Analyse et Implémentation d'un Modèle de Deep Learning

Show, *Attend and Tell*

Angelo Bou Tanous, Basile Joret, Hugo Tondenier

Insa Rouen

May 3, 2025

Plan de la présentation

Introduction

Contexte du projet

- Objectif : étude approfondie d'un modèle de Deep Learning
- Implémentation d'un article scientifique
- Analyse des performances et amélioration du modèle

Choix du papier

- Finance jugée trop complexe et données difficiles à traiter
- Préférance d'un article de Deep Learning pur

Carte d'identité de l'article

Références

- **Titre** : Show, Attend and Tell
- **Auteurs** : K. Xu et al.
- **Conférence** : ICML 2015
- **Citations** : 13455+

Visuel

Show, Attend and Tell: Neural Image Caption Generation with Visual Attention

Kelvin Xu[*] KELVIN.XU@UMONTREAL.CA
Jimmy Lei Ba[†] JIMMY@PSI.UTORONTO.CA
Ryan Kiros[†] RKIROS@CS.TORONTO.EDU
Kyunghyun Cho[‡] KYUNGHYUN.CHO@UMONTREAL.CA
Aaron Courville[‡] AARON.COURVILLE@UMONTREAL.CA
Ruslan Salakhutdinov[†*] RSALAKHU@CS.TORONTO.EDU
Richard S. Zemel[†] ZEMEL@CS.TORONTO.EDU
Yoshua Bengio[‡*] YOSHUA.BENGIO@UMONTREAL.CA
* Université de Montréal, † University of Toronto, ∘ CIFAR

Abstract

Inspired by recent work in machine translation and object detection, we introduce an attention based model that automatically learns to describe the content of images. We describe how we can train this model in a deterministic manner using standard backpropagation techniques and stochastically by maximizing a variational lower bound. We also show through visualization how the model is able to automatically learn to fix its gaze on salient objects while generating the corresponding words in the output sequence. We validate the use of attention with state-of-the-art performance on three benchmark datasets: Flickr8k, Flickr30k and MS COCO.

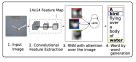


Figure 1. Our model learns a words/image alignment. The visualized attentional maps (3) are explained in Sections 3.1 & 5.4

1. Introduction

Automatically generating captions for an image is a task close to the heart of scene understanding — one of the primary goals of computer vision. Not only must caption generation models be powerful enough to solve the difficult images of determining what objects are in an image, but they must also be powerful enough to capture and express their relationships in natural language. For this reason, caption generation has long been seen as a difficult problem. It amounts to mimicking the remarkable human ability to compress huge amounts of salient visual information into descriptive language and is thus an important challenge for machine learning and AI research.

Yet despite the difficult nature of this task, there has been a recent surge of research interest in attacking the image caption generation problem. Aided by advances in training deep neural networks (Krizhevsky et al., 2012) and the availability of large classification datasets (Russakovsky et al., 2014), recent work has significantly improved the quality of caption generation using a combination of convolutional neural networks (convnets) to obtain vectorial representation of images and recurrent neural networks to decode those representations into natural language sentences (see Sec. 2). One of the most curious facets of the human visual system is the presence of attention (Rensink, 2000; Corbetta & Shulman, 2002). Rather than compress an entire image into a static representation, attention allows for salient features to dynamically come to the forefront as needed. This is especially important when there is a lot of clutter in an image. Using representations (such as those from the very top layer of a convnet) that distill information in image down to the most salient objects is one effective solution that has been widely adopted in previous work. Unfortunately, this has one potential drawback of losing information which could be useful for richer, more descrip-

Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

Page de titre de l'article

Contexte et objectif du papier

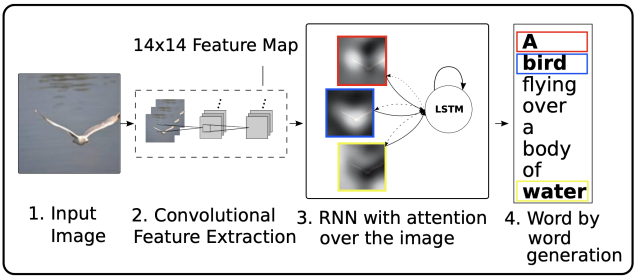
Problématique

- Génération automatique de légendes d'images
- Idée clé : vision + traitement automatique du langage naturel

Approche principale

- Attention sélective sur régions clés

Architecture du modèle



Architecture globale : encodeur-décodeur

Encodeur CNN

- VGG pré-entraîné
- Carte de caractéristiques $14 \times 14 \times 512$:
  - 196 régions spatiales
  - Vecteurs 512-d

Décodeur LSTM

- Génération mot à mot
- Entrées : mot précédent, état interne, contexte

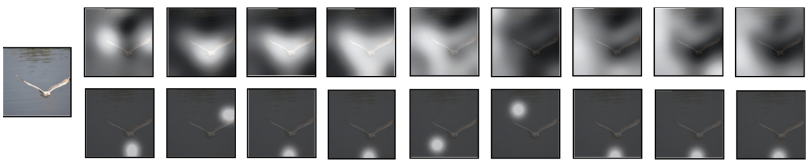
Mécanisme d'attention

Soft Attention

- Moyenne pondérée de toutes les régions
- Tout reste différentiable

Hard Attention

- Sélection d'une région via échantillonnage
- Regard plus "précis"



Comparaison Hard vs Soft Attention

Jeux de données et métriques

Datasets

- Flickr8k
- Flickr30k
- MS COCO

Évaluation

- Scores BLEU
- Scores METEOR

		BLEU				
Dataset	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Flickr8k	Google NIC(Vinyals et al., 2014)†Σ	63	41	27	—	—
	Log Bilinear (Kiros et al., 2014a)◦	65.6	42.4	27.7	17.7	17.31
	Soft-Attention	**67**	44.8	29.9	19.5	18.93
	Hard-Attention	**67**	**45.7**	**31.4**	**21.3**	**20.30**
Flickr30k	Google NIC†◇Σ	66.3	42.3	27.7	18.3	—
	Log Bilinear	60.0	38	25.4	17.1	16.88
	Soft-Attention	66.7	43.4	28.8	19.1	**18.49**
	Hard-Attention	**66.9**	**43.9**	**29.6**	**19.9**	18.46
COCO	CMU/MS Research (Chen & Zitnick, 2014)α	—	—	—	—	20.41
	MS Research (Fang et al., 2014)†α	—	—	—	—	20.71
	BRNN (Karpathy & Li, 2014)◦	64.2	45.1	30.4	20.3	—
	Google NIC†◇Σ	66.6	46.1	32.9	24.6	—
	Log Bilinear◇	70.8	48.9	34.4	24.3	20.03
	Soft-Attention	70.7	49.2	34.4	24.3	**23.90**
	Hard-Attention	**71.8**	**50.4**	**35.7**	**25.0**	23.04

Résultats obtenus par les chercheurs

Datasets et pré-traitement

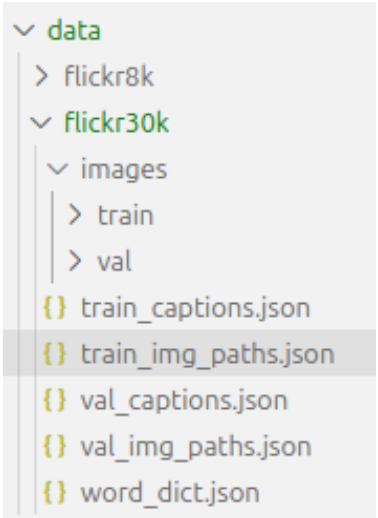
Choix et justification des jeux de données

- **Flickr8k** et **Flickr30k** utilisés ; splits de *Karpathy* (2015) pour une comparabilité maximale.
- **Pas d'entraînement sur MS COCO** : gain certain en BLEU mais coût temps / énergie prohibitif au regard des objectifs pédagogiques.
- Données brutes : chaque image associée à **5 légendes** descriptives.

Datasets et pré-traitement 2

Format et reproductibilité

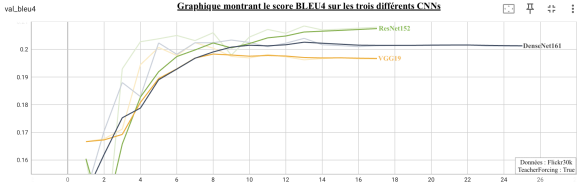
- Application des **splits de Karpathy (2015)** (train/val) afin de garantir la comparabilité avec l'étude originale.
- Génération d'un dataset final :
  - `datasets/train/`, `datasets/val/`
  - `train_captions.json`, `train_img_paths.json`
  - `val_captions.json`, `val_img_paths.json`
  - `word_dict.json` (mapping mot → indice)
- C'est sur ces splits que nous avons entraîné nos modèles, assurant ainsi une **reproductibilité** exacte des résultats par rapport au papier.

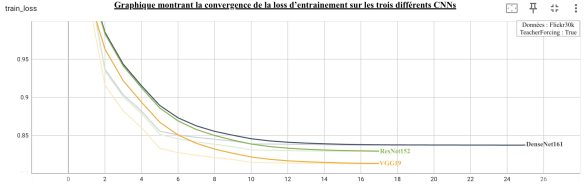


Encodeur CNN testés (backbone gelé)

Backbones comparés

Backbone	Sortie	Paramètres
VGG19	$14 \times 14 \times 512$	$\sim$144 M
ResNet-152	$14 \times 14 \times 2048$	$\sim$60 M
DenseNet161	$14 \times 14 \times 2208$	$\sim$29 M





Décodeur attentif

Architecture LSTM + Attention

- **Embedding** des mots : dimension 512
- **LSTM** uni-directionnel à 512 unités
- Mécanisme **Soft Attention** :
  - Calcul des scores d'attention : $e_{t,i} = w^\top \tanh\left(W_h h_{t-1} + W_a a_i\right)$
  - Poids $\alpha_{t,i} = \mathrm{softmax}(e_{t,i})$
  - Contexte : $c_t = \sum_i \alpha_{t,i}\, a_i$
- Concaténation $(h_{t-1},\, c_t) \rightarrow$ projection $\rightarrow$ entrée LSTM
- Dropout 0.5 sur embeddings et sorties LSTM (par defaut pytorch)

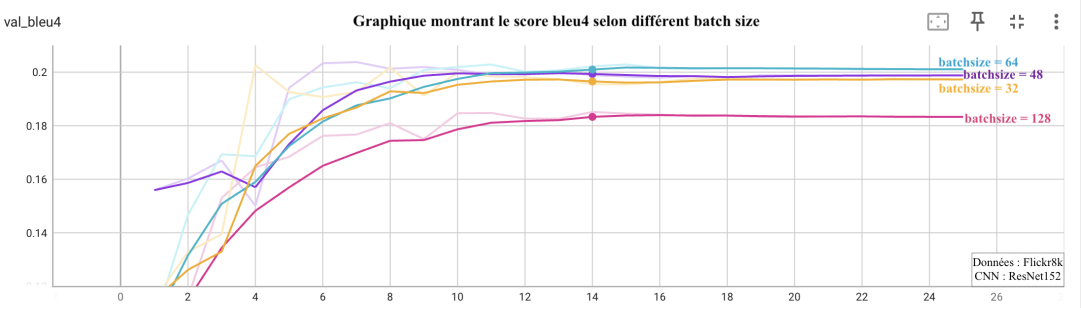
Point Technique

- Fonction de loss : Cross-Entropy sur vocabulaire
- Critère de sélection du meilleur modèle : score BLEU-4 sur `val`

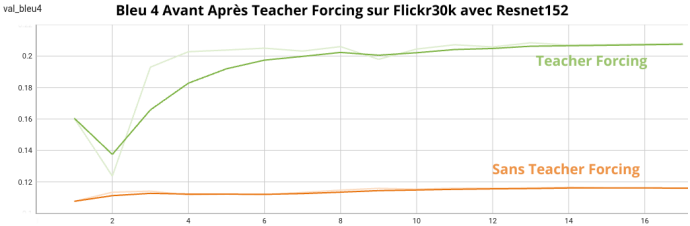
Stratégie d'entraînement

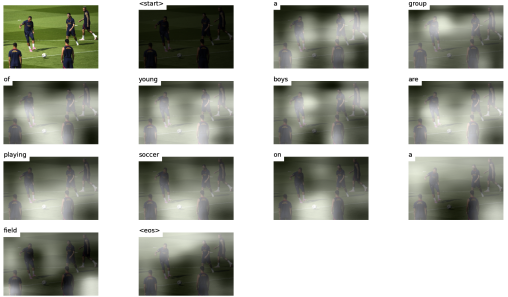
Configuration d'entraînement

- Taille des batchs : 64
- Nombre d'époques : variable
- Learning rate : 0.005 choisi après avoir eu des entraînements instables avec 0.05



Teacher Forcing: Révolution de notre entraînement





Résultats expérimentaux

Hyperparamètres retenus

batch_size	64
epochs	15
lr	0.0005
step_size	5
alpha_c	1
log_interval	100
network	resnet152
teacher_forcing	true
dataset	flickr30k

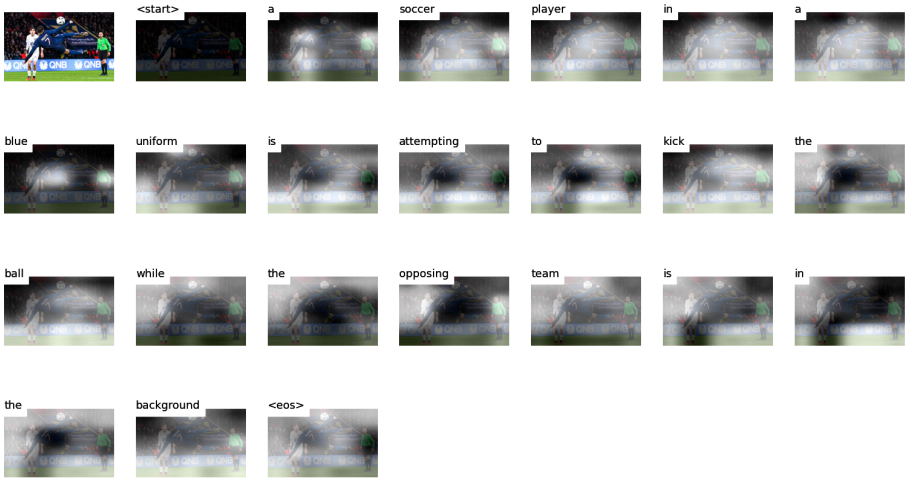
Comparatif des métriques

Métrique	Train 1	Papier	Final
BLEU-1	0.48	0.67	**0.64**
BLEU-2	0.26	0.44	**0.44**
BLEU-3	0.14	0.29	**0.31**
BLEU-4	0.07	0.20	**0.21**

Temps de training nécessaire pour notre modèle : 7h01

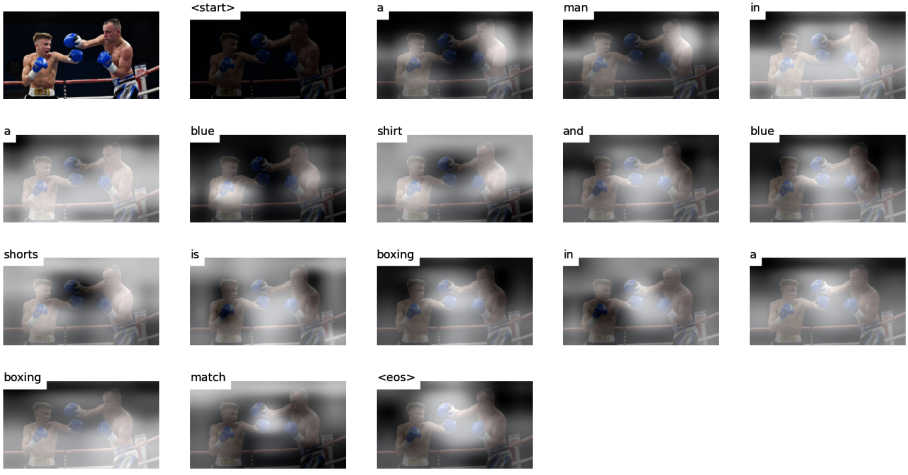
Visualisation des résultats

Visualisation des résultats



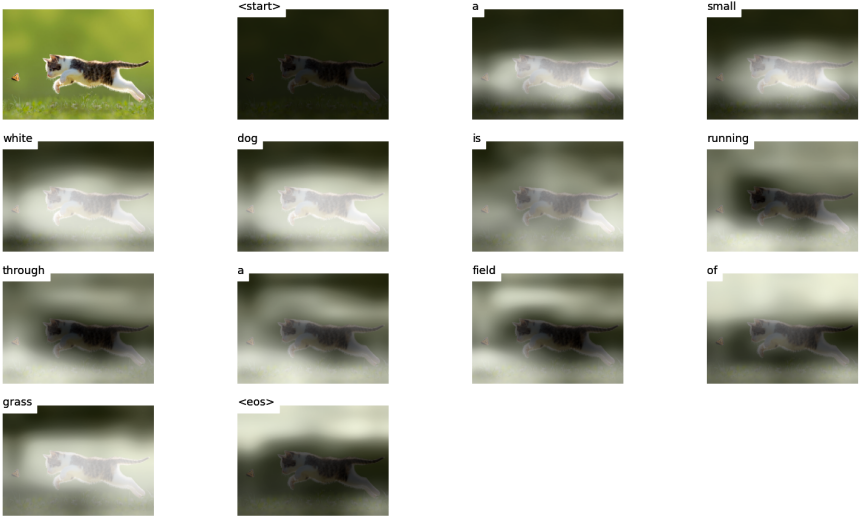
Visualisation des résultats

Visualisation des résultats



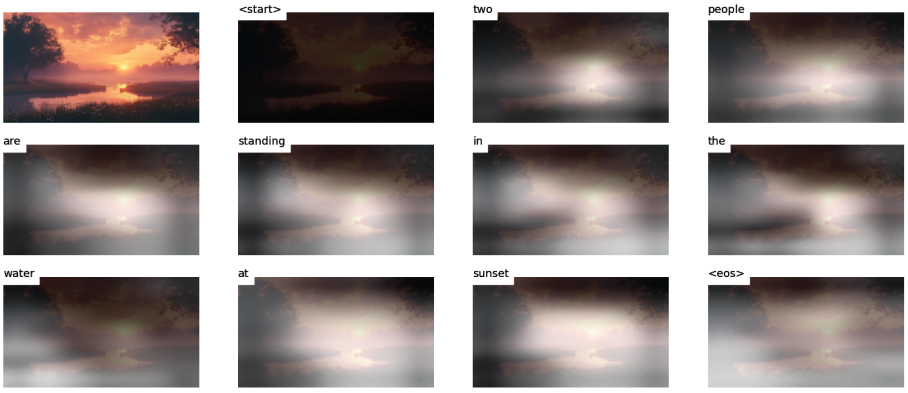
Visualisation des résultats

Visualisation des résultats



Visualisation des résultats

Visualisation des résultats



Analyse critique du modèle

Forces du modèle

- Très performant sur les images "d'action"
- Très entraîné sur des images d'humains
- Produit en sortie des légendes syntaxiquement correctes dans la quasi totalité des cas.

Limites constatées

- Peu performant sur les images ne comportant pas d'actions ou d'humains
- A du mal à déduire l'information importante d'une image complexe.

Améliorations proposées

Changement de dataset

Training sur COCO dataset : images plus nombreuses et plus variées.

Exploration sur les hyper-paramètres existants

- step size pour le learning rate adaptatif
- alpha c pour la soft attention
- dégel du backbone en entrée du modèle

Augmentation de la robustesse au bruit

- Data augmentation
- Exploration de nouvelles méthodes de régularisation (dropout, label smoothing, ...)

Impacts sociétaux et environnementaux

Consommation énergétique

- Entraîner un modèle attentionnel profond (ex. ResNet152 + LSTM) nécessite plusieurs heures de calcul sur GPU, parfois sur supercalculateur.
- Cela engendre une empreinte carbone non négligeable (ex. $\sim$7h d'entraînement dans notre cas).
- Inférence relativement légère en terme de ressources et de temps

Enjeux sociétaux

- Risques de biais dans les légendes générées (stéréotypes liés aux données d'entraînement).
- Possibles dérives dans l'usage (ex. surveillance, annotation automatisée à grande échelle).
- Nécessité de transparence et d'évaluation éthique des usages.

Conclusion

Synthèse du projet

- Reproduction du modèle combinant CNN et LSTM avec mécanisme d'attention douce.
- Entraînement réalisé sur le dataset `flickr30k`, avec ajustement des hyperparamètres et régularisation.
- Évaluation selon les métriques BLEU : résultats globalement cohérents avec ceux du papier, malgré des écarts.

Apports de l'implémentation

- Compréhension approfondie du fonctionnement de l'attention visuelle et de sa mise en œuvre pratique.
- Expérience concrète d'entraînement sur un supercalculateur : gestion des ressources, temps de calcul, monitoring.
- Meilleure maîtrise des problématiques liées au traitement du langage naturel et à la vision par ordinateur.

Merci de votre attention

Questions?