SemStyle: Learning to Generate Stylised Image Captions using Unaligned Text

Alexander Mathews*†, Lexing Xie*†, Xuming He‡
Australian National University*, Data to Decision CRC†, ShanghaiTech University‡
alex.mathews@anu.edu.au, lexing.xie@anu.edu.au, hexm@shanghaitech.edu.cn

Abstract

Linguistic style is an essential part of written communication, with the power to affect both clarity and attractiveness. With recent advances in vision and language, we can start to tackle the problem of generating image captions that are both visually grounded and appropriately styled. Existing approaches either require styled training captions aligned to images or generate captions with low relevance. We develop a model that learns to generate visually relevant styled captions from a large corpus of styled text without aligned images. The core idea of this model, called SemStyle, is to separate semantics and style. One key component is a novel and concise semantic term representation generated using natural language processing techniques and frame semantics. In addition, we develop a unified language model that decodes sentences with diverse word choices and syntax for different styles. Evaluations, both automatic and manual, show captions from SemStyle preserve image semantics, are descriptive, and are style shifted. More broadly, this work provides possibilities to learn richer image descriptions from the plethora of linguistic data available on the web.

1. Introduction

An image can be described in different styles, for example, from a first-person or third-person perspective, with a positive or neutral sentiment, in a formal or informal voice. Style is an essential part of written communication that reflects personality [42], influences purchasing decisions [30] and fosters social interactions [10, 37]. The analysis of linguistic styles [10, 39] and generating natural language descriptions [54] are two fast developing topics in language understanding and computer vision, but an open challenge remains at their intersection: writing a visually relevant sentence in a given style. Incorporating style into image captions will help to communicate image content clearly, attractively, and in a way that is emotionally appropriate. For this work, we focus on writing one sentence to describe an image.

Style traditionally refers [42] to linguistic aspects other than the message content. It can be defined in terms of a fixed set of attributes [39, 14] such as formality and complexity, or implicitly with a document collection from a single author [46] or genre [24]. The early works on stylistic image captioning model word changes [36], and transformation of word embeddings [15], but do not explicitly separate content and style. These works also require manually created, style specific, image caption datasets [36, 15], and are unable to use large collections of styled text that does not describe images. We aim to address three gaps in the current solutions. The first is human-like style transfer: using large amounts of unrelated text in a given style to compose styled image captions. This is in contrast to existing systems that require aligned images and styled text. The second is representing an image so that the semantics are preserved while allowing flexible word and syntax use. The third is ensuring stylistic text remains descriptive and relevant to the image.

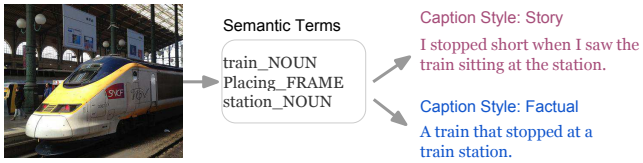


Figure 1. *SemStyle* distils an image into a set of semantic terms, which are then used to form captions of different styles.

We develop a model, dubbed *SemStyle*, for generating stylistically interesting and semantically relevant image captions by learning from a large corpus of stylised text without aligned images. Central to our approach is a separation of concerns regarding semantic relevance and style. We propose a novel semantic terms representation that is concise and promotes flexibility in word choice. This term representation consists of normalised words with part-of-speech tag, and verbs generalised using the lexical database FrameNet [5]. Further, we develop a *term generator* for obtaining a list of terms related to an image, and a *language generator* that decodes the ordered set of semantic terms into a stylised sentence. The *term generator* is trained on images and terms derived from factual captions. The *language generator* is trained on sentence collections and is conditioned to generate the desired style. As illustrated in

Figure 1, the *term generator* produces train_NOUN, Placing_FRAME, station_NOUN from the image, and the *language generator* produces sentences of different styles from this set of terms. Evaluated on both MSCOCO [8] and a corpus of romance novels [61], the SemStyle system produced distinctively styled captions in 58.8% of cases, while retaining visual semantics as judged by the SPICE metric [1]. Evaluated subjectively by the crowd, SemStyle achieved an average descriptiveness of 2.97 (out of 4, larger is more descriptive), which is competitive with a purely descriptive baseline at 2.95. Since this descriptive baseline is the basis of the *term generator* we can conclude that SemStyle retains the descriptive accuracy of the underlying semantic model. Moreover, 41.9% of captions from SemStyle were judged to be telling a story about the associated image. The main contributions of this paper are as follows:

- A concise semantic term representation for image and language semantics, implemented with a neural-network based *term generator*.
- A method that uses semantic terms to generate relevant captions with wording flexibility.
- A training strategy for learning to mimic sentence-level style using both styled and descriptive corpora.
- Competitive results in human and automatic evaluations with existing, and two novel, automated metrics for style. Dataset, models and evaluation results are released online [1].

2. Related work

This work is closely related to recent work on recognising linguistic style, captioning images with additional information, and the new topic of generating stylistic captions.

The problem of identifying writing style to identify authors has received much interest. Many features have been proposed [46] to disentangle style and content including: lexical features [4, 3, 50], bag of functional words [3] and Internet slang terms [4]. Paraphrasing, and word choice are shown to be important indicators of style [59, 39]. In online communities, writing style is shown to be indicative of personality [45], vary across different online fora, and change throughout a discussion [40]. Synthesising text in a particular style is an emerging problem in natural language generation [16, 22], but the quality of results is limited by the size of parallel text collection in two different styles [19]. In other cases the semantic content is not controlled and so may not be relevant to the subject such as an image or movie [14]. In this work we address both problems with respect to stylised image captioning.

Current state-of-the-art image captioning models consist of a Convolutional Neural Network [28] (CNN) for object detection, and a Recurrent Neural Network [13, 18] (RNN) for caption generation [12, 21, 25, 34, 54, 29, 60, 41, 49]. These two components can be composed and learnt jointly through back-propagation. Training requires hundreds of thousands of aligned image-caption pairs, and the resulting captions reflect the purely descriptive style of the training data. Generalising image caption generators beyond the standard caption datasets (such as MSCOCO [8]) is thus of interest. Some captioning systems are designed to generalise to unseen objects [33, 2, 52]; Luong *et al*. [31] exploit other linguistic tasks via multi-task sequence learning; Venugopalan [53] show that additional text improves the grammar of video captions. While this work leverages the general idea of multi-task learning, our specific proposal for separating semantics and style is new, as is our model for generating stylistic text.

Neural-storyteller [24] generates styled captions by retrieving candidate factual captions [25], embedding them in a vector space [26], shifting the embeddings by the mean of the target style (e.g. romance novels) before finally decoding. The resulting captions are representative of the target style but only loosely related to the image. Our earlier SentiCap system [36] generates captions expressing positive or negative sentiment. Training employs a switching RNN to adapt a language decoder using a small number of training sentences. This approach needs an aligned dataset of image captions with sentiment, and word-level annotations to emphasize words carrying sentiment. The StyleNet system [15] uses a factored weight matrix to project word embeddings. Style is encoded in this factored representation while all other parameters are shared across different styles. This approach uses styled image-captions pairs for learning. To best of our knowledge, no stylistic image caption system exists that is grounded on images, adapts both word choice and syntactic elements, and is able to learn on large collections of text without paired images.

3. Our Approach

We propose a novel encoder-decoder model for generating semantically relevant styled captions. First this model maps the image into a semantic term representation via the *term generator*, then the *language generator* uses these terms to generate a caption in the target style. This is illustrated in Figure 2.

The lower left of Figure 2 describes the *term generator*, which takes an image as input, extracts features using a CNN (Convolutional Neural Network) and then generates an ordered term sequence summarising the image semantics. The upper right of Figure 2 describes the *language generator*, which takes the term sequence as input, encodes it with an RNN (Recurrent Neural Network) and then using an attention based RNN decodes it into natural language with a specific style. We design a two-stage learning strategy enabling us to learn the *term generator* network using only a standard image caption dataset, such as

[1] https://github.com/computationalmedia/semstyle

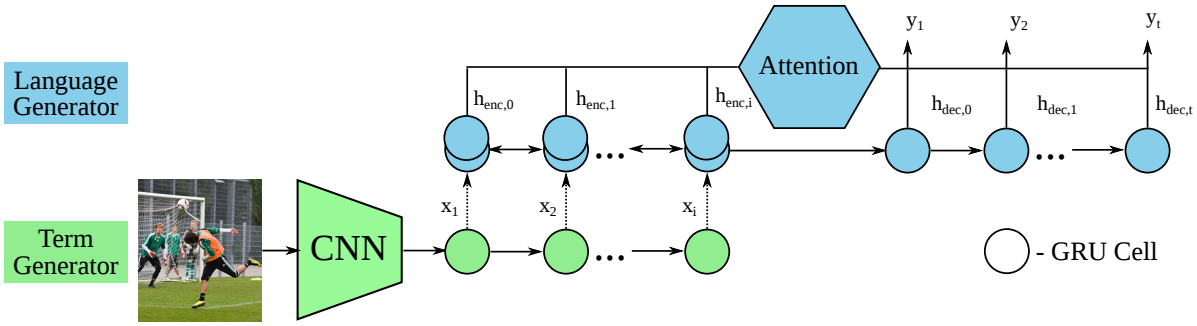


Figure 2. An overview of the *SemStyle* model. The *term generator* network (in green) is shown in the lower left. The *language generator* network is in the upper right (in blue) .

MSCOCO [8], and learn the *language generator* network on styled text data, such as romantic novels. The remainder of this section introduces our semantic representation and encoder-decoder neural network, while the learning method is discussed in Section 4.

3.1. Semantic term representation

To generate image captions that are both semantically relevant and appropriately styled, our structured semantic representation should capture visual semantics and be independent of linguistic style. We would like all semantics to be represented in the image, while language constructs that are stylistic in nature can be freely chosen by the *language generator*. Our representation also needs to fully capture the semantics, to avoid teaching the *language generator* to invent semantics. Since we also wish to train the *language generator* without images, we need a representation that can be extracted from text alone. In Section 6, we show this semantic sequence preserves the majority of real-world image and text semantics.

Formally, given a sentence $\mathbf{w} = \{w_1, w_2, \cdots, w_n\}$ with $w_i \in \mathcal{V}^{in}$, we define a set of rules mapping it to our ordered semantic terms $\mathbf{x} = \{x_1, x_2, \cdots, x_M\}, x_i \in \mathcal{V}^{term}$. Our goal is to define a set of semantic terms and mapping rules broad enough to encompass the semantics of both images and stylistic texts, and yet specific enough to avoid encoding style. Inspired by computational stylistics we construct three sets of rules:

A. Filtering non-semantic words. Function words are known to encode style rather than semantics, and are often used in authorship identification models [3, 50, 4]. Here we remove function words in order to encode semantics and strip out style. From input sentence s, we filter English stopwords and a small list of additional terms, either informal e.g. "nah", the result of tokenization e.g. "nt", or numbers e.g. "one", "two". Using Parts Of Speech (POS) tags we further remove: punctuation, adverbs, adjectives, pronouns and conjunctions. This importance ordering of POS types is derived from a data-driven perplexity evaluation described in the supplement [35]. Throughout this process we preserve common collocations such as "hot dog" and "fire hydrant". These collocations are from a pre-defined list, but automatic approaches [56] could also be used.

B. Lemmatization and tagging. Words from a sentence are converted to semantic terms to remove common surface variations. For most words we choose to lemmatize and concatenate with the POS tag, e.g. "rock" becomes "rock_NOUN". Lemmatization allows terms to be used more freely by the *language generator*, enabling stylistic choices such as tense and active/passive voice. POS tags distinguish among different senses of the same word, for example the verb "rock" and the noun "rock" are disparate. We use the spaCy[2] natural language toolkit for lemmatization and POS tagging.

C. Verb abstraction. Verbs are replaced with a FrameNet [5] frame, preserving much of the semantics without enforcing a particular word choice. FrameNet is a lexical database of semantic frames, which are a conceptual structure for describing events, relations, or objects along with their participants. For example, *sitting*, *laying*, *parking* all map to the *Placing* semantic frame. Table 1 contains five commonly used verb frames. We use the semantic role labelling tool SEMAFOR [27] to annotate frames. We then map these raw frames into a reduced frame vocabulary, consisting of frames occurring over 200 times in the MSCOCO training set. Out-of-vocabulary frames are mapped to an in-vocabulary ancestor via the FrameNet hierarchy. Failing this, the frame is filtered out. Intuitively, frames not occurring frequently in the MSCOCO set, and with no frequent ancestors, are unlikely to be visually grounded – for example the frame *Certainty*, with word lemmas *believe* and *trust*, is a frame with no obvious visual grounding.

The order of semantic terms is identical to the order in the original sentence. Results (Sec 6) show training with this ground truth ordering helps performance.

3.2. Generating semantic terms from images

We design a *term generator* network that maps an input image, denoted I, to an ordered sequences of seman-

[2] https://github.com/explosion/spaCy/tree/v1.9.0

Frame (count)	Common MSCOCO verbs
Placing (86,262)	sitting, parked, laying, hanging, leaning
Posture (45,150)	standing, lying, seated, kneeling, bends
Containing (32,040)	holding, holds, held, hold
Motion (22,378)	flying, going, swinging, fly, floating
Self motion (21,118)	walking, walks, walk, swimming

Table 1. The most common frames in the MSCOCO training set with frequency counts (in 596K training captions) and the most common verbs which instantiate them.

tic terms $\mathbf{x} = \{x_1, x_2, x_i, ..., x_M\}, x_i \in \mathcal{V}^{term}$. This is achieved with a CNN+RNN structure inspired by Show and Tell [54], and illustrated in the lower left of Figure 2. The image feature is extracted from the second last layer of the Inception-v3 [48] CNN pre-trained on ImageNet [43]. It passes through a densely connected layer, and then is provided as input to an RNN with Gated Recurrent Unit (GRU) cells [9]. The term list $\mathbf{x}$ is shorter than a full sentence, which speeds up training and alleviates the effect of forgetting long sequences.

At each time-step i, there are two inputs to the GRU cell. The first is the previous hidden state $\mathbf{h}_{i-1}$ summarising the image I and term history $x_1, ..., x_{i-1}$, the second is the embedding vector $\mathbf{E}_{x_i}$ of the current term. A fully connected layer with softmax non-linearity takes the output $\mathbf{h}_i$ and produces a categorical distribution for the next term in the sequence x_{i+1}. Argmax decoding can be used to recover the entire term sequence from the conditional probabilities:

$$x_{i+1} = \underset{j \in \mathcal{V}^{term}}{\operatorname{argmax}} P(x_{i+1} = j | I, x_i ... x_1) \quad (1)$$

We set x_1 to be a beginning-of-sequence token and terminate when the sequence exceeds a maximum length or the end-of-sequence token is generated.

3.3. Generating styled descriptions

The *language generator*, shown in the upper right of Figure 2, maps from a list of semantic terms to a sentence with a specific style. For example, given the term list "*dog_NOUN*", "*Self_motion_FRAME*", "*grass_NOUN*", a suitable target can be "*The dog bounded through the fresh grass.*". Given the list of semantic terms $\mathbf{x}$, we generate an output caption $\mathbf{y} = \{y_1, y_2, y_t, ..., y_L\}, y_t \in \mathcal{V}^{out}$ – where $\mathcal{V}^{out}$ is the output word vocabulary. To do so, we learn an RNN sequence-to-sequence *language generator* network with attention over the input sequence, using styled text without corresponding paired images.

The encoder component for sequence $\mathbf{x}$ consists of a Bidirectional RNN [44] with GRU cells and a learn-able term to vector embedding. The Bidirectional RNN is implemented as two independent RNNs running in opposite directions with shared term embeddings. Hidden outputs from the forward RNN $\mathbf{h}_{fwd,i}$ and the backward RNN $\mathbf{h}_{bak,i}$ are concatenated to form the hidden outputs of the encoder $\mathbf{h}_{enc,i} = [\mathbf{h}_{fwd,i}, \mathbf{h}_{bak,i}]$. The last of these hidden outputs is used to initialise the hidden state of the decoder $\mathbf{h}_{dec,0} = \mathbf{h}_{enc,M}$. The decoder itself is a unidirectional RNN (only a single forwards RNN) with GRU cells, learnable word embeddings, attention layer, and a softmax output layer.

The attention layer connects selectively weighted encoder hidden states directly to decoder cells, using weightings defined by a learnt similarity (Equations 2 & 3). This avoids compressing the entire sequence into a single fixed length vector which improves performance in sequence-to-sequence modelling [58, 47, 32]. Attention vector $\mathbf{a}_t = (a_{t,1}, ..., a_{t,i}, ..., a_{t,M})$ quantifies the importance of the input term i to the current output time-step t. We compute the attention vector as a softmax over similarity $\mathbf{v}_t$ with learnt weight matrix W^a, defined as:

$$\begin{aligned} v_{t,i} &= \mathbf{h}_{enc,i}^{\top} W^a \mathbf{h}_{dec,t} \\ a_{t,i} &= \exp(v_{t,i}) / \sum_{j=1}^{M} \exp(v_{t,j}) \end{aligned} \quad (2)$$

Using the attention we compute a context vector that summarises the important hidden outputs of the encoder for the current decoder time step. The context vector at step t is defined as a weighted sum of the hidden outputs:

$$\mathbf{c}_t = \sum_{i=1}^{M} a_{t,i} \mathbf{h}_{enc,i} \quad (3)$$

To produce the output distribution we concatenate the context vector $\mathbf{c}_t$ with the hidden output of the decoder component $\mathbf{h}_{dec,t}$, and apply a fully connected layer with softmax non-linearity:

$$\begin{aligned} \mathbf{h}_{out,t} &= W^{out}[\mathbf{c}_t, \mathbf{h}_{dec,t}] + \mathbf{b}^{out} \\ p(y_t = k | \mathbf{x}) &= \exp(h_{out,t,k}) / \sum_{j=1}^{|\mathcal{V}^{out}|} \exp(h_{out,t,j}) \end{aligned} \quad (4)$$

Here $|\mathcal{V}^{out}|$ denotes the output vocabulary size, $[\mathbf{c}_t, \mathbf{h}_{dec,t}]$ denotes vector concatenation, $W^{out}, \mathbf{b}^{out}$ are both learnt parameter of the output layer, and t is an index to the current element of the decoded sequence.

4. Learning with Unpaired Styled Texts

The SemStyle network learns on existing image caption datasets with only factual descriptions, plus a large set of styled texts without aligned images. To achieve this, we develop a two-stage training strategy for the *term generator* and *language generator*.

4.1. Training the term generator

We train the *term generator* network on an image caption dataset with factual descriptions, such as MSCOCO.

The ground truth semantic sequence for each image is constructed from the corresponding ground truth descriptive captions by following the steps in Section 3.1.

For each image, the loss function is the mean categorical cross entropy over semantic terms in the sequence:

$$\mathcal{L} = -\frac{1}{M}\sum_{i=1}^{M}\log p(x_i = \hat{x}_i | I, \hat{x}_{i-1}...\hat{x}_1) \quad (5)$$

Here $\hat{x}$ denotes ground truth terms. At training time the input terms $\hat{x}_{i-1}...\hat{x}_1$ are ground truth – this is the common teacher forcing technique [57]. We found that schedule sampling [6] – where sampled outputs are fed as inputs during training – did not improve performance, despite recent work on longer sequences achieving small gains [55].

4.2. Training the language generator

The *language generator* described in Section 3.3 takes a semantic term sequence $\mathbf{x}$ as input and generates a sentence $\mathbf{y}$ in the desired style. To create training data we take a training sentence $\mathbf{y}$ and map it to a semantic sequence $\mathbf{x}$ according to the steps in Section 3.1. The loss function is categorical cross entropy.

We train the *language generator* with both styled and descriptive sentences. This produces a richer language model able to use descriptive terms infrequent in styled sentences. Training only requires text, making it adaptable to many different datasets.

Concatenating both datasets leads to two possible output styles; however, we wish to specify the style. Our solution is to provide a *target-style term* during training and testing. Specifically, our *language generator* network is trained on both the descriptive captions and the styled text with a *target-style term*, indicating provenance, appended to each input sequence. As our encoder is bidirectional we expect it is not sensitive to term placement at the beginning or end of the sequence, while a term at every time step would increase model complexity. This technique has previously been used is sequence-to-sequence models for many-to-many translation [20]. In Section 6 we demonstrate that purely descriptive or styled captions can be generated from a single trained model by changing the *target-style term*.

5. Evaluation settings

Both the *term generator* and *language generator* use separate 512 dimensional GRUs and term or word embedding vectors. The *term generator* has a vocabulary of 10000 terms while the *language generator* has two vocabularies: one for encoder input another for the decoder – both vocabularies have 20000 entries to account for a broader scope. The number of intersecting terms between the *term generator* and the *language generator* is 8266 with both datasets, and 6736 without. Image embeddings come from the second last layer of the Inception-v3 CNN [48] and are 2048 dimensional.

Learning uses mini-batch stochastic gradient decent method ADAM [23] with learning rate 0.001. We clip gradients to $[-5, 5]$ and apply dropout to image and sentence embeddings. The mini-batch size is 128 for both the *term generator* and the *language generator*. For the *language generator* each mini-batch is composed of 64 styled sentences and 64 image captions. To achieve this one-to-one ratio we randomly down-sample the larger of the two datasets at the start of each epoch.

At test time both the *term generator* and the *language generator* use greedy decoding: the most likely word is chosen as input for the next time step. The code and trained models are released online [1].

5.1. Datasets

Descriptive image captions come from the MSCOCO dataset [8] of 82783 training images and 40504 validation images, with 5 descriptive captions each. It is common practice [54] to merge a large portion of this validation set into the training set to improve captioning performance. We reserve 4000 images in the validation set as a test set, the rest we merged into training set. The resulting training set has 119287 images and 596435 captions.

The styled text consists of 1567 romance novels from bookcorpus [61] – comprising 596MB of text and 9.3 million sentences. We filter out sentences with less than 10 characters, less than 4 words, or more than 20 words. We further filter sentences not containing any of the 300 most frequent non stop-words from the MSCOCO dataset – leaving 2.5 million sentences that are more likely to be relevant for captioning images. Our stop-word list is from NLTK [7] and comparisons are on stemmed words. For faster training and to balance the styled and descriptive datasets we further down-sample to 578,717 sentences, with preference given to sentences containing the most frequent MSCOCO words. We remove all but the most basic punctuation (commas, full stops and apostrophes), convert to lower-case, tokenise and replace numbers with a special token.

The StyleNet [15] test set was not released publicly at the time of writing, so we could not use it for comparisons.

5.2. Compared approaches

We evaluate 6 state-of-the-art baselines and 7 variants of *SemStyle* – extended details are in the supplement [35].

CNN+RNN-coco is based on the descriptive Show+Tell model [54], **TermRetrieval** uses the *term generator* to generate a list of words, which are then used to retrieve sentences from the styled text corpus. **StyleNet** generates styled captions, while **StyleNet-coco** generates descriptive captions. Both are a re-implementation of the Gan *et al*. [15] model with minor modifications to ensure convergence on our dataset. **neural-storyteller** is a model trained on romance text (from the same source as ours) and re-

leased by Kiros [24]. **JointEmbedding** maps images and sentences to a continuous multi-modal vector space [25], and uses a separate decoder, trained on the romance text, to decode from this space.

SemStyle is our full model. **SemStyle-unordered** is a variant of *SemStyle* with a randomised semantic term ordering; **SemStyle-words** is a variant where the semantic terms are raw words – they are not POS tagged, lemmatized or mapped to FrameNet frames; **SemStyle-lempos** is a variant where the semantic terms are lemmatized and POS tagged, but verbs are not mapped to FrameNet frames; **SemStyle-romonly** is *SemStyle* with the language generator trained only on the romantic novel dataset. **SemStyle-cocoonly** is *SemStyle* trained only on the MSCOCO dataset. **SemStyle-coco** is *SemStyle* trained on both datasets but with a MSCOCO *target-style term* used at test time to indicate descriptive captions should be generated.

5.3. Evaluation metrics

Automatic relevance metrics. Widely-used captioning metrics such as BLEU [38], METEOR [11] and CIDEr [51] are based on n-gram overlap. They are less relevant to stylised captioning since the goal is to change wording while preserving semantics. We include them for descriptive captions (Table 2); results for stylised captions are in the supplement [35]. The SPICE [1] metric computes an f-score over semantic tuples extracted from MSCOCO reference sentences [8]. This is less dependent on exact n-gram overlap, and is strongly correlated with human judgements of descriptiveness. In the following we interchangably use the terms descriptiveness and relevance.

Automatic style metrics. To the best of our knowledge, there are no well-recognised measures for style adherence. We propose three metrics, the first two use a language model in the target style, the second is a high-accuracy style classifier. LM is our first language model metric, it is the average perplexity in bits per word under a 4-gram model [17] built on the romance novels. Lower scores indicate stronger style. The GRULM metric is the bits per word under a GRU language model, with the structure of the *language generator* decoder without attention. The CLassifier Fraction (CLF) metric, is the fraction of generated captions classified as styled by a binary classifier. This classifier is logistic regression with 1,2-gram occurrence features trained on styled sentences and MSCOCO training captions. Its cross-validation precision is 0.992 at a recall of 0.991. We have released all three models [1].

Human evaluations of relevance and style. Automatic evaluation does not give a full picture of a captioning systems performance [8]; human evaluation can help us to better understand its strengths and weaknesses, with the end user in mind. We evaluate each image-caption pair with two crowd-sourced tasks on the CrowdFlower[3] platform. The first measures how descriptive a caption is to an image on a four point scale – from unrelated (1) to clear and accurate (4). The second task evaluates the degree of style transfer. We ask the evaluator to choose among three mutually exclusive options – that the caption: is likely to be part of a story related to the image (*story*), is from someone trying to describe the image to you (*desc*), or is completely unrelated to the image (*unrelated*). Note that most sentences in a romance novel are not identifiably romantic once taken out of context. Being part of a story is an identifiable property for a single sentence. We choose this over shareability, as used by Gan *et al*. [15], since being part of a story more concisely captures the literary quality of the styled text. We separate the descriptiveness and story aspects of human evaluation, after pilot runs found that the answer to descriptiveness interferes with the judgement about being part of a story.

Using each method we caption the same 300 random test images, and evaluate each with 3 workers – a total of 900 judgements per method. More details on the crowd-sourced evaluation, including a complete list of questions and guideline text, can be found in the supplement [35].

6. Results

Table 2 summarizes measurements of content relevance against factual (MSCOCO) captions. Table 3 and Figure 3 report automatic and human evaluations on caption style learned from romance novels.

Evaluating relevance. *SemStyle-coco* generates descriptive captions because the descriptive *target-style term* is used. It achieves semantic relevance scores comparable to the *CNN+RNN-coco*, with a SPICE of 0.157 vs 0.154, and BLEU-4 of 0.238 for both. This demonstrates that using semantic terms is a competitive way to distil image semantics, and that the *term generator* and *language generator* constitute an effective vision-to-language pipeline. Moreover, *SemStyle* can be configured to generate different caption styles just by changing the *target-style term* at test time – the complement of the CLF metric shows *SemStyle-coco* captions are classified as descriptive in 99.7% of cases.

Evaluating style. *SemStyle* succeeds in generating styled captions in 58.9% of cases, as judged by CLF, and receives a SPICE score of 0.144. The baselines *TermRetrieval*, *neural-storyteller* and *JointEmbedding* have significantly higher CLF scores, but much lower SPICE scores. *TermRetrieval* produces weakly descriptive sentences (SPICE of 0.088) because it is limited to reproducing the exact text of the styled dataset which yields lower recall for image semantics. Both *neural-storyteller* (SPICE 0.057), and *JointEmbedding* (SPICE 0.046), decode from a single embedding vector allowing less control over semantics than SemStyle.

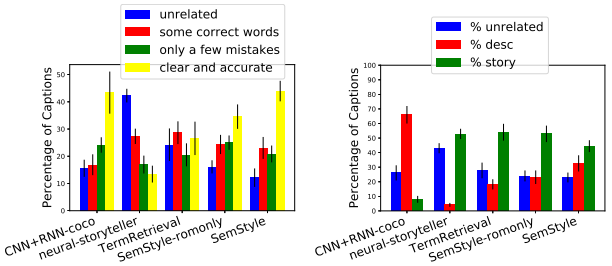
[3] https://www.crowdflower.com

Model	*BLEU-1*	*BLEU-4*	*METEOR*	*CIDEr*	*SPICE*	*CLF*	*LM*	*GRULM*
CNN+RNN-coco	0.667	0.238	0.224	0.772	0.154	0.001	6.591	6.270
StyleNet-coco	0.643	0.212	0.205	0.664	0.135	0.0	6.349	5.977
SemStyle-cocoonly	0.651	0.235	0.218	0.764	0.159	0.002	6.876	6.507
SemStyle-coco	0.653	0.238	0.219	0.769	0.157	0.003	6.905	6.691

Table 2. Evaluating caption relevance on the MSCOCO dataset. For metrics see Sec. 5.3, for approaches see Sec. 5.2.

Model	*SPICE*	*CLF*	*LM*	*GRULM*
StyleNet	0.010	0.415	7.487	6.830
TermRetrieval	0.088	0.945	3.758	4.438
neural-storyteller	0.057	0.983	5.349	5.342
JointEmbedding	0.046	0.99	3.978	3.790
SemStyle-unordered	0.134	0.501	5.560	5.201
SemStyle-words	0.146	0.407	5.208	5.096
SemStyle-lempos	0.148	0.533	5.240	5.090
SemStyle-romonly	0.138	0.770	4.853	4.699
SemStyle	0.144	0.589	4.937	4.759

Table 3. Evaluating styled captions with automated metrics. For *SPICE* and *CLF* larger is better, for *LM* and *GRULM* smaller is better. For metrics and baselines see Sec. 5.3 and Sec. 5.2.



(a) (b)

Figure 3. Human evaluations for SemStyle and selected baselines. (a) relevance measured on a four point scale, reported as percentage of generated captions at each level with 0.95 confidence interval error bars. (b) style conformity as a percentage of captions: unrelated to the image content, a basic description of the image, or part of a story relating to the image.

This leads to weaker caption relevance. *StyleNet-coco* produces factual sentences with comparable BLEU and SPICE scores. However, *StyleNet* produces styled sentences less frequently (CLF 41.5%) and with significantly lower semantic relevance – SPICE of 0.010 compared to 0.144 for *SemStyle*. We observe that the original *StyleNet* dataset [15] mostly consists of factual captions re-written by adding or editing a few words. The romance novels in the book corpus, on the other hand, have very different linguistic patterns to COCO captions. We posit that the factored input weights in *StyleNet* work well for small edits, but have difficulty capturing richer and more drastic changes. For *SemStyle*, the semantic term space and a separate *language generator* make it amenable to larger stylistic changes.

Coverage of semantic terms. We find that most of the terms generated by the *term generator* are represented in the final caption by the *language generator*. Of the Non-FrameNet terms 94% are represented, while 96% of verb frames are represented. Evaluating the full *SemStyle* pipeline involves mapping generated and ground truth captions to our semantic term space and then calculating multi-reference precision (BLEU-1) and recall (ROUGE-1). *SemStyle* gets a higher precision 0.626 and recall 0.517 than all styled caption baselines, and is close to the best descriptive model *SemStyle-cocoonly* with precision 0.636 and recall 0.531. Full results are in the supplement [35].

Human evaluations are summarised in Figure 3, tabular results and statistical significance testing are in the supplement [35]. Figure 3(a) shows image-caption relevance judged on a scale of 1 (unrelated) to 4 (clear and accurate). *StyleNet* was not included in the human evaluations since it scored significantly worse than others in the automatic metrics, especially SPICE and LM. *SemStyle* has a mean relevance of 2.97 while *CNN+RNN-coco* has 2.95. In addition, only 12.2% of *SemStyle* captions are judged as *unrelated*, the lowest among all approaches. *SemStyle* produces *clear and accurate* captions 43.8% of the time, while *CNN+RNN-coco* produces them 43.4% of the time – significantly higher than other approaches. As the CNN+RNN architecture is the basis of the *term generator*, this indicates our semantic term mapping and separate styled *language generator* do not reduce the relevance of the generated captions. *TermRetrieval* has mean relevance 2.50, and *neural-storyteller* 2.02 – both significantly lower than *SemStyle*. *neural-storyteller* generates a large fraction of completely unrelated captions (42.3%) while *TermRetrieval* avoids doing so (24.4%). *SemStyle-romonly* produces fewer clear and accurate captions than *SemStyle* (34.7% vs 43.8%), which demonstrates improved caption relevance when both training datasets are combined.

Figure 3(b) summarises crowd-worker choices among being *story-like*, *descriptive*, or *unrelated*. The two *SemStyle* variants have the lowest ($< 25\%$) fraction of captions that are judged *unrelated*. *SemStyle* generates story like captions 41.9% of the time, which is far more frequently than the *CNN+RNN-coco* trained on MSCOCO at 6.2%. *neural-storyteller* produces captions that are judged as story like 52.6% of the time, but at the expense of 44.2% completely unrelated captions. *TermRetrieval* produces captions that are story like 55.5% of the time and unrelated only 26.0% of the time; however, as shown in Figure 3(a), the relevance to images is low.

We calculate the correlation between the three new style

Figure 4. Example results, includes styled (**Story**) output from *SemStyle* and descriptive (**Desc**) output from *SemStyle-coco*. Four success cases are on the left (a,b,c,d), and two failures are on the right (e,f).

metrics (in Section 5.3) and human *story-like* judgements. Following the method of Anderson *et al*. [1], Kendall's τ correlation co-efficient is: 0.434 for CLF, 0.150 for LM, and 0.091 for GRULM.

Evaluating modeling choices of SemStyle. The last 5 rows of Table 3 highlight trade-offs among variants of *SemStyle*. Randomly ordering the semantic terms during training and testing, *SemStyle-unordered*, leads to captions with less semantic relevance, shown by a SPICE of 0.134 compared to 0.144 for the full model. They also conform less to the target style with a CLF of 0.501 compared to 0.589.

Using a raw word term space *SemStyle-words* (without FrameNet, lemmatization or POS tags) gives similar semantic relevance, SPICE of 0.146 to the full models 0.144, but less styling with CLF at 0.407. Using verb lemmas rather than FrameNet terms as in *SemStyle-lempos*, has a similar effect, with a slight increase in SPICE to 0.148 and a decrease in style to a CLF of 0.533. This clearly demonstrates the three components FrameNet, lemmatization and POS tags all contribute to remove style from the intermediate representation, and thus lead to output in the target style.

Learning from both datasets improves caption relevance. If we only train on the romantic novel corpus as in *SemStyle-romonly*, we find strong conformity to the target style (CLF 0.770) but less semantic relevance, SPICE 0.138. Without the joint training some semantics terms from the MSCOCO dataset are never seen by the language generator at training time – meaning their semantic content is inaccessible at test time. Our joint training approach avoids these issues and allows style selection at test time.

Example captions. Figure 4 shows four success cases on the left (a,b,c,d) and two failures on the right (e,f). The success cases are story-like, such as "The woman stepped underneath her umbrella and walked in the rain." rather than "A woman walking with an umbrella in the rain.". They also tend to use more interesting verbs (due to FrameNet) – "He pulled out a horse carriage and **charged** down the street". We note *SemStyle* examples use more past tense (c), definite articles (b,d), and first person view (e,f) – statistics are included in the supplement [35]. The failures are caused by the *term generator* incorrectly identifying cows in the image (top row), or word use ("juicer") by the *language generator* that is grammatically correct, but contradicts commonsense (bottom row).

7. Conclusion

We propose *SemStyle*, a method to learn visually grounded style generation from texts without paired images. We develop a novel semantic term representation to disentangle content and style in descriptions. This allows us to learn a mapping from an image to a sequence of semantic terms that preserves visual content, and a decoding scheme that generates a styled description. Future work includes learning from a richer set of styles, and developing a recognised set of automated and subjective metrics for styled captions.

Acknowledgments This work is supported, in part, by the Australian Research Council via project DP180101985. The Tesla K40 used for this research was donated by the NVIDIA Corporation. We thank S. Mishra, S. Toyer, A. Tran, and S. Wu for helpful suggestions.

References

[1] P. Anderson, B. Fernando, M. Johnson, and S. Gould. SPICE : Semantic Propositional Image Caption Evaluation. *ECCV'16*, 1:382–398, 2016.

[2] L. Anne Hendricks, S. Venugopalan, M. Rohrbach, R. Mooney, K. Saenko, and T. Darrell. Deep compositional captioning: Describing novel object categories without paired training data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–10, 2016.

[3] S. Argamon and S. Levitan. Measuring the usefulness of function words for authorship attribution. In *Proc. of the ACH/ALLC*, pages 1–3, 2005.

[4] S. Argamon, M. Šarić, and S. Stein. Style mining of electronic messages for multiple authorship discrimination: first results. *Knowledge discovery and data mining*, pages 475–480, 2003.

[5] C. F. Baker, C. J. Fillmore, and J. B. Lowe. The Berkeley FrameNet Project. In *Proceedings of the 36th annual meeting on Association for Computational Linguistics -*, volume 1, page 86, Morristown, NJ, USA, 1998. Association for Computational Linguistics.

[6] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer. Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks. *NIPS*, 2015.

[7] S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python*, volume 43. 2009.

[8] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollar, and C. L. Zitnick. Microsoft COCO captions: Data collection and evaluation server. *arXiv:1504.00325*, 2015.

[9] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, 2014.

[10] C. Danescu-Niculescu-Mizil, M. Gamon, and S. Dumais. Mark my words!: linguistic style accommodation in social media. In *Proceedings of the 20th international conference on World wide web*, pages 745–754. ACM, 2011.

[11] M. Denkowski and A. Lavie. Meteor Universal: Language Specific Translation Evaluation for Any Target Language. In *EACL 2014 Workshop on Statistical Machine Translation*, pages 376–380, 2014.

[12] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. *CVPR*, jun 2015.

[13] J. L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.

[14] J. Ficler and Y. Goldberg. Controlling linguistic style aspects in neural language generation. In *EMNLP Workshop on Stylistic Variation*, pages 94–104, 2017.

[15] C. Gan, Z. Gan, X. He, J. Gao, and L. Deng. StyleNet: Generating Attractive Visual Captions with Styles. *CVPR*, 2017.

[16] M. Ghazvininejad, X. Shi, Y. Choi, and K. Knight. Generating Topical Poetry. *EMNLP*, pages 1183–1191, 2016.

[17] K. Heafield, I. Pouzyrevsky, J. H. Clark, and P. Koehn. Scalable Modified Kneser-Ney Language Model Estimation. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 690–696, 2013.

[18] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[19] H. Jhamtani, V. Gangal, E. Hovy, and E. Nyberg. Shakespearizing modern language using copy-enriched sequence-to-sequence models. In *EMNLP Workshop on Stylistic Variation*, 2017.

[20] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, M. Hughes, and J. Dean. Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

[21] A. Karpathy and L. Fei-Fei. Deep Visual-Semantic Alignments for Generating Image Descriptions. *CVPR*, 2015.

[22] C. Kiddon, L. Zettlemoyer, and Y. Choi. Globally coherent text generation with neural checklist models. *EMNLP*, pages 329–339, 2016.

[23] D. P. Kingma and J. L. Ba. ADAM: A Method For Stochastic Optimization. *ICLR'15*, 2015.

[24] R. Kiros. neural-storyteller, 2015. `https://github.com/ryankiros/neural-storyteller`.

[25] R. Kiros, R. Salakhutdinov, and R. Zemel. Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models. *arXiv preprint arXiv:1411.2539*, pages 1–13, 2014.

[26] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler. Skip-Thought Vectors. *NIPS*, (786):1–9, 2015.

[27] M. Kshirsagar, S. Thomson, N. Schneider, J. Carbonell, N. A. Smith, and C. Dyer. Frame-Semantic Role Labeling with Heterogeneous Annotations. *Proceedings of ACL*, pages 218–224, 2015.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2323, 1998.

[29] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy. Improved image captioning via policy gradient optimization of spider. 3, 2017.

[30] S. Ludwig, K. De Ruyter, M. Friedman, E. C. Brüggen, M. Wetzels, and G. Pfann. More than words: The influence of affective content and linguistic style matches in online reviews on conversion rates. *Journal of Marketing*, 77(1):87–103, 2013.

[31] M.-T. Luong, Q. V. Le, I. Sutskever, O. Vinyals, and L. Kaiser. Multi-task Sequence to Sequence Learning. *ICLR*, 2016.

[32] M.-t. Luong, H. Pham, and C. D. Manning. Effective Approaches to Attention-based Neural Machine Translation. *EMNLP'15*, 2015.

[33] J. Mao, X. Wei, Y. Yang, J. Wang, Z. Huang, and A. L. Yuille. Learning like a child: Fast novel visual concept learning from sentence descriptions of images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2533–2541, 2015.

[34] J. Mao, W. Xu, Y. Yang, J. Wang, H. Huangzhi, and A. Yuille. Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). *ICLR'15*, 2015.

[35] A. Mathews, L. Xie, and X. He. Supplemental Material – SemStyle: Learning to Generate Stylised Image Captions using Unaligned Text, 2018. `https://github.com/computationalmedia/semstyle/blob/master/doc/supplement.pdf`.

[36] A. P. Mathews, L. Xie, and X. He. Senticap: Generating image descriptions with sentiments. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[37] K. G. Niederhoffer and J. W. Pennebaker. Linguistic style matching in social interaction. *Journal of Language and Social Psychology*, 21(4):337–360, 2002.

[38] K. Papineni, S. Roukos, T. Ward, and W. Zhu. BLEU: a method for automatic evaluation of machine translation. *ACL'02*, pages 311–318, 2002.

[39] E. Pavlick and A. Nenkova. Inducing lexical style properties for paraphrase and genre differentiation. In *HLT-NAACL*, pages 218–224, 2015.

[40] E. Pavlick and J. Tetreault. An empirical analysis of formality in online communication. *Transactions of the Association for Computational Linguistics*, 4:61–74, 2016.

[41] M. Pedersoli, T. Lucas, C. Schmid, and J. Verbeek. Areas of attention for image captioning. 2017.

[42] J. W. Pennebaker and L. A. King. Linguistic styles: language use as an individual difference. *Journal of personality and social psychology*, 77(6):1296, 1999.

[43] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[44] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[45] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, M. E. Seligman, et al. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, 8(9):e73791, 2013.

[46] E. Stamatatos. A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556, 2009.

[47] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to Sequence Learning with Neural Networks. *NIPS'14*, pages 3104–3112, 2014.

[48] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the Inception Architecture for Computer Vision. *CVPR*, 2016.

[49] Y. H. Tan and C. S. Chan. phi-lstm: a phrase-based hierarchical lstm model for image captioning. pages 101–117, 2016.

[50] H. Van Halteren, H. Baayen, F. Tweedie, M. Haverkort, and A. Neijt. New machine learning methods demonstrate the existence of a human stylome. *Journal of Quantitative Linguistics*, 12(1):65–77, 2005.

[51] R. Vedantam, C. L. Zitnick, and D. Parikh. CIDEr: Consensus-based image description evaluation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 07-12-June, pages 4566–4575, 2015.

[52] S. Venugopalan, L. Anne Hendricks, M. Rohrbach, R. Mooney, T. Darrell, and K. Saenko. Captioning images with diverse objects. July 2017.

[53] S. Venugopalan, L. A. Hendricks, R. Mooney, and K. Saenko. Improving lstm-based video description with linguistic knowledge mined from text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP-16)*, pages 1961–1966, Austin, Texas, 2016.

[54] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. *CVPR'15*, 2015.

[55] O. Vinyals, A. Toshev, S. Bengio, and D. 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, 2017.

[56] E. Wehrli, V. Seretan, and L. Nerima. Sentence Analysis and Collocation Identification. *Proceedings of the Multiword Expressions: From Theory to Applications (MWE 2010)*, pages 28–36, 2010.

[57] R. J. Williams and D. Zipser. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural Computation*, 1(2):270–280, 1989.

[58] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Ł. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *Arxiv*, pages 1–23, 2016.

[59] W. Xu, A. Ritter, B. Dolan, R. Grishman, and C. Cherry. Paraphrasing for style. *Proceedings of COLING 2012*, pages 2899–2914, 2012.

[60] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei. Boosting image captioning with attributes. *ICCV*, 2(5):8, 2017.

[61] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books. 2015.