UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Generating Image Descriptions via Sequential Cross-Modal Alignment Guided by Human Gaze

Takmaz, E.; Pezzelle, S.; Beinborn, L.; Fernández, R.

DOI
10.18653/v1/2020.emnlp-main.377

Publication date
2020

Document Version
Final published version

Published in
2020 Conference on Empirical Methods in Natural Language Processing

License
CC BY

Link to publication

Citation for published version (APA):
Takmaz, E., Pezzelle, S., Beinborn, L., & Fernández, R. (2020). Generating Image Descriptions via Sequential Cross-Modal Alignment Guided by Human Gaze. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *2020 Conference on Empirical Methods in Natural Language Processing: EMNLP 2020 : proceedings of the conference : November 16-20, 2020* (pp. 4664–4677). The Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.377

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:14 Oct 2026

Generating Image Descriptions via Sequential Cross-Modal Alignment Guided by Human Gaze

Ece Takmaz[1], **Sandro Pezzelle**[1], **Lisa Beinborn**[2], **Raquel Fernández**[1]
[1]Institute for Logic, Language and Computation, University of Amsterdam
[2]Vrije Universiteit Amsterdam
{e.takmaz|s.pezzelle|raquel.fernandez}@uva.nl,
l.m.beinborn@vu.nl

Abstract

When speakers describe an image, they tend to look at objects before mentioning them. In this paper, we investigate such sequential cross-modal alignment by modelling the image description generation process computationally. We take as our starting point a state-of-the-art image captioning system and develop several model variants that exploit information from human gaze patterns recorded during language production. In particular, we propose the first approach to image description generation where visual processing is modelled *sequentially*. Our experiments and analyses confirm that better descriptions can be obtained by exploiting gaze-driven attention and shed light on human cognitive processes by comparing different ways of aligning the gaze modality with language production. We find that processing gaze data sequentially leads to descriptions that are better aligned to those produced by speakers, more diverse, and more natural—particularly when gaze is encoded with a dedicated recurrent component.

1 Introduction

Describing an image requires the coordination of different modalities. There is a long tradition of cognitive studies showing that the interplay between language and vision is complex. On the one hand, eye movements are influenced by the task at hand, such as locating objects or verbally describing an image (Buswell, 1935; Yarbus, 1967). On the other hand, visual information processing plays a role in guiding linguistic production (e.g., Griffin, 2004; Gleitman et al., 2007). Such cross-modal coordination unfolds sequentially in the specific task of image description (Coco and Keller, 2012)—i.e., objects tend to be looked at before being mentioned. Yet, the temporal alignment between the two modalities is not straightforward (Griffin and Bock, 2000; Vaidyanathan et al., 2015)

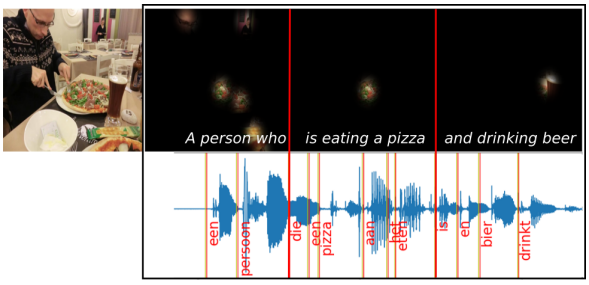

Figure 1: In our approach, an image captioning model is fed with a *sequence* of masked images encoding the gaze fixations of a *single* human speaker during language production. This diagram is a toy illustration.

In this paper, we follow up on these findings and investigate cross-modal alignment in image description by modelling the description generation process computationally. We take a state-of-the-art system for automatic image captioning (Anderson et al., 2018) and develop several model variants that exploit information derived from eye-tracking data. To train these models, we use a relatively small dataset of image descriptions in Dutch (DIDEC; van Miltenburg et al., 2018) that includes information on gaze patterns collected during language production. We hypothesise that a system that encodes gaze data as a proxy for human visual attention will lead to better, more human-like descriptions. In particular, we propose that training such a system with eye-movements sequentially aligned with utterances (see Figure 1) will produce descriptions that reflect the complex coordination across modalities observed in cognitive studies.

We develop a novel metric that measures the level of semantic and sequential alignment between descriptions and use it in two ways. First, we analyse cross-modal coordination in the DIDEC data, finding that the product of content and sequentiality better captures cross-modal correlations than content alone. Second, we test whether our

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 4664–4677,
November 16–20, 2020. ©2020 Association for Computational Linguistics

models generate captions that capture sequential alignment. Our experiments show that exploiting gaze-driven attention helps enhance image caption generation, and that processing gaze patterns sequentially results in descriptions that are better aligned with those produced by speakers, as well as being more diverse—both in terms of variability per image and overall vocabulary—particularly when gaze is encoded with a dedicated recurrent component that can better capture the complexity of the temporal alignment across modalities. Our data and code are publicly available at https://github.com/dmg-illc/didec-seq-gen.

Overall, this work presents the first computational model of image description generation where both visual and linguistic processing are modelled sequentially, and lends further support to cognitive theories of sequential cross-modal coordination.

2 Related Work

Image captioning Various models have been proposed to tackle the challenging task of generating a caption for a visual scene (Bernardi et al., 2016). Contemporary approaches make use of deep neural networks and encoder-decoder architectures (Sutskever et al., 2014). In the influential model by Vinyals et al. (2015), a Convolutional Neural Network (CNN) is used to encode the input image into a feature representation, which is then decoded by a Long Short-Term Memory network (LSTM; Hochreiter and Schmidhuber, 1997) that acts as a generative language model. In recent years, there have been many proposals to enhance this basic architecture. For instance, via extracting features from a lower layer of a CNN, Xu et al. (2015) obtain representations for multiple regions of an image over which attention can be applied by the LSTM decoder. The 'Bottom-up and Top-down Attention' model by Anderson et al. (2018) further refines this idea by extracting multiple image features with the help of Faster R-CNN (Ren et al., 2015), which results in the ability to focus on regions of different sizes better aligned with the objects in the image. Other models based on unsupervised methods (e.g., Feng et al., 2019) and Generative Adversarial Networks (Chen et al., 2019) have also been proposed recently.

We take as our starting point the model by Anderson et al. (2018) for two main reasons: first, it is among the best-performing architectures on standard image captioning benchmarks; second, its underlying idea (i.e., bottom-up and top-down attention) is explicitly inspired by human visual attention mechanisms (Buschman and Miller, 2007), which makes it suitable for investigating the impact of adding human gaze information.

Eye tracking In computer vision, human eye movements collected with eye-tracking methods have been exploited to model what is salient in an image or video for object detection (Papadopoulos et al., 2014), image classification (Karessli et al., 2017), image segmentation (Staudte et al., 2014), region labelling (Vaidyanathan et al., 2015, 2018), and action detection (Vasudevan et al., 2018). More relevant for the present study, gaze has also been used in automatic description generation tasks, such as video frame captioning (Yu et al., 2017) and image captioning (Sugano and Bulling, 2016; Chen and Zhao, 2018; He et al., 2019). In all these approaches, gaze data from different participants is *aggregated* into a *static* saliency map to represent an abstract notion of saliency. This aggregated gaze data is used as supervision to train models that predict generic visual saliency.

In contrast, in our approach, we model the production process of a *single* speaker by directly inputting information about where that speaker looks at during description production, and compare this to the aggregation approach. In addition, we exploit the *sequential* nature of gaze patterns, i.e., the so-called scanpath, and contrast this with the use of static saliency maps. Gaze scanpaths have been used in NLP for diverse purposes: For example, to aid part-of-speech tagging (Barrett et al., 2016) and chunking (Klerke and Plank, 2019); to act as a regulariser in sequence classification tasks (Barrett et al., 2018); as well as for automatic word acquisition (Qu and Chai, 2008) and reference resolution (Kennington et al., 2015). To our knowledge, the present study is the first attempt to investigate sequential gaze information for the specific task of image description generation.

3 Data

We utilise the Dutch Image Description and Eye-Tracking Corpus (DIDEC; van Miltenburg et al., 2018). In particular, we use the data collected as part of the description-view task in DIDEC, where participants utter a spoken description in Dutch for each image they look at. The gaze of the participants is recorded with an SMI RED 250 eye-tracking device while they describe an image. Over-

all, DIDEC consists of 4604 descriptions in Dutch (15 descriptions per image on average) for 307 MS COCO images (Lin et al., 2014). For each description, the audio, textual transcription, and the corresponding eye-tracking data are provided.

3.1 Preprocessing

We tokenise the raw captions, lowercase them, and exclude punctuation marks and information tokens indicating, e.g., repetitions (<rep>). We then use CMUSphinx[1] to obtain the time intervals of each word given an audio file and its transcription. See Appendix A for more details.

Gaze data in DIDEC is classified into gaze events such as fixations, saccades or blinks. We discard saccades and blinks (since there is no visual input during these events) and use only fixations that fall within the actual image. We treat consecutive occurrences of such fixations as belonging to the same fixation window.

3.2 Saliency maps

Using the extracted fixation windows, we create two types of saliency maps, *aggregated* and *sequential*, which indicate the prominence of certain image regions as signalled by human gaze.

Aggregated saliency maps *(per image)* The aggregated saliency map of an image is computed as the combination of all participants' gazes and represents what is generally prominent given the image description task. To create it, we first compute the saliency map of each participant who looked at the given image. Following Coco and Keller (2015), for each fixation window of the participant, we create a Gaussian mask centered at the window's centroid with a standard deviation of 1° of visual angle. Given the data collection setup of DIDEC, this standard deviation corresponds to 44 pixels. We sum up the masks weighted by relative fixation durations and normalise the resulting mask to have values in the range $[0, 1]$. Finally, we sum up and normalise the maps of all relevant participants to obtain the aggregated saliency map per image.

Sequential saliency maps *(per image-participant pair)* A sequential saliency map consists of a sequence of saliency maps aligned with the words in a description, and represents the scan pattern of a given participant over the course of description production. Using the temporal intervals extracted from the audio files, we align each word with the image regions fixated by the participant right before the word was uttered. For each word w_t—using the same method described above for aggregated maps—we combine all the fixation windows that took place between w_{t-1} and the onset of w_t and normalise them to obtain a word-level saliency map.[2] This way, we obtain a sequence of saliency maps per participant description.

3.3 Masked images and image features

The saliency maps are used to keep visible only the image regions that were highly attended by participants and to mask the image areas that were never or rarely looked at (see Figure 1). We create each masked image by calculating the element-wise multiplication between the corresponding 2D saliency map and each RGB channel in the original image. We then extract image features from the masked images using ResNet-101 (He et al., 2016) pre-trained on ImageNet (Deng et al., 2009). We take the output of the 2048-d average pooling layer as the image features to give as input to our models.

4 Evaluation Measures

We propose a novel metric to quantify the degree of both *semantic* and *sequential* alignment between two sentences. In our study, this metric will be leveraged in two ways: (1) to analyse cross-modal coordination in the DIDEC data (Section 5) and (2) to evaluate our generation models (Section 7). For context, we first briefly review several existing metrics for automatic image captioning.

Image Captioning metrics Image caption generation is evaluated by assessing some kind of similarity between the generated caption and one or more reference captions (i.e., those written by human annotators). One of the most commonly used metrics is CIDEr (Vedantam et al., 2015), which (a) computes the overlapping *n*-grams between the generated caption and the entire set of reference sentences for a given image, and (b) downweighs *n*-grams that are frequent in the entire corpus via *tf-idf* scores. Thus—regarding semantics and sequentiality—CIDEr scores can be affected by word order permutations, but not by the relative position of words in the entire caption nor

[1] https://cmusphinx.github.io/

[2] For the first word, we combine all the fixation windows that took place before its utterance. Some participants may look at an image before uttering the first word to obtain its gist (Oliva and Torralba, 2006). However, we do not encode these differences in behaviour explicitly.

by the presence of different but semantically similar words. Other metrics such as BLEU (which looks at n-gram precision; Papineni et al., 2002) and ROUGE-L (which considers n-gram recall; Lin, 2004) suffer from comparable limitations.

METEOR (Banerjee and Lavie, 2005) and SPICE (Anderson et al., 2016) also make use of n-grams (or tuples in a scene's graph, in the case of SPICE) and take into account semantic similarity by matching synonyms using WordNet (Pedersen et al., 2004). This allows for some flexibility, but can be too restrictive to grasp overall semantic similarity. To address this, Kilickaya et al. (2017) proposed using WMD, which builds on `word2vec` embeddings (Mikolov et al., 2013); more recently, several metrics capitalising on contextual embeddings (Devlin et al., 2019) were proposed, such as BERTScore (Zhang et al., 2020) and MoverScore (Zhao et al., 2019). However, these metrics neglect the sequential alignment of sentences.[3]

SSD We propose *Semantic and Sequential Distance* (SSD), a metric which takes into account both semantic similarity and the overall relative order of words. Regarding the latter, SSD is related to *Ordering-based Sequence Similarity* (OSS; Gómez-Alonso and Valls, 2008), a measure used by Coco and Keller (2010) to compare sequences of categories representing gaze patterns.[4] Given two sequences of words, i.e., a generated sentence G and a reference sentence R, SSD provides a single positive value representing the overall *dissimilarity* between G and R: the closer the value to 0, the higher the similarity between the two sentences (note that the value is unbounded). This single value is the average of two terms, *gr* and *rg*, which quantify the overall distance between G and R—the sum of their cosine (cos) and positional (pos) distance—from G to R and from R to G, respectively. The equation for *gr* is given below:

$$gr = \sum_{i=1}^{N} cos(G_i, R_s(i)) + pos(G_i, R_s(i)) \quad (1)$$

where $R_s(i)$ is the semantically closest element to G_i in R, and cos in our experiments is computed over `word2vec` embeddings trained on the 4B-token corpus in Dutch, COW (Tulkens et al., 2016).

[3]Moreover, metrics based on contextual embeddings have been shown to suffer with languages other than English.

[4]Despite its name, OSS is a *distance* measure. Note that it accounts for relative position, but not for semantic similarity.

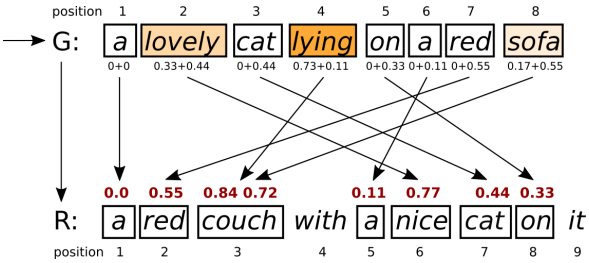

Figure 2: SSD. Computation of *gr* (Eq. 1). Sums below each word in G stand for $cos + pos$, darker shades of orange for higher cos distance. Value of gr is the sum of numbers in red (here 3.76). Best viewed in color.

Figure 2 illustrates how the metric works in practice. Full details are in Appendix B. For simplicity, the diagram only shows the computation in the gr direction. For example, consider the second element in G, 'lovely'. Its closest embedding in R is 'nice' ($cos = 0.33$). For each of these elements, we retrieve their position index (i.e., 2 for 'lovely' in G and 6 for 'nice' in R), compute their positional distance, and normalise it by the length of the longest sentence in the pair (here R), obtaining $|2-6|/9 \approx 0.44$. We then sum up the cosine distance and the positional distance to obtain a score for 'lovely': $0.33 + 0.44 = 0.77$. To obtain the overall *gr* value, we add up the scores for all words in G. We compute *rg* in a similar manner and obtain SSD as follows: $SSD = (gr + rg)/2$.

5 Cross-Modal Coordination Analysis

To empirically motivate our generation models, as a preliminary experiment we investigate the level of coordination between visual attention and linguistic production in the DIDEC dataset. In particular, we test whether scanpath similarity and sentence similarity are correlated and whether taking into account the sequential nature of the two modalities results in higher cross-modal alignment.

We transform gaze data into time-ordered sequences of object labels, i.e., scanpaths, (e.g., S = 'cat', 'person', 'cat', 'table') using the annotations of object bounding boxes in the MS COCO image dataset. On average, scanpaths have a length of 23.4 object labels. As for captions, we simply take the full sentences and treat them as sequences of words (e.g., C = 'a cute cat cuddled by a boy'). Descriptions contain an average of 12.8 tokens.

Order-sensitive analysis (*sequential*) For each image, we take the set of produced descriptions and compute all pairwise similarities by using SSD (see

Section 4). Similarly, we take the corresponding scanpaths and compute all pairwise similarities by using OSS (Gómez-Alonso and Valls, 2008). We then calculate Spearman's rank correlation (one-tailed) between the two similarity lists. This way, we obtain a correlation coefficient and p-value for each of the 307 images in the dataset.

Bag of Words analysis (*BoW*) We compare the correlation observed in the order-sensitive analysis with a BoW approach. Here, we represent a sentence as the average of the `word2vec` embeddings of the words it contains and a scanpath as a term-frequency vector. We then perform the same correlation analysis described above.

Random baseline (*random*) As a sanity check, using the stricter order-sensitive measures, for each image we re-compute the correlation between the two lists of similarities after randomly shuffling the sentences and corresponding scanpaths per image. We repeat this analysis 3 times.

Results As shown in Table 1, the highest level of alignment is observed in the *sequential* condition, where a significant positive correlation between scanpath and sentence similarities is found for 81 images out of 307 (26%). In BoW, the level of alignment is weaker: a positive correlation is found for 73 images (24%), with lower maximum correlation coefficients (0.65 vs. 0.49). Substantially weaker results can be seen in the *random* condition. These outcomes are in line with those obtained by Coco and Keller (2012) in a small dataset of 576 English sentences describing 24 images.

Overall, the results of the analysis indicate that the product of content and sequentiality better captures the coordination across modalities compared to content alone. Yet, the fact that positive correlations are present for only 26% of the images suggests that coordination across modalities is (not surprisingly) more complex than what can be captured by the present pairwise similarity computation, confirming the intricacy of the cross-modal temporal alignment (Griffin and Bock, 2000; Vaidyanathan et al., 2015). We take this aspect into account in our proposed generation models.

	sequential	*BoW*	*random*
# positively corr.	**81**	73	52.3 $\pm$ 5.774
% positively corr.	**0.26**	0.24	0.17 $\pm$ 0.015
Spearman's ρ (min)	0.15	0.15	0.15 $\pm$ 0.002
Spearman's ρ (max)	**0.65**	0.49	0.50 $\pm$ 0.042

Table 1: Results of the correlation analysis: number and percentage of images with statistically significant (p<0.05) positive correlations and range of coefficients in the three conditions. For random, avg. over 3 runs.

6 Models

The starting point for our models is the one by Anderson et al. (2018).[5] The main aspect that distinguishes this model from other image captioning systems is the use of Faster R-CNN (Ren et al., 2015) as image encoder, which identifies regions of the image that correspond to objects and are therefore more salient—the authors refer to this type of saliency detection as "bottom-up attention". Each object region i is transformed into an image feature vector v_i. The set of region vectors $\{v_1, \ldots, v_k\}$ is utilised in two ways by two LSTM modules: The first LSTM takes as input the mean-pooled image feature $\bar{\boldsymbol{v}}$ (i.e., the mean of all salient regions) at each time step, concatenated with the two standard elements of a language model, i.e., the previous hidden state and an embedding of the latest generated word. The hidden state of this first LSTM is then used by an attention mechanism to weight the vectors in $\{v_1, \ldots, v_k\}$—the authors refer to this kind of attention as "top-down". Finally, the resulting weighted average feature vector $\hat{v}_t$ is given as input to the second LSTM module, which generates the caption one word at a time. Note that the set of region vectors $\{v_1, \ldots, v_k\}$ and the mean-pooled vector $\bar{\boldsymbol{v}}$ are constant over the generation of a caption, while the weights over $\{v_1, \ldots, v_k\}$ and hence the weighted average feature vector $\hat{v}_t$ do change dynamically at each time step since they are influenced by the words generated so far.

We take the original model as our baseline and modify it to integrate visual attention defined by gaze behaviour. In particular, we replace the mean-pooled vector $\bar{\boldsymbol{v}}$ by a gaze vector g computed from masked images representing fixation patterns as explained in Section 3. We do not directly modify the set of object regions $\{v_1, \ldots, v_k\}$ present in the original model (i.e., bottom-up attention is still present in our proposed models). However, the top-down attention weights learned by the models

[5]The original implementation of this model can be found at: https://github.com/peteanderson80/bottom-up-attention. We developed our models building on the PyTorch re-implementation of the model available at: https://github.com/poojahira/image-captioning-bottom-up-top-down.

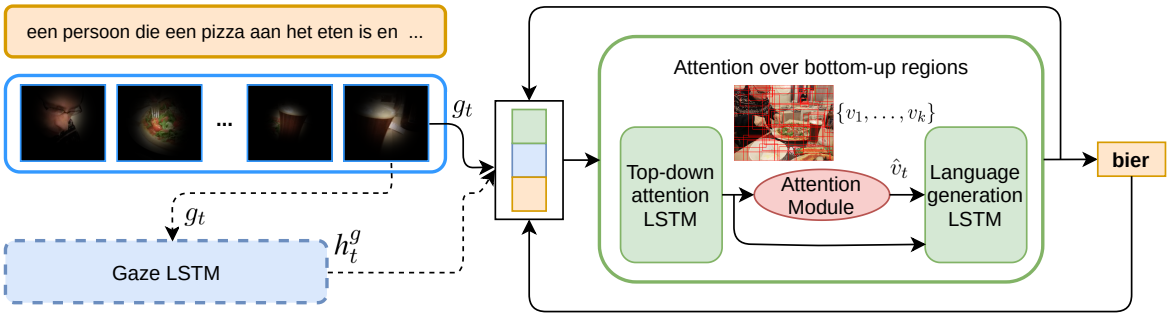

Figure 3: Architecture of the GAZE-SEQ and GAZE-2SEQ models. Dashed lines indicate that the connections to and from the Gaze LSTM are only present in the GAZE-2SEQ model.

are influenced by the gaze patterns given as input. Concretely, we test the following model conditions:

- NO-GAZE: The original model as described above, with exactly the same image feature vectors used by Anderson et al. (2018).
- GAZE-AGG: The mean-pooled vector $\overline{v}$ in the original model is replaced with a gaze image vector $\overline{g}$ computed on the image masked by the aggregated gaze saliency map. As explained in Section 3.2, this corresponds to the combination of all participants' fixations per image and hence remains constant over the course of generation.
- GAZE-SEQ: As depicted in Figure 3, we replace $\overline{v}$ with g_t, which are features computed for the image that was masked by the participant-specific sequential gaze saliency map at time t. Hence, g_t differs at each time step t. Building on the results of the correlation analysis, this sequential condition thus offers a model of the production process of a speaker where visual processing and language production are time-aligned.
- GAZE-2SEQ: Cross-modal coordination processes seem to go beyond simplistic content and temporal alignment (Griffin and Bock, 2000; Vaidyanathan et al., 2015). To allow for more flexibility, we add an extra gaze-dedicated LSTM component (labelled 'Gaze LSTM' in Figure 3), which processes the sequential gaze vector g_t and produces a hidden representation h_t^g. This dynamic hidden representation goes through a linear layer and then replaces $\overline{v}$ at each time step t.

For the three GAZE models, we also considered a version where $\overline{v}$ is concatenated with $\overline{g}$ or g_t as appropriate, rather than being replaced by the gaze vectors. Since they did not bring in better results, we do not discuss them further in the paper.

7 Experiments

We experiment with the proposed models using the DIDEC dataset and report results per model type.

7.1 Setup

We randomly split the DIDEC dataset at the image level, using 80% of the 307 images for training, 10% for validation, and 10% for testing. Further details are available in Appendix C.

Pre-training Since DIDEC is a relatively small dataset, we pre-train all our models using a translated version of train/val annotations of MS COCO 2017 version. We translated all the captions in the training and validation sets of MS COCO from English to Dutch using the Google Cloud Translation API.[6] We exclude all images present in our DIDEC validation and test sets from the training set of the translated MS COCO. We randomly split the original MS COCO validation set into validation and test. The final translated dataset in Dutch used for pre-training includes over 118k images for training, and 2.5k images for validation and testing, respectively, with an average of 5 captions per image.

Manual examination of a subset of translated captions showed that they are of good quality overall. Indeed, pre-training the NO-GAZE model with the translated corpus results in an improvement of about 21 CIDEr points (from 40.81 to 61.50) in the DIDEC validation set. Given that the MS COCO dataset is comprised of written captions compared to DIDEC, which includes spoken descriptions, these two datasets can have distinct characteristics. We expect the transfer learning approach to help mitigate this by allowing our models to learn the

[6] https://cloud.google.com/translate/

Model	*selected with SSD*			*selected with CIDEr*		
	SSD	CIDEr	BLEU-4	**CIDEr**	SSD	BLEU-4
NO-GAZE	5.86 (0.25)	55.04 (4.31)	39.09 (2.16)	52.45 (3.43)	6.09 (0.15)	35.60 (2.56)
GAZE-AGG	5.93 (0.10)	53.39 (3.56)	38.84 (1.70)	**55.74** (3.74)	5.97 (0.12)	37.69 (1.71)
GAZE-SEQ	5.82 (0.03)	56.16 (1.62)	39.80 (1.24)	53.59 (2.03)	6.10 (0.14)	36.09 (3.01)
GAZE-2SEQ	**5.81** (0.15)	53.55 (1.69)	38.05 (1.88)	52.94 (2.27)	5.93 (0.14)	36.27 (3.04)

Table 2: Test set results (average over 5 runs, with standard deviations in brackets) for the models selected with SSD and with CIDEr. Scores for BLEU-4 and SSD/CIDEr when not used for model selection are shown for reference only. For SSD, lower is better; for CIDEr and BLEU-4, higher is better.

features of spontaneous spoken descriptions during the fine-tuning phase.

All results reported below were obtained with pre-training (i.e., by initialising all models with the weights learned by the NO-GAZE model on the translated dataset and then fine-tuning on DIDEC).

Vocabulary and hyperparameters We use a vocabulary of 21,634 tokens consisting of the union of the entire DIDEC vocabulary and the translated MS COCO training set vocabulary. For all model types, we perform parameter search focusing on the learning rate, batch size, word embedding dimensions and the type of optimiser. The reported results refer to models trained with a learning rate of 0.0001 optimising the Cross-Entropy Loss with the Adam optimiser. The batch size is 64. The image features have 2048 dimensions and the hidden representations have 1024. The generations for the validation set were obtained through beam search with a beam width of 5. Best models were selected via either SSD or CIDEr scores on the validation set, with an early-stopping patience of 50 epochs.

More information regarding reproducibility can be found in Appendix D.

7.2 Results

The results obtained with different models are shown in Table 2. We report results on the test set, averaging over 5 runs with different random seeds. These scores are obtained with the best models selected on the validation set with either SSD or CIDEr.[7] For reference, we also include scores for other metrics not used for model selection. This allows us to check whether scores for other metrics are reasonably good when the models are optimised for a certain metric; however, only scores in the shaded columns allow us to extract conclusions on the relative performance of different model types.

[7]We use the library at https://github.com/Maluuba/nlg-eval to obtain corpus-level BLEU and CIDEr scores.

On average, the best GAZE models outperform the NO-GAZE model: 5.81 vs. 5.86 for SSD (lower is better) and 55.74 vs. 52.45 for CIDEr (higher is better). This indicates that eye-tracking data encodes patterns of attention that can contribute to the enhancement of image description generation. Zooming into the different gaze-injected conditions, we find that among the models selected with SSD, the sequential models perform better than GAZE-AGG (5.81 and 5.82 vs. 5.93). This shows that the proposed models succeed (to some extent) in capturing the *sequential* alignment across modalities, and that such alignment can be exploited for description generation. Interestingly, GAZE-2SEQ is the best-performing gaze model: it has the best average SSD across runs and the best absolute single run (5.70 vs. 5.79 and 5.80 by GAZE-SEQ and GAZE-AGG, respectively). This suggests that the higher flexibility and abstraction provided by the gaze-dedicated LSTM component offers a more adequate model of the intricate ways in which the two modalities are aligned.

As for the CIDEr-selected models, on average the gaze-injected models also perform better than NO-GAZE. The best results are obtained with GAZE-AGG (55.74). This is consistent with what CIDEr captures: it takes into account regularities across different descriptions of a given image; therefore, using a saliency map that combines the gaze patterns of several participants leads to higher scores than inputting sequential saliency maps, which model the path of fixations of each speaker independently. This variability seems to have a negative effect on CIDEr scores of sequential models, which are lower than GAZE-AGG; yet higher than NO-GAZE (53.59 and 52.94 vs. 52.45).

It is worth noting that CIDEr and BLEU-4 scores obtained with the SSD-selected models are sensible, which indicates that the generated descriptions do not suffer with respect to distinct aspects evaluated by other metrics when the models are optimised with SSD. Indeed, the highest CIDEr score

	specificity	**disfluency**	**compression**	**repetition**
NO-G	een vrouw die in de keuken staat… *(a woman who is standing in the kitchen…)*	een foto van een straat met een aantal vogels *(a photo of a street with a number of birds)*	een rode bus en een bus *(a red bus and a bus)*	een straat met **auto's** en **auto's** *(a street with **cars** and **cars**)*
2SEQ	een vrouw in een keuken met **donuts** *(a woman in the kitchen with **donuts**)*	**uh uh uh uh** met een aantal vogels *(**uh uh uh uh** with some birds)*	**twee** bussen die geparkeerd staan *(**two** buses that are parked)*	een straat in de stad met **auto's** en **auto's** *(a street in the city with **cars** and **cars**)*

Figure 4: Phenomena that are either particular to gaze models (specificity, disfluency, and compression) or common to all (repetition). Abbreviations NO-G and 2SEQ refer to NO-GAZE and GAZE-2SEQ, respectively.

obtained among models selected via SSD (GAZE-SEQ: 56.16) is even higher than that obtained by the best CIDEr-selected one (GAZE-AGG: 55.74). However, this is likely due to CIDEr being sensitive to lexical differences between the test set and the validation set used for model selection, which could lead to slightly different patterns.

8 Analysis

This section presents an analysis of the descriptions generated by the models on the test set (446 descriptions). We focus on one single run per model.

Cross-modal sequential alignment Given what SSD captures, our results indicate that the captions generated by GAZE-2SEQ are better aligned—in terms of semantic content and order of words—with the human captions than the ones generated by non-sequential models. Arguably, this enhanced alignment is driven by the specific information provided by the scanpath of each speaker. If this information is used effectively by the sequential models, then we should see more variation in their output. By definition, the non-sequential models generate only one single caption per image. Are the sequential models able to exploit the variation stemming from the speaker-specific scanpaths? Indeed, we find that GAZE-2SEQ generates an average of 4.4 different descriptions per image (i.e., 30% of the generated captions per image are unique).

Furthermore, we conjecture that tighter coordination between scanpaths and corresponding descriptions should give rise to more variation, since presumably the scanpath has a stronger causal effect on the description in such cases. To test this, we take the 30 images in the test set and divide them into two groups: (A) images for which a significant positive correlation was found in the cross-modal coordination analysis of Section 5; (B) all the others. These groups include, respectively, 10 and 20 images. As hypothesised, we observe a higher percentage of unique captions per image in A (35%) compared to B (27%).

Quantitative analysis We explore whether there are any quantitative differences across models regarding two aspects, i.e., the average length in tokens of the captions, and the size of the vocabulary produced. No striking differences are observed regarding caption length: NO-GAZE produces slightly shorter captions (avg. 7.5) compared to both GAZE-2SEQ (avg. 7.7) and GAZE-AGG (avg. 8.1). The difference, however, is negligible. Indeed, it appears that equipping models with gaze data does not make sentence length substantially closer to the length of reference captions (avg. 12.3 tokens).

In contrast, there are more pronounced differences regarding vocabulary. While GAZE-AGG has a similar vocabulary size (68 unique tokens produced) to NO-GAZE (63), GAZE-2SEQ is found to almost double it, with 109 unique tokens produced. Though this number is still far from the total size of the reference vocabulary (813), this trend suggests that a more diverse and perhaps 'targeted' language is encouraged when specific image regions are identified through gaze-based attention. The following qualitative analysis sheds some light on this hypothesis.

Qualitative analysis Manual inspection of all the captions generated by the models reveals interesting qualitative differences. First, captions generated by gaze-injected models are more likely to refer to objects—even when they are small and/or in the background—which are image-specific and thus very relevant for the caption. For example, when describing the leftmost image in Fig. 4, NO-GAZE does not mention the word *donuts*, which is produced by both GAZE-AGG and GAZE-2SEQ. Second, gaze-injected models produce language that seems to reflect uncertainty present in the visual input. For the second image of Fig. 4, e.g.,

both GAZE-AGG and GAZE-2SEQ generate disfluencies such as *uh* (interestingly, several participants' descriptions include similar disfluencies for this same image, which suggests some degree of uncertainty at the visual level); in contrast, in the entire test set no disfluencies are produced by NO-GAZE.

Finally, we find that GAZE-2SEQ is able to produce captions that somehow 'compress' a repetitive sequence (e.g., *a red bus and a bus*) into a shorter one, embedding a number (e.g., *two buses that are parked*; see third example in Fig. 4). This phenomenon is never observed in the output of other models (crucially, not even in GAZE-SEQ). We thus conjecture that this ability is due to the presence of the gaze-dedicated LSTM, which allows for a more abstract processing of the visual input. However, the presence of gaze data does not fully solve the issue of words being repeated within the same caption, as illustrated by the rightmost example in Fig. 4. Indeed, this weakness is common to all models, including the best performing GAZE-2SEQ.

9 Conclusions

We tackled the problem of automatically generating an image description from a novel perspective, by modelling the sequential visual processing of a speaker concurrently with language production. Our study shows that better descriptions—i.e., more aligned with speakers' productions in terms of content and order of words—can be obtained by equipping models with human gaze data. Moreover, this trend is more pronounced when gaze data is fed *sequentially*, in line with cognitive theories of sequential cross-modal alignment (e.g., Coco and Keller, 2012).

Our study was conducted using the Dutch language dataset DIDEC (van Miltenburg et al., 2018), which posed the additional challenges of dealing with a small amount of data and a low resource language. We believe, however, that there is value in conducting research with languages other than English. In the future, our approach and new evaluation measure could be applied to larger eye-tracking datasets, such as the English dataset by He et al. (2019). Since different eye-tracking datasets tend to make use of different gaze encodings and formats, the amount of pre-processing and analysis steps required to apply our method to other resources was beyond the scope of this paper. We leave testing whether the reported pattern of results holds across different languages to future work.

Despite the challenges mentioned above, our experiments show that a state-of-art image captioning model can be effectively extended to encode cognitive information present in human gaze behaviour. Comparing different ways of aligning the gaze modality with language production, as we have done in the present work, can shed light on how these processes unfold in human cognition. This type of computational modelling could help, for example, study the interaction between gaze and the production of filler words and repetitions, which we have not investigated in detail. Taken together, our results open the door to further work in this direction and support the case for computational approaches leveraging cognitive data.

Acknowledgments

We are grateful to Lieke Gelderloos for her help with the Dutch transcriptions, and to Jelle Zuidema and the participants of EurNLP 2019 for their feedback on a preliminary version of the work. Lisa Beinborn worked on the project mostly when being employed at the University of Amsterdam. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819455 awarded to Raquel Fernández).

References

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. SPICE: Semantic propositional image caption evaluation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Amsterdam, The Netherlands.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and VQA. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6077–6086.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Maria Barrett, Joachim Bingel, Nora Hollenstein, Marek Rei, and Anders Søgaard. 2018. Sequence classification with human attention. In *Proceedings of the 22nd Conference on Computational Natural*

Language Learning (CoNLL 2018), pages 302–312, Brussels, Belgium. Association for Computational Linguistics.

Maria Barrett, Joachim Bingel, Frank Keller, and Anders Søgaard. 2016. Weakly supervised part-of-speech tagging using eye-tracking data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 579–584, Berlin, Germany. Association for Computational Linguistics.

Raffaella Bernardi, Ruket Cakici, Desmond Elliott, Aykut Erdem, Erkut Erdem, Nazli Ikizler-Cinbis, Frank Keller, Adrian Muscat, and Barbara Plank. 2016. Automatic description generation from images: A survey of models, datasets, and evaluation measures. *Journal of Artificial Intelligence Research*, 55:409–442.

Timothy J. Buschman and Earl K. Miller. 2007. Top-down versus bottom-up control of attention in the prefrontal and posterior parietal cortices. *Science*, 315(5820):1860–1862.

Guy Thomas Buswell. 1935. *How people look at pictures: A study of the psychology and perception in art*. University Chicago Press.

Chen Chen, Shuai Mu, Wanpeng Xiao, Zexiong Ye, Liesi Wu, Fuming Ma, and Qi Ju. 2019. Improving image captioning with conditional generative adversarial nets. In *AAAI Conference on Artificial Intelligence*.

Shi Chen and Qi Zhao. 2018. Boosted attention: Leveraging human attention for image captioning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 68–84.

Moreno I. Coco and Frank Keller. 2010. Scan patterns on visual scenes predict sentence production. In *Proceedings of the 32nd Annual Meeting of the Cognitive Science Society*, pages 1934–1939.

Moreno I. Coco and Frank Keller. 2012. Scan patterns predict sentence production in the cross-modal processing of visual scenes. *Cognitive Science*, 36(7):1204–1223.

Moreno I. Coco and Frank Keller. 2015. Integrating mechanisms of visual guidance in naturalistic language production. *Cognitive Processing*, 6(2):131–150.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Yang Feng, Lin Ma, Wei Liu, and Jiebo Luo. 2019. Unsupervised image captioning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4120–4129.

Lila R Gleitman, David January, Rebecca Nappa, and John C Trueswell. 2007. On the give and take between event apprehension and utterance formulation. *Journal of Memory and Language*, 57(4):544–569.

Cristina Gómez-Alonso and Aida Valls. 2008. A similarity measure for sequences of categorical data based on the ordering of common elements. In *International Conference on Modeling Decisions for Artificial Intelligence*, pages 134–145. Springer.

Zenzi M. Griffin. 2004. *Why Look? Reasons for eye movements related to language production*, pages 213–248. Psychology Press.

Zenzi M. Griffin and Kathryn Bock. 2000. What the eyes say about speaking. *Psychological Science*, 11:274–9.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Sen He, Hamed R Tavakoli, Ali Borji, and Nicolas Pugeault. 2019. Human attention in image captioning: Dataset and analysis. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 8529–8538.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Nour Karessli, Zeynep Akata, Bernt Schiele, and Andreas Bulling. 2017. Gaze embeddings for zero-shot image classification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6412–6421.

Casey Kennington, Ryu Iida, Takenobu Tokunaga, and David Schlangen. 2015. Incrementally tracking reference in human/human dialogue using linguistic and extra-linguistic information. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 272–282, Denver, Colorado. Association for Computational Linguistics.

Mert Kilickaya, Aykut Erdem, Nazli Ikizler-Cinbis, and Erkut Erdem. 2017. Re-evaluating automatic metrics for image captioning. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 199–209. Association for Computational Linguistics.

Sigrid Klerke and Barbara Plank. 2019. At a glance: The impact of gaze aggregation views on syntactic tagging. In *Proceedings of the Beyond Vision and LANguage: inTEgrating Real-world kNowledge (LANTERN)*, pages 51–61, Hong Kong, China. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755. Springer International Publishing.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119.

Emiel van Miltenburg, Ákos Kádár, Ruud Koolen, and Emiel Krahmer. 2018. DIDEC: The Dutch Image Description and Eye-tracking Corpus. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 3658–3669. Association for Computational Linguistics.

Aude Oliva and Antonio Torralba. 2006. Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, 155:23–36.

Dim P. Papadopoulos, Alasdair D. F. Clarke, Frank Keller, and Vittorio Ferrari. 2014. Training object class detectors from eye tracking data. In *Proceedings of 13th European Conference on Computer Vision*, pages 361–376.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318. Association for Computational Linguistics.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. Wordnet:: Similarity: measuring the relatedness of concepts. In *Demonstration papers at HLT-NAACL 2004*, pages 38–41. Association for Computational Linguistics.

Shaolin Qu and Joyce Chai. 2008. Incorporating temporal and semantic information with eye gaze for automatic word acquisition in multimodal conversational systems. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 244–253.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, NIPS'15, pages 91–99.

Maria Staudte, Matthew W. Crocker, Alexis Héloir, and Michael Kipp. 2014. The influence of speaker gaze on listener comprehension: Contrasting visual versus intentional accounts. *Cognition*, 133(1):317–328.

Yusuke Sugano and Andreas Bulling. 2016. Seeing with humans: Gaze-assisted neural image captioning. *CoRR*, abs/1608.05203.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112.

Stephan Tulkens, Chris Emmery, and Walter Daelemans. 2016. Evaluating unsupervised dutch word embeddings as a linguistic resource. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Preethi Vaidyanathan, Emily Prudhommeaux, Cecilia Ovesdotter Alm, Jeff B Pelz, and Anne Haake. 2015. Alignment of eye movements and spoken language for semantic image understanding. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 76–81.

Preethi Vaidyanathan, Emily Prudhommeaux, Jeff B Pelz, and Cecilia Ovesdotter Alm. 2018. SNAG: Spoken narratives and gaze dataset. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 132–137.

Arun Balajee Vasudevan, Dengxin Dai, and Luc Van Gool. 2018. Object referring in videos with language and human gaze. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4129–4138.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: consensus-based image description evaluation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3156–3164.

Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *32nd International Conference on Machine Learning (ICML)*, pages 2048–2057.

Alfred L Yarbus. 1967. Eye movements during perception of complex objects. In *Eye movements and vision*, pages 171–211. Springer.

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

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 563–578.

Appendices

A Audio-Caption Alignment

In this appendix, we provide details on the pipeline used to time-align audio and descriptions. After processing a transcribed caption, we insert it as a grammar rule into a Java Speech Grammar Format (JSGF) file to be fed into CMUSphinx. As CMUSphinx supports English by default, we incorporated into the tool the phonetic and language models and the dictionary for Dutch as provided by the developers of CMUSphinx.[8]

Some words in our JSGF files were not in the VoxForge Dutch phonetic dictionary of CMUSphinx, which lists lexical items and their corresponding pronunciations in a format similar to ARPABET, adapted for Dutch.[9] To overcome this problem, we used eSpeak[10] to obtain the International Phonetic Alphabet (IPA) transcriptions of such out-of-vocabulary words. We obtained the set of IPA symbols existing in the transcriptions of out-of-vocabulary words and the set of ARPABET symbols in the dictionary. Then, a native speaker of Dutch, who is also a linguist, manually produced a mapping from these IPA symbols to ARPABET symbols of Dutch phonemes.[11] Given this mapping, we automatically converted out-of-vocabulary tokens into the required format and appended to the dictionary. A similar approach was also followed for numbers in numeric notation and certain English words.

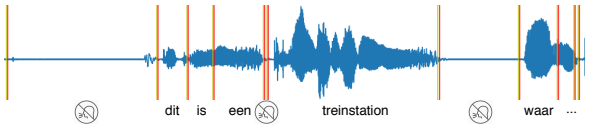

Figure 5: Temporal alignment of words in a transcribed caption and the corresponding audio file.

For some audio-caption pairs, the tool could not find an alignment matching the grammar. We turned off noise- and silence-removal and experimented with parameters related to beam-decoding in CMUSphinx to allow for a maximal number of complete alignments. However, we had to exclude some captions where there were unintelligible words in particular at the beginning or in the middle of the audio, since such an issue disrupts the alignment procedure.

Considering possible inter-participant differences in terms of pronunciation, the quality of audio files, and possible noise in the background of recordings, we assume that the time intervals of the words we obtained after these pre-processing steps are approximate indicators. Although there might be a few cases where the alignment is not quite accurate, we find this way of obtaining utterance timestamps reliable in general. An example audio-caption alignment is shown in Figure 5.

B SSD: Further Details

SSD is the average of two terms, *gr* and *rg*, which quantify the overall distance between a generated sentence (G) and a reference sentence (R). Eq. 2 (identical to Eq. 1 in Section 4) shows the calculation from G to R and Eq. 3 from R to G:

$$gr = \sum_{i=1}^{N} cos(G_{\mathrm{i}}, R_s(i)) + pos(G_{\mathrm{i}}, R_s(i)) \quad (2)$$

$$rg = \sum_{j=1}^{M} cos(R_{\mathrm{j}}, G_s(j)) + pos(R_{\mathrm{j}}, G_s(j)) \quad (3)$$

N and M refer to number of tokens in G and R, respectively. Cosine and positional distances are

[8] https://sourceforge.net/projects/cmusphinx/files/Acoustic%20and%20Language%20Models/Dutch/

[9] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[10] http://espeak.sourceforge.net/

[11] The mapping from IPA symbols to ARPABET symbols is provided in our GitHub repository.

Type	Description	SSD	gr	rg
R G	een dubbeldekker bus een dubbeldekker bus in een stad	1.41	2.82	0.00
R G	een dubbeldekkerbus in uh in engeland een dubbeldekker bus in een stad	2.64	2.72	2.55
R G	een rustige straat met een bus tegemoetkomend naar <unk> nummer 43 een dubbeldekker bus die op een weg rijdt	5.87	4.31	7.43
R G	een bus met lijn 43 die aan het rijden is waarvan uh de bus uit twee <unk> bestaat een dubbeldekker bus	8.62	0.43	16.81

Table 3: Examples of SSD scores for several descriptions generated (G) by GAZE-2SEQ compared to the reference description (R). gr and rg indicate the direction of the calculation. Lower SSD scores are better.

computed between the i_{th} element of G and another token, which is the most semantically similar word to G_i in R. $R_s(i)$ is the most semantically similar word to G_i and $G_s(j)$ is the most semantically similar word to R_j:

$$R_s(i) = \underset{j}{\operatorname{argmin}}(cos(G_{\text{i}}, R_{\text{j}})) \quad (4)$$

$$G_s(j) = \underset{i}{\operatorname{argmin}}(cos(R_{\text{j}}, G_{\text{i}})) \quad (5)$$

Table 3 shows some example descriptions generated by the GAZE-2SEQ model and corresponding references for a single image. We report the overall SSD scores along with gr and rg values separately.

C Data Split Statistics

Table 4 lists the number of images belonging to each split after we divide the DIDEC corpus (description-view partition) with respect to the images. In addition, the total number of captions in each split is provided.

	train	val	test	total
Images	247	30	30	307
Captions	3658	444	446	4548

Table 4: Number of images and captions.

The number of human descriptions per image varies in DIDEC and as we also removed some captions during preprocessing, images do not have an equal number of captions. Therefore, we report the average number of captions per image for each split, as well as their range, in Table 5.

	train	val	test	overall
Avg	14.81	14.80	14.87	14.81
Min	11	12	13	11
Max	16	16	16	16

Table 5: Number of captions per image.

D Reproducibility

We implemented and trained our models in Python version 3.6[12] and PyTorch version 0.4.1.[13] All models were run on a computer cluster with Debian Linux OS. Each model used a single GPU GeForce 1080Ti, 11GB GDDR5X, with NVIDIA driver version: 418.56 and CUDA version: 10.1.

Pre-training with the translated MS COCO dataset took approximately 5 days. NO-GAZE and GAZE-AGG took around 1.5 hours and GAZE-SEQ and GAZE-2SEQ models took 2 hours to fine-tune over the pre-trained model.

Since the pre-trained model and the fine-tuned NO-GAZE, GAZE-AGG and GAZE-SEQ models use essentially the same architecture, they have an equal number of parameters: 85 million. GAZE-2SEQ has more parameters due to the addition of the Gaze LSTM: 100 million.

In all the models, the biases in linear layers were set to 0 and the weights were uniformly sampled from the range (-0.1, 0.1). Embedding weights were initialised uniformly in the range (-0.1, 0.1). LSTM hidden states were initialised to 0.

Below we give details regarding the manually-tuned hyperparameters.

D.1 Hyperparameters for Pre-Training

We experimented with learning rate (0.001, 0.0001), dimensions for the word embeddings and

[12] https://www.python.org/downloads/release/python-360/

[13] https://pytorch.org/

hidden representations (512, 1024) and batch size (64, 128). The best pre-trained model is selected based on its CIDEr score on the validation split of our translated MS COCO dataset, with an early-stopping patience of 20 epochs. We use a learning rate of 0.0001 optimising the Cross-Entropy Loss with the Adam optimiser. The batch size is 128. The image features have 2048 dimensions and the hidden representations 1024. The generations for the validation set are obtained through beam search with a beam width of 5.

D.2 Hyperparameters for Fine-tuning

We experimented with the same set of hyperparameters as in pre-training. The details of the hyperparameters for the selected models were given in the main text. We select the models separately based on CIDEr scores and SSD scores. We train each model type with their selected configuration with 5 different random seeds to set the random behaviour of PyTorch and NumPy. We also turn off the cuDNN benchmark and also set cuDNN to deterministic.