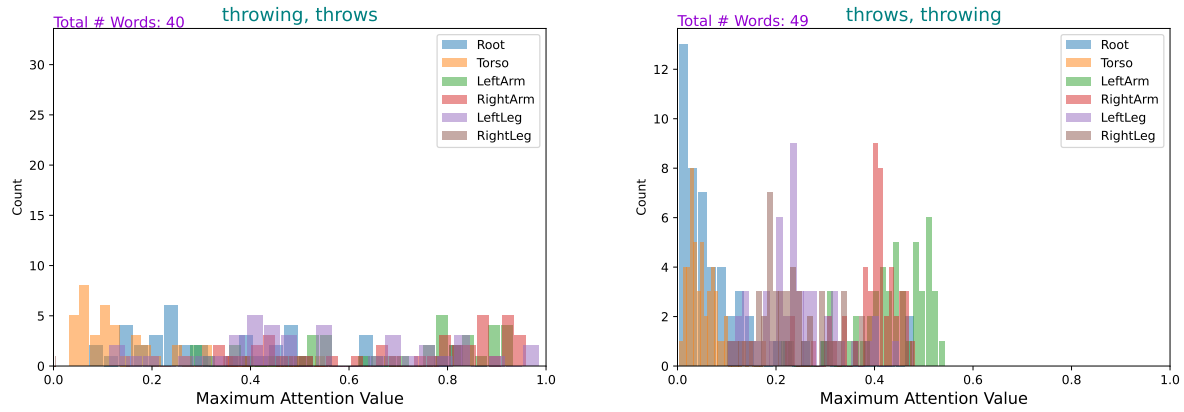
This behavior degrades performance, as seen in Table 2 for HumanML3D. We hypothesize that this leads the weights of the spatio-temporal attention to receive more gradient updates for non-motion words through $\bar{c}_t$, which can make picking-out the more important motion words more difficult.

However, when we introduce adaptive gate supervision (cf. Figure 3b), the model more frequently assigns less weight β to non-motion words and begins to learn how to make decisions automatically, as illustrated also in Figure title 4b. This allows the model to focus on using the context motion information only for motion words. The Table 7 gives a concrete demonstration. Additionally, β values tend to be higher correctly for motion words (trajectory, direction, action, body part...) as illustrated in Figure 5a.

Spat+adapt attention The adaptive attention forces the model to use the context vector for the prediction, while guided spatial attention enhances the learned attention maps. We draw the hypothesis that both supervision types lead to learning attention map that better match human perception.

To demonstrate the role of each of the context vectors c_t and LSTMs hidden states ($\bar{h}_t$,h_t), we fix the β value at 1 and show a representative examples compared to adaptive gate in Table 7. The context vector is only used to recognize motion characteristics (interestingly synonym motion words are grouped e.g. *boxing, punching*). While the hidden states provides the language structure and context.

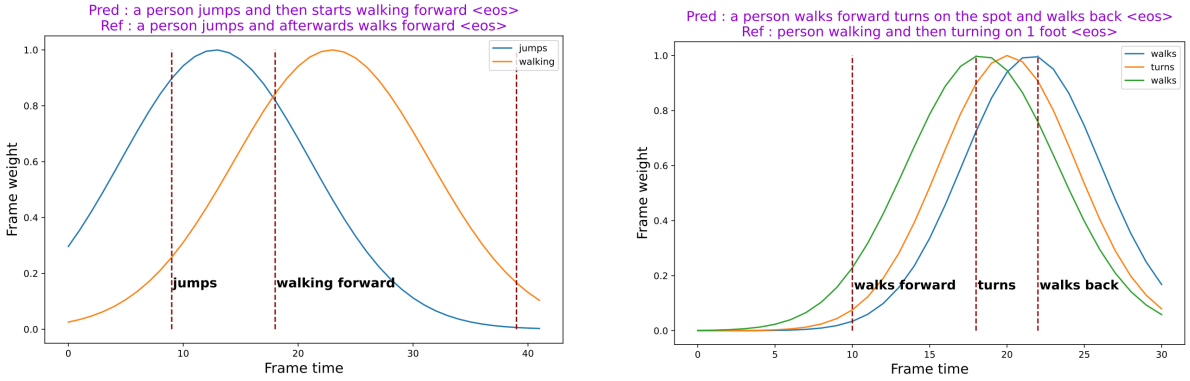
Regarding spatio-temporal attention maps, the model effectively focuses on relevant parts for motion word prediction, as can be seen in the example in Figure 4 for motion words *kicks* and *waves*.



(a) Without spatial supervision.

(b) With spatial supervision.

Figure 7: Effect of spatial supervision on HumanML3D.



(a) Multi-action.

(b) U-turn motion.

Figure 8: Temporal Gaussian window displayed for different motion words given a prediction on KIT-ML.

5 Applications

Designing a motion captioning framework while maintaining the interpretability may be very challenging, but has several advantages and consequences that go beyond focusing solely on increasing NLP metrics. In addition to overcome bias in captioning, thus to be right for right reason, we can derive other information leveraging the interpretability provided by attention mechanisms. In this section, we propose methods for inferring information such as body part identification and action localization. Moreover, we discuss the transferability of our methodologies to other tasks and provide concrete solutions.

Body part identification. We can illustrate the effectiveness of our architecture in learning a correct body part association though spatio-temporal attention by viewing the density distribution for maximum attention across time per each body part for some motion words.

For KIT-ML, Figure 6 illustrates the compared spatial attention part weights with supervision 6b vs. without 6a for the action *kick*. In the former attention is concentrated on the legs, while in the latter it focuses on *Root* and *Left Arm*, which doesn't suit the action. Quantitatively, this leads to an increase in performance as previously seen (cf. Table 2).

For Human-ML3D, in Figure 7, the spatial attention correctly focuses on arms for *throw* in both cases (w and w/o supervision). We believe that with more training data, containing more diverse descriptions (HML3D) body part encoding alone may be sufficient to implicitly learn correct attention maps (see other examples in appendix A.5). However, it's still required for small size datasets, as seen for KIT-ML.

Action localization. Another aspect that emerges from temporal Gaussian attention weights is action localization. The architecture shows some ability to identify motion onset without temporal supervision. We can derive the action onset

from spatio-temporal attention maps as illustrated in Figure 8 where we also show their actual onset time (identified manually trough manual visualization).

Transfer to adjacent tasks. Similar tasks such as action recognition and localization can benefit from the proposed formulations. For instance, our proposed motion encoder and skeleton partitioning could be used for skeleton based action recognition, and the number of layer could be adapted regarding the size and the nature of the considered dataset.

In the case of action localization given a continuous stream, this task could be also cast as sequence to sequence learning. In this context, attention weights could be used to infer the action start/end times as an unsupervised learning without the need of action time labels. If the time annotation is available, this can be used to supervise the spread of temporal weights, further enhancing the accuracy of action location and spatio-temporal attention maps.

Moreover, our formulation of spatial weights supervision could be used leveraging the prior knowledge about the body parts involved in a given action. Looking into other scenarios such as vision based captioning, for image input for each given visual words in the caption, our spatial supervision could be transformed into maximizing the attention weights on the spatial regions corresponding to the object described by the given word. While in the case of video input, the temporal attention block will be added. In both cases, the encoder could simply use a pretrained CNN network that gives spatial grid image features as analogy to skeleton body parts, then using the spatio-temporal formulation along with the supervision of adaptive learnable gate, that should put emphasis on visual words.

Finally, the interpretability could be evaluated using the proposed density function for adaptive attention and histograms for attention distribution on spatial locations in other captioning context.

6 Conclusion

We have introduced guided attention with adaptive gate for motion captioning. After evaluating the influence of different weighting schemes for the main loss terms, we have found that our approach leads to interpretable captioning while improving performance over SOTA. Interpretability is very important to consider when designing an architecture, it's gives insights on model capability to perform a true reasoning. This insures the ability of generalizing instead of memorizing. The proposed model addressed the two challenges, given an interpretable result with accurate semantic captions. Our first perspective is to propose a reference protocol for a systematic evaluation of interpretability based on human assessors, the lack of which is likely the most significant limitation for quantitative evaluation of interpretability. We have used a rule-based language processing approach for word categorization. However, we can likely improve performance further by proposing a more sophisticated and fine-grained semantic analysis of motion words in their context. The model and proposed methodology can be transposed to other captioning tasks, such as supervision of spatial attention weights in action recognition tasks.

References

Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT motion-language dataset. *Big Data*, 4(4):236–252, dec 2016. doi:10.1089/big.2016.0028. URL `http://dx.doi.org/10.1089/big.2016.0028`.

Mathis Petrovich, Michael J. Black, and Gül Varol. Temos: Generating diverse human motions from textual descriptions. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 480–497, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-20047-2.

W. Takano, K. Yamane, T. Sugihara, Kou Yamamoto, and Y. Nakamura. Primitive communication based on motion recognition and generation with hierarchical mimesis model. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 3602–3609, 2006. doi:10.1109/ROBOT.2006.1642252.

Angkoon Phinyomark, Robyn Larracy, and Erik Scheme. Fractal analysis of human gait variability via stride interval time series. *Frontiers in Physiology*, 11, 2020. ISSN 1664-042X. doi:10.3389/fphys.2020.00333. URL `https://www.frontiersin.org/articles/10.3389/fphys.2020.00333`.

Pierre Gilfriche, Laurent M. Arsac, Estelle Blons, and Véronique Deschodt-Arsac. Fractal properties and short-term correlations in motor control in cycling: influence of a cognitive challenge. *Human Movement Science*, 67:102518, 2019. ISSN 0167-9457. doi:https://doi.org/10.1016/j.humov.2019.102518. URL `https://www.sciencedirect.com/science/article/pii/S0167945719300326`.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5152–5161, June 2022a.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3242–3250, 2017. doi:10.1109/CVPR.2017.345.

Chenxi Liu, Junhua Mao, Fei Sha, and Alan Yuille. Attention correctness in neural image captioning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4176–4182. AAAI Press, 2017.

Youngjae Yu, Jongwook Choi, Yeonhwa Kim, Kyung Yoo, Sang-Hun Lee, and Gunhee Kim. Supervising neural attention models for video captioning by human gaze data. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6119–6127, 2017. doi:10.1109/CVPR.2017.648.

Matthias Plappert, Christian Mandery, and Tamim Asfour. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. *Robotics and Autonomous Systems*, 109: 13–26, 5 2017. doi:10.1016/j.robot.2018.07.006.

Angela S. Lin, Lemeng Wuk, Rodolfo Corona, Kevin Tai, Qixing Huang, and Raymond J. Mooney. Generating animated videos of human activities from natural language descriptions. In *Proceedings of the Visually Grounded Interaction and Language Workshop at NeurIPS 2018*, December 2018. URL `http://www.cs.utexas.edu/users/ai-labpub-view.php?PubID=127730`.

Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1376–1386, 2021. doi:10.1109/ICCV48922.2021.00143.

Yusuke Goutsu and Tetsunari Inamura. Linguistic descriptions of human motion with generative adversarial seq2seq learning. In *Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4281–4287. IEEE, 2021. ISBN 978-1-7281-9077-8. doi:10.1109/ICRA48506.2021.9561519. URL `https://ieeexplore.ieee.org/document/9561519/`.

Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 580–597, Cham, 2022b. Springer Nature Switzerland. ISBN 978-3-031-19833-5.

Jingkuan Song, Lianli Gao, Zhao Guo, Wu Liu, Dongxiang Zhang, and Heng Tao Shen. Hierarchical lstm with adjusted temporal attention for video captioning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 2737–2743. AAAI Press, 2017. ISBN 9780999241103.

Tatsuro Yamada, Hiroyuki Matsunaga, and Tetsuya Ogata. Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. *IEEE Robotics and Automation Letters*, 3:3441–3448, 10 2018. ISSN 2377-3766. doi:10.1109/LRA.2018.2852838. URL `https://ieeexplore.ieee.org/document/8403309/`.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2017. URL `https://arxiv.org/abs/1711.05101`.

Lukas Biewald. Experiment tracking with weights and biases, 2020. URL `https://www.wandb.com/`. Software available from wandb.com.

A Appendix

This appendix provides more details on the method implementation, and more visualization for global evaluation of interpretability. Furthermore, we discuss the effectiveness of architecture design. All following analysis were conducted on the Test set.

A.1 Motivation

Our approach is focusing on interpretability while ameliorating motion captioning performance. This comes with additional challenging question on accurate methods for interpretability evaluations. To address this question, a first attempt is to draw multiple visualizations. However, for a global evaluation on test set, this become infeasible. To overcome this limitation, in addition, a simple solution, yet effective, is to display histogram and density distributions for attention weights across all test set instead of just sample wise visualizations.

Recalling the main contributions of our paper: interpretable architecture design, supervision of adaptive and spatial attention and effective tools for global interpretability evaluation. Thus, regarding each contribution aspect, we will show the concrete effectiveness of associated theoretical formulations.

A.2 Ground Truth generation for supervision

Predefined dictionary. We manually define a dictionary based on representative words in the dataset describing different motion characteristics. Intentionally the dictionary doesn't cover all datasets actions with their synonyms, we want the model to be able to generalize to remaining unsupervised words for their spatial and gate attention. We will see later that the model effectively converges for this intended behavior.

Category	Words	Body part
Trajectory	circle, circuit, clockwise, anticlockwise, forward, backward	Root
Local motion	open, waves, wipe, throw, punch, pick, boxing, clean, swipe, catch, handstand, draw	Arms
	kick, stomp, lift, kneel, squat, squad, stand, stumble, rotate	Legs
	bend, bow	Torso
Connection words	a, is, the, of, his, her, its, on, their	-
Subject	person, human, man	-

Table 5: Predefined dictionary for both datasets.

During training, the words in the Table 5, and targets words, are stemmed to find correspondence for spatial weight supervision.

Spatial attention supervision. The ground truth spatial attention weights α_{ti} are generated based on the predefined dictionary and it's same for all frames, the temporal attention is the responsible for temporal filtering.

Adaptive attention supervision. The ground truth β_t is generated based on the Part Of Speech (POS) tagging.

A.3 Hyperparameters selection

We run experiments for different values of $(\lambda_{spat}, \lambda_{adapt})$. The quantitative results are reported in the Table 6.

Dataset	λ_{spat}	λ_{adapt}	BLEU@1	BLEU@4	CIDEr	ROUGE_L	BERTScore
KIT-ML	0	0	57.3	23.6	109.9	57.8	41.1
	0	3	56.3	22.5	108.4	56.5	39.8
	1	3	57.6	23.5	102.6	57.2	40.1
	2	3	**58.4**	**24.4**	**112.1**	**58.3**	**41.2**
	3	5	57.6	23.7	105.7	57.5	40.9
	5	5	56.5	22.0	99.4	56.8	39.9
HML3D	0	0	69.3	24.0	58.8	54.8	38.7
	0	3	**69.9**	**25.0**	61.6	**55.3**	**40.3**
	0.1	3	69.5	23.8	58.7	55.0	38.9
	0.25	3	68.7	23.8	59.7	54.7	39.3
	0.5	3	68.8	23.8	60.0	55.0	38.6
	1	3	68.7	23.7	58.2	54.6	39.0
	2	3	69.2	24.4	**61.7**	55.0	40.3
	3	3	68.3	23.2	56.5	54.5	37.1

Table 6: Spat+adapt supervision impact w.r.t each corresponding weights.

A.4 Architecture compounds effectiveness

We aim in the following visualizations to demonstrate the global effectiveness of architecture design of each separate compound:

- Functionality of gating mechanism.
- Impact of Part based motion encoding.
- Spatio-temporal attention blocks.

Gating mechanism. The gate variable β allows the model to use or not the motion information given the word time step. To visualize this internal process of switching between motion and language, we display predictions for the best model on KIT-ML (Table. 7). As we see in the following Table, the context vector ($\beta = 1$) is successfully used for all motion characteristics, *action*, *speed*, *body parts*, *trajectory*, *direction*.... Particularly, we note that the end token `<eos>` is also motion related, as outputting this word depends on the end of the relevant human motion range. The gating process is illustrated by Figure 9, this mechanism prevents the decoder from attending to motion for non-motion word. Consequently, the motion encoder is prevented from receiving important gradients updates for non motion words.

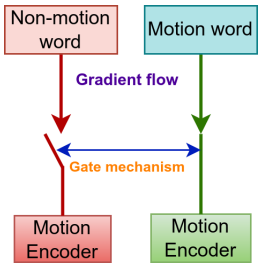
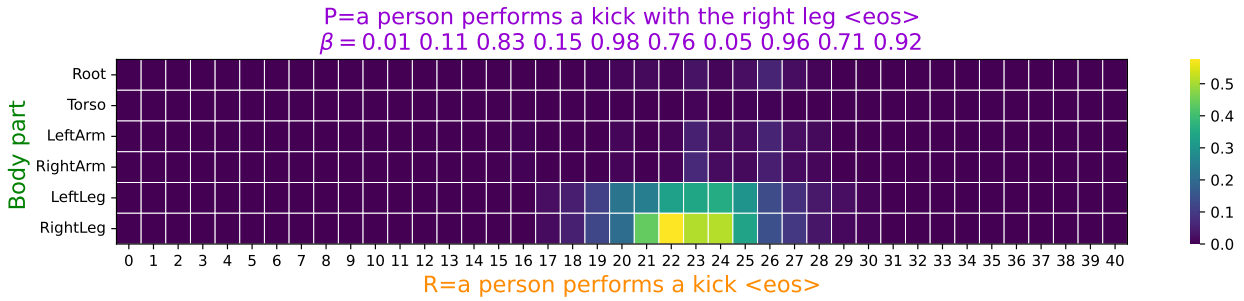


Figure 9: Illustration of our gating mechanism during training.

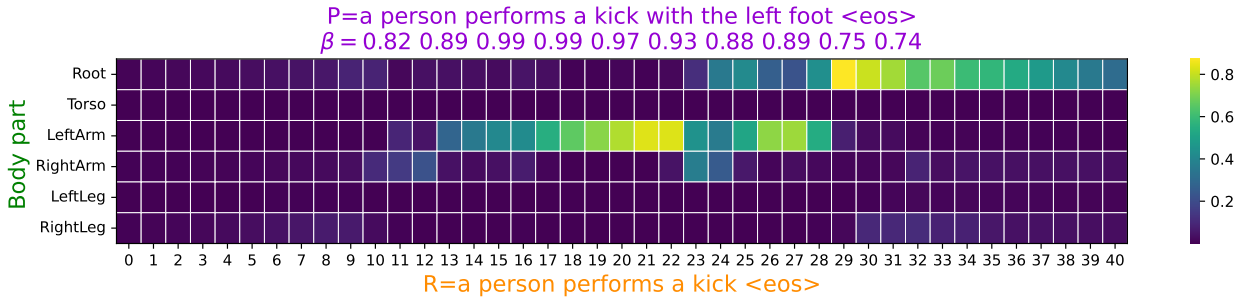
β (gate)	Prediction
1	waves waves waving with both hands <eos>
Adaptive	the person is waving both hands <eos>
REF	the person is waving both hands <eos>
1	walks walks slowly <eos>
Adaptive	a person walks slowly <eos>
REF	a person walks forwards quite slowly <eos>
1	kicking kicking kicking with left leg <eos>
Adaptive	a person kicks something with its left foot <eos>
REF	a human kicks something with his left foot <eos>
1	turns walking and turning on walking <eos>
Adaptive	a person turns on his right foot <eos>
REF	a human turns abruptly <eos>
1	jumping jumps forward <eos>
Adaptive	a person jumps with both legs <eos>
REF	a person jumping 1 step <eos>

Table 7: Comparison of the prediction when setting $\beta = 1$ and adaptive on KIT-ML (Spat+adapt $(2, 3)$).

Spatial+adapt attention supervision [KIT]. We show comparison of spatio-temporal attention maps and text generated between the case of supervision and without supervision :



(a) With supervision KIT-(2,3) (action range [19,28]/right kick).

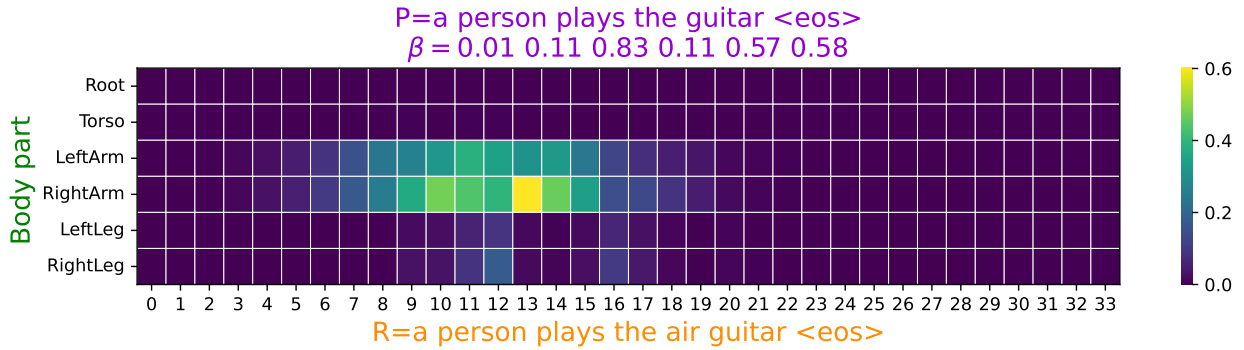


(b) Without supervision KIT-(0,0) (action range [19,27]/right kick).

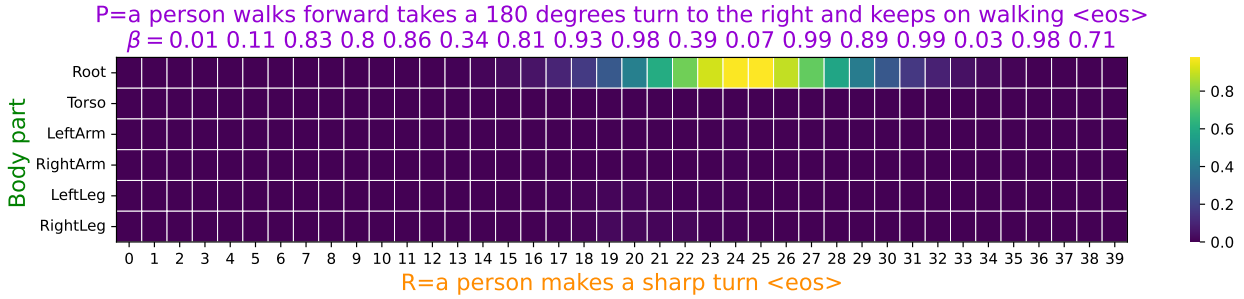
Figure 10: Impact of attention supervision.

As we see in the case of supervision (Fig.10a) the part was correctly identified and perfectly localized in the range $[20, 26]$ with corresponding manually identified range $[19, 28]$ and small β values are associated with non-motion words. Without supervision (Fig.10b), the model focuses on irrelevant part and consequently the range of action was not precisely localized. Additionally, the β values are high for all kind of words.

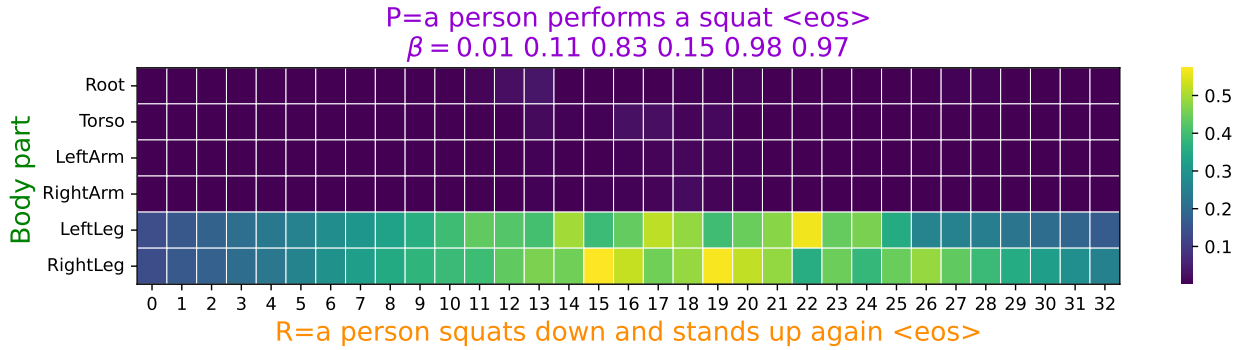
We visualize more samples (Fig.11) with Spatial+adapt supervision. Temporal range is mentioned for comparison, even if action localization wasn't the main focus in the captioning task, the model was able to learn implicitly a temporal location through the temporal Gaussian attention mechanism.



(a) Play (action range [10,20]).



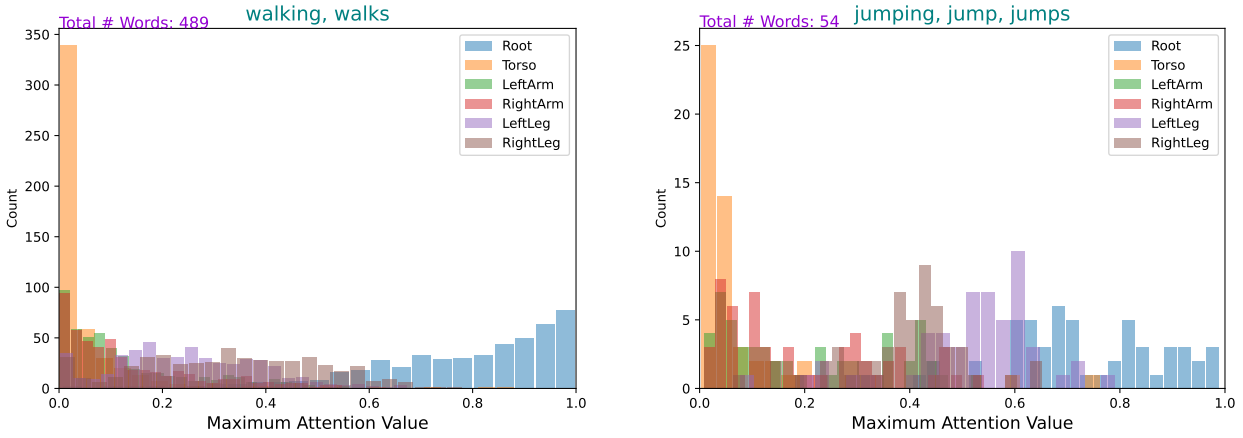
(b) Turn (action range [22,27]).



(c) Squat (action range [10,28]).

Figure 11: Spatio-temporal attention for different motion words on KIT-ML.

Trajectory and global motion. The attention was supervised only for words describing trajectory, but the model generalize successfully to motion words highly depending on global trajectory. This result on maximum attention distributed toward the *Root* body part, as we see in the Figure 12.

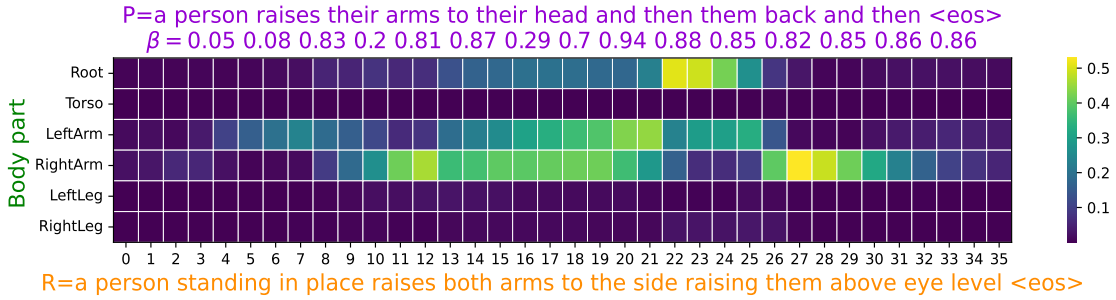


(a) Walk. (b) Jump.

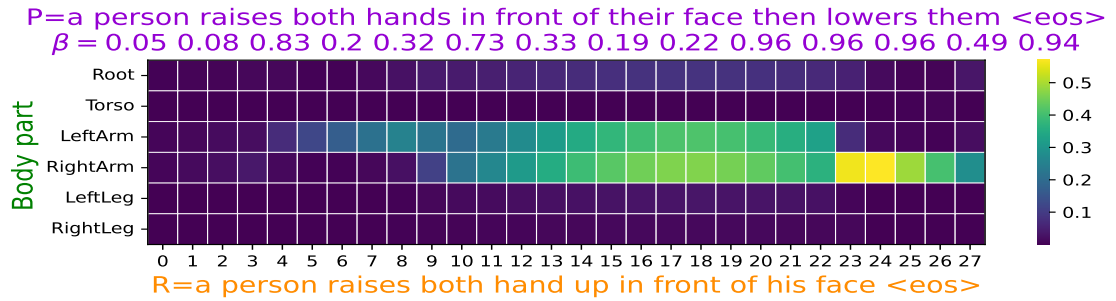
Figure 12: [KIT-(2,3)]: Body part distribution (spat+adapt).

A.5 Part based encoding & spatio-temporal attention

As mentioned before, our architecture design could be sufficient in learning a correct spatial attention maps using larger dataset with rich semantic descriptions. For demonstration, we will use the model with no spatial supervision, to show that part based encoding and spatio-temporal can work solely and correctly together for focusing on relevant body parts w.r.t the associated generated motion word. To this purpose, we propose to display the histogram distribution of temporal maximum attention weights for each body part over all test set and given a different motion words. This allows for an effective global evaluation of interpretability over all test set.



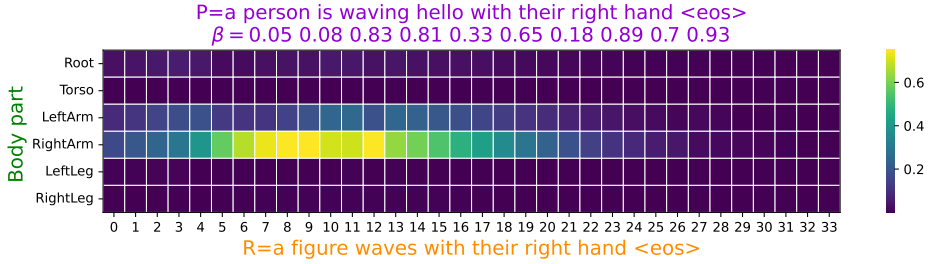
(a) Shakes.



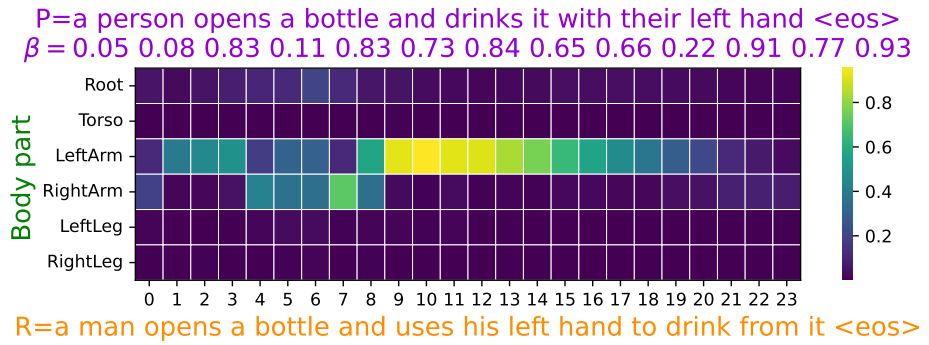
(b) Lowers.

Figure 13: Bi-part based motion.

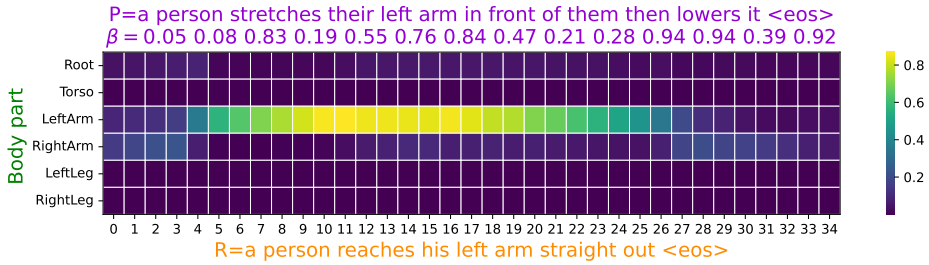
Spatio-temporal attention map. In the case of the model without spatial supervision, we have found that the model performs a correct attention focus. When an action is performed using right leg/arm, the model focuses correctly on the corresponding parts. Moreover, for actions performed with both arms/legs, the model focus on both parts. For all cases, body part words (left/right/both) are always accurately identified into the generated text. These observations are common across different representative samples (from different actions).



(a) Waving.



(b) Opens.

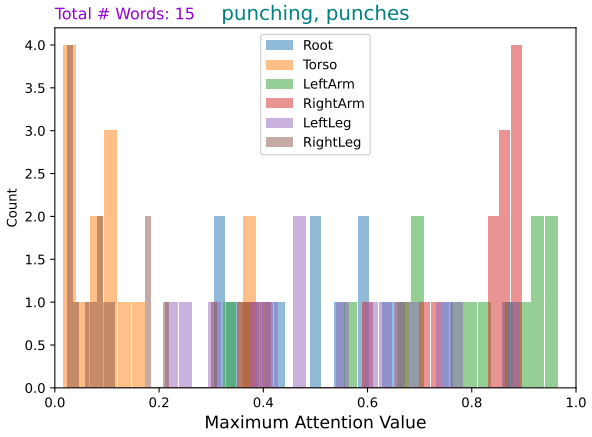


(c) Stretches.

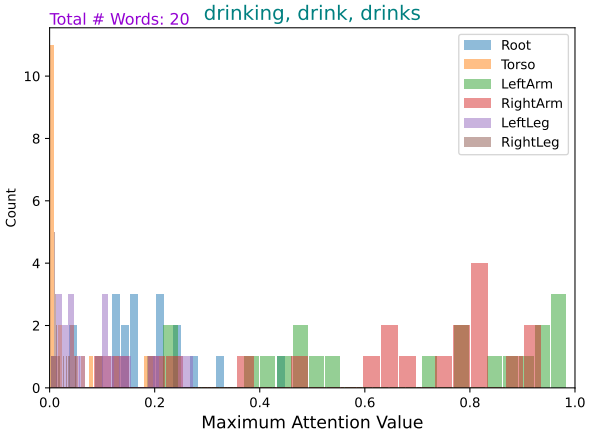
Figure 14: Single part-based motion.

Histograms. In the following, we display the body parts histogram distribution across the test set for different motion words on the model with ***no spatial supervision*** as demonstration for the effectiveness of part-based encoding along with spatio-temporal attention in finding relevant parts to focus on. This is only in the case of the larger dataset HumanML3D. The KIT-ML small dataset still requires spatial supervision to help the architecture focusing on relevant part, as the vocabulary and its size are limited. As demonstrated in all following Figures, depending on the motion word, arms-based/legs-based actions, and particularly some motions with an emphasis on Torso body part.

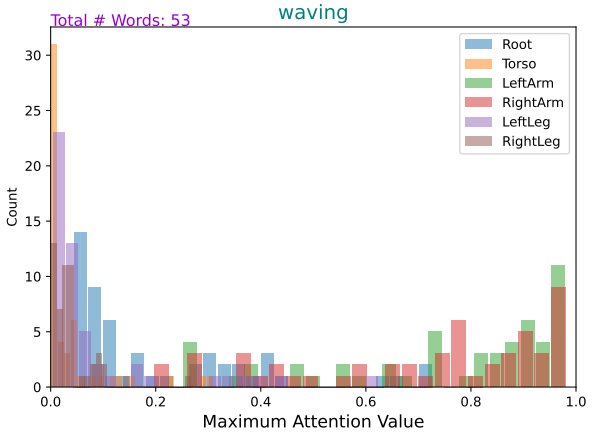
Analysis. As illustrated by the histograms below, in the case of arm-based motion, the attention model successfully focuses on arm body parts (as shown in Figure 15 with the red and green bar plot) for all test set samples. As complementary, Figure 16 provides examples of actions involving other body parts, such as *kick* (with the model focusing on legs), *crawl*, *cartwheel* (with attention directed towards the torso body part), and *clockwise* (with maximal attention on the root due to the global trajectory). These visualizations align with common sense and demonstrate the capability of our model design to achieve correct spatial and temporal alignment in the case of HumanML3D, even without spatial supervision.



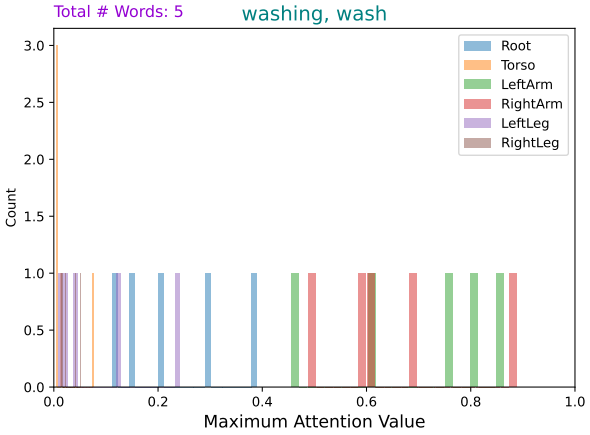
(a) Punch.



(b) Drink.

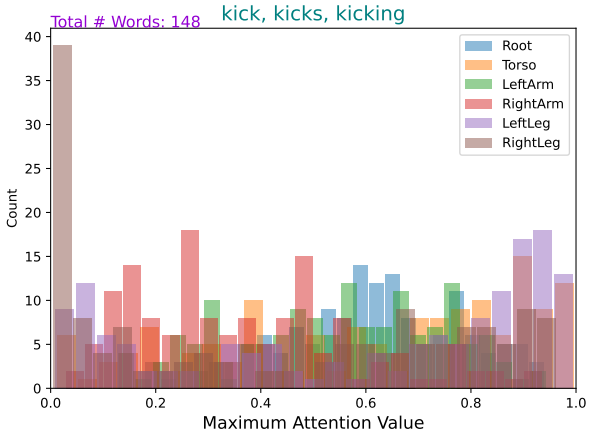


(c) Wave.

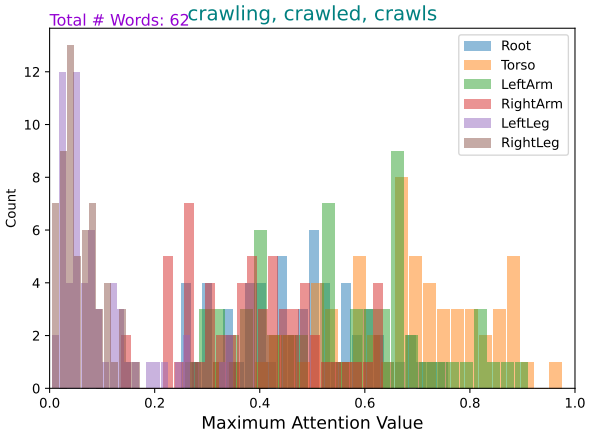


(d) Wash.

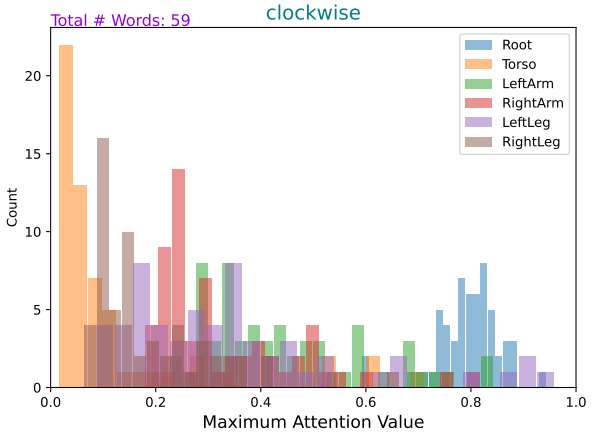
Figure 15: Arms based motion: Histogram generated on the HML3D with the config (0,3).



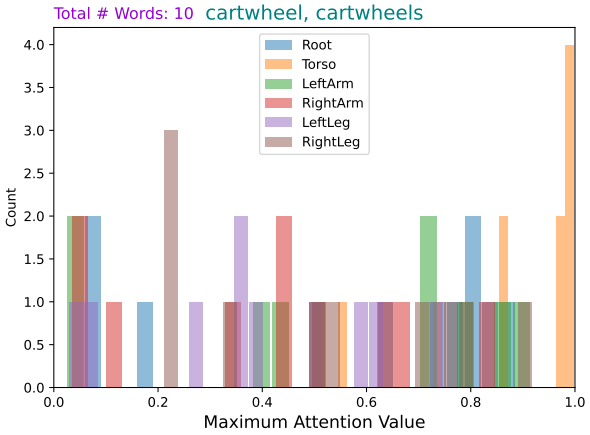
(a) Clockwise.



(b) Crawl.



(c) Clockwise.



(d) Cartwheel.

Figure 16: Histogram generated on HML3D with the config (0,3) for actions involving importantly other body parts than arms.