RATT: Recurrent Attention to Transient Tasks for Continual Image Captioning

Riccardo Del Chiaro [1] **Bartłomiej Twardowski** [2] **Andrew D. Bagdanov** [1] **Joost van de Weijer** [2]

Abstract

Research on continual learning has led to a variety of approaches to mitigating catastrophic forgetting in feed-forward classification networks. Until now surprisingly little attention has been focused on continual learning of recurrent models applied to problems like image captioning. In this paper we take a systematic look at continual learning of LSTM-based models for image captioning. We propose an attention-based approach that explicitly accommodates the *transient* nature of vocabularies in continual image captioning tasks – i.e. that task vocabularies are not disjoint. We call our method Recurrent Attention to Transient Tasks (RATT), and also show how to adapt continual learning approaches based on weight regularization and knowledge distillation to recurrent continual learning problems. We apply our approaches to incremental image captioning problem on two new continual learning benchmarks we define using the MS-COCO and Flickr30 datasets. Our results demonstrate that RATT is able to sequentially learn five captioning tasks while incurring *no* forgetting of previously learned ones.

1. Introduction

Classical supervised learning systems acquire knowledge by providing them with a set of annotated training samples from a task, which for classifiers is a single set of classes to learn. This view of supervised learning stands in stark contrast with how humans acquire knowledge, which is instead *continual* in the sense that mastering new tasks builds upon previous knowledge acquired when learning previous ones. This type of learning is referred to as *continual* learning (sometimes *incremental* or *lifelong* learning), and continual learning systems instead consume a sequence of tasks, each containing its own set of classes to be learned. Through a sequence of *learning sessions*, in which the learner has access only to labeled examples from the current task, the learning system should integrate knowledge from past and current tasks in order to accurately master them all in the end. A principal shortcoming of state-of-the-art learning systems in the continual learning regime is the phenomenon of *catastrophic forgetting* (Goodfellow et al., 2013; Kirkpatrick et al., 2017): in the absence of training samples from previous tasks, the learner is likely to *forget* them in the process of acquiring new ones.

Continual learning research has until now concentrated primarily on classification problems modeled with deep, feed-forward neural networks (De Lange et al., 2019; Parisi et al., 2019). Given the importance of recurrent networks for many learning problems, it is surprising that continual learning of recurrent networks has received so little attention (Coop & Arel, 2013; Sodhani et al., 2019). A recent study on catastrophic forgetting in deep LSTM networks (Schak & Gepperth, 2019) observes that forgetting is more pronounced than in feed-forward networks. This is caused by the recurrent connections which amplify each small change in the weights. In this paper, we consider continual learning for captioning, where a recurrent network (LSTM) is used to produce the output sentence describing and image. Rather than having access to all captions jointly during training, we consider different captioning tasks which are learned in a sequential manner (examples of tasks could be captioning of sports, weddings, news, etc).

Most continual learning settings consider tasks that each contain a set of classes, and these sets are disjoint (Pfülb & Gepperth, 2019; Rebuffi et al., 2017; Serra et al., 2018). A key aspect of continual learning for image captioning is the fact that tasks are naturally split into overlapping vocabularies. Task vocabularies might contain nouns and some verbs which are specific to a task, however many of the words (adjectives, adverbs, and articles) are *shared* among tasks. Moreover, the presence of homonyms in different tasks might directly lead to forgetting of previously acquired concepts. This *transient* nature of words in task vocabularies makes continual learning in image captioning networks different from traditional continual learning.

[1] Media Integration and Communication Center, University of Florence, 50134 Florence, FI, Italy. [2] Computer Vision Center, Department of Informatics, Universitat Autónoma de Barcelona, 08193 Barcelona, Spain. Correspondence to: Riccardo Del Chiaro <riccardo.delchiaro@unifi.it>.

Proceedings of the 37th International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020. Copyright 2020 by the author(s).

In this paper we take a systematic look at continual learning for image captioning problems using recurrent, LSTM networks. We consider three of the principal classes of approaches to exemplar-free continual learning: weight-regularization approaches, exemplified by Elastic Weight Consolidation (EWC) (Kirkpatrick et al., 2017); knowledge distillation approaches, exemplified by Learning without Forgetting (LwF) (Li & Hoiem, 2017); and attention-based approached like Hard Attention to the Task (HAT) (Serra et al., 2018). For each we propose modifications specific to their application to recurrent LSTM networks, in general, and more specifically to image captioning in the presence of transient task vocabularies.

The contributions of this work are threefold: (1) we propose a new framework and splitting methodologies for modeling continual learning of sequential generation problems like image captioning; (2) we propose an approach to continual learning in recurrent networks based on transient attention masks that reflect the transient nature of the vocabularies underlying continual image captioning; and (3) we support our conclusions with extensive experimental evaluation on our new continual image captioning benchmarks and compare our proposed approach to continual learning baselines based on weight regularization and knowledge distillation. To the best of our knowledge we are the first to consider continual learning of sequential models in the presence of *transient tasks vocabularies* whose classes may appear in some learning sessions, then disappear, only to reappear in later ones.

2. Continual LSTMs for transient tasks

We first describe our image captioning model and some details LSTM networks. Then we describe how to apply classical continual learning approaches to LSTM networks.

2.1. Image captioning Model

We use a captioning model similar to Neural Image Captioning (NIC) (Vinyals et al., 2015). It is an encoder-decoder network that "translates" an image into a natural language description. It is trained end-to-end, directly maximizing the probability of correct sequential generation:

$$\hat{\theta} = \arg\max_{\theta} \sum_{(I,s)} \log p(s_N | I, s_1, \ldots, s_{N-1}; \theta),$$

where $s = [s_1, \ldots s_N]$ is the target sentence for image I, θ are the model parameters.

The decoder is an LSTM network in which words $s_1, \ldots, s_{n-1}$ are encoded in the hidden state h_n and a linear classifier is used to predict the next word time step n:

$$x_0 = V\text{CNN}(I) \quad (1) \qquad h_n = \text{LSTM}(x_n, h_{n-1}) \quad (3)$$

$$x_n = S s_n \quad (2) \qquad p_{n+1} = C h_n \quad (4)$$

where S is a word embedding matrix, s_n is the n-th word of the ground-truth sentence for image I, C is a linear classifier, and V is the visual projection matrix that projects image features from the CNN encoder into the embedding space at time $n = 0$.

The LSTM network is defined by the following equations (for which we omit the bias terms):

$$i_n = \sigma(W_{ix} x_n + W_{ih} h_{n-1}) \quad (5)$$

$$o_n = \sigma(W_{ox} x_n + W_{oh} h_{n-1}) \quad (6)$$

$$f_n = \sigma(W_{fx} x_n + W_{fh} h_{n-1}) \quad (7)$$

$$g_n = \tanh(W_{gx} x_n + W_{gh} h_{n-1}) \quad (8)$$

$$h_n = o_n \odot c_n \quad (9)$$

$$c_n = f_n \odot c_{n-1} + i_n \odot g_n \quad (10)$$

where $\odot$ is the Hadamard (element-wise) product, σ the logistic function, c the LSTM cell hidden state. The W matrices are the trainable LSTM parameters related to input W_i, the hidden state W_h, and for each gate i, f, o, g. The loss used to train the network is the sum of the negative log likelihood of the correct word at each step:

$$\mathcal{L}(x, s) = -\sum_{n=1}^{N} \log p_n(s_n), \quad (11)$$

Inference. During training we perform teacher forcing using n-th word of the target sentence as input to predict word $n+1$. At inference time, since we have no target caption, we use the word predicted by the model at the previous step $\arg\max\ p_n$ as input to the word embedding matrix S.

2.2. Continual learning of recurrent models

Normally catastrophic forgetting is highlighted in continual learning benchmarks by defining tasks that are mutually disjoint in the classes they contain (i.e. no class belongs to more than one task). For sequential problems like image captioning, however, this is not so easy: sequential learners must classify *words* at each decoding step, and a large vocabulary of *common* words are needed for any practical captioning task.

Incremental model. Our models are trained on sequences of captioning tasks, each having different vocabularies. For this reason any captioning model must be able to enlarge its vocabulary. When a new task arrives we add a new column for each new word in the classifier and word embedding matrices. The recurrent network remains untouched because the embedding projects inputs into the same space. The basic approach to adapt to the new task is to fine-tune the network over the new training set. To manage the different classes (words) of each task we have two possibilities: (1) Use different classifier and word embedding matrices for each task; or (2) Use a common, growing classifier and a common, growing word embedding matrix.

The first option has the advantage that each task can benefit from ad hoc weights for the task, potentially initializing from the previous task for the common words. However, it also increases decoder network size considerably with each new task. The second option has the opposite advantage of having a controlled growth of the decoder, sharing weights for all common words. Because of the nature of the captioning problem, many words will be shared and duplicating both word embedding matrix and classifier for all the common words seems wasteful. Thus we adopt the second alternative. With this approach, the key trick is to *deactivate* classifier weights for words not present in the current task vocabulary.

We use $\hat{\theta}^t$ to denote optimal weights learned for task t on dataset D_t. After training on task t, we create a new model for task $t+1$ with expanded weights for classifier and word embedding matrices. We use weights from $\hat{\theta}^t$ to initialize the shared weights of the new model.

2.3. Recurrent continual learning baselines

We describe how to adapt two common continual learning approaches, one based on weight regularization and the other on knowledge distillation. We will use these as baselines in our comparison.

Weight regularization. A common method to prevent catastrophic forgetting is to apply regularization to important model weights before proceeding to learn a new task (Aljundi et al., 2018; Chaudhry et al., 2018; Kirkpatrick et al., 2017; Zenke et al., 2017). Such methods can be directly applied to recurrent models with little effort. We choose Elastic Weight Consolidation (EWC) (Serra et al., 2018) as a regularization-based baseline. The key idea of EWC is to limit change to model parameters vital to previously-learned tasks by applying a quadratic penalty to them depending on their importance. Parameter importance is estimated using a diagonal approximation of the Fisher Information Matrix. The additional loss function we minimize when learning task t is:

$$\mathcal{L}^t_{\mathrm{EWC}}(x,S;\theta^t) = \mathcal{L}(x,S) + \lambda \sum_i \frac{1}{2}\, F_i^{t-1}(\theta_i^t - \hat{\theta}_i^{t-1})^2, \tag{12}$$

where $\hat{\theta}^{t-1}$ are the estimated model parameters for the previous task, θ^t are the model parameters at the current task t, $\mathcal{L}(x,S)$ is the standard loss used for fine-tuning the network on task t, i indexes the model parameters shared between tasks t and $t-1$, F_i^{t-1} is the i-th element of a diagonal approximation of the Fisher Information Matrix for model after training on task $t-1$, and λ weights the importance of the previous task. We apply Eq. 12 to all trainable weights. Due to the transient nature of words across tasks, we do not expect weight regularization to be optimal since some words are shared and regularization limits the plasticity needed to adjust to a new task.

Recurrent Learning without Forgetting. We also apply a knowledge distillation (Hinton et al., 2015) approach inspired by Learning without Forgetting (LwF) (Li & Hoiem, 2017) on the LSTM decoder network to prevent catastrophic forgetting. The model after training task $t-1$ is used as a teacher network when fine-tuning on task t. The aim is to let the new network freely learn how to classify new words appearing in task t while keeping stable the predicted probabilities for words from previous tasks.

To do this, at each step n of the decoder network the previous decoder is also fed with the data coming from the new task t. Note that the input to the LSTM at each step n is the embedding of the $n-$th word in the target caption, and the same embedding is given as input to both teacher and student networks – i.e. the student network's embedding of word n is also used as input for the teacher, while each network uses its own hidden state h_{n-1} and cell state c_{n-1} to decode the next word. At each decoding step we define the output probabilities from the student LSTM network corresponding only to words encountered up to the previous task as $\tilde{p}^t_{n+1}$. These are compared with the output probabilities p^{t-1}_{n+1} predicted by the teacher network. A distillation loss ensures that the student network does not deviate from the teacher:

$$\mathcal{L}^t_{\mathrm{LwF}}(\tilde{p}^t, p^{t-1}) = -\sum_n H(\gamma(\tilde{p}^t_n), \gamma(p^{t-1}_n))$$

where $\gamma(\cdot)$ rescales a probability vector p -with temperature parameter T. This loss is combined with the LSTM training loss (see Eq. 11). Note that differently from (Li & Hoiem, 2017), we do not fine-tune the classifier of the old network because we use a single, incremental word classifier.

3. Attention for continual learning of transient tasks

Inspired by the Hard Attention to the Task (HAT) method (Serra et al., 2018), we developed an attention-based technique applicable to recurrent networks. We name it *Recurrent Attention to Transient Tasks (RATT)*, since it is specifically designed for recurrent networks with task transience. The key idea is to use an attention mechanism to allocate a portion of the activations of each layer to a *specific* task t. An overview of RATT is provided in Fig. 1.

Attention masks. The number of neurons used for a task

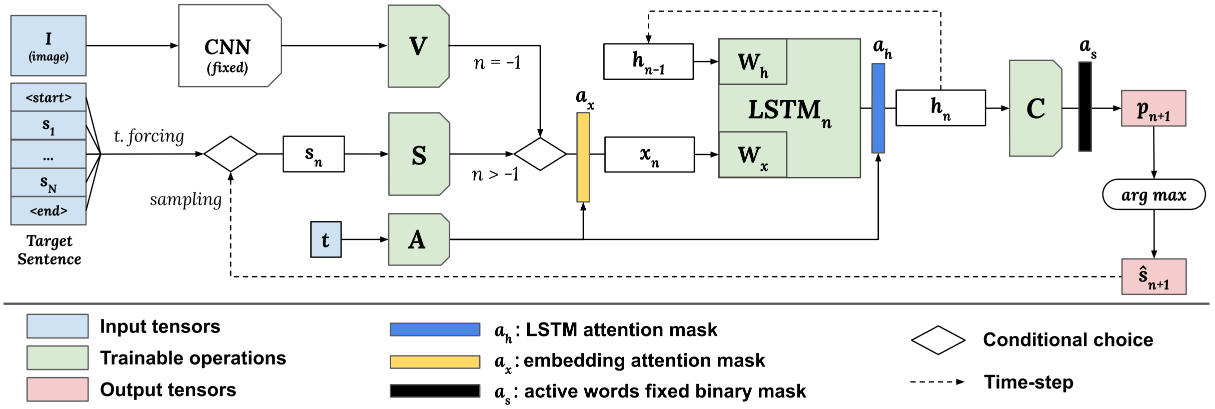


Figure 1. Recurrent Attention to Transient Tasks (RATT). See section 3 for a detailed description of each component of the network.

is limited by two task-conditioned attention masks: embedding attention $a_x^t \in [0,1]^{|X|}$ and hidden state attention $a_h^t \in [0,1]^{|H|}$, where $|X|$ and $|H|$ are the dimension of the embedding and hidden space respectively. These masks are computed with a sigmoid activation σ and a positive scaling factor s according to:

$$a_x^t = \sigma(sA_x t^T) \ , \ a_h^t = \sigma(sA_h t^T), \tag{13}$$

where t is a one-hot task vector, and A_x and A_h are embedding matrices. Next to the two attention mask, we have a vocabulary mask a_s^t which is a binary mask identifying the words of the vocabulary used in task t: $a_{s,i}^t = 1$ if word i is part of the vocabulary of task t and is zero otherwise. The forward pass (see Eqs.1 and 4) of the network is modulated with the attention masks according to:

$$\begin{aligned} \bar{x}_0 &= x_0 \odot a_x^t \\ \bar{x}_n &= x_n \odot a_x^t \\ \bar{h}_n &= h_n \odot a_h^t \\ \bar{p}_{n+1} &= p_{n+1} \odot a_s^t \end{aligned}$$

Attention masks act as an inhibitor when their value is near 0. The main idea is to learn attention masks during training, and as such learn a limited set of neurons for each task. Neurons used in previous tasks can still be used in subsequent ones, however the weights which were important for previous tasks have reduced plasticity (depending on the amount of attention to for previous tasks).

Training. For training we define the cumulative mask, that picks the highest values from the masks of previous tasks:

$$a_x^{<t} = \max(a_x^{t-1}, a_x^{<t-1}), \tag{14}$$

$a_h^{<t}$ and $a_s^{<t}$ are similarly defined. We now define the following backward masks which have the dimensionality of the weight matrices of the network and are used to selectively backpropagate the gradient to the layers:

$$B_{h,ij}^t = 1 - \min(a_{h,i}^{<t}, a_{h,j}^{<t}) \tag{15}$$

$$B_{x,ij}^t = 1 - \min(a_{h,i}^{<t}, a_{x,j}^{<t}) \tag{16}$$

Note that we use $a_{h,i}$ refer to the i-th element of vector a_h, etc. The backpropagation with learning rate λ is then done according to

$$W_h \leftarrow W_h - \lambda B_h^t \odot \frac{\partial \mathcal{L}^t}{\partial W_h} \tag{17}$$

$$W_x \leftarrow W_x - \lambda B_x^t \odot \frac{\partial \mathcal{L}^t}{\partial W_x}. \tag{18}$$

The only difference from standard backpropagation are the backward matrices B which prevents the gradient from changing those weights that were highly attended in previous tasks. The backpropagation updates to the other matrices in Eqs. 5-8 are similar.

We define backward masks for the word embedding S, the linear classifier C, and the image-projection matrix V:

$$B_{S,ij}^t = 1 - \min(a_{x,i}^{<t}, a_{s,j}^{<t}) \tag{19}$$

$$B_{C,ij}^t = 1 - \min(a_{s,i}^{<t}, a_{h,j}^{<t}) \tag{20}$$

$$B_{V,ij}^t = 1 - a_{x,i}^{<t}. \tag{21}$$

The corresponding backpropagation updates are:

$$S \leftarrow S - \lambda B_S^t \odot \frac{\partial \mathcal{L}^t}{\partial S} \tag{22}$$

$$C \leftarrow C - \lambda B_C^t \odot \frac{\partial \mathcal{L}^t}{\partial C} \tag{23}$$

$$V \leftarrow V - \lambda B_V^t \odot \frac{\partial \mathcal{L}^t}{\partial V}. \tag{24}$$

The backward mask B_V^t modulates the backpropagation to the image features. Since we do not define a mask on the output of the fixed image encoder, this is only defined by $a_x^{<t}$. Differently than (Serra et al., 2018), when computing B_S^t we take into account the recurrency of the network, considering the classifier C to be the previous layer of S.

Linearly annealing the scaling parameter s, used in Eq. 13, during training (like (Serra et al., 2018)) was found to be beneficial. We apply $s = \frac{1}{s_{max}} + \left(s_{max} - \frac{1}{s_{max}}\right)\frac{b-1}{B-1}$

where b is the batch index and B is the total number of batches for the epoch.

The additional loss used to promote low network usage and to keep some neurons available for future tasks is:

$$\mathcal{L}_a^t = \frac{\sum_i a^t_{x,i}(1-a^{<t}_{x,i})}{\sum_i(1-a^{<t}_{x,i})} + \frac{\sum_i a^t_{h,i}(1-a^{<t}_{h,i})}{\sum_i(1-a^{<t}_{h,i})}. \qquad (25)$$

This loss is combined with Eq. 11 for training. The loss encourages attention to only a few new neurons. However, tasks can attend to previously attended neurons without any penalty. This encourages forward transfer during training. If the attention masks are binary, the system would not suffer from any forgetting, however it would lose its backward transfer ability.

4. Related work

Catastrophic forgetting. Early works demonstrating the inability of networks to retain knowledge from previously task when learning new ones are (McCloskey & Cohen, 1989) and (Goodfellow et al., 2013). Approaches include methods that mitigate catastrophic forgetting via replay of examplars (iCarl (Rebuffi et al., 2017), EEIL (Castro et al., 2018), and GEM (Lopez-Paz & Ranzato, 2017)) or by performing pseudo-replay with GAN-generated data (Shin et al., 2017; Wu et al., 2018). Weight regularization has also been investigated (Aljundi et al., 2018; Kirkpatrick et al., 2017; Zenke et al., 2017; Liu et al., 2018). Output regularization via knowledge distillation was investigated in LwF (Li & Hoiem, 2017), as well as architectures based on network growing (Rusu et al., 2016; Schwarz et al., 2018) and feature masking (Serra et al., 2018; Masana et al., 2020) or PiggyBack (Mallya et al., 2018). For more details we refer to recent surveys on continual learning (Parisi et al., 2019; De Lange et al., 2019). Recent work investigated estimating the drift in neurons, and proposes a way to compensate for it (Yu et al., 2020).

In this work we only focus on methods that do neither exemplar replay nor model expansion. We compare our approach described in section 3 with baselines derived from LwF (Li & Hoiem, 2017) and EWC (Kirkpatrick et al., 2017) and adapted these for use on recurrent image captioning models.

Image captioning. Captioning models are typically based on a CNN encoder, an RNN caption generator, and additional modules like attention or auxiliary classification heads for prediction (Hossain et al., 2019). We consider a model based on (Vinyals et al., 2015) as a starting point to investigating forgetting in recurrent image captioning networks. The model is not over-complicated or extended with techniques that might make forgetting analysis harder or inconclusive.

Task	**Train**	**Valid**	**Test**	**Vocab (words)**
transport	14,266	3,431	3,431	3,116
animals	9,314	2,273	2,273	2,178
sports	10,077	2,384	2,384	1,967
food	7,814	1,890	1,890	2,235
interior	17,541	4,340	4,340	3,741
total	59,012	14,318	14,318	6,344

(a) MS-COCO task split statistics.

Task	**Train**	**Valid**	**Test**	**Vocab (words)**
scene	7,500	250	250	3,242
animals	3,312	107	113	1,631
vehicles	4,084	123	149	2,169
instruments	1,290	42	42	848
total	16,186	522	554	4,057

(b) Flickr30k task split statistics.

Table 1. Number of images and words per task for our MS-COCO and Flickr30K splits.

Continual learning of recurrent networks. A fixed expansion layer technique was proposed to mitigate forgetting in RNNs in (Coop & Arel, 2013). A dedicated network layer that exploits sparse coding of RNN hidden state is used to reduce the overlap of pattern representations. In this method the network grows with each new task. A Net2Net technique was used for expanding the RNN in (Sodhani et al., 2019). The method uses GEM(Lopez-Paz & Ranzato, 2017) for training on a new task, but has several shortcomings: model weights continue to grow and it must retain previous task data in the memory.

Experiments on four synthetic datasets were conducted in (Schak & Gepperth, 2019) to investigate forgetting in LSTM networks. The authors concluded that the LSTM topology has no influence on forgetting. This observations motivated us to take a close look to continual image captioning where the network architecture is more complex and an LSTM is used as a output decoder.

5. Experimental results

All experiments use the same architecture: for the encoder network we used ResNet151 (He et al., 2016) pre-trained on ImageNet (Russakovsky et al., 2015). Note that the image encoder is frozen and is not trained during continual learning, as is common in many image captioning systems.The decoder consists of the word embedding matrix S that projects the input words into a 256-dimensional space, an LSTM cell with hidden size 512 that takes the word (or image feature for the first step) embeddings as input, and a final fully connected layer C that take as input the hidden state h_n at each LSTM step n and outputs a probability distribution p_{n+1} over the $|V^t|$ words in the vocabulary for current task t. The model has ~$59.69M$ parameters and needs extra 768 parameter per each word in the vocabulary. The only extra parameters in RATT are given by the task embedding matrices A_x and A_h, that are $256 \times T$ and $512 \times T$ respectively, where T is the total number of tasks on which is trained.

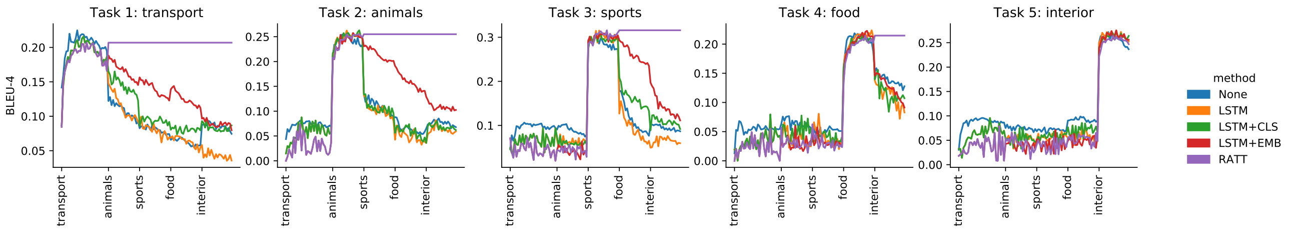


Figure 2. BLEU-4 performance at each epoch over the whole sequence of MS-COCO tasks.

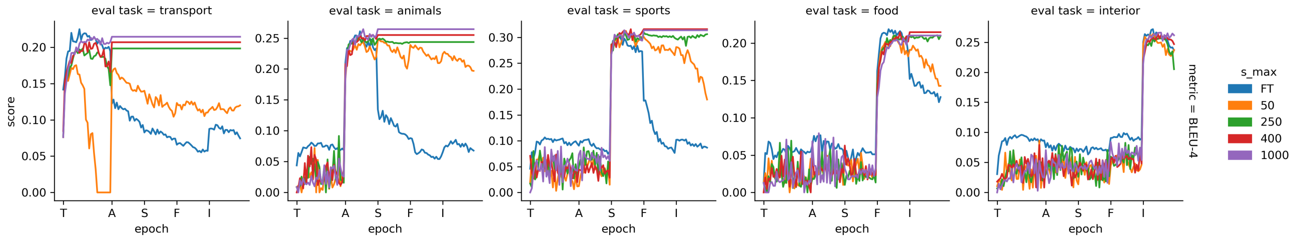


Figure 3. RATT comparison with different s_{max} values and finetuning baseline. BLEU-4 validation performance for each training epoch on MS-COCO dataset is reported.

We applied all techniques on the Flickr30K (Plummer et al., 2017) and MS-COCO (Lin et al., 2014) captioning datasets (see next section for task splits). All experiments were conducted using PyTorch, networks were trained using the Adam (Kingma & Ba, 2014) optimizer, all hyperparameters were tuned over validation sets. [1] Batch size, learning rate and max-decode length for evaluation were set, respectively, to 128, 4e-4, and 26 for MS-COCO, and 32, 1e-4 and 40 for Flickr30k. These differences are due to the size of the training set and by the average caption lengths in the two datasets.

Inference at test time is *task-aware* for all methods. For EWC and LwF this means that we consider only the word classifier outputs corresponding to the correct task, and for RATT that we use the fixed output masks for the correct task. All metrics where computed using the nlg-eval toolkit (Sharma et al., 2017). Models where trained for a fixed number of epochs and the best model according to BLEU-4 performance on the validation set were chosen for each task. When proceeding to the next task, the best model from the previous task were used as a starting point.

5.1. Datasets and task splits

For our experiments we use two different captioning datasets: MS-COCO (Plummer et al., 2017) and Flickr30k (Lin et al., 2014). We split MS-COCO into tasks using a *disjoint visual categories* procedure. For this we defined five tasks based on disjoint MS-COCO supercategories containing related classes. For Flickr30K we instead used an *incremental visual categories* procedure. Using the visual entities, phrase types, and splits from (Plummer et al., 2017) we identified four tasks. In this approach the first task contains a set of visual concepts that can also be appear in future tasks.

Some statistics on number of images and vocabulary size for each task are given in table 1 for both datasets. MS-COCO does not provide a test set, so we randomly selected half of the validation set images and used them for testing only. Since images have at least five captions, we used the first five captions for each image as the target.

5.2. Ablation study

We conducted a preliminary study on our split of MS-COCO to evaluate the impact of our proposed Recurrent Attention to Transient Tasks (RATT) approach. In this experiment we progressively introduce the attention masks described in section 3. We start with the basic captioning model with no forgetting mitigation, and so is equivalent to *fine-tuning*. Then we introduce the mask on hidden state h_n of the LSTM (along with the corresponding backward mask), and then the constant binary mask on the classifier that depends on the words of the current task, then the visual and word embedding masks, and finally the combination of all masks.

In figure 2 we plot the BLEU-4 performance of these configurations for each training epoch and each of the five MS-COCO tasks. Note that for later tasks the performance on early epochs (i.e. *before* encountering the task) is noisy as expected – we are evaluating performance on *future* tasks. These results clearly show that applying the mask to LSTM decreases forgetting in the early epochs when learning a new task. However, performance continues to decrease and in some tasks the result is similar to fine-tuning. Even if the LSTM is forced to remember how to manage hidden states

[1] Code for all models: https://github.com/delchiaro/RATT

	Transport				Animals				Sports				Food				Interior			
	FT	**EWC**	**LwF**	**RATT**	**FT**	**EWC**	**LwF**	**RATT**	**FT**	**EWC**	**LwF**	**RATT**	**FT**	**EWC**	**LwF**	**RATT**	**FT**	**EWC**	**LwF**	**RATT**
BLEU-4	.0928	.1559	.1277	**.2126**	.0816	.1545	.1050	**.2468**	.0980	.2182	.1491	**.3161**	.1510	.1416	.1623	**.2169**	.2712	.2107	.2537	**.2727**
METEOR	.1472	.1919	.1708	**.2169**	.1396	.1779	.1577	**.2349**	.1639	.2209	.1918	**.2707**	.1768	.1597	.1962	**.2110**	**.2351**	.1967	.2286	.2257
CIDEr	.2067	.4273	.3187	**.6349**	.1480	.4043	.2158	**.7249**	.1680	.5146	.3277	**.8085**	.2668	.2523	.3816	**.5195**	**.6979**	.4878	.6554	.6536
% forgetting	59.1	31.2	43.7	0.0	67.5	33.8	45.0	0.0	68.9	23.6	45.0	0.0	32.8	14.6	16.5	0.0	N/A	N/A	N/A	N/A

Table 2. Performance on MS-COCO. Numbers are the per-task performance after training on the *last* task. Per-task forgetting in the last row is the BLEU-4 performance after the last task divided by the BLEU-4 performance measured immediately after learning each task.

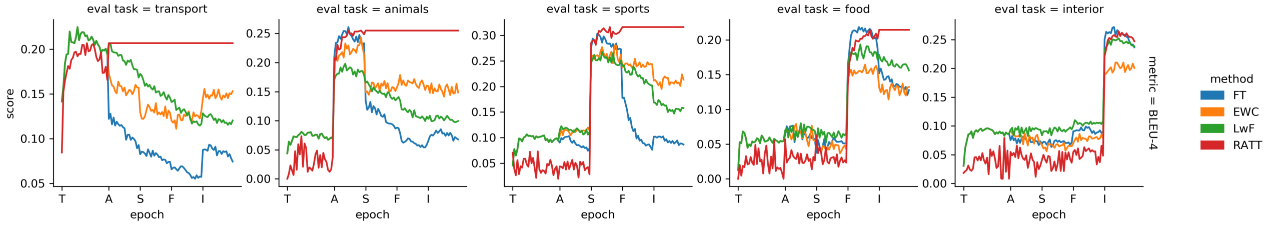


Figure 4. Comparison of RATT and baselines validation performance during training across all MS-COCO tasks.

for previous tasks, the other parts of the network suffer from catastrophic forgetting. Adding the classifier mask improves the situation, but the main contribution comes from applying the mask to the embedding. Applying all masks we obtain zero or nearly-zero forgetting. This, of course, depends on the S_{max} value used during training and in figure 3 we plot learning curves for RATT all MS-COCO tasks for varying values of s_{max} on the validation set. From these curves we see that for low values of s_{max} the network suffers from forgetting of early tasks. We used $s_{max} = 2000$ and $s_{max} = 400$ for experiments on Flickr30k and MS-COCO respectively, resulting in zero forgetting for the latter.

5.3. Results on MS-COCO

In table 2 we report the performance of a fine-tuning baseline with no forgetting mitigation (FT), EWC, LwF, and RATT on our splits for the test set of the MS-COCO captioning dataset. The forgetting percentage is computed by taking the BLEU-4 score for each model after training on the last task and dividing it by the BLEU-4 score at the end of the training of each individual task. From the results we see that all techniques consistently improve performance on previous tasks when compared to the FT baseline. Despite the simplicity of EWC, the improvement over fine-tuning is clear, but it struggles to learn a good model for the last task. LwF instead shows the opposite behavior: it is more capable of learning the last task, but forgetting is more noticeable. RATT achieves *zero* forgetting on MS-COCO, although at the cost of some performance on the final task. This is to be expected, though, as our approach deliberately and progressively limits network capacity to prevent forgetting of old tasks.

To illustrate the differences in forgetting between fine-tuning, RATT, EWC, and LwF, in figure 4 we plot the learning curves on the validation set for all three approaches across all MS-COCO tasks. From these curves we see that all methods except RATT suffer varying degrees of catastrophic forgetting. These curves also illustrate several cases of *negative* forgetting (i.e. positive backward transfer) between tasks.

Some qualitative captioning results on MS-COCO are provided in figure 5. From these examples we see that all techniques except RATT suffer from concept drift after training on the final task, and their captions reflect this.

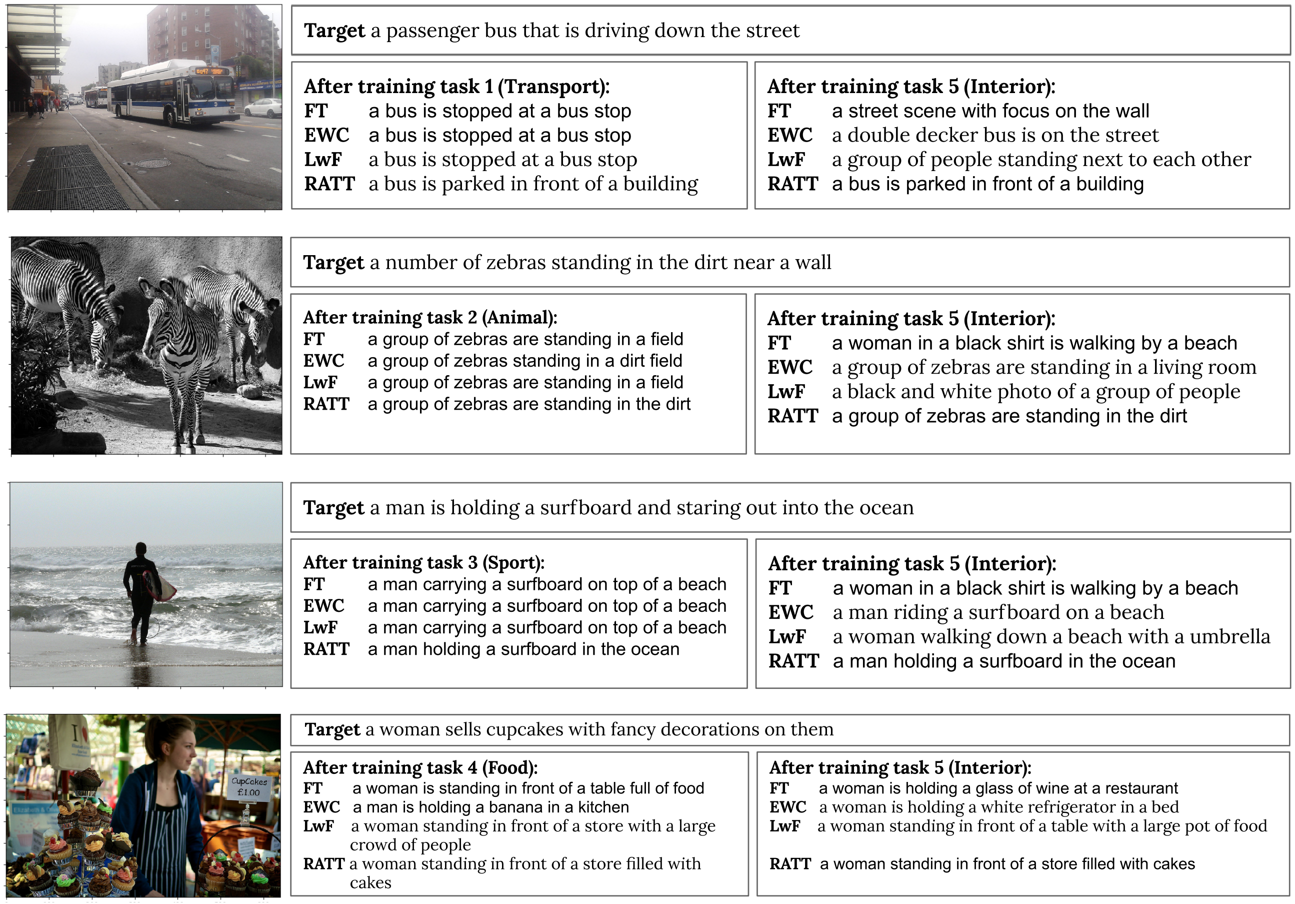
5.4. Results on Flickr30k

In table 3 we report performance of a fine-tuning baseline with no forgetting mitigation (FT), EWC, LwF, and RATT on our Flickr30k task splits. Because these splits are based on *incremental visual categories*, it does not reflect a classical continual-learning setup that enforce disjoint categories to maximize catastrophic forgetting: not only there are common words that share the same meaning between different tasks, but some of the visual categories in early tasks are also present in future ones. For this reason, learning how to describe task $t = 1$ also implies learning at least how to partially describe future tasks, so forward and backward transfer is significant.

Despite this we see that all approaches increase performance on old tasks while retaining good performance on the last one. Note that both RATT and LwF result in *negative forgetting*: in these cases the training of a new task results in backward transfer that increases performance on an old one. EWC improvement is marginal, and LwF behaves a bit better and seems more capable of exploiting backward transfer. RATT backward transfer is instead limited by the choice of a high S_{max}, which however guarantees nearly zero forgetting.

	Scene				Animals				Vehicles				Instruments			
	FT	EWC	LwF	RATT	FT	EWC	LwF	RATT	FT	EWC	LwF	RATT	FT	EWC	LwF	RATT
BLEU-4	.1074	.1370	.1504	**.1548**	.1255	.1381	.1384	**.1921**	.1083	.1332	.1450	**.1724**	.1909	.2313	.1862	**.2386**
METEOR	.1570	.1722	**.1851**	.1710	.2046	.1833	.1954	**.2107**	.1625	.1770	**.1847**	.1750	.1933	.1714	**.1876**	.1782
CIDEr	.1222	.1688	.2402	**.2766**	.2460	.2755	.2756	**.4708**	.1586	.1315	.1748	**.2988**	.2525	.2611	**.2822**	.2329
% forgetting	31.1	11.3	2.7	-2.5	38.7	19.2	-15.1	0.0	35.6	4.9	-1.5	0.0	N/A	N/A	N/A	N/A

Table 3. Performance on Flickr30K. Evaluation is the same as for MS-COCO.



Target a passenger bus that is driving down the street

After training task 1 (Transport):
FT a bus is stopped at a bus stop
EWC a bus is stopped at a bus stop
LwF a bus is stopped at a bus stop
RATT a bus is parked in front of a building

After training task 5 (Interior):
FT a street scene with focus on the wall
EWC a double decker bus is on the street
LwF a group of people standing next to each other
RATT a bus is parked in front of a building

Target a number of zebras standing in the dirt near a wall

After training task 2 (Animal):
FT a group of zebras are standing in a field
EWC a group of zebras standing in a dirt field
LwF a group of zebras are standing in a field
RATT a group of zebras are standing in the dirt

After training task 5 (Interior):
FT a woman in a black shirt is walking by a beach
EWC a group of zebras are standing in a living room
LwF a black and white photo of a group of people
RATT a group of zebras are standing in the dirt

Target a man is holding a surfboard and staring out into the ocean

After training task 3 (Sport):
FT a man carrying a surfboard on top of a beach
EWC a man carrying a surfboard on top of a beach
LwF a man carrying a surfboard on top of a beach
RATT a man holding a surfboard in the ocean

After training task 5 (Interior):
FT a woman in a black shirt is walking by a beach
EWC a man riding a surfboard on a beach
LwF a woman walking down a beach with a umbrella
RATT a man holding a surfboard in the ocean

Target a woman sells cupcakes with fancy decorations on them

After training task 4 (Food):
FT a woman is standing in front of a table full of food
EWC a man is holding a banana in a kitchen
LwF a woman standing in front of a store with a large crowd of people
RATT a woman standing in front of a store filled with cakes

After training task 5 (Interior):
FT a woman is holding a glass of wine at a restaurant
EWC a woman is holding a white refrigerator in a bed
LwF a woman standing in front of a table with a large pot of food
RATT a woman standing in front of a store filled with cakes

Figure 5. Captioning results for all methods on MS-COCO. Images and target captions belong to a specific task and captions are generated by all techniques after training the correct task (left) and a later task (right). Approaches except RATT contextualize to some degree generated captions with respect to the most recently learned task.

6. Conclusions

In this paper we proposed a technique for continual learning of image captioning networks based on Recurrent Attention to Transient Tasks (RATT). Our approach is motivated by a feature of image captioning not shared with other continual learning problems: tasks are composed of *transient* classes (words) that can be shared across tasks. We also showed how to adapt Elastic Weight Consolidation and Learning without Forgetting, two representative approaches of continual learning, to the recurrent image captioning networks. We proposed task splits for the MS-COCO and Flickr30k image captioning datasets, and our experimental evaluation confirms the need for recurrent task attention in order to mitigate forgetting in continual learning with sequential, transient tasks.

References

Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., and Tuytelaars, T. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 139–154, 2018.

Castro, F. M., Marín-Jiménez, M. J., Guil, N., Schmid, C., and Alahari, K. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 233–248, 2018.

Chaudhry, A., Dokania, P. K., Ajanthan, T., and Torr, P. H. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 532–547, 2018.

Coop, R. and Arel, I. Mitigation of catastrophic forgetting in recurrent neural networks using a fixed expansion layer. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, Dallas, TX, USA, aug 2013. IEEE.

De Lange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., and Tuytelaars, T. Continual learning: A comparative study on how to defy forgetting in classification tasks. *arXiv preprint arXiv:1909.08383*, 2019.

Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A., and Bengio, Y. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Hossain, M. Z., Sohel, F., Shiratuddin, M. F., and Laga, H. A comprehensive survey of deep learning for image captioning. *ACM Comput. Surv.*, 51(6), February 2019. ISSN 0360-0300. doi: 10.1145/3295748. URL https://doi.org/10.1145/3295748.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Liu, X., Masana, M., Herranz, L., Van de Weijer, J., Lopez, A. M., and Bagdanov, A. D. Rotate your networks: Better weight consolidation and less catastrophic forgetting. In *International Conference on Pattern Recognition (ICPR)*, 2018.

Lopez-Paz, D. and Ranzato, M. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pp. 6467–6476, 2017.

Mallya, A., Davis, D., and Lazebnik, S. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

Masana, M., Tuytelaars, T., and van de Weijer, J. Ternary feature masks: continual learning without any forgetting. *arXiv preprint arXiv:2001.08714*, 2020.

McCloskey, M. and Cohen, N. J. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., and Wermter, S. Continual lifelong learning with neural networks: A review. *Neural Networks*, 2019.

Pfülb, B. and Gepperth, A. A comprehensive, application-oriented study of catastrophic forgetting in dnns. In *ICLR*, 2019.

Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *IJCV*, 123(1):74–93, 2017.

Rebuffi, S.-A., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3): 211–252, 2015.

Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., and Hadsell, R. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

Schak, M. and Gepperth, A. A study on catastrophic forgetting in deep lstm networks. In Tetko, I. V., Kůrková, V., Karpov, P., and Theis, F. (eds.), *Artificial Neural Networks and Machine Learning – ICANN 2019: Deep Learning*, Lecture Notes in Computer Science, pp. 714–728, Cham, 2019. Springer International Publishing. ISBN 978-3-030-30484-3. doi: 10.1007/978-3-030-30484-3_56.

Schwarz, J., Luketina, J., Czarnecki, W. M., Grabska-Barwinska, A., Teh, Y. W., Pascanu, R., and Hadsell, R. Progress & compress: A scalable framework for continual learning. In *International Conference on Machine Learning (ICML)*, 2018.

Serra, J., Suris, D., Miron, M., and Karatzoglou, A. Overcoming catastrophic forgetting with hard attention to the task. In *International Conference on Machine Learning (ICML)*, 2018.

Sharma, S., El Asri, L., Schulz, H., and Zumer, J. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *CoRR*, abs/1706.09799, 2017. URL http://arxiv.org/abs/1706.09799.

Shin, H., Lee, J. K., Kim, J., and Kim, J. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems*, pp. 2990–2999, 2017.

Sodhani, S., Chandar, S., and Bengio, Y. Toward training recurrent neural networks for lifelong learning. *Neural Computation*, 32:1–34, 11 2019.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3156–3164, 2015.

Wu, C., Herranz, L., Liu, X., Wang, Y., van de Weijer, J., and Raducanu, B. Memory replay GANs: learning to generate images from new categories without forgetting. In *Advances in Neural Information Processing Systems*, 2018.

Yu, L., Twardowski, B., Liu, X., Herranz, L., Wang, K., Cheng, Y., Jui, S., and Weijer, J. v. d. Semantic drift compensation for class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6982–6991, 2020.

Zenke, F., Poole, B., and Ganguli, S. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 3987–3995. JMLR. org, 2017.