

Picture It In Your Mind: Generating High Level Visual Representations From Textual Descriptions

Fabio Carrara
ISTI-CNR
via G. Moruzzi, 1
56124 Pisa, Italy
fabio.carrara@isti.cnr.it

Andrea Esuli
ISTI-CNR
via G. Moruzzi, 1
56124 Pisa, Italy
andrea.esuli@isti.cnr.it

Tiziano Fagni
ISTI-CNR
via G. Moruzzi, 1
56124 Pisa, Italy
tiziano.fagni@isti.cnr.it

Fabrizio Falchi
ISTI-CNR
via G. Moruzzi, 1
56124 Pisa, Italy
fabrizio.falchi@isti.cnr.it

Alejandro Moreo
Fernández
ISTI-CNR
via G. Moruzzi, 1
56124 Pisa, Italy
alejandro.moreo@isti.cnr.it

ABSTRACT

In this paper we tackle the problem of image search when the query is a short textual description of the image the user is looking for. We choose to implement the actual search process as a similarity search in a visual feature space, by learning to translate a textual query into a visual representation. Searching in the visual feature space has the advantage that any update to the translation model does not require to reprocess the, typically huge, image collection on which the search is performed. We propose *Text2Vis*, a neural network that generates a visual representation, in the visual feature space of the fc6-fc7 layers of ImageNet, from a short descriptive text. *Text2Vis* optimizes two loss functions, using a stochastic loss-selection method. A visual-focused loss is aimed at learning the actual text-to-visual feature mapping, while a text-focused loss is aimed at modeling the higher-level semantic concepts expressed in language and countering the overfit on non-relevant visual components of the visual loss. We report preliminary results on the MS-COCO dataset.

Keywords

image retrieval; cross-media retrieval; text representation

1. INTRODUCTION

Using a textual query to retrieve images is a very common cross-media search task, as text is the most efficient media to describe the kind of image the user is searching for. The actual retrieval process can be implemented in a number of ways, depending on how the shared search space between text and images is defined. The search space can be based

on textual features, visual features, or a joint space in which textual and visual features are projected into.

Using textual features is the most common solution, especially at the Web scale. Each image is associated with a set of textual features extracted from its context of use (e.g., the text surrounding the image in the Web page, description fields in metadata), and eventually enriched by means of classifiers that assign textual labels related to the presence or certain relevant entities or abstract properties in the image. The textual search space model can exploit the actual visual content of the image only when classifiers for the concepts of interest are available, thus requiring a relevant number of classifiers; this also requires to reprocess the entire image collection whenever a new classifier is made available.

On the other side, the visual and joint search spaces represent each image through visual features extracted from its actual content. The method we propose in this paper adopts a visual space search model. A textual query is converted into a visual representation in a visual space, where the search is performed by similarity. An advantage of this model is that any improvement in the text representation model, and its conversion to visual features, has immediate benefits on the image retrieval process, without requiring to reprocess the whole image collection.

A joint space model requires instead a reprocessing of all images whenever the textual model is updated, since the projection of images into the joint space is influenced also by the textual model part. It also requires managing and storing the additional joint space representations that are used only for the cross-media search.

In this paper we present the preliminary results on learning *Text2Vis*, a neural network model that converts textual descriptions into visual representations in the same space of those extracted from deep Convolutional Neural Networks (CNN) such as ImageNet [15]. *Text2Vis* achieves its goal by using a stochastic loss choice on two separate loss functions (as detailed in Section 3), one for textual representations autoencoding, and one for visual representations generation. Preliminary results show that the produced visual representations capture the high level concepts expressed in the textual description.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Neu-IR '16 SIGIR Workshop on Neural Information Retrieval July 21, 2016, Pisa, Italy

© 2016 Copyright held by the owner/author(s).

2. RELATED WORK

Deep Learning and Deep Convolutional Neural Networks (DCNNs) in particular, have recently shown impressive performance on a number of multimedia information retrieval tasks [15, 23, 10]. Deep Learning methods learn representations of data with multiple levels of abstraction. As a result, the activation of the hidden layers has been used in the context of transfer learning and content-based image retrieval [5, 22] as high-level representations of the visual content. Somewhat similarly, distributional semantic models, such as those produced by Word2Vec [18], or GloVe [21], have been found useful in modeling semantic similarities among words by establishing a correlation between *word meaning* and *position* in a vector space.

In order to perform cross-media retrieval, the two feature spaces (text and images in our case) should become comparable, typically by learning how to properly map the different sources. This problem has been attempted in different manners so far, which could be roughly grouped into three main variants, depending on whether the mapping is performed into a common space, the textual space, or the visual space.

Mapping into a common space: The idea of comparing texts and images in a shared space has been investigated by means of Cross-modal Factor Analysis and (Kernel) Canonical Correlation Analysis in [4]. In a similar vein, Corr-AE was proposed for cross-modal retrieval, allowing the search to be performed in both directions, i.e., from text-to-image and viceversa [8]. The idea is to train two autoencoders, one for the image domain and another for the textual domain, imposing restrictions between the two. As will be seen, the architecture we are presenting here bears resemblance to one of the architectures investigated in [8], the so-called *Correspondence full-modal autoencoder* (which is inspired by the multimodal deep learning method [19]). Contrarily to the multimodal architectures though, we apply a stochastic criterion to jointly optimize for the two modals, thus refraining from combining them into a parametric single loss.

Mapping into the textual space: The BoWDNN method trains a deep neural network (DNN) to map images directly into a bag-of-words (BoW) space, where the cosine similarity between BoWs representations is used to generate the ranking [1]. Somehow similarly, a dedicated area of related research is focused on generating captions describing the salient information of an image (see, e.g., [13, 7]).

Two other important examples along these lines are DeViSE [9] and ConSE [20]. Both methods build upon the higher layers of the convolutional neural network of [15]; the main difference lies on the way both methods treat the last layer of the net. Whereas DeViSE replaces this last layer with a linear mapping (thus fine-tuning the whole network) ConSE, on the other side, directly takes the outputs of the last layer and learns a projection to the textual embedding space.

Mapping into the visual space: Our proposal *Text2Vis* belongs to this group where, to the best of our knowledge, the only example up to now was a method dubbed *Word2VisualVec* [6], which was reported just very recently. There are some fundamental points where their method and ours differ, though. On the one hand, their *Word2VisualVec* takes combinations of Word2Vec-like vectors as a starting point, thus reducing the dimensionality of the input space; we directly start from

the bag-of-words vector encoding of the textual space, as we did not observe any improvement in pre-training the textual part. On the other, they build a deep network on top of the textual representation. As shall be seen, our *Text2Vis* is much shallower, as we found the net to be capable of mapping textual vectors into the visual space quite efficiently, provided that the model is properly regularized; an issue on which we focused our attention.

3. GENERATING VISUAL REPRESENTATIONS OF TEXT

In this section we describe the architecture of our *Text2Vis* network. Our idea is to map textual descriptions to high-level visual representations. As the visual space we used the *fc6* and *fc7* layers of the Hybrid network [24] (i.e., an AlexNet [15] trained on both ImageNet¹ and Places² datasets). We tested two vectorial representations for the textual descriptions: *Text2Vis*₁ uses simple bag-of-words vectors that mark with a value of one the positions that are relative to words that appear in the textual description and leave to zero all the others; *Text2Vis*_N adds a bit text structure info by considering also N-grams for a selection of part-of-speech patterns³. *Text2Vis*_N is a first approach at modeling text structure into the input vectorial representation, which differentiates the task of search from detailed/complex textual description we aim at from the traditional keyword search.

We have also investigated the use of pre-trained word embeddings, representing the textual description as the average of the embeddings of the words composing the description (see Equation 1 in [6]), but we have not observed any improvement. Generating the word embeddings is an additional cost, and the fitness of the embeddings for the task depends on the type of documents they are learned from. For example, an 11% improvement in MAP is reported in [2] from learning embedding from Flickr tags compared to learning them from Wikipedia pages. The direct use of bag-of-words vectors in *Text2Vis* removes the variable of selecting an appropriate document collection to learn the embedding and its learning cost.

As described in the following, *Text2Vis* actually learns a description embedding space that is able to reconstruct both the original description and the visual description. To reach this, we started with a simple regressor model (Figure 1, left) trained to directly predict the visual representation of the image associated with the textual input. We observed a strong tendency to overfit (Figure 1, right), thus degrading the applicability of the method to unseen images.

We explained this overfitting with the fact that a visual representation keeps track of every element that appears in the image, regardless of their semantic relevance within the image, while a (short) textual description is more likely focused on the visually relevant information, disregarding the secondary content of the image, as shown in Figure 4.6. As the learning iterations proceed, the simple regressor model starts capturing secondary elements of the images that are not relevant for the main represented concept, but are somewhat characteristic in the training data.

¹<http://image-net.org>

²<http://places.csail.mit.edu/index.html>

³We considered the part-of-speech patterns: ‘NOUN-VERB’, ‘NOUN-VERB-VERB’, ‘ADJ-NOUN’, ‘VERB-PRT’, ‘VERB-VERB’, ‘NUM-NOUN’, and ‘NOUN-NOUN’.

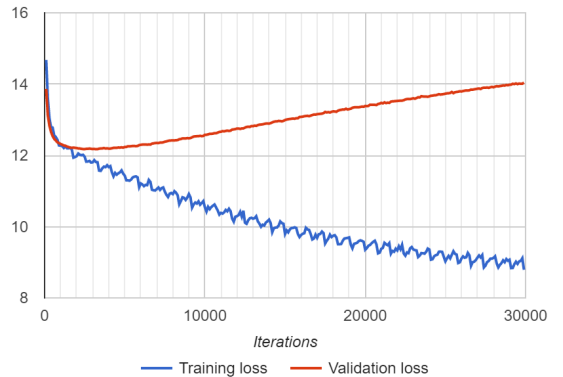
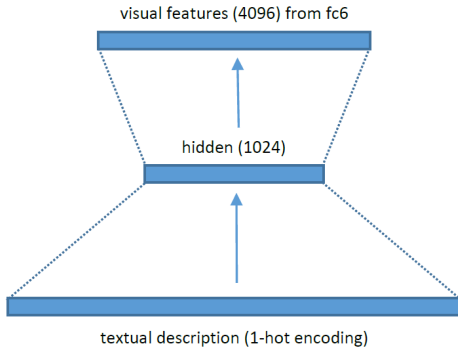


Figure 1: Overfitting of a simple regressor model with one hidden layer of size 1024.

Our *Text2Vis* proposal to contrast such overfitting is to add a text-to-text autoencoding branch to the hidden layer (Figure 2, left), forcing the model to satisfy two losses: one visual (text-to-visual regression) and one linguistic (text-to-text autoencoder). The linguistic loss works at higher level of abstraction than the visual one, acting as a regularization constraint on the model, and preventing, as confirmed by our experiments, overfitting on the visual loss (Figure 2, right). As detailed in the next section, we implemented the use of the two losses with a stochastic process, in which at each iteration one of the two is selected for optimization.

3.1 Text2Vis

Text2Vis consists of two overlapped feedforward neural nets with a shared hidden layer. The shared hidden layer causes a regularization effect during the combined optimization; i.e., the hidden state is constrained to be a good representation to accomplish with two different goals. The feedforward computation is described by the following equations:

$$z = ReLU(W_1 t_{in} + b_1) \quad (1)$$

$$t' = ReLU(W_2 z + b_2) \quad (2)$$

$$v' = ReLU(W_3 z + b_3) \quad (3)$$

where t_{in} represents the bag-of-words encoding for the textual descriptor given as input to the net, z is the hidden representation, v' and t' are the visual and textual predictions, respectively, obtained from the hidden representation z , $\Theta = \{W_i, b_i\}_{i \in \{1,2,3\}}$ are the model parameters to be learned, and $ReLU$ is the activation function, defined by $ReLU(x) = \max\{0, x\}$.

Both predictions v' and t' are then confronted with the expected outputs (i) the visual representation v corresponding to the *fc6* or *fc7* layers of [15], and (ii) a textual descriptor t_{out} that is semantically equivalent to t_{in} . We used the *mean squared error* (MSE) as the loss function in both cases:

$$\mathcal{L}(x, y; \Theta') = MSE(x, y) = \frac{1}{n} \sum_{i=1}^n (x_i - y_i)^2 \quad (4)$$

The model is thus multi-objective, and many alternative strategies could be followed at this point in order to set the Θ parameters so that both criteria are jointly minimized. We rather propose a much simpler, yet effective, way for carrying out the optimization search, that consists of considering both branches of the net as independent, and randomly de-

cidating in each iteration which of them is to be used for the gradient descent optimization.

Let thus define $\Theta_t = \{W_i, b_i\}_{i \in \{1,2\}}$ and $\Theta_v = \{W_i, b_i\}_{i \in \{1,3\}}$ as the model parameters of each independent branch. The optimization problem has two objectives (Equations 5 and 6), and at each iteration, a random choice decides which of them is to be optimized. We call this heuristic the *Stochastic Loss* (SL) optimization.

$$\Theta_{\hat{t}} = \underset{\Theta_t}{\operatorname{argmin}} \mathcal{L}_t(t_{out}, t'; \Theta_t) \quad (5)$$

$$\Theta_{\hat{v}} = \underset{\Theta_v}{\operatorname{argmin}} \mathcal{L}_v(v, v'; \Theta_v) \quad (6)$$

Note that the net is fed with a triple $\langle v, t_{in}, t_{out} \rangle$ at each iteration. When $t_{out} = t_{in}$ the text-to-text branch is an *autoencoder*. It is also possible to have $t_{in} \neq t_{out}$, with the two pieces of text been semantically equivalent (e.g., $t_{in} =$ “a woman cutting a pizza with a knife”, $t_{out} =$ “a woman holds a knife to cut pizza”) then the text-to-text branch might be reminiscent of the *Skip-gram*- and *CBOW*-like architectures. The text-to-image branch is, in any case, a regressor. The SL causes the model to be co-regularized. Notwithstanding, since our final goal is to project the textual descriptor into the visual space, the text-to-text branch might be thought as a regularization to the visual reconstruction (and, more specifically, to its internal encoding) which responds to constrains of linguistic nature.

4. EXPERIMENTS

4.1 Datasets

We used the *Microsoft COCO* dataset (MsCOCO⁴ [17]). MsCOCO was originally proposed for image recognition, segmentation, and caption generation. Although other datasets for image retrieval exist (e.g., the one proposed in [11]), they are more oriented to keyword-based queries. We believe MsCOCO to be more fit to the scenario we want to explore, since the captions associated to the images are expressed in natural language, thus semantically richer than a short list of keywords composing a query.

MsCOCO contains 82.783 training images (*Train2014*), 40.504 validation images (*Val2014*), and about 40K and 80K test images corresponding to two different competitions [3] (*Test2014* and *Test2015*). Because MsCOCO was proposed for caption generation, the captions are only accessible in

⁴Publicly available at <http://mscoco.org/>

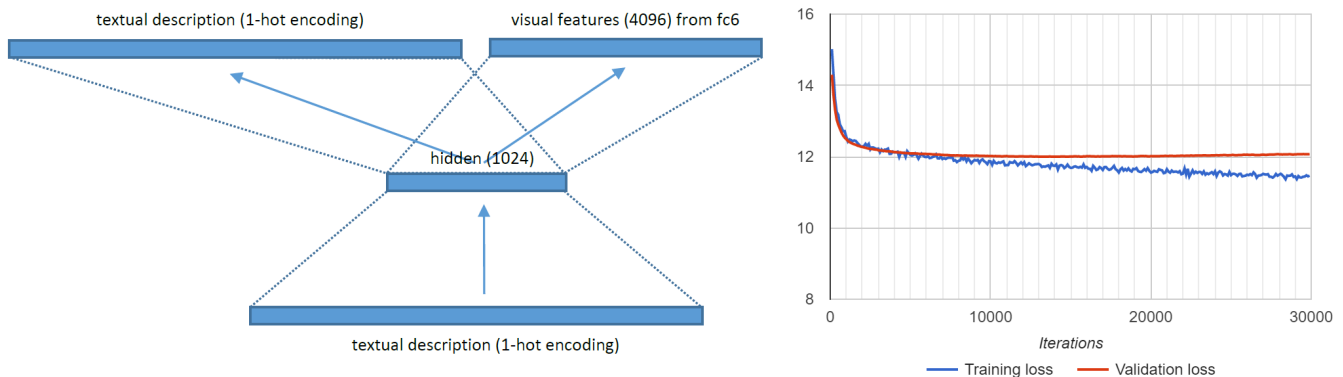


Figure 2: Our proposed *Text2Vis* which controls overfitting by adding an autoencoding constraint on the hidden state.

the *Train2014* and *Val2014* sets, while they are not yet released for *Test2014* and *Test2015*. We have thus taken the *Train2014* set for training, and split the *Val2014* into two disjoint sets of 20K images each for validation and test.

Each image in MsCOCO has 5 different captions associated. Let $\langle I, C \rangle$ be any labeled instance in MsCOCO, where I is an image and $C = \{c_1..c_5\}$ is a set of captions describing the content of I . Given a $\langle I, C \rangle$ pair, we define a labeled instance in our model as $\langle v, t_{in}, t_{out} \rangle$, where $v \in \mathbb{R}^{4096}$ is the visual representation of the image I taken from the *fc6* layer (or *fc7*, in separate experiments) of the Hybrid network [24]; t_{in} and t_{out} are two textual descriptors from C representing the input and output descriptors for the model, respectively. During training, t_{in} and t_{out} are uniformly chosen at random from C (thus t_{in} and t_{out} are not imposed to be different). Note that the number of training instances one could extract from a given $\langle I, C \rangle$ amounts to 25, which increases the variability of the training set along the different epochs.

4.2 Training

We solve the optimization problems of Equations 5 and 6, using the *Adam* method [14] for stochastic optimization, with default parameters (learning rate $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e^{-0.8}$). Note that there are two independent instances of the Adam optimizer, one associated to \mathcal{L}_t (Equation 5) and other for \mathcal{L}_v (Equation 6). In this preliminary study we decided to set for the SL an equal selection probability to both \mathcal{L}_t and \mathcal{L}_v ; different distributions will be investigated in future research.

We set the size of the training batch to 100 examples. We set the maximum number of iterations to 300,000, but apply an early stop when the model starts overfitting (as reflected in the validation error). The training set is shuffled each time a complete pass over all images is completed.

All the Θ parameters have been initialized at random according to a truncated normal distribution centered in zero with standard deviation of $\frac{1}{\sqrt{n}}$, where n is the number of columns. The biases have all been initialized to 0.

The vocabulary size is 10,358 for *Text2Vis*₁ after removing terms appearing in less than 5 captions. For *Text2Vis*_N we considered the 23,968 uni-grams and N-grams appearing at least in 10 captions. Since the number of units in the hidden and output layers are 1024 and 4096, respectively, the total number of parameters of the models amount to 25.4M in *Text2Vis*₁ and 53.3M in *Text2Vis*_N.

A Tensorflow implementation of *Text2Vis* is available at <https://github.com/AlexMoreo/tensorflow-Text2Vis>.

4.3 Evaluation Measures

Image retrieval is performed by similarity search in the visual space, using Euclidean distance on the l2-normalized visual vectors to generate a ranking of images, sorted by closeness. We measure the retrieval effectiveness of the visual representations produced from textual descriptions by our *Text2Vis* network by means of the *Discounted Cumulative Gain* (DCG [12]), defined as:

$$DCG_p = \sum_{i=1}^p \frac{2^{rel_i} - 1}{\log_2(i + 1)} \quad (7)$$

where rel_i quantifies the *relevance* of the retrieved element at rank position i with respect to the query, and p is the rank at which the metric is computed; we set $p = 25$ in our experiments, as was done in related research [11, 6].

Because the *rel* values are not provided in the MsCOCO, we estimate them by using the *ROUGE_L* [16] metric. *ROUGE_L* is one of the evaluation measures for the MsCOCO caption generation competition⁵ [3]. We compute $rel_i = ROUGE_L(t_{in}, C_i)$, where t_{in} is the query caption, and C_i are the 5 captions associated to the retrieved image at rank i . This caption-to-caption relevance model is thus aimed at measuring how much the concepts expressed in the query appear as relevant parts of the retrieved images.

4.4 Results

We compared the performance of *Text2Vis*₁ and *Text2Vis*_N models against: *RRank*, a lower bound baseline that produces a random ranking of images, for any query; *VisSim*, a direct similarity method that computes the Euclidean distances using the original *fc6*, or *fc7*, features for the image that is associated to query caption in MsCOCO; and *VisReg*, the text-to-image regressor described in Figure 1.

Table 4.4 reports the averaged DCG scores obtained by the compared methods. These results show a significant improvement of our proposal with respect to the compared methods. When using *fc6* as the visual space, *Text2Vis*₁ obtains a 8.51% relative improvement with respect to *VisSim* and 1.40% over *VisReg*. The improvements of *Text2Vis*_N are respectively of 8.08% and 0.94%. When using *fc7* as the

⁵<https://github.com/tylin/coco-caption>

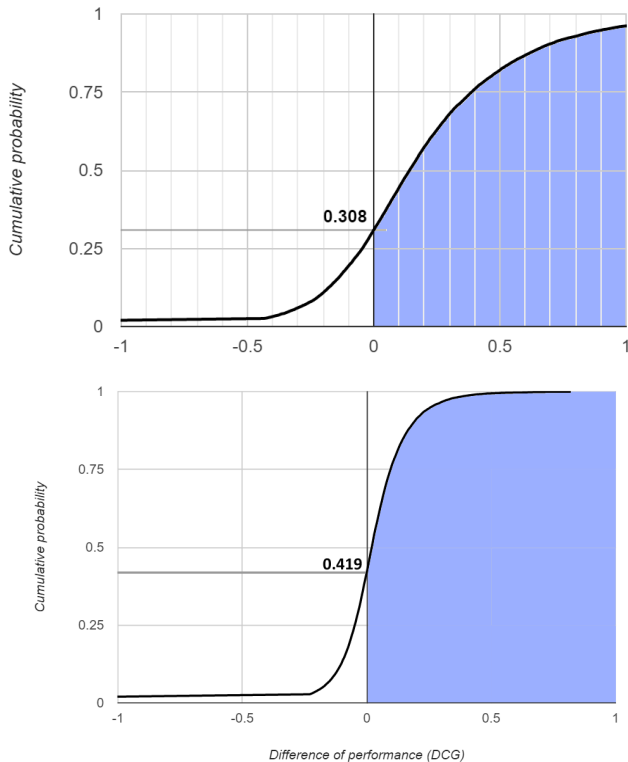


Figure 3: Cumulative probability distribution of the difference in performance of our $Text2Vis_1$ with respect to $VisSim$ (upper plot) and $VisReg$ (lower plot), on $fc6$. Positive differences mean $Text2Vis$ obtained a better ranking score than $VisSim$ or $VisReg$ (resp. 69.2% and 58.1% of cases, shadowed region).

visual space it is $Text2Vis_N$ that obtains, yet by a small margin, the best result. The relative improvements of $Text2Vis_1$ over $emphVisSim$ and $VisReg$ are respectively of 8.48% and 0.97%, and for $Text2Vis_N$ respectively of 8.60% and 1.09%.

Method	fc6	fc7
$RRank$	1.524	1.524
$VisSim$	2.150	2.180
$VisReg$	2.317	2.359
$Text2Vis_1$	2.350	2.382
$Text2Vis_N$	2.339	2.385

Table 1: Performance comparison of the different methods in terms of averaged DCG

In addition to the averaged performance, we also investigated how often the ranking produced by $Text2Vis$ is more relevant (according to DCG) than those produced by $VisSim$ and $VisReg$. Figure 3 indicates that in 69.2% of the cases, the ranking of $Text2Vis_1$ was found more relevant than $VisSim$ (see Figure 3). The same happens in 58.1% of the cases when comparing $Text2Vis$ to $VisReg$.

4.5 Why Stochastic Loss?

$Text2Vis$ uses two independent optimizers to optimize the visual (\mathcal{L}_v) and the textual (\mathcal{L}_t) losses, based on a stochas-

tic choice at each iteration (SL, section 3.1). Previous approaches to multimodal learning relied instead on a unique aggregated loss (typically of the form $\mathcal{L} = \mathcal{L}_v + \lambda\mathcal{L}_t$) that is minimized by a single optimizer [8, 19]. We compared the two approaches on the case of equal relevance of the two losses ($\lambda = 1$, uniform distribution for SL). SL better optimizes the two losses (Figure 4), and is less prone to overfit.

We deem that SL allows to model in a more natural way the relative relevance of the various losses that are combined, i.e., by selecting the losses in proportion to the assigned relevance, whereas the numeric aggregation is affected by the relative values of losses and the differences in their variation during the optimization (e.g., a loss that has a large improvement may compensate for another loss getting worse). SL is also computationally lighter than the aggregated loss, as SL updates only a part of the model on each iteration.

4.6 Visual comparison

Figure 5 show a few samples⁶ that highlight the differences in results from the three compared methods. In all the cases results from the $VisSim$ method are dominated by the main visual features of the images: a face for the first query, the content of the screen for the second query, an outdoor image with a light lower part, plants, people and a bit of sky in the third one. The two text based methods obtains results that

⁶More results at <https://github.com/AlexMoreo/tensorflow-Text2Vis>

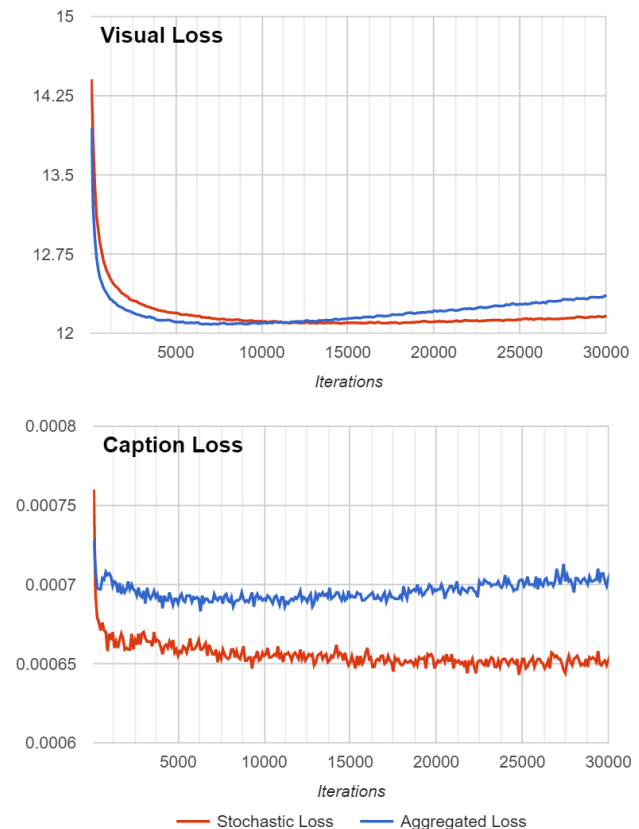


Figure 4: Validation loss for \mathcal{L}_v and \mathcal{L}_t , optimizing on a linear combination of losses (blue) or using two optimizers with stochastic loss selection (SL, red).

Query	Method	1st	2nd	3rd	4th	5th
	<i>VisSim</i>					
<i>a chubby toddler is chewing on a toothbrush</i>	<i>VisReg</i>					
	<i>Text2Vis</i>					
	<i>VisSim</i>					
<i>a large screen monitor on a desk hooked up to a laptop</i>	<i>VisReg</i>					
	<i>Text2Vis</i>					
	<i>VisSim</i>					
<i>a man gets ready to throw a frisbee</i>	<i>VisReg</i>					
	<i>Text2Vis</i>					

Figure 5: Examples of search results from the three compared methods.

more often contain the key elements of the description. For the first query, *Text2Vis* retrieves four relevant images out of five, one more than *VisReg*. For the other two queries the results are pretty similar, with *Text2Vis* placing in second position an image that is a perfect match for the query, while *VisReg* places it in fifth position.

5. CONCLUSIONS

The preliminary experiments indicate our method produces more relevant rankings than those produced by similarity search directly on the visual features of a query image. This is an indication that our text-to-image mapping produces better prototypical representations of the desired scene than the representation of a sample image itself. A simple explanation of this result is that textual descriptions

strictly emphasize the relevant aspects of the scene the user has in mind, whereas the visual features, directly extracted from the query image, are keeping track of all the information that is contained in that image, causing the similarity search to be potentially confused by secondary elements of the scene. The *Text2Vis* model also improved, yet by a smaller margin, over the *VisReg* model, showing that an auto-encoding branch in the network is useful to avoid overfitting on visual features. We also found that combining losses in a stochastic fashion, rather than numerically, improves both the effectiveness and efficiency of the system. In the future we plan to compare *Text2Vis* against the recently proposed *Word2VisualVec* [6] model. We also intend to improve the modeling word order information in *Text2Vis*, likely by adding a recurrent component to the network architecture.

6. REFERENCES

- [1] Y. Bai, W. Yu, T. Xiao, C. Xu, K. Yang, W.-Y. Ma, and T. Zhao. Bag-of-words based deep neural network for image retrieval. In *Proceedings of the ACM International Conference on Multimedia*, pages 229–232. ACM, 2014.
- [2] S. Cappallo, T. Mensink, and C. G. Snoek. Image2emoji: Zero-shot emoji prediction for visual media. In *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, pages 1311–1314, New York, NY, USA, 2015. ACM.
- [3] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.
- [4] J. Costa Pereira, E. Coviello, G. Doyle, N. Rasiwasia, G. R. Lanckriet, R. Levy, and N. Vasconcelos. On the role of correlation and abstraction in cross-modal multimedia retrieval. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(3):521–535, 2014.
- [5] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.
- [6] J. Dong, X. Li, and C. G. M. Snoek. Word2VisualVec: Cross-Media Retrieval by Visual Feature Prediction. *ArXiv e-prints*, Apr. 2016.
- [7] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, et al. From captions to visual concepts and back. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1473–1482, 2015.
- [8] F. Feng, X. Wang, and R. Li. Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the ACM International Conference on Multimedia*, pages 7–16. ACM, 2014.
- [9] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems*, pages 2121–2129, 2013.
- [10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.
- [11] X.-S. Hua, L. Yang, J. Wang, J. Wang, M. Ye, K. Wang, Y. Rui, and J. Li. Clickage: Towards bridging semantic and intent gaps via mining click logs of search engines. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 243–252. ACM, 2013.
- [12] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.
- [13] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.
- [14] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [16] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In S. S. Marie-Francine Moens, editor, *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.
- [17] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014*, pages 740–755. Springer, 2014.
- [18] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
- [19] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 689–696, 2011.
- [20] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.
- [21] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543, 2014.
- [22] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 806–813, 2014.
- [23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [24] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014.