

Notes: “Aesthetic Image Captioning From Weakly-Labelled Photographs”

Sun

April 27, 2020

1 Notes:

1.1 Summary:

1. Total meme of a paper. Included very neat pretraining and data preparation steps, though.
2. Focused on training an img2text model on a task where there is no real dataset, paper details the steps they took to create one.
3. Weakly supervised model.

1.2 Medium level:

The goal of this paper was to generate critical textual feedback for photographs of many different modalities (but all lying within ‘artistic photography’). There are datasets for similar purposes (namely images paired together with comments), however, the vast majority of these comments aren’t critical, or are grammatically incorrect. The authors seem to delete or automagically correct these, but it’s not exactly clear in the paper.

The authors decide to quantify captions for a given image based off how ‘informative’ the caption is. They first broke the vocabulary into unigrams and bigrams; bigrams consisted of ‘descriptor-object’ pairs, such as ‘nice colors’, ‘too small’, ‘distracting background’, etc.. They then look at the TF-IDF of words in the vocabulary.

Each n-gram, ω , is then assigned a probability P , as:

$$P(\omega) = \frac{C_\omega}{\sum_{i=1}^D C_i} \quad (1)$$

where D is the vocabulary size C_ω is the corpus frequency of the n-gram ω .

The authors then represent a comment as the union of the unigrams (u_i) and bigrams (b_i), with a given sequence $S = (u_1, \dots, u_N) \cup (b_1, \dots, b_M) = S_u \cup S_b$. A comment is assigned an informativeness score, ρ , as:

$$\rho_s = -\frac{1}{2} \left[\log \prod_i^N P(u_i) + \log \prod_j^M P(b_j) \right] \quad (2)$$

(2) is just the average of the negative log probability of S_u and S_b .





Training Strategy				
(a) Noisy Data & Supervised CNN (NS)	i like the angle and the composition	i like the colors and the composition	i like the composition and the lighting	i like the composition and the bw
(b) Clean Data & Supervised CNN (CS)	i like the idea , but i think it would have been better if the door was in focus .	i like the colors and the water . the water is a little distracting .	i like the way the light hits the face and the background .	i like this shot . i like the way the lines lead the eye into the photo .
(c) Clean Data & Weakly Supervised CNN (CWS)	i like the composition , but i think it would have been better if you could have gotten a little more of the building	i like the composition and the colors . the water is a little too bright .	this is a great shot . i love the way the light is coming from the left .	i like the composition and the bw conversion .

Figure 1: Example outputs.

The score of a comment is created under the assumption that all n-grams are independent. As such, if the n-grams in a sentence have higher corpus probabilities, then the corresponding ρ score is low, due to the negative logarithm, and vice versa.

ρ will be higher for longer comments than others, however, long comments without ‘informative’ words are still discarded.

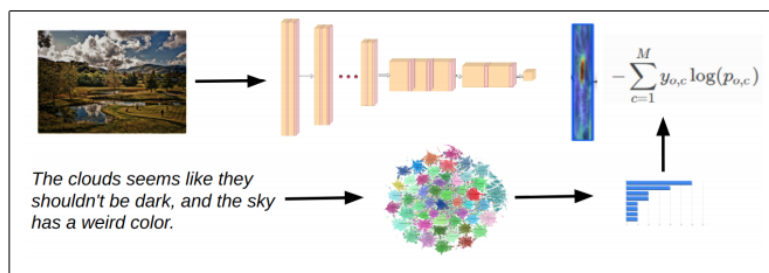
Comments below a certain threshold ($\rho = 20$) were deleted. Roughly 55% of the $\sim 3m$ corpus were deleted. These comments in hand, we still cannot train the CNN efficiently. There are $\sim 25k$ n-grams, many redundant. The authors then cluster semantically similar n-grams, such as ‘face’ and ‘ear’, ‘sky’ and ‘cloud’, etc..

The authors then use a technique called latent Dirichlet allocation.

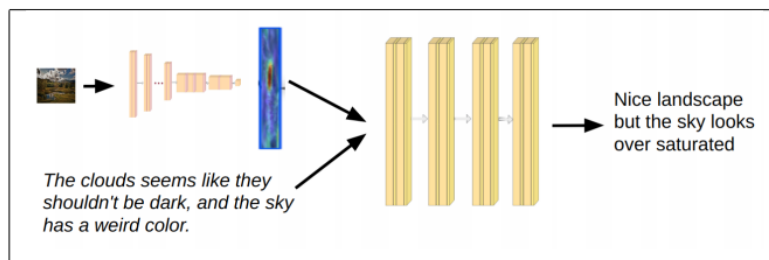
1.2.1 Latent Dirichlet Allocation (LDA):

Given a set of documents, $\mathcal{D} = \{D_1, \dots, D_N\}$, vocabulary of words, $\mathcal{W} = \{w_1, \dots, w_M\}$, the task is to infer K latent topics $\mathcal{T} = \{T_1, \dots, T_k\}$, where each topic can be seen as a collection of words, and each document can be seen as a collection of topics. This is usually done via a variational approximation.

The authors set $K = 200$, then used these topics as labels for the CNN. Once the CNN was trained, they simply used it as a feature extractor for the LSTM, which took it as input and tried to predict the ground truth caption.



(a) Weakly-supervised training of the CNN: Images and comments are provided as input. The image is fed to the CNN and the comment is fed to the inferred topic model. The topic model predicts a distribution over the topics which is used as a label for computing the loss for the CNN.



(b) Training the LSTM: Visual features extracted using the CNN and the comment is fed as an input to the LSTM which predicts a candidate caption.