A Additional Experimental Details

In this section, we discuss additional experimental details for interested readers.

A.1 Code

We make all code/data publicly available for use at https://s3.us-west-1.wasabisys.com/anon-neurips2022/neurips.tar.gz (Github link in camera ready). We hope that releasing our code, along with the JSON files containing test-set predictions for the models in question will help inspire further research and examination into the evaluation of models for visual description.

A.2 Datasets

MSR-VTT Dataset: The MSR-VTT dataset (Xu et al., 2016) is a dataset for video description consisting of 10,000 videos, with 20 reference ground truth descriptions for each video. It was collected by downloading 118 videos for each of 257 queries from a popular video sharing website. MSR-VTT contains 41.2 hours of video, with an average clip length lying between 10 to 30 seconds. It has a vocabulary size of 21,913. For more details about the diversity of the language present in the dataset, we refer readers to Chan et al. (2022).

MS-COCO Dataset: The MS-COCO dataset (Lin et al., 2014) is a large-scale dataset for image description, object detection and segmentation. MS-COCO contains 328K images, each with 5 ground truth descriptions generated by human AMT workers. For more details about the diversity of the language present in the dataset, we refer readers to Chan et al. (2022). MS-COCO is licensed under a Creative Commons Attribution 4.0 license.

A.3 Models

This paper explores the performance of our metrics over several models: two video captioning models, and two image captioning models.

TVT The Two-View Transformer (Chen et al., 2018) is a baseline method for video description, which consists of a transformer encoder/decoder structure. While we did not have access to the original code, we trained our own version of the model on the MSR-VTT dataset (standard splits), leveraging features from Perez-Martin et al. (2021). The model was trained for 300 epochs, with a batch size of 64, model hidden dimension of 512, 4 transformer encoder and decoder layers with 8 heads each, and dropout of 0.5. For optimization, we leveraged the Adam optimizer with a learning rate of $3e^{-4}$ and weight decay of $1e^{-5}$ with exponential learning rate decay with gamma 0.99. This model achieves a $CIDEr$ score of 56.39 on the test dataset. The model was trained using a Titan RTX-8000 GPU over the course of several hours.

O2NA O2NA (Liu et al., 2021) is a recent approach for non-auto-regressive generation of video captions. While the method had available code and checkpoints which we used for this experiment, the method is not designed to sample more than one candidate caption at any given time. To adjust the model to sample multiple candidate captions, we made several adjustments. First, the model was modified to sample a length according to a softmax distribution over the length likelihoods (instead of using a greedy choice of length, or beam search over lengths, as proposed in the paper). Second, the model was modified to sample tokens at each non-autoregressive step from a temperature-adjusted softmax distribution instead of greedily sampling tokens. We make our modified code available as a patch to the original repository, in the hopes that other users will continue to build on these alterations.

CLIPCap CLIPCap (Mokady et al., 2021) is a recent model for image description based on using the CLIP (Radford et al., 2021a) model for large vision and language pre-training as a feature encoder, and GPT (Brown et al., 2020) as a natural language decoder. CLIPCap code and MS-COCO trained model checkpoints are publicly available from the authors, however we made some alterations to support temperature-based and nucleus sampling. We make our modified code available as a patch to the original repository, in the hopes that other users will continue to build on these alterations. CLIPCap is licensed under the MIT license.

VLP VLP (Zhou et al., 2020) is a unified vision and language pre-training model, designed to perform both image captioning and visual question answering. The model is pre-trained on the Conceptual Captions (Sharma et al., 2018) dataset, and fine-tuned on the MS-COCO captions dataset for image description. The authors make code and pre-trained models publicly available, however we modified the code somewhat to support additional sampling methods. We make our modified code available as a patch to the original repository, in the hopes that other users will continue to build on these alterations. VLP is licensed under the Apache License 2.0.

A.4 Distance Metrics

In this paper, we explore three base semantic metrics as distance underlying our TRM methods, CIDEr-D (Vedantam et al., 2015), METEOR (Agarwal & Lavie, 2008), and BERT Distance (Zhang* et al., 2020).

CIDEr-D CIDEr-D (Vedantam et al., 2015) is a n-gram-based metric designed for visual description, and based on the idea that common words are less useful in practice than uncommon words. In practice, this takes the form of a cosine similarity between TF-IDF weighted vectors representing the sentences. Because CIDEr-D is a score, and not a distance, we create a distance function: $d(c, r) = 10 - C(c, r)$, which works as CIDEr-D is bounded by 10. Note that because CIDEr-D is 10 if and only if and only if the two sentences are equal, this fulfills the TRM requirements.

METEOR METEOR (Agarwal & Lavie, 2008) is a score which evaluates the semantic distance between two text utterances based on one-to-one matches between tokens in the candidate and reference text. The score first computes an alignment between the reference and candidate, and computes a score based on the quality of the alignment. Because METEOR is a score, and not a distance function, we use the distance $d(c, r) = 1 - M(c, r)$, where M is the METEOR score of the reference. Because METEOR is bounded at 1 if and only if the two utterances are identical, this simple transformation satisfies the requirements of the TRM adjustment. While we could explore other ways of deriving a distance from METEOR, we found that this simple approach was sufficient to demonstrate the performance of our methods.

BERT Distance A recent method for determining the semantic distance between two samples is to leverage a pre-trained BERT embedding model to create a semantic embedding of the text, and computing the cosine distance between the test samples. In our work, we leverage the `MiniLM-L6-v2` model from the sentence-transformers package by Reimers & Gurevych (2019) to embed our descriptions. Because cosine distance is already a distance function, no additional transformation is necessary.

A.5 P-value Computations

For our experiments, our null hypothesis is that the candidate samples and the ground truth samples are drawn from the same distribution. Because most of the methods do not have an analytical way to compute the p-values (in fact, the TRMs are the only method which has an analytic p-value computation given in Liu & Modarres (2011)), we instead must compute the p-values though sampling. We thus enumerate the value of the statistic across all of the possible candidate/reference partitions given the joint set of candidates and references, and determine the probability of observing the sampled value, or some value more extreme.

The values in Table 1 represent the p-value obtained with a single candidate sentence, and 4 ground truth candidates for MS-COCO, or 19 ground truth candidates for MSR-VTT. We reserve one gorund truth description in both datasets to serve as the "Human" performance description. For TVT, CLIPCap and VLP, we sample the descriptions using beam search with 16 beams. For O2NA, which is a non-autoregressive model, we sample according to the method suggested in the original work (see Liu et al. (2021)). Because there are several thousand videos per dataset, computing all possible combinations across the dataset would be far from tractable. Thus, the p-values were computed on a per-visual-input basis, and then aggregated across videos using the harmonic mean, as suggested by Wilson (2019). Such an aggregation method is valid when the experiments are not independent (which they are not), unlike Fischer's method (Fisher, 1992).

Figure 3 demonstrates the log p-values for the proposed methods across several candidate samples. For MS-COCO, we use all five reference captions, and between one and ten candidate captions

sampled from CLIPCap using Nucleus Sampling (Holtzman et al., 2019) with a temperature of 1.0, top-p of 0.9 and top-k of 20. The caption set is generated once, meaning that the two-candidate set consists of the one-candidate set and one more additional caption. For MSR-VTT, we use 10 reference captions, and between one and seven candidate captions sampled from O2NA as described in appendix A.3 with a temperature of 1.0 for both the length and token samples. We do not go to the full 10 candidate captions for MSR-VTT due to tractability concerns, since adding an additional caption forces twice the number of partitions to be evaluated when computing p-values.

The above experiments were performed on several n2d-standard-32 cloud GCP instances, containing 32vCPUs and 128GB of RAM.

A.6 Frechet BERT Distance

The Frechet Inception Distance, originally proposed in Salimans et al. (2016), has often been used for the evaluation of the distance between samples of images generated by GANs. Images are first embedded in a latent space using a pre-trained inception network, and then the Frechet distance between the generated samples and the reference samples is computed. In our work, we replace the images with text, and the inception network with a pre-trained BERT embedding network (Devlin et al., 2018). For a set of candidate samples $(c_1, \ldots, c_n) = C$, a set of reference samples $(r_1, \ldots, r_m) \in R$, and a BERT embedding function $\phi_{\text{BERT}} : C \cup R \to \mathbb{R}^k$, we compute the Frechet BERT Distance as:

$$d^2 = \left|\left| \frac{1}{n}\sum_{i=1}^{n} \phi_{BERT}(c_i) - \frac{1}{m}\sum_{i=1}^{n} \phi_{BERT}(r_i) \right|\right| + \text{Tr}\left(C_C + C_R - 2\sqrt{C_C C_R}\right) \tag{5}$$

where C_C andC_R are the covariance matrices of the C and R sets embedded with ϕ_{BERT} respectively.

To get the BERT embedding, we leverage the CLS token of a large pre-trained model, in this case, the `MiniLM-L6-v2` model from the sentence-transformers package by Reimers & Gurevych (2019).

The computation of p-values for the Frechet-BERT distance is largely bottle-necked by the slow performance of the `sqrtm` function, which, because the matrices are not symmetric, has no efficient algorithm for computation. Additionally, unlike the feature computation, this operation must occur for every partition, leading to significantly reduced efficiency compared to the other measures presented in this paper.

A.7 MMD-BERT

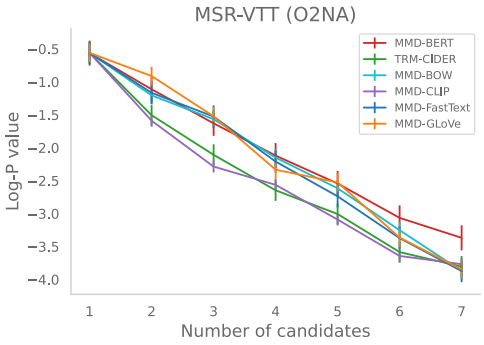
Another common metric in the GAN literature is the computation of a maximum-mean discrepancy between kernel-estimates of the samples introduced by Li et al. (2017). For a set of candidate samples $(c_1, \ldots, c_n) = C$, a set of reference samples $(r_1, \ldots, r_m) \in R$, and a BERT embedding function $\phi_{\text{BERT}} : C \cup R \to \mathbb{R}^k$, we compute the MMD-BERT distance as:

$$\begin{aligned} M\hat{M}D = &\sum_{i=1}^{N}\sum_{j=1}^{N} K(\phi_{\text{BERT}}(c_i), \phi_{\text{BERT}}(c_j)) + \\ &\sum_{i=1}^{M}\sum_{j=1}^{M} K(\phi_{\text{BERT}}(r_i), \phi_{\text{BERT}}(r_j)) + \sum_{i=1}^{N}\sum_{j=1}^{M} K(\phi_{\text{BERT}}(c_i), \phi_{\text{BERT}}(r_j)) \end{aligned} \tag{6}$$

where K is a kernel function. In our experiments, we use an RBF kernel function with σ equal to the median distance pairwise distance divided by two.

A.8 Search Techniques

In section 3, Figure 6, we explore the performance of several different search techniques for our two-view transformer model on the MSR-VTT dataset. In this figure, we explore four decoding search techniques: Greedy Search, Beam Search, Temperature-Based Sampling, and Nucleus Sampling. For each method, and for each video in the test set, we sample 10 descriptions. For Greedy Search, we sample 10 repeated sentences. For beam search we sample the top beam search candidate, and repeat this ten times. While we did explore using the top 10 results from a larger beam search, we found that a smaller beam search and repeated values produced better METEOR scores, so we chose to compare



Sensitivity and performance on Human-Generated Captions

Method	Log-P	Samples/Sec
TRM-CIDEr	-1.596	88.93
MMD-BERT	-1.786	56.68
MMD-CLIP	-1.887	14.41
MMD-GLoVe	-1.952	54.8
MMD-FastText	-1.954	57.45
MMD-BOW	-2.022	49.41

Figure 7: Performance of several different embedding functions for the MMD-* family of metrics. Left: Sensitivity when evaluated on the MSR-VTT dataset with ten reference captions and between one and seven candidate captions generated by O2NA. Right: Sensitivity and speed when evaluated on human reference samples with 5 references and 5 candidates.

against this. Wider beam searches did produce higher $\text{TRM}_{\text{METEOR}}$ scores, but because optimizing for METEOR would be the current paradigm, we decided to include that in the referenced figure. For standard temperature based sampling, we sampled 10 results at each temperature. For Nucleus sampling, we sample 10 results at each temperature, however we freeze they hyper-paramters of top-p at 0.9 and top-k at 20, as we found these values to generate the best scores under the standard pairwise metrics. It remains relevant future work to perform a deep-dive into the different generative methods with respect to TRMs, as there are likely many interesting lessons that can be learned.

B Additional Results

In this section we present several additional interesting results to augment those in the main discussion.

B.1 Embedding Methods for KBMs

In the main work, we primarily explore a BERT-based embedding method for the kernel-based methods. Such an exploration does not preclude the use of other embedding methods, each of which has different trade-offs, when looking at the quality of the resulting metric, what the resulting metric measures, the time required to compute the embedding, and the performance when the reference distribution is limited to small numbers of human samples (such as happens in practice). Figure Figure 7 shows a quick look at several possible choices for embedding methods in the MMD-* family, including Bag of words (with a 5K vocab), GLoVe (Pennington et al., 2014), FastText (Bojanowski et al., 2017), and CLIP (Radford et al., 2021b).

While we can see that some of the methods are more sensitive to deviations in the image distributions, such methods come with additional trade-offs. CLIP-style embeddings are the most sensitive to human versus generated captions with fewer captions created, but are significantly slower to evaluate at test time (almost 4x slower) than MMD-BERT, and also produce a higher p-value when computing the leave-one scores on the human captions (which is less desirable, as the human captions are drawn from the same distribution).

B.2 Unique vs. Correct Descriptions

In Figure 8, we explicitly demonstrate how TRMs enable evaluation of both caption diversity and quality. We artificially generate candidates for the MSR-VTT dataset by mixing human-generated exact descriptions with human-generated descriptions from other videos. On one axis we have the number of unique descriptions and on the other axis we have the number of correct (exactly-matching) descriptions. Clearly, unlike METEOR alone, $\text{TRM}_{\text{METEOR}}$ scores are affected by both correctness and diversity.

Each experiment consisted of 10 candidate captions from the MSR-VTT dataset, and 10 reference captions from the MSR-VTT dataset. We first split the 20 MSR-VTT reference captions into two sets of 10. One set of 10 captions formed the references. To select the candidate captions, we first sampled k unique captions from the remaining reference set (which formed the "correct pool"), and

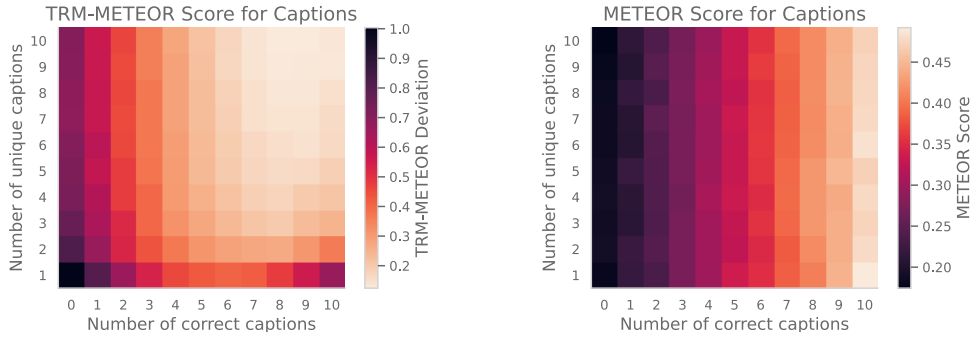


Figure 8: Plots showing how TRMs evaluate both diversity and quality. Left: $\text{TRM}_{\text{METEOR}}$, Right: METEOR. Lighter colors represent better scores. While $\text{TRM}_{\text{METEOR}}$ trades off between diversity and quality, METEOR focuses only on quality not diversity.

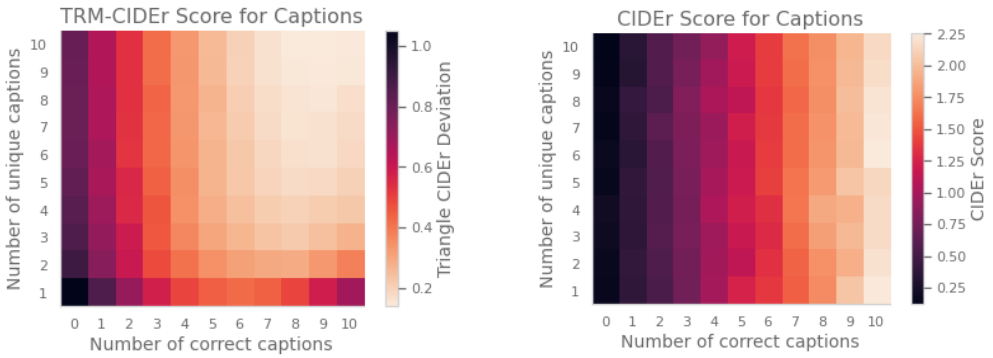


Figure 9: Plots showing diversity vs. quality tradeoffs. Left: $\text{TRM}_{\text{CIDEr}}$, Right: CIDEr. Lighter colors represent better scores. While $\text{TRM}_{\text{CIDEr}}$ trades off between diversity and quality, CIDEr focuses only on quality not diversity.

k unique captions from other videos in the dataset at random (forming the "incorrect pool"). We then selected m correct captions, from the correct pool (at random) and $10 - m$ captions from the incorrect pool (at random). This was then plotted with m on the x-axis, and k on the y-axis, as a heat-map, where lighter colors represent better scores (higher METEOR, or lower TRM-METEOR), and darker colors represent poor scores.

We also explored the performance of the CIDEr metric across the same axes, the results of which are shown in Figure 9. We can see that they are largely similar to those from the METEOR metric, suggesting that regardless of the underlying metric, we are still making similar trade-offs between diversity and correctness.

B.3 Visualizing Central Descriptions

We have found that descriptions which minimize the expected distance to the ground truth distribution are relatively sparse in detail compared to other descriptions. Figures 10, 11, 12 and 13 show qualitative examples of such descriptions for the MS-COCO dataset. Each plot shows qualitative examples of "central" captions. The caption marked with arrows is the ground truth caption which minimizes the expected METEOR distance to the other reference captions, and the other captions are the additional references in the MS-COCO dataset. Images are selected at random, and do not represent cherry-picked samples from MS-COCO.

B.4 Human p-values

Strong metrics for distributional comparison will have high sensitivity to samples coming from distinct distributions, and will produce high p-values for samples which come from the same distribution. To check that such a relationship holds, we also perform leave-one-out experiments using human-generated captions from the reference set for both MSR-VTT and MS-COCO. For MSR-VTT, we

Two hot dogs sitting side by side with condiments.
Two hot dogs are laden with relish, ketchup, and mustard.
>>> two hot dogs on a plate loaded with condiments
Two hot dogs covered with ketchup and relish on a plate.
Two hot dogs in buns are smothered with condiments.

The meal is ready on the tray to be eaten.
A breakfast was delivered to a hotel room on a tray.
a bunch of food and stuff is laying on a tray
>>> Bananas, cereal, juice and other breakfast foods on a tray.
This tray includes several different items for a full breakfast.

Figure 10: Qualitative example of "central" captions. The caption marked with arrows is the ground truth caption which minimizes the expected METEOR distance to the other reference captions.

A photo taken from a boat with a long bridge in the background.
A view of the coast from within a boat
The side of a boat and a bridge going over the ocean.
>>> A view of the lake, taken from a boat.
A boat flies its flag while sailing just off a pier.

a microwave on a kitchen counter above a dishwasher
this micro wave is black and silver and is on the counter
>>> A microwave oven sitting on top of a counter.
A microwave sitting on a counter, its stainless steel.
a silver microwave oven on a tan counter and a window

Figure 11: Qualitative example of "central" captions. The caption marked with arrows is the ground truth caption which minimizes the expected METEOR distance to the other reference captions.

A narrow city street has a leaning one way sign.
>>> a street with a line of cars parked on the side
Cars are parked alongside the road and a man is standing next to a sign.
A man is standing next to a road sign with a line of parked cars across the street in an urban area
A crooked one way sign pointing into the ground

A person pressing a button on a Wii controller.
A hand holds a remote that operates a video game.
There are no image to describe on this page..
>>> A person is holding a white Wii control
someone that is holding a wii remote in their hand

Figure 12: Qualitative example of "central" captions. The caption marked with arrows is the ground truth caption which minimizes the expected METEOR distance to the other reference captions.

a blue truck and a male in a purple shirt and a tree
Blue pickup truck filled with scrap pieces of household items.
A man has filled his truck with wheelchairs.
>>> A blue truck parked next to a tree and a man.
a man standing next to a truck full of bikes and a wheel chair

Mom gives her daughter a lesson in using her baseball glove.
>>> Mother and her son playing in a few
two girls in red shirts grass and a baseball glove
A woman playing catch with her young child.
The mom is teaching her daughter to play baseball

Figure 13: Qualitative example of "central" captions. The caption marked with arrows is the ground truth caption which minimizes the expected METEOR distance to the other reference captions.

Table 3: Log P-Values on human leave-one our samples. We can see that, surprisingly, none of the methods (even the standard aggregations) produce statistically signficant differences. That being said, TRMs often produce higher p-values, indicating that they may be more robust to noise in human caption sets. We do not compute the Frechet-BERT values for humans here, as it was prohibitively expensive.

	METEOR	$\text{TRM}_{\text{METEOR}}$	CIDEr	$\text{TRM}_{\text{CIDEr}}$	BERT	TRM_{BERT}	MMD-BERT
MSCOCO	-0.6303	-0.5941	-0.5957	-0.4742	-0.6230	-0.5633	-0.6550
MSR-VTT	-1.0046	-0.9613	-1.0224	-0.9777	-1.0172	-1.040	-1.0374

split the reference data into sets of 10 candidate samples and 10 reference samples, and compute the deviations using this partitioning. For MS-COCO, we leverage the c40 split which has 40 reference descriptions for 5000 samples of the ground truth. We partition the references for each video into groups of ten descriptions, and compute the p-values from pairs of these partitions. Table 3 gives the performance of the metrics on this human data.

B.5 MAUVE PERFORMANCE

In the main work, we found that MAUVE was prohibitively slow to use to compute p-values for the training data. Because our p-values were computed with 10 reference sentences, and up to 10 candidate sentences, at the existing rate, it could take several years to compute the MAUVE p-values for the 50,000 sample MS-COCO dataset. In Table 4, we present several high-variance estimates of the MAUVE p-values (computed using only 100 samples).

Table 4: Log p-value estimates for MAUVE using five candidates, five references, and 100 samples (at nucleus sampling temperature 1.0 for O2NA, CLIPCap and VLP models). We can see that Log p-values for MSR-VTT and MS-COCO are signficantly worse than METEOR even with aggregation, likely due to the method using k-means to approximate the text distributions with only 5 samples.

Dataset	MAUVE Log p-value	METEOR Log p-value
MSR-VTT (O2NA)	-0.4414	-1.7881
MSR-VTT (Human Captions)	-0.1441	-0.6037
MS-COCO (CLIPCap)	-0.3980	-2.5585
MS-COCO (VLP)	-0.3234	-2.8609
MS-COCO (Human Captions)	-0.2189	-0.7233

B.6 ADDITIONAL QUALITATIVE SAMPLES

MSCOCO Image 421060

Candidate Set 1
METEOR (↑): 0.236
TRM-METEOR (↓): 0.912

A person on a snowboard does a trick in the air.
A person on a snowboard does a trick in the air.
A person on a snowboard does a trick in the air.
A person on a snowboard does a trick in the air.
...

Candidate Set 2
METEOR (↑): 0.264
TRM-METEOR (↓): 0.362

A person in mid air on a snowboard in front of a TV.
A snowboarder getting some air after a jump.
A man is performing a ski jump on a green slope.
A person on skis going down a ramp.
...

References

Competitive spirit during a competition in mid air
A skier races down the track at a competition.
A person is skiing on a slope covered in snow.
The skier is jumping into the air in a half pipe.
Skier performing aerial jump during outdoor competition.

Figure 14: A qualitative sample from CLIPcap. Candidate set one uses beam search (8 beams), while candidate set two uses nucleus sampling (with temperature one, top-k of 20 and top-p of 0.9).