Amlioration de la gnration de lgendes d'images avec RBM sur le module de captioning de BLIP (blip-image-captioning-base)

Fortun HOUESSOU
Sous la supervision de M. Jrme Lacaille

Mars 2025

Abstract

Le Image captioning, qui consiste gnrer des descriptions textuelles partir de contenus visuels, a connu des avances majeures grce aux modles Vision-Language comme BLIP. Cependant, ces approches rencontrent encore des difficults produire des lgendes la fois prcises et contextuellement cohrentes, notamment pour des images complexes ou ambigus. Dans ce travail, nous proposons une amlioration innovante en intgrant une Machine de Boltzmann Restreinte (RBM) dans le processus de gnration de lgendes de BLIP. Les RBM, reconnues pour leur capacit apprendre des reprsentations latentes profondes, permettent daffiner la cohrence smantique des lgendes gnres sans ncessiter un rentranement massif sur de grandes bases de donnes. Nous explorons diffrentes stratgies dintgration, o la RBM intervient pour rvaluer ou ajuster les sorties de BLIP, afin damliorer la pertinence et la richesse des descriptions. Des expriences approfondies menes sur des benchmarks standard montrent que cette approche hybride permet de gnrer des lgendes plus naturelles et contextuellement adaptes, ouvrant ainsi de nouvelles perspectives pour lamlioration des modles Vision-Language.

Mots-cls : Captioning dimages, Modles Vision-Langage (VLM), BLIP, Machine de Boltzmann Restreinte (RBM), Apprentissage profond.

Contents

1 Introduction

Lassociation entre la vision par ordinateur et le traitement automatique du langage naturel a rvolutionn la manire dont les machines interprtent et dcrivent le contenu visuel. Cette convergence a donn naissance des modles Vision-Language Models (VLMs), capables danalyser une image et de gnrer une description textuelle cohrente et pertinente. Lun des modles les plus performants dans ce domaine est le Bootstrapping Language-Image Pre-training (BLIP), qui exploite des techniques avances dapprentissage auto-supervis pour amliorer la comprhension des relations entre les images et le texte.

Cependant, malgr son efficacit, BLIP souffre de certaines limitations, notamment une tendance produire des descriptions gnriques, parfois trop peu informatives ou manquant de diversit smantique. Cette limitation est due en partie aux biais inhrents aux donnes dentranement et aux mcanismes doptimisation utiliss. Pour remdier ces faiblesses, il est crucial dexplorer des mthodes complmentaires qui puissent affiner les reprsentations latentes du modle et amliorer la richesse du langage gnr.

Dans cette optique, lintgration dun modle probabiliste comme la Machine de Boltzmann Restreinte (RBM) reprsente une piste prometteuse. Les RBMs sont largement utilises dans lapprentissage non supervis et la modlisation des distributions complexes. Elles offrent un moyen damliorer les reprsentations caches en capturant des interactions non linaires subtiles dans les donnes. En les combinant avec le module de captioning de BLIP, nous visons enrichir la qualit des descriptions gnres en affinant les reprsentations smantiques sous-jacentes.

Ce travail explore donc lintgration dune RBM au sein du pipeline de gnration de descriptions dimages de BLIP, en valuant son impact sur la prcision, la diversit et la pertinence linguistique des lgendes produites. Cette approche pourrait ouvrir de nouvelles perspectives pour lamlioration des modles Vision-Language et leur application des domaines exigeant une comprhension fine du contenu visuel, comme laccessibilit numrique ou lannotation automatique dimages.

2 Image - Captioning : tat de l'Art

Le **captioning d'images**, ou annotation automatique d'images, est une tche en vision par ordinateur et de langage consistant gnrer une description textuelle cohrente et prcise du contenu d'une image. Cette tche est complexe car elle ncessite une comprhension approfondie des objets prsents, de leurs actions, de leurs interactions et du contexte global de l'image.

2.1 Approches classiques du captioning d'images

Plusieurs approches ont t explores parmi lesquelles :

2.1.1 Rseaux de Neurones Rcurrents (RNN)

Les premires approches utilisaient des RNN, notamment des LSTM (Long Short-Term Memory), pour gnrer des descriptions squentielles des images. Un pipeline typique impliquait l'utilisation d'un rseau de neurones convolutifs (CNN) pour extraire des caractristiques visuelles de l'image, suivie d'un RNN pour gnrer la lgende base sur ces caractris-

tiques. Par exemple, l'article *Language Models for Image Captioning: The Quirks and What Works* explore l'utilisation des RNN pour cette tche. [**?**].

2.1.2 Transformers

Les modles *Transformers*, introduits initialement pour le traitement du langage naturel, ont t adapts au captioning d'images en raison de leur capacit modliser des dpendances longue porte dans les donnes squentielles. Ces modles utilisent des mcanismes d'attention pour se concentrer sur diffrentes parties de l'image lors de la gnration de chaque mot de la lgende. L'article *A Review of Transformer-Based Approaches for Image Captioning* fournit une revue exhaustive des modles de captioning d'images bass sur les Transformers[**?**].

2.1.3 Modles Vision-Langage

Les modles *Vision-Langage* intgrent des informations visuelles et textuelles pour amliorer la performance du captioning d'images. Ces modles sont pr-entrans sur de grandes quantits de donnes d'images et de textes associs, puis affins pour des tches spcifiques comme le captioning. Le modle blip-image-captioning [8] qui fera l'objet de notre tude est un exemple concret des performances atteintes avec les VLMs.
Mais c'est quoi rellement un VLM ?

3 Vision-Language Models (VLM)

Les modles de Vision Langage (VLM) sont des architectures multimodales qui combinent le traitement visuel et linguistique pour accomplir des tches varies telles que la gnration de lgendes d'images, la recherche d'images par texte, ou l'association image-texte. Ces modles ont gagn en popularit grce leur capacit fusionner les informations textuelles et visuelles, permettant ainsi de rsoudre des problmes complexes en vision par ordinateur et en traitement du langage naturel. Parmi les modles les plus connus dans ce domaine figurent CLIP [11] et BLIP [8].

BLIP, en particulier, adopte une approche de pr-entranement utilisant des donnes web massives et combine un encodeur d'image avec un dcodeur de texte pour gnrer des descriptions adaptes aux images. Cette approche a dmontr son efficacit en gnration de lgendes et interprtation multimodale. De mme, CLIP [11] se distingue par sa capacit associer des images et des descriptions textuelles, facilitant ainsi des applications comme la recherche d'images par texte et l'analyse d'images partir de requtes textuelles.

Les VLMs sont galement prsents dans des travaux comme VisualBERT [9], qui fusionne les informations visuelles et textuelles dans un modle transformeur pour des tches de comprhension multimodale, telles que la rponse des questions visuelles. Ce modle a montr qu'il est possible d'aligner les reprsentations visuelles et textuelles dans un espace partag pour des performances solides sur des tches de vision et de langage.

Un autre modle significatif dans ce domaine est ViLT [7], qui a introduit une approche simplifie, ne recourant ni aux convolutions ni une supervision rgionale, permettant de traiter simultanment les images et les textes tout en rduisant la complexit du modle. Cela a ouvert la voie des applications plus lgres tout en maintenant des performances comptitives.

Dans le cadre de la conduite autonome et d'autres applications de scurit, des travaux comme VLM-RL [6] ont explor l'intgration des VLMs avec l'apprentissage par renforcement pour amliorer la prise de dcision base sur des informations multimodales, notamment en gestion de scnarios visuels et textuels complexes.

Les capacits de raisonnement spatial sont galement un domaine d'amlioration important pour les VLMs, comme le montre SpatialVLM [3], qui cherche doter ces modles de la capacit effectuer des raisonnements gomtriques et spatiaux, ce qui est essentiel pour des applications telles que la navigation autonome et la manipulation d'objets en 3D.

Enfin, des modles comme VILA [5] et Xmodel-VLM [10] se concentrent sur des techniques de pr-entranement et d'optimisation pour rendre les VLMs plus efficaces et adapts aux environnements de production, en rduisant la taille des modles tout en prservant leur capacit traiter des informations visuelles et textuelles de manire prcise.

Dans la suite de ce rapport sur l'tat de l'Art des VLMs, de Blip et RBM, nous commenons par prsenter de faon gnrale, le fonctionnement d'un VLM puis le role des mathmatiques derrire ces modles, et enfin nous traiterons les diffrents modles qui feront l'objet de ce sujet savoir BLIP (Bootstrapping Language-Image Pre-training) et RBM (Restricted Boltzmann machine).

Les **modles de Vision et Langage** (VLMs) sont des systmes qui essaient de comprendre la fois des **images** et des **mots**. Imaginons qu'on montre une image de chat un enfant et qu'on lui dise "C'est un chat". Le cerveau de l'enfant va faire le lien entre l'image du chat et le mots "chat". C'est exactement ce que font les VLMs, mais d'une manire beaucoup plus complexe et dans un ordinateur !

Comment a marche concrtement ?

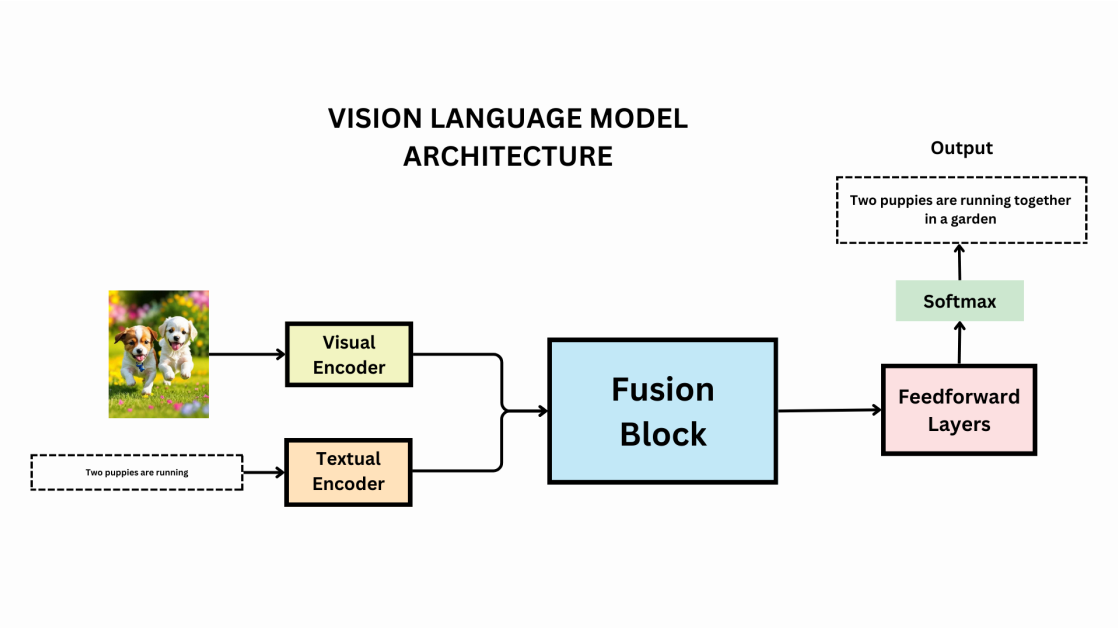

Figure 1: VLM

Le **VLM** combine la comprhension des **images** et des **mots**. Il analyse d'abord l'image pour en extraire des informations visuelles (formes, objets, etc.) via un rseau de neurones. Ensuite, les mots sont transforms en reprsentations que le modle peut comprendre.

Enfin, le VLM fusionne les informations des images et des mots pour accomplir des tches comme l'association image-texte ou la gnration de descriptions d'images. L'objectif est d'apprendre lier efficacement ces deux modalits pour mieux comprendre et interagir avec le monde visuel et linguistique simultanment. C'est comme un Homme qui entends le mot "chat" et sait un peu quelle forme s'attendre, parce que des chats il en a beaucoup vu, de diffrentes couleurs et dans diffrents horizons.

Un peu de formalisme :

L'entrainement du modle BLIP s'ffectue en plusieurs tapes :

3.1 Prtraitement et Encodage

Soit x une entre visuelle (image) et t une entre textuelle (texte). Dans la plupart des VLMs, l'image x est d'abord passe travers un encodeur d'image f_{image}, typiquement un rseau de neurones convolutifs (CNN) ou un modle transformer ou vision encodeur adapt l'image (par exemple, Vision Transformer, ViT). L'objectif est de projeter l'image dans un espace de reprsentation latente $\mathcal{Z}_{\text{image}} \in R^d$.

$$\mathcal{Z}_{\text{image}} = f_{\text{image}}(x)$$

De mme, le texte t est encod par un modle de langage, tel que le Transformer, en une reprsentation vectorielle $\mathcal{Z}_{\text{texte}} \in R^d$. Le processus de traitement du texte suit un encodage similaire avec une fonction f_{texte}, qu'on ne connait pas priori (BERT par exemple).

$$\mathcal{Z}_{\text{texte}} = f_{\text{texte}}(t)$$

Dans le processus d'embedding, certains modles de VLM utilisent du : self-attention.

3.2 Fusion des Reprsentations : Alignement Visuel-Linguistique

Pour regrouper ces deux embeddings dans un mme espace latent, il y a deux grandes stratgies :

A. Projection directe

On passe Z_{image} et Z_{texte} dans des Multi-Layer Perceptrons (MLP) [4] ou des couches linaires afin de les projeter dans un **mme espace latent**. Une fois cette projection effectue, on calcule simplement la distance entre ces deux reprsentations, par exemple via une similarit:

$$\text{similarit}(Z_{\text{image}}, Z_{\text{texte}})$$

Les fonctions de similarit font l'objet d'une estimation de la distance entre deux embedding. C'est une faon de reprsenter par un nombre les diffrences ou rapprochements entre entits.

Fonctions de similarit

Bases sur la distance ou l'angle

- **Similarit cosinus (Cosine Similarity)** :

$$\text{sim}(x, y) = \frac{x \cdot y}{\|x\|\|y\|}$$

- **Distance euclidienne (Euclidean Distance)** :

$$d(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$$

  Souvent transforme en :

$$\text{sim}(x, y) = \frac{1}{1 + d(x, y)}$$

Bases sur des probabilits

- **Divergence de Kullback-Leibler (KL Divergence)** :
  Pour deux distributions de probabilits P et Q :

$$D_{KL}(P \parallel Q) = \sum_i P(i) \log \left(\frac{P(i)}{Q(i)} \right)$$

  Pas symtrique et non borne.

- **Divergence de Jensen-Shannon (JS Divergence)** :
  Variante symtrique et plus stable du KL :

$$D_{JS}(P, Q) = \frac{1}{2} D_{KL}(P \parallel M) + \frac{1}{2} D_{KL}(Q \parallel M)$$

  o

$$M = \frac{1}{2}(P + Q)$$

Bases sur des ensembles

- **Similarit de Jaccard (Jaccard Similarity)** :
  Pour des ensembles A et B :

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

- **Coefficient de recouvrement (Overlap Coefficient)** :

$$\text{Overlap}(A, B) = \frac{|A \cap B|}{\min(|A|, |B|)}$$

C'est prcisment l'approche adopte par le modle CLIP.

B. Cross-Attention

Dans certains modles comme BLIP, avant la projection des donnes bimodales (image/texte) dans l'espace latent,un mecanisme d'attention croise (cross-attention) est utilis dans les architectures de type transformer. Ce mcanisme permet de calculer une attention mutuelle entre les reprsentations des images et du texte afin d'extraire des correspondances pertinentes entre les modalits. Les articles [2] et [1] introduisent et expliquent dans les dtails le mcanisme d'attention. Le calcul de l'attention croise peut tre formalis comme suit :

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

o Q (Query) est la requte (environnement textuel ou visuel), K (Key) est la cl (respectivement l'information visuelle ou textuelle), et V (Value) est la valeur : [14] et [13]. Ce mcanisme est appliqu pour apprendre les relations entre $\mathcal{Z}_{\text{image}}$ et $\mathcal{Z}_{\text{texte}}$, en ajustant les poids des connexions neuronales selon l'importance de chaque lment textuel ou visuel pour le modle.

Optimisation et Apprentissage

L'entranement des VLMs repose sur la minimisation d'une fonction de perte $\mathcal{L}$ qui permet au modle d'apprendre associer de manire optimale les informations visuelles et textuelles. Par exemple, une fonction de perte typique dans le cas d'une tche de classification d'images par texte est la suivante :

$$\mathcal{L} = \sum_i \left[\log P(y_i|\mathcal{Z}_{\text{image}}) + \log P(y_i|\mathcal{Z}_{\text{texte}})\right]$$

o y_i est l'tiquette associe l'entre x_i ou t_i. Cette formulation implique une double minimisation des erreurs sur les prdictions visuelles et textuelles.

Un autre type de perte, utilis dans les modles de type *contrastive learning* (comme CLIP [11]), repose sur l'ide de maximiser la similarit entre les reprsentations textuelles et visuelles d'une mme instance tout en minimisant la similarit entre les paires d'images et de textes diffrentes. La fonction de perte contraste peut tre formule ainsi :

$$\mathcal{L}_{\text{contrastive}} = -\sum_i \log \frac{\exp(\text{sim}(\mathcal{Z}^i_{\text{image}}, \mathcal{Z}^i_{\text{texte}}))}{\sum_j \exp(\text{sim}(\mathcal{Z}^i_{\text{image}}, \mathcal{Z}^j_{\text{texte}}))}$$

o $\text{sim}(a, b)$ reprsente la similarit entre les vecteurs a et b, souvent mesure par un produit scalaire ou un cosinus.

Architecture gnrale dun VLM (Vision-Language Model)

1 Embedding

- Image → Patchs (ViT) ou CNN → Embeddings visuels.
- Texte → Tokenisation → Embeddings textuels.

2 Self-Attention(selon le modle)

- Appliqu sparment sur les embeddings :
- Image → Self-attention entre patchs.
- Texte → Self-attention entre tokens.

3 Cross-Attention (si fusion, cas de BLIP)

- Texte attend sur limage ou inversement.
- Q = texte, K, V = image (ou linverse selon le modle).
- Permet dassocier finement les informations. (On dit que le texte et l'image se regardent)

4 Projection

- Alignement dans un mme espace vectoriel via couches linaires.
- Image → projection.
- Texte → projection.

5 Similarit

- Calcul par produit scalaire ou cosin similarity entre vecteurs projets.

6 Contrastive Loss

- But :
  - Maximiser la similarit des bonnes paires (image, texte lis).
  - Minimiser celle des paires incorrectes.

"""

4 BLIP (Bootstrapping Language-Image Pre-training)

BLIP (*Bootstrapping Language-Image Pre-training*) [8] est un framework conu pour amliorer les performances sur les tches vision-langage. Il se distingue par sa capacit exploiter la fois des **donnes bruites web-scale** et des annotations de qualit, grce une architecture flexible et un apprentissage par bootstrapping.

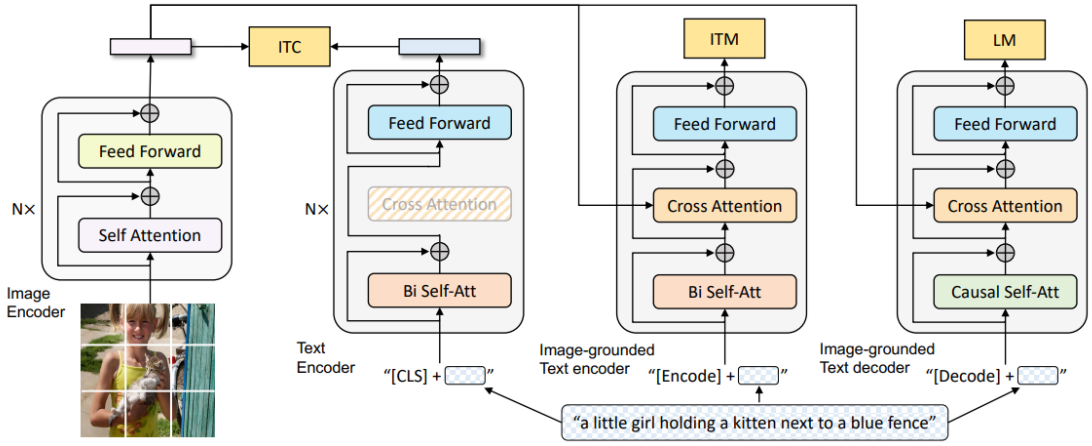

Figure 2: BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation

4.0.1 Fonctionnement gnral

BLIP repose principalement sur trois tches de pr-entranement complmentaires :

- **Image-Text Contrastive Learning (ITC)** : maximise la similarit entre les reprsentations dune image et de sa lgende correspondante, et minimise celles des paires non correspondantes.
  L'ITC est une fonction de perte utilise pour aligner les reprsentations des images et des textes dans un espace commun. Elle est inspire de la InfoNCE loss et fonctionne comme une perte contrastive entre les paires image-texte positives et ngatives.

$$\mathcal{L}_{ITC} = -\frac{1}{N}\sum_{i=1}^{N}\left(\log\frac{e^{\text{sim}(z_i^I, z_i^T)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(z_i^I, z_j^T)/\tau}}\right)$$

  o z_i^I et z_i^T sont les embeddings dimage et de texte, sim est la similarit cosinus, τ est la temprature (temperature scaling) et N la taille du lot des donnes traites.

- **Image-Text Matching (ITM)** : discrimine si une paire image-texte est correcte ou incorrecte via une classification binaire, souvent couple une fonction de perte d'entropie croise classique.

$$\mathcal{L}_{ITM} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log P(y_i|I_i, T_i) + (1-y_i)\log(1 - P(y_i|I_i, T_i))\right]$$

- **Image-Conditioned Language Modeling (LM)** : gnre du texte conditionnellement une image.

$$\mathcal{L}_{LM} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T_i}\log P(w_t^i | w_{<t}^i, I_i)$$

  o :

  - N est le nombre d'exemples dans le batch.
  - T_i est la longueur de la squence de texte pour l'exemple i. (nombre de token)
  - w_t^i est le mot la position t dans la squence i.
  - $w_{<t}^i$ reprsente tous les mots prcdents dans la squence.
  - I_i est l'image associe.
  - $P(w_t^i \mid w_{<t}^i, I_i)$ est la probabilit prdite du mot w_t^i tant donn l'image et les mots prcdents.

4.0.2 Architecture globale

BLIP sappuie sur une combinaison de :

- **Encoders visuels** (souvent des Vision Transformers - ViT)
- **Encoders textuels** (type BERT)
- **Dcoders textuels** (transformer auto-rgressif)

Pendant l'entrainement, les modles apprennent dans un espace latent partag o les embeddings dimages et de textes sont projets, permettant une compatibilit smantique entre les deux modalits.

Dans ce projet, nous nous interessons plus prcisment au modle BLIP-Captionner du framework BLIP, qui en sortie, gnre une description textuelle en rponse une image reue en entre.

5 Blip-Image-Captioning-Base

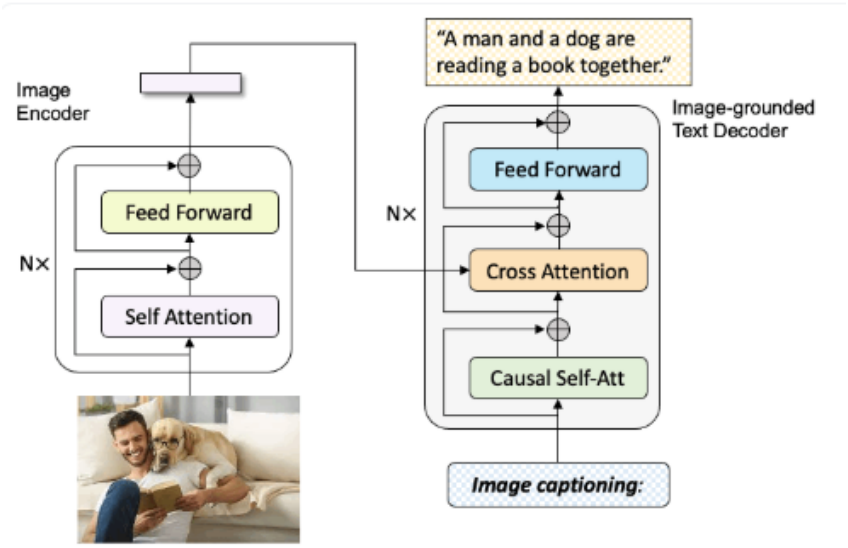

Figure 3: Blip-image-captioning-base

Pour la gnration de lgende d'image (image captioning) le modele blip-image-captioning-base utilse :

- Un encodeur d'image (ViT) compos de couches d'auto attention (SA) et d'un reseau Feed Forward (FF) le tout entrain sur des bases de donnes annotes par des humains comme Coco et des donnes bruites du web et nttoyes par bootstrapping via la mthode CapFilt. CapFilt est un ensemble form d'un Captioner(Image-grounded Text Decoder)[8] et d'un Flilter(Image-grounded Text Encoder) qui sont des modules du framework BLIP et qui servent reannoter (gnration de lgende) les images bruites obtenues sur le web.

- Le dcodeur de Blip-image-captioning-base est compos d'un rseau FF, pr-entrain avec de l'auto attention bidirectionnelle sur coco dataset. La perte utilise est l'ITC (confre figure 2). Il n'y a pas d'attention croise (le module "Cross Attention" est gel dans un premier temps).Dans un second temps, le meme modle pr-entrain est repris et cette fois ci avec l'attention croise(image-grounded Text encodeur sur la figure 2) et reentrain sur Coco dataset pour une tache de classification binaire [8]. La perte utilise est l'ITM. Enfin le dcodeur de blip-image-captioning est obtenu en remplaant les couches d'auto attention bidirectionnelle par des couches d'auto-attention causale comme sur la figure 2 (image-grounded Text Decoder). La perte utilise lors de l'entrainement est la LM.

Notre objectif est d'augmenter les performances de blip-image-captioning pour affiner la gnration de lgende d'image par insertion d'une RBM (Machine de Blotzmann Restreinte) entre l'encodeur et le dcodeur du modele.

6 La Machine de Boltzmann Restreinte (RBM)

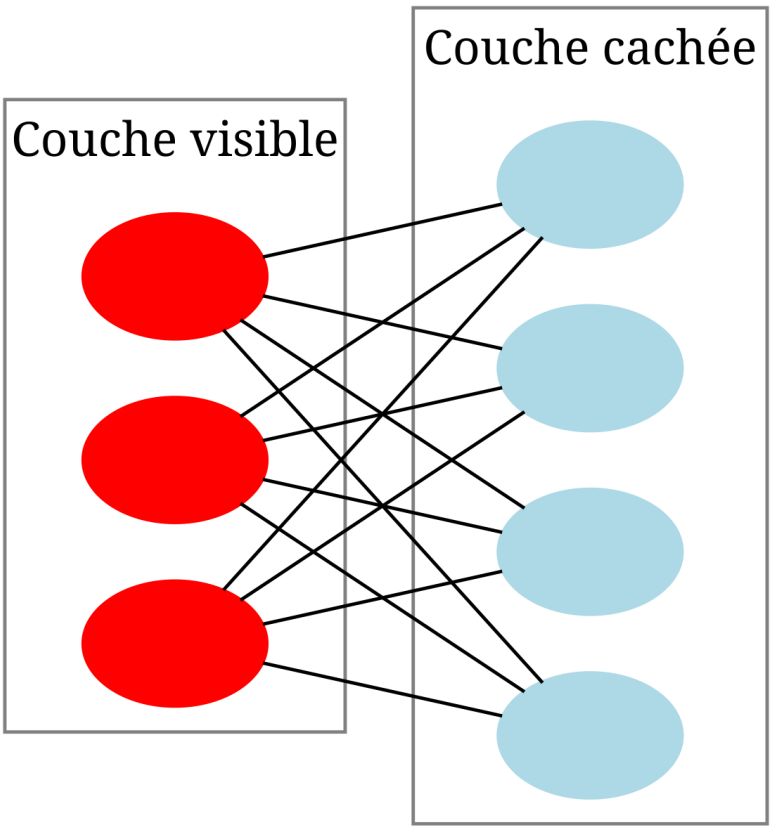

Figure 4: Restricted Boltzmann machine

La Machine de Boltzmann Restreinte (RBM), introduite par [12] et popularise pour l'apprentissage non supervis, est un modle probabiliste gnratif appartenant la famille des modles nergtiques. Apparues dans les annes 1980, les Machines de Boltzmann Restreintes (RBM) ont bnfici dune meilleure attention avec lessor des capacits de calcul

et leur intgration dans des structures dapprentissage profondes, en particulier les Deep Belief Networks (DBN). En superposant plusieurs RBM, il devient possible dextraire des reprsentations successives des donnes, passant de caractristiques lmentaires des concepts plus abstraits. Par ailleurs, lorsquune RBM est exploite pour transformer des entres en reprsentations latentes, elle peut tre assimile un rseau de neurones propagation avant, optimisant ainsi son intgration dans des modles superviss pour amliorer lefficacit de lapprentissage.

6.1 Pourquoi la machine de Boltzmann restreinte ?

L'utilisation d'un **Restricted Boltzmann Machine (RBM)**pour amliorer l**image captioning** est un choix intuitif et judicieux, notamment en raison de sa capacit apprendre des reprsentations riches et compactes des donnes visuelles de manire **non supervise**. En effet, un RBM permet de modliser efficacement une distribution de probabilit inconnue en dcouvrant des caractristiques latentes pertinentes dans les images. Grce sa structure bipartite compose de **variables visibles** (les pixels ou des features extraites dun rseau de neurones) et de **variables latentes** (invisibles), il est capable de capturer des **dpendances complexes** entre les pixels dune image et dencoder des relations de haut niveau, facilitant ainsi la gnration de descriptions prcises et cohrentes.

Dans un cadre dapprentissage probabiliste, le RBM optimise ses **paramtres** θ via des mthodes comme **lestimation du maximum de vraisemblance**, ce qui lui permet dencoder efficacement la structure sous-jacente des images. Cette approche est particulirement bnfique pour limage captioning, o il est crucial de comprendre non seulement les objets prsents dans une image, mais aussi leurs relations contextuelles et smantiques. En apprenant des reprsentations de manire non supervise, le RBM permet de gnraliser sur de nouvelles images et damliorer la robustesse du modle de gnration de descriptions textuelles.

Enfin, en intgrant un RBM en amont dun modle de gnration de texte, on bnficie dune reprsentation dentre plus expressive et informative, ce qui amliore la qualit des lgendes gnres. Cette synergie entre **modlisation probabiliste** et **apprentissage profond** fait du RBM un choix judicieux et intuitif pour lamlioration des systmes dimage captioning.

6.2 Structure d'une RBM

Une RBM est un rseau biparti compos de :

- **Une couche visible** $\mathbf{v} = (v_1, v_2, \ldots, v_m)$ reprsentant les donnes observables.
- **Une couche cache** $\mathbf{h} = (h_1, h_2, \ldots, h_n)$ permettant de capturer des dpendances complexes (caches).

Contrairement aux Machines de Boltzmann standards, il n'y a **pas de connexions intra-couche** :

- Aucune connexion entre les neurones visibles.
- Aucune connexion entre les neurones cachs.

6.3 Fonction nergtique

Le cur de la RBM repose sur une fonction dnergie dfinie par :

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{v}^\top \mathbf{W}\mathbf{h} - \mathbf{b}^\top \mathbf{v} - \mathbf{c}^\top \mathbf{h}$$

o :

- $\mathbf{W}$ est la matrice des poids entre les couches visible et cache.
- $\mathbf{b}$ et $\mathbf{c}$ sont les biais des couches visible et cache respectivement.
- $\mathbf{v}$ et $\mathbf{h}$ sont les tats des neurones sur les couches visible et cache respectivement.

6.4 La RBM comme modle probabiliste

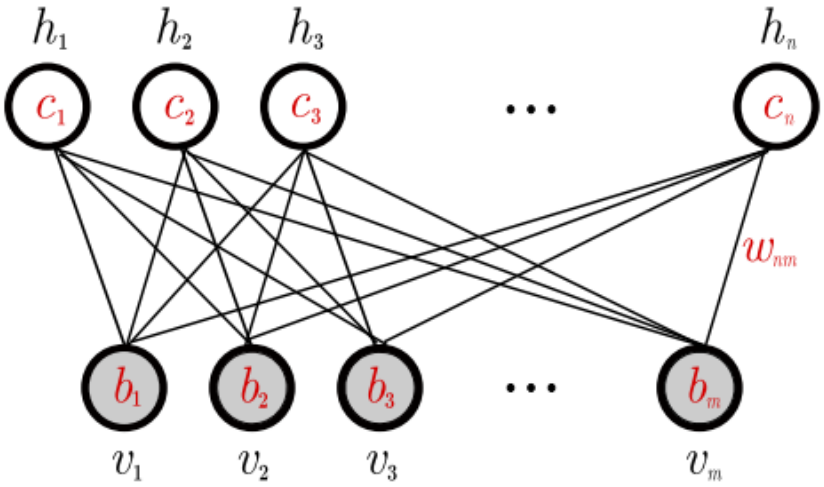

Figure 5: Restricted Boltzmann machine

Une Machine de Boltzmann Restreinte (RBM) est un modle probabiliste bas sur un champ de Markov alatoire structur sous forme dun graphe biparti non orient. Elle est constitue de m units visibles $V = (V_1, ..., V_m)$, qui reprsentent les donnes observables, et de n units caches $H = (H_1, ..., H_n)$, charges de capturer les interactions entre ces donnes.

Dans le cadre des RBM binaires, qui sont lobjet de cette tude, les variables alatoires (V, H) prennent des valeurs dans $\{0, 1\}^{m+n}$. La distribution conjointe associe ce modle suit la loi de Gibbs :

$$p(v, h) = \frac{1}{Z} e^{-E(v,h)} \tag{1}$$

avec l'nergie dfinie par :

$$E(v, h) = -\sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} h_i v_j - \sum_{j=1}^{m} b_j v_j - \sum_{i=1}^{n} c_i h_i. \tag{2}$$

Ici, pour tout $i \in \{1, ..., n\}$ et $j \in \{1, ..., m\}$, w_{ij} dsigne un coefficient rel associ la connexion entre lunit cache H_i et lunit visible V_j. Les paramtres b_j et c_i correspondent respectivement aux biais appliqus aux units visibles et caches.

L'entranement d'une **Machine de Boltzmann Restreinte (RBM)** repose principalement sur des mthodes d'optimisation adaptes aux modles probabilistes base d'nergie. L'algorithme d'apprentissage le plus couramment utilis est **Contrastive Divergence (CD)**, introduit par Geoffrey Hinton. D'autres algorithmes comme : Parallel Tempering et Persistent Contrastive Divergence (PCD) sont aussi utiliss.

Lapprentissage des paramtres $(\mathbf{W}, \mathbf{b}, \mathbf{c})$ repose sur une estimation approche du gradient du logarithme de la vraisemblance :

$$\Delta w_{ij} = \epsilon \left(\langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \right) \tag{3}$$

$$\Delta W = \epsilon \cdot \left(\langle v h^T \rangle_{\text{donnes}} - \langle v h^T \rangle_{\text{modle}} \right) \tag{4}$$

o :

- ϵ est le taux dapprentissage.
- $\langle \cdot \rangle_{data}$ est l'esprance sous la distribution empirique des donnes.
- $\langle \cdot \rangle_{model}$ est l'esprance sous la distribution modlise.

Le processus implique des tapes de **Gibbs Sampling**, alternant entre la mise jour de $\mathbf{h}$ et $\mathbf{v}$:

$$P(h_j = 1 | \mathbf{v}) = \sigma \left(\sum_i w_{ij} v_i + c_j \right) \tag{5}$$

et

$$P(v_i = 1 | \mathbf{h}) = \sigma \left(\sum_j w_{ij} h_j + b_i \right) \tag{6}$$

o $\sigma(x)$ est la fonction sigmode.

6.5 Formalisme

vraissemblance du modle
Posons

$$\mathcal{L}(\theta | v) = p(v | \theta) = \frac{1}{Z} \sum_h e^{-E(v,h)} \tag{7}$$

La log vraissemblance :

$$\ln \mathcal{L}(\theta | v) = \ln p(v | \theta) = \ln \frac{1}{Z} \sum_h e^{-E(v,h)} = \ln \sum_h e^{-E(v,h)} - \ln \sum_{v,h} e^{-E(v,h)}$$

Gradient de la log vraissemblance :

$$\frac{\partial \ln \mathcal{L}(\theta | v)}{\partial \theta} = \frac{\partial}{\partial \theta} \left(\ln \sum_h e^{-E(v,h)} \right) - \frac{\partial}{\partial \theta} \left(\ln \sum_{v,h} e^{-E(v,h)} \right)$$

$$= -\frac{1}{\sum_h e^{-E(v,h)}} \sum_h e^{-E(v,h)} \frac{\partial E(v,h)}{\partial \theta} + \frac{1}{\sum_{v,h} e^{-E(v,h)}} \sum_{v,h} e^{-E(v,h)} \frac{\partial E(v,h)}{\partial \theta}$$

et en utilisant l'galit :

$$p(h|v) = \frac{p(v,h)}{p(v)} = \frac{\frac{1}{Z}e^{-E(v,h)}}{\frac{1}{Z}\sum_h e^{-E(v,h)}} = \frac{e^{-E(v,h)}}{\sum_h e^{-E(v,h)}}$$

on a :

$$\frac{\partial \ln \mathcal{L}(\theta|v)}{\partial \theta} = -\sum_h p(h|v)\frac{\partial E(v,h)}{\partial \theta} + \sum_{v,h} p(v,h)\frac{\partial E(v,h)}{\partial \theta} \tag{8}$$

Le passage un lot S de donnes (batch) de taille N donne :

$$\frac{1}{N}\sum_{v\in S}\frac{\partial \ln \mathcal{L}(\boldsymbol{\theta}|v)}{\partial \theta} = \frac{1}{N}\sum_{v\in S}\left[-\mathbb{E}_{p(h|v)}\left[\frac{\partial E(v,h)}{\partial \theta}\right] + \mathbb{E}_{p(h,v)}\left[\frac{\partial E(v,h)}{\partial \theta}\right]\right] \tag{9}$$

$$\frac{1}{N}\sum_{v\in S}\frac{\partial \ln \mathcal{L}(\boldsymbol{\theta}|v)}{\partial \theta} = \frac{1}{N}\sum_{v\in S}\left[-\mathbb{E}_{\text{donnes}}\left[\frac{\partial E(v,h)}{\partial \theta}\right] + \mathbb{E}_{\text{modle}}\left[\frac{\partial E(v,h)}{\partial \theta}\right]\right] \tag{10}$$

Le modele : $\theta = (W, b, c)$

On remplace $\theta = W$ dans le premier terme de l'quation (8) et on a :

$$-\sum_h p(h \mid v)\frac{\partial E(v,h)}{\partial w_{ij}} = \sum_h p(h \mid v) h_i v_j$$

Par indpendance des variables conditionnelles $(h_i \mid v)$ on a : $p(h \mid v) = \prod_{k=1}^{n} p(h_k \mid v)$ et il s'ensuit que ;

$$-\sum_h p(h \mid v)\frac{\partial E(v,h)}{\partial w_{ij}} = \sum_h \prod_{k=1}^{n} p(h_k \mid v) h_i v_j = \sum_{h_i}\sum_{h_{-i}} p(h_i \mid v) p(h_{-i} \mid v) h_i v_j$$

$$-\sum_h p(h \mid v)\frac{\partial E(v,h)}{\partial w_{ij}} = \sum_{h_i} p(h_i \mid v) h_i v_j \sum_{h_{-i}} \underbrace{p(h_{-i} \mid v)}_{=1}$$

$$-\sum_h p(h \mid v)\frac{\partial E(v,h)}{\partial w_{ij}} = p(H_i = 1 \mid v) v_j = \sigma\left(\sum_{j=1}^{m} w_{ij} v_j + c_i\right) v_j$$

On reprend l'equation (8) et on a :

$$\frac{\partial \ln \mathcal{L}(\theta|v)}{\partial w_{ij}} = -\sum_h p(h|v)\frac{\partial E(v,h)}{\partial w_{ij}} + \sum_{v,h} p(v,h)\frac{\partial E(v,h)}{\partial w_{ij}}$$

$$= \sum_h p(h|v) h_i v_j - \sum_v p(v) \sum_h p(h|v) h_i v_j$$

$$= p(h_i = 1|v) v_j - \sum_v p(v) p(h_i = 1|v) v_j. \tag{11}$$

Moyenne du gradient de la log vraissemblance sur un ensemble d'apprentissage $S = \{\mathbf{v}_1, \ldots, \mathbf{v}_N)$
Les notations suivantes sont souvent utilises :

$$\frac{1}{N}\sum_{v\in S}\frac{\partial \ln \mathcal{L}(\boldsymbol{\theta} \mid v)}{\partial w_{ij}} = \frac{1}{N}\sum_{v\in S}\left[-\mathbb{E}_{p(h|v)}\left[\frac{\partial E(v,h)}{\partial w_{ij}}\right] + \mathbb{E}_{p(h,v)}\left[\frac{\partial E(v,h)}{\partial w_{ij}}\right]\right]$$

$$= \frac{1}{N}\sum_{v\in S}\left[\mathbb{E}_{p(h|v)}[v_i h_j] - \mathbb{E}_{p(h,v)}[v_i h_j]\right] (5) \tag{12}$$

Par analogie on a;

$$\frac{\partial \ln \mathcal{L}(\boldsymbol{\theta} \mid v)}{\partial b_i} = v_j - \sum_v p(v) v_j \tag{13}$$

et

$$\frac{\partial \ln \mathcal{L}(\boldsymbol{\theta} \mid v)}{\partial c_i} = p(H_i = 1 \mid v) - \sum_v p(v) p(H_i = 1 \mid v) \tag{14}$$

Approximation de la log vraissemblance:

L'ide ici est d'viter un calcul trop complexe qui ncessiterait de parcourir toutes les valeurs possibles des variables visibles (complexit exponentielle). Ce problme apparat notamment lorsqu'on calcule le gradient du log-vraisemblance, en particulier pour certains termes spcifiques des quations (5), (6) et (7). Les embeddings rcuprer en sortie de l'encodeur de Blip sont de tailles $(577)(768) = 443136$, donc il faudrait valuer 2^{443136} possibilits. Cel est juste impossible car meme le supercalculateur le plus puissant de notre re ne finirait ce calcul avant des milliers d'annes.

Solution possible

Plutt que d'effectuer ces sommes directement, on peut estimer ces valeurs en prenant des chantillons gnrs partir du modle lui-mme. Pour obtenir ces chantillons, on utilise une mthode appele chantillonnage de Gibbs, qui repose sur une chane de Markov. Cette chane doit tourner assez longtemps pour atteindre un tat stable.

Cependant, mme avec cette mthode, le cot de calcul reste trop lev pour un apprentissage efficace des RBM. C'est pourquoi des approximations supplmentaires sont souvent utilises, comme la CD-k (k-Contrastive Divergence).

$$\mathrm{CD}_k(\boldsymbol{\theta}, \boldsymbol{v}^{(0)}) = -\sum_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v}^{(0)})\frac{\partial E(\boldsymbol{v}^{(0)}, \boldsymbol{h})}{\partial \boldsymbol{\theta}} + \sum_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v}^{(k)})\frac{\partial E(\boldsymbol{v}^{(k)}, \boldsymbol{h})}{\partial \boldsymbol{\theta}}.$$

La divergence contrastive (CD) simplifie l'entranement des RBM en limitant le nombre d'itrations de l'chantillonnage. La chane est initialise avec un exemple d'entranement $v(0)$ et gnre un chantillon $v(k)$ aprs k tapes. chaque tape, on commence par chantillonner $h(t)$ partir de $p(h|v(t))$, puis $v(t+1)$ partir de $p(v|h(t))$. Cette approximation permet d'acclrer le calcul du gradient et l'optimisation du modle.

Algorithm 1 k-step contrastive divergence

Input: RBM $(V_1, \ldots, V_m, H_1, \ldots, H_n)$, training batch S
Output: gradient approximation Δw_{ij}, Δb_j, and Δc_i for $i = 1, \ldots, n$, $j = 1, \ldots, m$

Function `k-step contrastive divergence`$(V_1, \ldots, V_m, H_1, \ldots, H_n, S)$**:**
  for $i = 1, \ldots, n$, $j = 1, \ldots, m$ **do**
    $\Delta w_{ij} = 0$, $\Delta b_j = 0$, $\Delta c_i = 0$
  end
  for *each* $v \in S$ **do**
    $v^{(0)} \leftarrow v$
    for $t = 0, \ldots, k-1$ **do**
      for $i = 1, \ldots, n$ **do**
        sample $h_i^t \sim p(h_i|v^{(t)})$
      end
      for $j = 1, \ldots, m$ **do**
        sample $v_j^{(t+1)} \sim p(v_j|h^{(t)})$
      end
    end
    for $i = 1, \ldots, n$, $j = 1, \ldots, m$ **do**
      $\Delta w_{ij} \leftarrow \Delta w_{ij} + p(H_i = 1|v^{(0)}) \cdot v_j - p(H_i = 1|v^{(k)}) \cdot v_j$ $\Delta b_j \leftarrow \Delta b_j + v_j^{(0)} - v_j^{(k)}$
      $\Delta c_i \leftarrow \Delta c_i + p(H_i = 1|v^{(0)}) - p(H_i = 1|v^{(k)})$
    end
  end

Nous avons introduire la RBM binaire car cest de l que part toutes les variantes existantes de la Machine de Boltzmann Restreinte. Les embeddings obtenues en sortie de lencodeur dimage du modle Blip-Image-Captioning tant des valeurs continues (pas que des 0 et 1), nous allons travailler avec la RBM gaussienne.

7 Quest-ce quune RBM Gaussienne-Binaire?

Cest une **extension** de la RBM binaire qui permet de **traiter des donnes continues** (comme des embeddings, des signaux audio, ou des images en niveaux de gris).

Diffrence cl :

- Dans une RBM **binaire**, les neurones visibles prennent **0 ou 1**.
- Dans une RBM **gaussienne**, les neurones visibles prennent **des valeurs continues**.

Pourquoi cest utile ? Parce que dans ce sujet les embeddings sont **des vecteurs de nombres rels**, donc une RBM gaussienne serait plus adapte quune RBM binaire.

8 Fonctionnement dune RBM-GB

Elle garde la **mme structure** quune RBM classique, mais avec une **modification sur les tats des neurones visibles**.

8.1 Activation des neurones cachs (identique la RBM Binaire-Binaire)

Chaque neurone cach h_j reoit une somme pondre des entres v_i et applique une **sigmode** :

$$P(h_j = 1|v) = \sigma\left(\sum_i w_{ij} v_i + b_j\right)$$

Donc les **neurones cachs restent binaires (0 ou 1)**, comme dans une RBM classique.

8.2 Activation des neurones visibles (modifi pour les valeurs continues)

Dans une RBM binaire, on n'applique pas une sigmode aux **neurones visibles**. Mais ici, on considre quils suivent une **distribution gaussienne** :

$$v_i \sim \mathcal{N}\left(c_i + \sum_j w_{ij} h_j, \sigma^2\right)$$

(σ est un hyperparametre.)
Chaque v_i est un **nombre rel** tir dune **distribution normale (gaussienne)** centre autour de $\sum_j w_{ij} h_j + c_i$, avec une variance σ^2.

Ce que a signifie :

- Au lieu dtre **0 ou 1**, les neurones visibles prennent **des valeurs relles**.
- On peut interprter a comme une **reconstruction bruite, dcompresse ou raffine** des embeddings.

9 Entranement dune RBM gaussienne

Lapprentissage suit la mme logique quune RBM binaire :

1. On passe une **donne relle** dans la RBM.
2. On **calcule les activations caches** avec la sigmode.
3. On **reconstruit les neurones visibles** avec une distribution gaussienne.
4. On ajuste les poids avec le **Contraste de Divergence (CD-k)**.

Petite diffrence mais importante : Comme les neurones visibles sont continus, la mise jour des poids prend en compte la variance σ^2, ce qui change un peu les quations dapprentissage.
(cette partie est completer par les nouvelles quations et l'algo proprement dit de la RBM gaussienne.)

10 Une fois entrane, comment on utilise la RBM gaussienne ?

Aprs lentranement, on peut utiliser la RBM gaussienne pour :

- **Transformer des embeddings** : on donne un **vecteur dembeddings** en entre, et la RBM gnre une **version transforme** des embeddings partir des activations caches.
- **Gnrer de nouvelles donnes** : on initialise un vecteur au hasard et on le fait passer plusieurs fois dans la RBM, ce qui produit des donnes similaires celles du dataset dentranement.
- **Extraire des features** : on ne garde que les activations caches h comme **reprsentation compacte** des donnes visibles.

C'est le premier cas d'utilisation qui fait l'objet de ce sujet.

11 Limites dune RBM gaussienne

- Peut tre difficile entraner car la variance doit tre bien rgle.
- Complexit computationnelle (paralllisation recommande)
- Moins populaire car souvent remplace par des autoencodeurs variationnels (VAE).
- Dans le cas de donnes complexes (distributions complexes),la RBM Gaussienne peut s'avrer impuissante car elle suppose que les donnes suivent une gaussienne.

References

[1] Noam Shazeer Ashish Vaswani et al. Attention is all you need. *https://arxiv.org/abs/1706.03762*, 2023.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[3] Yul Choi et al. Spatialvlm: Integrating spatial reasoning with vision-language models. *arXiv preprint arXiv:2105.08752*, 2021.

[4] Simon Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1998.

[5] Jialei Hou et al. Vila: Vision-and-language pre-training with large-scale datasets. *arXiv preprint arXiv:2103.12824*, 2021.

[6] Haesu Hwang et al. Vlm-rl: Visual-language models for reinforcement learning. *arXiv preprint arXiv:2004.06592*, 2020.

[7] Wonjae Kim et al. Vilt: Vision-and-language transformer without convolution or region supervision. *arXiv preprint arXiv:2102.03334*, 2021.

[8] Junnan Li et al. Blip: Bootstrapping language-image pre-training. *arXiv preprint arXiv:2201.12086*, 2022.

[9] Luowei Li et al. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019.

[10] Wei Liu et al. Xmodel-vlm: Efficient cross-modal representation learning. *arXiv preprint arXiv:2104.06923*, 2021.

[11] Alec Radford et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

[12] Paul Smolensky. Information processing in dynamical systems: Foundations of harmony theory. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1:194–281, 1986.

[13] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019.

[14] Yao-Hung Hubert Tsai et al. Multimodal transformer for unaligned multimodal language sequences. *arXiv preprint arXiv:1906.00295*, 2019.