

# Pre-gen metrics: Predicting caption quality metrics without generating captions\*

Marc Tanti      Albert Gatt

Adrian Muscat

University of Malta, Msida MSD 2080, Malta

{marc.tanti.06, albert.gatt, adrian.muscat}@um.edu.mt

## Abstract

Image caption generation systems are typically evaluated against reference outputs. We show that it is possible to predict output quality without generating the captions, based on the probability assigned by the neural model to the reference captions. Such pre-gen metrics are strongly correlated to standard evaluation metrics.

## 1 Introduction

Automatic metrics for image description generation (IDG) compare  $c$ , a generated caption, to a set of reference sentences,  $R_1 \dots R_n$ . We therefore refer to these as **post-gen**(eration) metrics. In most neural IDG architectures generation is performed by an algorithm such as beam search that samples the vocabulary at every timestep, selecting a likely next word after a given sentence prefix (according to the neural network) and attaching it to the end of the prefix, and repeating this procedure until the entire caption is produced. Given that the output thus generated is evaluated against a gold standard, post-gen metrics actually evaluate the neural network’s ability to predict the words in the reference captions given an image. Unfortunately, generating sentences is a time consuming process due to the fact that every word in a sentence requires its own forward pass through the neural network. This means that generating a 20-word sentence requires calling the neural network 20 times. As an indicative example, it takes 20.8 minutes to generate captions for every image in the MSCOCO test set on a standard hardware setup (GeForce GTX 760) using a beam width of just 1.

Our question is whether a system’s performance can be assessed *prior* to the generation step, by exploiting the fact that the output is ultimately based on this core sampling mechanism. We envisage a scenario in which a neural

---

\*This publication will appear in the Proceedings of the First Workshop on Shortcomings in Vision and Language (2018). DOI to be inserted later.

caption generator is evaluated based on the extent to which its estimated softmax probabilities over the vocabulary maximise the probability of the words in the reference sentences  $R_1 \dots R_n$ . We refer to this as a **pre-gen**(eration) evaluation metric, as it can be computed prior to generating any captions. A well-known example of a pre-gen metric is language model perplexity although, as we show below, this metric is not the best pre-gen candidate in terms of its correlation to standard evaluation measures for IDG systems.

From a development perspective, the advantage of a pre-gen metric lies in that all the word probabilities in a sentence are immediately available to the network in one forward pass, whereas a post-gen metric can only be computed following a relatively expensive process of word-by-word generation requiring repeated calls to a neural network. To return to the earlier example, on the same hardware setup it only takes 28 seconds to compute model perplexity.

Thus, if pre-gen metrics can be shown to correlate strongly with established post-gen metrics, they could serve as a proxy for such metrics. This would speed up processes requiring repeated caption quality measurement such as during hyperparameter tuning.

Finally, from a theoretical and empirical perspective, if caption quality, as measured by one or more post-gen metric(s), can be predicted prior to generation, this would shed further light on the underlying reasons for the observed correlations of such metrics [14].

All code used in these experiments is publicly available.<sup>1</sup>

The rest of this paper is organised as follows; background on metrics is covered in section 2, the methodology and experimental setup in section 3, and the results are given in section 4; the paper is concluded in section 5.

## 2 Background: *Post-gen* metrics for image captioning

In IDG, automatic metrics originally developed for Machine Translation or Summarisation, such as BLEU [21], ROUGE [18], and METEOR [2], were initially adopted, followed by metrics specifically designed for image description, notably CIDEr [24] and SPICE [1]. Lately, Word Mover’s Distance (WMD) [17], originally from the document similarity community, has also been suggested for IDG [14]. Like BLEU, ROUGE and METEOR, CIDEr makes use of n-gram similarities, while WMD measures the semantic distance between texts on the basis of word2vec [20] embeddings. All of these metrics are purely linguistically informed. By contrast, SPICE computes similarity between sentences from scene graphs [13], obtained by parsing reference sentences. This method is also linguistically informed; however the intuition behind it is that the human authored sentences should be an accurate reflection of image content.

A typical IDG experiment reports several post-gen metrics. One reason is that the metrics correlate differently with human judgments, depending on task

---

<sup>1</sup>See: <https://github.com/mtanti/pregen-metrics>

and dataset [3], echoing similar findings in other areas of NLP [7, 5, 6, 22, 4, 11, 10, 26]. Thus, BLEU, METEOR, and ROUGE correlate weakly [16, 12, 15] and yield different system rankings compared to human judgments [25]. METEOR has a reportedly higher correlation than BLEU/ROUGE [8, 9], with stronger relationships reported for CIDEr [24] and SPICE [1]. Meta-evaluation of the ability of metrics to discriminate between captions have also been somewhat inconsistent [24, 14].

The extent to which post-gen metrics correlate with each other also varies, with stronger relationships among those based on n-grams on the one hand, and more semantically-oriented ones on the other [14], suggesting that these groups assess complementary aspects of quality, and partially explaining their variable relationship to human judgments in addition to variations due to dataset.

For neural IDG architectures, post-gen metrics have one fundamental property in common: they compare reference outputs to generated sentences which are based on sampling at each time-step from a probability distribution. Our hypothesis is that it is possible to exploit this, using the probability distribution itself to directly estimate the quality of captions, prior to generation.

### 3 Pre-gen metrics

Given a prefix, a neural caption generator predicts the next word by sampling from the softmax’s probabilities estimated over the vocabulary. Let  $R$  be a reference caption of length  $m$ . Given a prefix  $R^{0\dots k}$  (where  $R^0$  is the start token),  $k \leq m$ , a neural caption generator can be used to estimate the probability of the next word (or the end token) in the reference caption,  $R^{k+1}$ . The intuition underlying pre-gen metrics is that the higher the estimated probability of  $R^{k+1}$ , for all  $k \leq m$ , the more likely it is that the generator will approximate the reference caption. Note that the idea is to estimate the probability of *reference* captions based on a trained IDG model.

Pre-gen metrics produce a score by aggregating the word probabilities predicted by the generator for all reference captions (combined with their respective image) over prefixes of different lengths. The way a caption generator predicts these word probabilities is illustrated in Figure 1. To find the best way to aggregate the word probabilities, we define a search space by setting options at four different algorithmic steps which we call ‘tiers’. Each tier represents a function and the composition of all four tiers together constitutes a pre-gen function. We try several different options for each tier in order to find the best pre-gen function. Figure 2 shows an example of how tiers form a pre-gen function.

Given a set of images with their corresponding reference captions, the process starts by computing each reference caption’s individual word probabilities (given the image) according to the model. Note that the model may not predict every word in a reference caption as the most likely in the vocabulary.

The first tier is a filter that selects which predicted word probabilities should be considered in the next tier. We consider three possible filters: (a) the filter *none* passes all probabilities; (b) *filter0* filters out the word probabilities that

		Reference sentence						
		a	dog	eating	a	pine	cone	<END>
Vocabulary	<END>	0.022	0.003	0.008	0.024	0.003	0.015	<u>0.743</u>
	a	<b>0.714</b>	0.001	0.005	<b>0.580</b>	0.001	0.020	0.007
	at	0.005	0.016	0.027	0.048	0.016	0.010	0.017
	cone	0.029	0.000	0.032	0.002	0.000	<b>0.739</b>	0.002
	cow	0.007	0.017	0.015	0.054	0.017	0.015	0.029
	dog	0.018	<u>0.438</u>	0.020	0.020	0.438	0.009	0.003
	eating	0.008	0.002	<b>0.364</b>	0.020	0.002	0.018	0.019
	feet	0.003	0.002	0.015	0.001	0.002	0.011	0.011
	grass	0.030	0.015	0.032	0.010	0.015	0.013	0.021
	nipping	0.034	0.005	0.019	0.023	0.005	0.019	0.035
	of	0.034	0.009	0.028	0.053	0.009	0.027	0.024
	on	0.032	0.012	0.030	0.006	0.012	0.012	0.001
	pine	0.027	<b>0.454</b>	0.027	0.053	<b>0.454</b>	0.033	0.016
	plays	0.012	0.016	0.340	0.049	0.016	0.031	0.019
	pounces	0.003	0.006	0.021	0.026	0.006	0.002	0.020
	the	0.014	0.003	0.013	0.018	0.003	0.009	0.025
	with	0.008	0.001	0.002	0.013	0.001	0.016	0.010

Figure 1: An example illustrating the output of a neural network that is predicting the probability for every word in a sentence. Given a single sentence, the caption generator will immediately output a matrix of probabilities, such that for every word position in the sentence, the matrix contains the probabilities for every word in the vocabulary being in that position given the image and the previous words (the first word has the start token as a previous word). In the above example, underlined probabilities are of the correct words being in the designated word position whilst bold probabilities are of the word with the maximum probability in the vocabulary for the designated word position. Note how the maximum probability is not always assigned to the correct word and that it might do so only intermittently.

are not ranked as most probable in the vocabulary by the model, i.e are not predicted to be maximally probable continuations of the current prefix; and (c) *prefix0* selects the longest prefix of the caption such that the model predicts all words in the prefix as being the most likely in the vocabulary.

At the second tier, we aggregate the selected word probabilities in each reference sentence into a single score for each sentence. We define four possible functions: (a) *prob* multiplies all probabilities; (b) *pplx* computes the perplexity; (c) *count* counts the number of word probabilities that were selected in the first tier; and (d) *normcount* normalises *count* by the total number of words in the reference sentence.

The third tier aggregates the scores obtained for all reference sentences into a single score for each image. We explore six possibilities: (a) *sum*; (b) *mean*; (c) *median*; (d) *geomean*, the geometric mean; (e) *max*; and (f) *min*. We also consider (g) *join*, whereby all the image-sentence scores are joined into a single list without aggregation so that they are all aggregated together in the next tier.

The fourth tier aggregates the image scores into a single dataset score, which is the final pre-gen score of the caption generator. For this aggregation, we use the same functions in the previous tier excluding *join*.

The above possibilities result in 504 unique combinations. In what follows, we adopt the convention of denoting a pre-gen metric by the sequence of function names that compose it, starting from tier four e.g. *mean\_max\_normcount\_prefix0*.

For every reference sentence in every image, get the words that are to be predicted by the caption generator in each sentence	image 1	<ul style="list-style-type: none"> <li>• a dog nipping at the feet of a cow &lt;END&gt;</li> <li>• a dog pounces on the grass &lt;END&gt;</li> </ul>
	image 2	<ul style="list-style-type: none"> <li>• a dog eating a pine cone &lt;END&gt;</li> <li>• a dog plays with a pine cone &lt;END&gt;</li> </ul>
Find probabilities of each word according to caption generator (probabilities in bold are maximum in vocabulary)	image 1	<ul style="list-style-type: none"> <li>• <b>0.711</b> <b>0.507</b> <b>0.380</b> <b>0.563</b> <b>0.577</b> <u>0.352</u> <b>0.557</b> <b>0.574</b> <u>0.268</u> <b>0.615</b></li> <li>• <b>0.711</b> <b>0.507</b> <u>0.369</u> <b>0.561</b> <b>0.605</b> <b>0.384</b> <b>0.520</b></li> </ul>
	image 2	<ul style="list-style-type: none"> <li>• <b>0.714</b> <u>0.438</u> <b>0.364</b> <b>0.580</b> <b>0.454</b> <b>0.739</b> <b>0.743</b></li> <li>• <b>0.714</b> <u>0.438</u> <u>0.340</u> <b>0.592</b> <b>0.536</b> <b>0.454</b> <b>0.739</b> <b>0.743</b></li> </ul>
Tier 1: Take longest prefix of maximal probabilities	image 1	<ul style="list-style-type: none"> <li>• <b>0.711</b> <b>0.507</b> <b>0.380</b> <b>0.563</b> <b>0.577</b></li> <li>• <b>0.711</b> <b>0.507</b></li> </ul>
	image 2	<ul style="list-style-type: none"> <li>• <b>0.714</b></li> <li>• <b>0.714</b></li> </ul>
Tier 2: Calculate the length of each filtered sequence divided by the original length	image 1	<ul style="list-style-type: none"> <li>• <math>5/10 = 0.500</math></li> <li>• <math>2/7 = 0.286</math></li> </ul>
	image 2	<ul style="list-style-type: none"> <li>• <math>2/7 = 0.286</math></li> <li>• <math>2/8 = 0.250</math></li> </ul>
Tier 3: Calculate the maximum of each image's normalized length	image 1	0.500
	image 2	0.286
Tier 4: Calculate the mean of the image scores		0.393

Figure 2: An example illustrating how tiers work. This illustration shows the best pre-gen metric found: *mean\_max\_normcount\_prefix0*.

In our experiments, we compute all of these different combinations and compare their predictions to standard post-gen metrics, namely METEOR, CIDEr, SPICE, and WMD. All metrics except WMD were computed using the MSCOCO Evaluation toolkit<sup>2</sup>. Since the toolkit does not include WMD, we created a fork of the repository that includes it.<sup>3</sup>

### 3.1 Experimental setup

For our experiments, we used a variety of pre-trained neural caption generators (36 in all) from [23].<sup>4</sup> These models are based on four different caption generator architectures. Each was trained and tested over three runs on Flickr8K [12], Flickr30k [27], and MSCOCO [19]. The four architectures differ in terms of how the CNN image encoder is combined with the RNN: *init* architectures use the image vector as the initial hidden state of the RNN; *pre* architectures treat the image vector as the first word of a caption; *par* architectures are trained on

<sup>2</sup>See: <https://github.com/tylin/coco-caption>

<sup>3</sup>See: <https://github.com/mtanti/coco-caption>

<sup>4</sup>See: <https://github.com/mtanti/where-image2>

Generated captions	Individual CIDEr		
caption 1	0.580695	} Final CIDEr 0.263	} Stratum 1 final CIDEr 0.447
caption 2	0.505971		
caption 3	0.443425		
caption 4	0.25617		
caption 5	0.14919		} Stratum 2 final CIDEr 0.081
caption 6	0.113116		
caption 7	0.03518		
caption 8	0.025599		

Figure 3: An example illustrating how a dataset is broken into strata in order to create a variety of performance scores using the same neural networks and thus be able to test the correlation between the post-gen and pre-gen metrics on many different scores.

captions where each word is coupled with an image vector at each time-step; and **merge** architectures keep the image out of the the RNN entirely, merging the image vector with the RNN hidden state in a final feedforward layer, prior to prediction.

Since only the final trained versions of the models are available, there is a bias towards good quality post-gen metric results. This renders the values of the post-gen metrics rather similar and concentrated in a small range. A pre-gen metric is useful if it makes good predictions on models of any quality not just good ones. Rather than re-training all the models and saving the parameters at different intervals during training, we opted to stratify the dataset on the basis of how well each individual image is rated by the CIDEr metric. This is illustrated in Figure 3.

We grouped images into the best and worst halves on the basis of the CIDEr score (since CIDEr is the post-gen metric that best correlates with the other post-gen metrics [14]) of their sentences as generated by a model. This creates two datasets, one where the model performs well and one where the model performs badly. We stratified the dataset into different numbers of equal parts and not just two, namely: 1 (whole), 2, 3, 4 and 5, resulting in a 15-fold increase over the original 36 averaged results and more importantly, over a wide dynamic range in CIDEr scores, which we required to study the correlation in between pre- and post-gen metrics.

## 4 Results

We evaluate the correlation between pre- and post-gen metrics using the Coefficient of Determination, or  $R^2$ , defined as the square of the Pearson correlation coefficient. The reason for this is twofold. First,  $R^2$  reflects the magnitude of a correlation, irrespective of whether it is positive or negative (the pre-gen metrics based on perplexity would be expected to be negatively correlated with post-gen

metrics). Second, given a linear model in which a pre-gen metric is used to predict the value on a post-gen metric,  $R^2$  indicates the proportion of the variance in the latter that the pre-gen metric predicts.

As a baseline, we show the scatter plot for the relationship between language model perplexity and the post-gen metrics in Figure 4. In terms of the description in the previous section, perplexity is defined as *geomean\_join\_pplx\_none*. As can be seen, perplexity performs somewhat poorly on low scoring captions. Our question is whether a better pre-gen metric can be found.

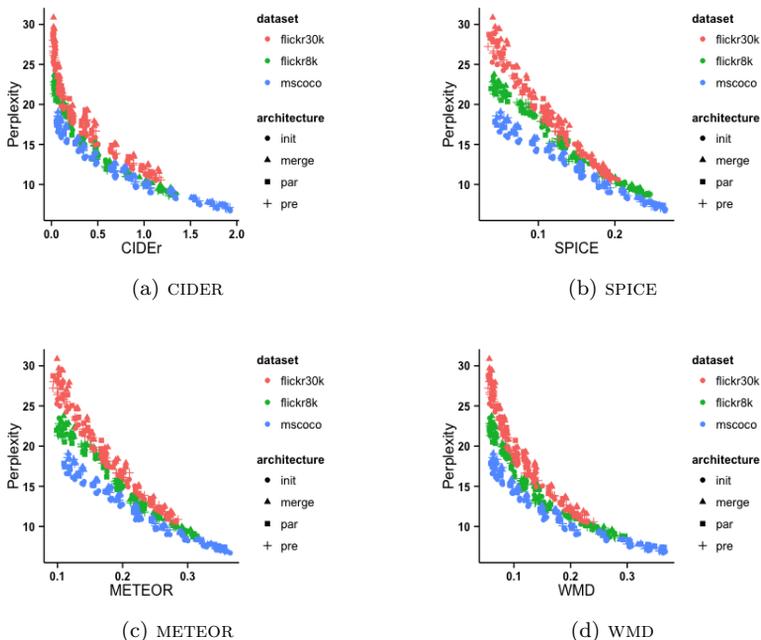


Figure 4: Relationship between perplexity and post-gen metrics by dataset and architecture. The overall correlation has an  $R^2$  of 0.76. (Best viewed in colour.)

For each of the 4 post-gen metrics, we identified the top 5 best correlated pre-gen metrics, based on the  $R^2$  value computed over all the data (i.e. aggregating scores across architectures and datasets). The top 4 pre-gen metrics were the same for all post-gen metrics, namely:

1. *mean\_max\_normcount\_prefix0*;
2. *mean\_mean\_normcount\_prefix0*;
3. *mean\_join\_normcount\_prefix0*;
4. *mean\_sum\_normcount\_prefix0*

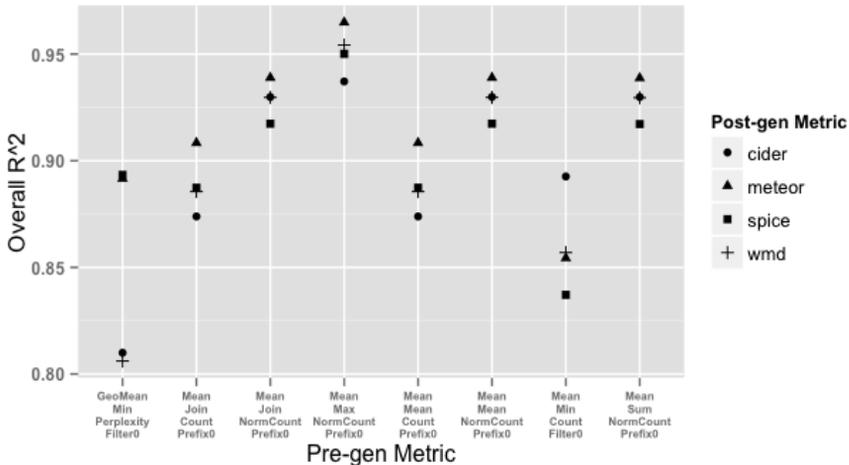


Figure 5: Overall  $R^2$  between the 4 post-gen metrics and their 5 most highly correlated pre-gen metrics. Scores average over architectures and datasets.

Note that all the best performing metrics are based on the variable *prefix0*. This is not surprising since when generating a sentence, it is probably the word with the maximum probability in the vocabulary which gets selected as a next word in a prefix. Hence if the most probable next word is an incorrect one then it will likely send the rest of the caption generation process off the rails as it will misinform the next words as well. Hence, *prefix0* is a measure of how likely this is to happen.

On the other hand, the fifth most highly correlated pre-gen metric differed for each post-gen metric, as follows:

- CIDER: *mean\_min\_count\_filter0*;
- METEOR: *mean\_mean\_count\_prefix0*;
- SPICE: *geomean\_min\_pplx\_filter0*;
- WMD: *mean\_join\_count\_prefix0*

Figure 5 displays the correlation between these pre-gen metrics and the post-gen scores. Note that all  $R^2$  scores are above 0.8, indicating a very strong correlation.<sup>5</sup> The top 4 scores have  $R^2 \geq 0.9$ .

To investigate the relationship between pre- and post-gen metrics more closely, we focus on the best pre-gen metric (that is, *mean\_max\_normcount\_prefix0*) and consider its relationship to each post-gen metric individually. This is shown in Figure 6. Irrespective of architecture and/or dataset, we observe a broadly linear relationship, despite some evidence of non-linearity at the lower ends of the

<sup>5</sup>All correlations are significant at  $p < 0.001$ .

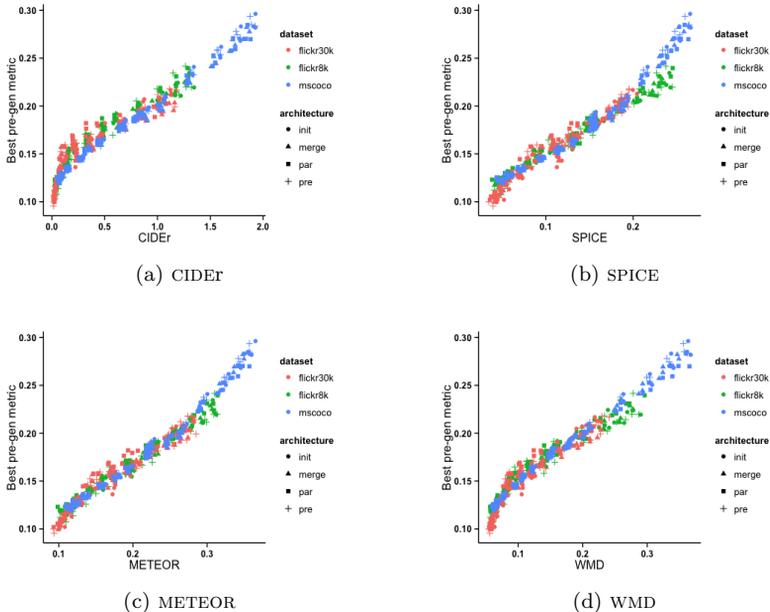


Figure 6: Best pre-gen metric (*mean\_max\_normcount\_prefix0*) vs post-gen metrics. The overall correlation has an  $R^2$  of 0.94. (Best viewed in colour.)

scale, especially for CIDEr and WMD. This supports the hypothesis made at the outset, namely, that it is possible to predict the quality of captions, as measured by a standard metric, by considering the probability of the reference captions in the test set, without the need to generate the captions themselves.

## 5 Discussion and conclusion

We have introduced and defined the concept of pre-gen metrics and described a methodology to search for useful variants of these metrics. Our results show that pre-gen metrics closely approximate a variety of standard evaluation measures.

These results can be attributed to the fact that neural captioning models share core assumptions about the sampling mechanisms that underlie generation, and that standard evaluation metrics ultimately assess the output of this sampling process. Thus, it is possible to predict the quality of the output, as measured by a post-gen metric, using the probability distribution that a trained model predicts over prefixes of varying length in the reference captions. The practical implication is that pre-gen metrics can act as quick and efficient evaluation proxies during development. The theoretical implication is that the correlations among standard evaluation metrics reported in the literature are due, at least in part, to core sampling mechanisms shared by most neural generation

architectures.

In future work, we plan to experiment with tuning captioning models using pre-gen metrics. We also wish to compare pre-gen metrics directly to human judgments.

## Acknowledgments

The research in this paper is partially funded by the Endeavour Scholarship Scheme (Malta). Scholarships are part-financed by the European Union - European Social Fund (ESF) - Operational Programme II Cohesion Policy 2014-2020 Investing in human capital to create more opportunities and promote the well-being of society.

## References

- [1] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, 2016.
- [2] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proc. Workshop on Intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72, 2005.
- [3] Raffaella Bernardi, Ruket Cakici, Desmond Elliott, Aykut Erdem, Erkut Erdem, Nazli Ikizler-Cinbis, Frank Keller, Adrian Muscat, and Barbara Plank. Automatic Description Generation from Images: A Survey of Models, Datasets, and Evaluation Measures. *JAIR*, 55:409–442, 2016.
- [4] Aoife Cahill. Correlating Human and Automatic Evaluation of a German Surface Realiser. In *Proc. ACL-IJCNLP’09*, pages 97–100, 2009.
- [5] Chris Callison-Burch, Miles Osborne, and Philipp Koehn. Re-evaluating the Role of BLEU in Machine Translation Research. In *Proc. EACL’06*, pages 249–256, 2006.
- [6] J. Gregory Caporaso, Nita Deshpande, J. Lynn Fink, Philip E. Bourne, Kevin Bretonnel Cohen, and Lawrence Hunter. Intrinsic evaluation of text mining tools may not predict performance on realistic tasks. *Pacific Symposium on Biocomputing*, 13:640–651, 2008.
- [7] B Dorr, Christof Monz, Douglas Oard, Stacy President, David Zajic, and Richard Schwartz. Extrinsic Evaluation of Automatic Metrics. Technical report, Instititue for Advanced Computer Studies, Univ of Maryland, College Park, College Park, MD, 2004.
- [8] Desmond Elliott and Frank Keller. Image description using visual dependency representations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1292–1302, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.
- [9] Desmond Elliott and Frank Keller. Comparing Automatic Evaluation Measures for Image Description. In *Proc. ACL’14*, pages 452–457, 2014.
- [10] Dominic Espinosa, Rajakrishnan Rajkumar, Michael White, and Shoshana Berleant. Further Meta-Evaluation of Broad-Coverage Surface Realization. In *Proc. EMNLP’10*, pages 564–574, 2010.
- [11] Albert Gatt and Anja Belz. Introducing shared task evaluation to NLG: The TUNA shared task evaluation challenges. In Emiel Krahmer and Mariët Theune, editors, *Empirical methods in natural language generation*. Springer, Berlin and Heidelberg, 2010.

- [12] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing Image Description as a Ranking Task: Data, Models and Evaluation Metrics. *JAIR*, 47(1):853–899, 2013.
- [13] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2015.
- [14] Mert Kilickaya, Aykut Erdem, Nazli Ikizler-Cinbis, and Erkut Erdem. Re-evaluating automatic metrics for image captioning. In *Proceedings of the 15<sup>th</sup> Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Association for Computational Linguistics, 2017.
- [15] Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *CoRR*, 1411.2539, 2014.
- [16] Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, and Tamara L. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR 2011*. IEEE, June 2011.
- [17] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In Francis Bach and David Blei, editors, *Proceedings of the 32<sup>nd</sup> International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France, 07–09 Jul 2015. PMLR.
- [18] Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proc. ACL’04*, 2004.
- [19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proc. ECCV’14*, pages 740–755, 2014.
- [20] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *CoRR*, 1301.3781, 2013.
- [21] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A method for automatic evaluation of machine translation. In *Proc. ACL’02*, pages 311–318, 2002.
- [22] Ehud Reiter and Anja Belz. An Investigation into the Validity of Some Metrics for Automatically Evaluating Natural Language Generation Systems. *Computational Linguistics*, 35(4):529–558, 2009.

- [23] Marc Tanti, Albert Gatt, and Kenneth P. Camilleri. Where to put the image in an image caption generator. *Natural Language Engineering*, 24(3):467–489, April 2018.
- [24] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based image description evaluation. In *Proc. CVPR’15*, 2015.
- [25] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):652–663, April 2017.
- [26] Sander Wubben, Antal van den Bosch, and Emiel Krahmer. Sentence Simplification by Monolingual Machine Translation. In *Proc. ACL’12*, pages 1015–1024, 2012.
- [27] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2:67–78, 2014.