Assessing and Efficiently Leveraging the Generalisation Abilities of Multimodal Models

Romain Bielawski

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 3 - Paul Sabatier

Présentée et soutenue par

Romain BIELAWSKI

Le 5 décembre 2022

Evaluation et Utilisation Efficace des Capacités de Généralisation des Modèles Multimodaux

---

Ecole doctorale : **EDMITT - Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse**

Spécialité : **Informatique et Télécommunications**

Unité de recherche :
CERCO - Centre de Recherche Cerveau et Cognition

Thèse dirigée par
Rufin VANRULLEN et Tim VAN DE CRUYS

Jury

Mme Elisabeth ANDRé, Rapporteure
M. Emmanuel DELLANDRéA, Rapporteur
M. Rufin VANRULLEN, Directeur de thèse
M. Tim VAN DE CRUYS, Co-directeur de thèse
Mme Justine CASSELL, Présidente

Avant de commencer, permettez-moi de vous dire que je suis ravi d'avoir terminé cette thèse. Il était temps. Et maintenant, je vais devoir remercier les gens qui m'ont aidé à en arriver là.

Mes parents. Merci de m'avoir donné la vie, et merci de m'avoir donné de bonnes raisons de ne pas la quitter quand j'ai commencé à travailler sur cette thèse. Votre soutien moral et financier pendant mes longues années d'études a été très apprécié.

Titouan. Merci de m'avoir laissé la plus grande chambre dans la coloc. Merci de m'avoir supporté pendant le confinement. Merci d'avoir écouté mes interminables élucubrations théoriques (je ne parle pas d'informatique).

Mon directeur de thèse, Rufin. Merci d'un jour être entré dans le bureau où j'effectuais mon stage de fin d'études, me questionnant sur mon avenir, et de m'avoir dit : « J'ai une thèse. Tu veux une thèse ? » J'ai failli refuser. Merci pour nos immanquables discussions hebdomadaires, pour m'avoir laissé suivre mes intuitions et mes envies, et – c'est ce qui fait de toi un directeur de thèse exceptionnel – pour avoir toujours su comment me guider et me conseiller. Merci pour le wakeboard, le beach-volley et les barbecues. Merci pour l'inspiration capillaire.

Mon autre directeur de thèse, Tim. Merci pour ta gentillesse, tes conseils, et merci d'avoir continué à m'aider après ton déménagement en Belgique.

Evidemment, Benjamin. Premier auteur de mon premier article scientifique publié. Merci d'avoir travaillé avec moi pendant presque trois ans. Sans toi, j'aurais sans doute beaucoup plus cherché et beaucoup moins trouvé. Et qui aurait pu prendre si bien soin de notre chère Anita ? Merci d'avoir été là jusqu'à la fin de cette épreuve.

Enfin, merci à tout le CerCo pour son accueil, pour le télétravail, et surtout pour les mémorables écoles d'été et d'hiver. Merci à tous mes anciens collègues, dont certains sont aujourd'hui des amis. Merci à mes autres amis, j'espère ne jamais vous avoir comme collègues.

Voilà, mes chers compatriote, mes chers collègues, mes chers amis, mes chers ennemis (s'il y en a), mes chères ex, mes chers inconnus qui sont arrivés ici par hasard, merci de m'avoir accompagné dans cette aventure et d'avoir lu autant de fois le mot « merci » sans vous lasser. Et si vous décidez de vous priver du plaisir de lire ce manuscrit en entier, ce n'est pas grave. On ne peut pas tout avoir dans la vie. En tout cas, moi j'ai un doctorat. C'est pas trop tôt.

Abstract

As larger multimodal datasets are becoming available on the web, the possibility for better, more human-like multimodal models grows. My research goal is to evaluate what multimodality brings to machine representation of data, especially when it comes to generalizing in one or two modalities (image and/or text), as well as to find ways of improving the quality of the latent space of multimodal algorithms. Bigger datasets and larger computational power enable better algorithms to be developed, but in this project, I aim at using as little data as possible, with as few annotations as possible, to improve the multimodal representation of pretrained algorithms.

There has been great progress in multimodal dataset availability, mostly due to the possibility of extracting information from big unstructured data on the web. The attention networks, originally designed for text only, have proven successful in their capacity for merging data. Most recently, the contrastive learning objective applied on hundreds of millions of annotated images has provided State-of-the-Art (SOTA) results. However, the standard methods and evaluations in the multimodal field have two shortcomings: The generalisation abilities of models trained multimodally are yet to be determined and there is no computationally cheap way, both in terms of data and power, to improve or leverage the latent space abilities of these cost-expensive algorithm on a tasks such as image captioning

In this thesis, after an introductory chapter on the unimodal and the multimodal field (Chapter 3), the first shortcoming is addressed by our evaluation tasks, that can be applied to other networks in order to compare the generalisation ability of any image and/or text model, and that are presented in Chapter 4 and 5. Part of the

second issue is dealt with using our Latent CycleGAN in Chapter 6, which is very cost-effective, and which improves a more straightfoward captioning pipeline with unmatched multimodal data.

Contents

List of Figures

List of Tables

Acronyms

AI Artificial Intelligence. 19, 21, 31, 35, 41, 43, 139, 159, 164

CNN Convolutional Neural Network. 7, 50, 51, 62, 86

CV Computer Vision. 19, 41, 47

GAN Generative Adversarial Network. 11, 13, 51, 61–63, 71, 147, 149

LSTM Long Short-Term Memory. 7, 44–47, 59, 60, 82

NLP Natural Language Processing. 19–21, 41, 42, 47

RDM Representational Dissimilarity Matrix. 105–108

RNN Recurrent Neural Network. 11, 44, 46, 61, 62, 86

RPN Region Proposal Network. 7, 50, 51, 54

RSA Representational Similarity Analysis. 104

SOTA State-of-the-Art. 4, 18, 20, 21, 27, 28, 31–33, 38, 41, 42, 55, 57, 59, 75, 83, 122, 123, 131, 132, 144, 153, 158–161

VQA Visual Question Answering. 51, 54, 60

Chapter 1

Résumé en Français

1.1 Aperçu

À mesure que de plus grands ensembles de données multimodaux deviennent disponibles sur le Web, la possibilité de développer de meilleurs modèles multimodaux, plus humains, augmente. Mon objectif de recherche est d'évaluer ce que la multimodalité apporte à la représentation des données par les machines, notamment lorsqu'il s'agit de généraliser dans une ou deux modalités (image et/ou texte), ainsi que de trouver des moyens d'améliorer la qualité de l'espace latent des algorithmes multimodaux. De plus grands ensembles de données et une plus grande puissance de calcul permettent certes de développer de meilleurs algorithmes, mais dans ce projet, je vise à utiliser le moins de données possible, avec le moins d'annotations possible, pour améliorer la représentation multimodale d'algorithmes préentraînés.

De grands progrès ont été faits en ce qui concerne la disponibilité des ensembles de données multimodaux, principalement en raison de la possibilité d'extraire des

informations à partir de données volumineuses, non structurées, sur le Web. Les réseaux attentionnels, conçus à l'origine uniquement pour le texte, ont fait leurs preuves dans leur capacité à fusionner les données. Plus récemment, l'objectif d'apprentissage contrastif appliqué sur des centaines de millions d'images annotées a fourni des résultats State-of-the-Art (SOTA). Cependant, les méthodes et les évaluations standards dans le domaine multimodal présentent deux lacunes :

- Les capacités de généralisation des modèles formés de manière multimodale restent à déterminer
- Il n'existe aucun moyen de calcul bon marché, à la fois en termes de données et de puissance, pour améliorer ou exploiter les capacités des espaces latents de ces algorithmes sur des tâches telles que la description d'images.

Dans cette thèse, la première lacune est abordée par nos tâches d'évaluation, qui peuvent être appliquées à d'autres réseaux afin de comparer la capacité de généralisation de n'importe quel modèle d'image et/ou de texte. Une partie du deuxième problème est traitée à l'aide de notre CycleGAN Latent (Latent CycleGAN), qui est très rentable et qui améliore une méthode de description plus simple avec des données multimodales non-appairées.

1.2 Apports de cette thèse

La plupart des modèles d'apprentissage profond multimodaux sont conçus sans tenir compte de ce qui a été réellement « appris » par le modèle. En effet, lorsqu'un modèle a été entraîné sur des centaines de millions d'échantillons, il est évalué

sur certaines tâches standards, et s'il fonctionne bien, alors le modèle est dit satisfaisant. C'est une façon légitime d'évaluer la qualité d'une représentation, mais dans cette thèse, nous voulons sonder plus profondément le cerveau des machines. C'est-à-dire que nous allons évaluer de manière plus fine les capacités de généralisation des modèles multimodaux (versus unimodaux). De plus, lorsqu'un modèle est utilisé pour une tâche en aval, la quantité de données et d'annotations nécessaires au finetuning (affinement des paramètres du réseau) est souvent considérée comme non pertinente, tant que la tâche peut être effectuée avec le meilleur score possible. Nous voulons également dans cette thèse maximiser le rapport coût-efficacité en tirant parti des propriétés de l'espace latent obtenues avec un préentraînement déjà coûteux en calcul, en développant des méthodes qui nécessitent peu de données et peu d'annotations. Pour ces deux objectifs, nous allons utiliser un algorithme bimodal qui est devenu, dès sa sortie, un classique : CLIP.

1.3 Des modèles unimodaux aux modèles multimodaux

L'expérience humaine est essentiellement multimodale. Pourtant, la plupart des algorithmes actuels ne peuvent traiter qu'un seul type de données. Cela donne les différents domaines de l'Artificial Intelligence (AI) : Computer Vision (CV), Natural Language Processing (NLP)... Dans ces domaines, les algorithmes sont entraînés sur des jeux de données unimodaux (MNIST [Deng, 2012], ImageNet [Deng et al., 2009] , CIFAR [Krizhevsky et al.,] pour la CV ; Wikipedia, Com-

mon Crawl, Book Corpus pour le NLP), généralement avec une seule tâche d'apprentissage. Parfois, leur compréhension est spécialisée pour cette seule tâche et ne peut pas être transférée à une autre – ou avec de très mauvais résultats. De nos jours, la plupart des algorithmes SOTA sont en fait préentraînés sur un grand ensemble de données (plusieurs millions d'échantillons) et sur une tâche générale (reconnaissance ou segmentation d'objets pour la vision [Kolesnikov et al., 2019, He et al., 2015], inférer un mot manquant dans une phrase ou évaluer si deux phrases se succèdent dans un texte pour le langage [Devlin et al., 2018]) qui leur permettent de comprendre de nombreux aspects de la modalité. Après ce préentraînement, les algorithmes sont ensuite affinés (fine-tuned), pour spécialiser leur représentation, pour d'autres tâches plus spécifiques. De grands ensembles de données unimodales sont disponibles depuis des années, permettant aux scientifiques de développer des modèles avec des caractéristiques robustes, qui peuvent généraliser à de nombreuses tâches dans leur modalité.

Si nous voulons aller plus loin dans la création d'algorithmes capables d'apprendre des fonctionnalités qui s'adaptent à diverses conditions – tout comme l'homme peut le faire – nous devons introduire de la multimodalité. Ce n'est qu'une étape sur la voie de la création de machines intelligentes comme les robots. [Bisk et al., 2020] expose les étapes passées et futures en vue de la création d'une intelligence de type humain, en partant d'une perspective de traitement du langage naturel. Elles sont au nombre de cinq, et sont appelées World Scopes. Chacune constitue une extension des composantes du monde auxquels un algorithme peut accéder afin de les traiter conjointement avec les autres. Ce sont les suivantes :

- WS1: Corpus *(Notre passé)*

- WS2: Internet *(La plupart du NLP récent)*
- WS3: Perception *Le NLP multimodal)*
- WS4: Incarnation
- WS5: Social

Les deux premiers World Scope sont dans le domaine du NLP unimodal. Les modèles SOTA NLP actuels ont un World Scope qui s'arrête à l'étape 2, ce qui signifie qu'ils sont entraînés avec un grand ensemble de données textuelles extraites du Web.

Le domaine de l'AI évolue désormais vers la multimodalité. La réalisation de cette étape aboutira à des modèles capables d'ancrer leur représentation linguistique dans d'autres domaines perceptifs, conduisant à une compréhension plus riche, plus subtile et plus robuste du monde.

Au-delà de ce point, nous entrons dans la science-fiction. Le World Scope incarnation correspond à la robotique, qui est maintenant principalement un domaine distinct, avec peu d'interactions (mais celles-ci augmentent) avec l'AI [Mehlmann et al., 2014] multimodale. Un corps physique rendra l'AI plus humaine ou animale, avec une proprioception et peut-être des sensations positives et négatives comme la douleur et le plaisir. La portée sociale correspond au moment où le modèle va apprendre à interagir directement avec d'autres êtres sociaux, avec des notions de réactions émotionnelles, d'empathie, de hiérarchie sociale, et à développer des comportements sociaux tel le regard social [Bee et al., 2010, Cassell et al., 1994, Cassell and Thórisson, 1999].

La disponibilité de grands jeux de données multimodaux étant très récente, ce n'est que maintenant que nous pouvons créer des modèles multimodaux comparables, en terme de nombre d'échantillons d'apprentissage [Lai,], à des modèles unimodaux (voir les chapitres 4 et 5). Avant cela, l'ensemble d'apprentissage de l'algorithme multimodal était très restreint (COCO [Lin et al., 2014], Conceptual Caption [Sharma et al., 2018a], LIRIS-ACCEDE [Baveye et al., 2015]) et bien que les modèles puissent obtenir de bons résultats sur certaines tâches, les fonctionnalités multimodales apprises par eux n'étaient souvent pas utilisables pour d'autres tâches [Devillers et al., 2021]. Ce type de modèles comprend : Virtex [Desai and Johnson, 2020], SimVLM [Wang et al., 2021], Frozen [Tsimpoukelli et al., 2021], BUTD [Anderson et al., 2017] et Dall-E 2 [Ramesh et al., 2022] entre autres.

1.4 Problématique

Lors de l'évaluation des capacités de généralisation des modèles multimodaux, il convient de déterminer sur quelle tâche et dans quel cadre celles-ci s'effectuent. La comparaison que nous voulons effectuer, car nous la considérons comme la comparaison de base, est entre les modèles unimodaux et multimodaux. Pour ce faire, nous voulons des modèles multimodaux capables de représenter des données unimodales avec un minimum de dégradation lors de l'extraction des caractéristiques. Cela nous guide vers des modèles de représentation coordonnés, où deux pipelines unimodales s'informent mutuellement lors de l'apprentissage, mais où à l'inférence, chaque pipeline est en fait indépendante, ce qui n'est pas le

cas dans les modèles de fusion, où il faut des échantillons issus des deux modalités pour créer une représentation fonctionnelle. C'est pourquoi CLIP sera si central dans notre thèse.

De plus, nous verrons avec MirrorGAN que la cohérence de cycle peut être utilisée dans un cadre multimodal. Même si dans le cas de MirrorGAN, les données sont appairées, nous verrons dans l'article CycleGAN (qui est conçu pour la traduction unimodale entre deux distributions d'images) qu'en fait, les données appariées ne sont pas nécessaires et que le principe de cohérence du cycle fonctionne en lui-même. Ces deux modèles nous aideront à concevoir un modèle de traduction multimodale dans l'espace latent de CLIP, qui nous permettra d'effectuer de la description d'images avec un apprentissage non supervisé, en tirant parti de la multimodalité déjà présente dans les représentations latentes de CLIP.

1.5 Evaluation des capacités de généralisation sur des tâches visuelles standards

L'apprentissage des modèles de vision à l'aide de la supervision linguistique a gagné en popularité [Quattoni et al., 2007, Srivastava et al., 2012, Frome et al., 2013, Joulin et al., 2016b, Pham et al., 2019, Desai and Johnson, 2020, Hu and Singh, 2021, Radford et al., 2021, Sariyildiz et al., 2020a] pour deux raisons principales : premièrement l'entraînement visio-linguistique permet de créer des ensembles de données d'entraînement massifs à partir de données en ligne facilement disponibles, sans annotation manuelle ; deuxièmement, le langage fournit des

informations sémantiques supplémentaires qui ne peuvent pas être déduites à partir d'ensembles de données uniquement visuels, ce qui pourrait aider à l'ancrage sémantique des caractéristiques visuelles.

Récemment [Radford et al., 2021] a introduit CLIP, un modèle de langage et de vision qui montre des capacités d'apprentissage instantanées (zero-shot) exceptionnelles sur de nombreuses tâches et des capacités d'apprentissage par transfert (transfer learning) convaincantes. Un rapport récent [Goh et al., 2021] a montré que CLIP produit des schémas de sélectivité neuronale comparables aux cellules conceptuelles « multimodales » observées dans le cerveau humain [Quiroga et al., 2005, Reddy and Thorpe, 2014]. À partir de ces résultats, il est tentant de supposer que les propriétés de généralisation du CLIP découlent de l'ancrage sémantique fourni par la formation conjointe vision-langage.

Dans cette thèse, nous montrons que CLIP et d'autres modèles de langage de vision ne fonctionnent pas mieux que les modèles de vision uniquement, entièrement supervisés sur un certain nombre de paramètres de généralisation et d'ensembles de données. L'analyse de la similarité des représentations [Kriegeskorte et al., 2008] révèle que les représentations multimodales qui émergent à travers l'apprentissage du langage visuel sont différentes *à la fois* des représentations linguistiques et visuelles et donc peut-être inadaptées à l'apprentissage par transfert pour nouvelles tâches visuelles. En conclusion, des travaux supplémentaires sur les fondements linguistiques sont encore nécessaires, s'il s'agit d'améliorer les capacités de généralisation des modèles de vision.

1.6 Evaluation des capacités de généralisation sur des tâches centrées sur l'humain

La préformation du langage de la vision dans les réseaux de neurones gagne en popularité en raison de l'intérêt croissant pour les tâches multimodales telles que le Visual Question Answering (Réponse à des question visuelles) ou la description d'images [Anderson et al., 2017, Lu et al., 2019, Li et al., 2019, Singh et al., 2019], mais aussi de la disponibilité de ressources en ligne qui permettent de construire des ensembles de données d'entraînement à grande échelle sans annotations manuelles [Radford et al., 2021, Jia et al., 2021]. En théorie, entraîner un modèle sur des données multimodales devrait permettre d'améliorer sa représentation des données de chacune des modalités. Pour un modèle image-texte, par exemple, les caractéristiques de l'image pourraient être enrichies par l'abstraction des données linguistiques – la propriété d'ancrage sémantique –, et inversement, les caractéristiques linguistiques pourraient gagner en information grâce à l'ancrage visuel [Harnad, 1990].

Malheureusement, cela ne se produit pas toujours dans la pratique.

Récemment, [Devillers et al., 2021] a évalué les capacités de généralisation visuelle de CLIP [Radford et al., 2021], un réseau populaire entraîné avec un objectif d'apprentissage contrastif sur plus de 400 millions de paires de légendes d'images extraites du Web, et d'autres modèles multimodaux [Sariyildiz et al., 2020b, Desai and Johnson, 2020]. Ils ont montré que pour les tâches de classification d'objets standard (par exemple, classification de chiffres, d'articles de mode ou d'images naturelles), les réseaux multimodaux comme

CLIP étaient sous-performants par rapport à d'autres modèles unimodaux (vision uniquement) comme BiT-M [Kolesnikov et al., 2019] dans l'apprentissage par transfert, l'apprentissage avec peu d'exemples et l'apprentissage non supervisé. Ici, nous revisitons cette question en utilisant des ensembles de données se concentrant sur des concepts plus « centrés sur l'humain ».

L'apprentissage humain implique généralement une interaction avec des données multimodales. Ainsi, on pourrait s'attendre à ce que les représentations CLIP des images et du texte soient d'une certaine façon plus proches des représentations humaines que celles apprises par les modèles unimodaux. De plus, étant donné que CLIP a été entraîné sur des paires image-description provenant de diverses sources sur Internet (y compris les réseaux sociaux), nous pouvons supposer qu'une partie importante de ses légendes d'entraînement a été écrite par des humains pour d'autres humains. Ceci est différent des ensembles de données de vision standard, dans lesquels les étiquettes ou les annotations sont parfois générées par l'homme (par exemple via le Mechanical Turk d'Amazon), mais toujours produites à des fins d'apprentissage automatique. Encore une fois, cette différence devrait rapprocher les représentations de CLIP des représentations humaines par rapport aux modèles unimodaux. Ainsi, il devrait exister au moins *quelques tâches spécifiques* pour lesquelles l'entraînement multimodal de CLIP offre des avantages par rapport aux modèles unimodaux. Par exemple, considérons la tâche consistant à attribuer un genre à un film en fonction de son affiche et de son titre. Cela nécessite de récupérer des informations fines sur, entre autres, l'aspect artistique, émotionnel ou stylistique d'une image ou d'un texte (ou les deux). Cela ne peut être correctement réalisé que si la formation du modèle offrait une exposition appropriée à ces concepts centrés sur l'humain. Ici, nous utilisons le terme *centré sur l'humain*

chaque fois qu'un concept fait référence à des composantes culturelles, sociales, esthétiques et/ou affectives du monde.

Nous ferons donc l'hypothèse que CLIP devrait être plus performant que les modèles unimodaux dans les tâches de généralisation impliquant des concepts centrés sur l'humain. Nous évaluerons cette hypothèse sur trois tâches impliquant de tels concepts centrés sur l'humain : l'analyse des sentiments sur les tweets ; la classification des genres de livres; la classification de genre des films. Toutes les tâches peuvent être effectuées sur la base de données visuelles (images), de données textuelles (tweet, titre de livre ou de film, résumé de l'intrigue du film), ou les deux. Pour la classification des genres de films, nous introduirons un nouveau jeu de données multimodal à grande échelle obtenu par un balayage sur The Movie Database (TMDb). Comme détaillé ci-après, nous constaterons que CLIP surpasse les modèles unimodaux dans la classification visuelle et textuelle, ainsi que les combinaisons par paires de ces modèles unimodaux dans le cas de la classification multimodale (image + texte). Par conséquent, CLIP établit un nouveau SOTA sur ces tâches.

1.7 Amélioration de la description d'image grâce au CycleGAN Latent

La multimodalité gagne en popularité grâce aux ressources en ligne récemment disponibles qui permettent la création d'énormes ensembles de données visio-linguistiques [Jia et al., 2021]. De nombreux modèles ont été créés pour effectuer des tâches bimodales spécifiques telles que le Visual Question Answering ou

la description d'images [Anderson et al., 2017, Lu et al., 2019, Li et al., 2019, Singh et al., 2019], mais certains ont été conçus avec un objectif plus général : produire un espace vectoriel latent multimodal où les images et le texte peuvent être représentés et comparés. Parmi ces modèles, CLIP – un algorithme formé avec un objectif contrastif multimodal sur un grand ensemble de données (400 millions d'échantillons) de paires de légendes et d'images – a montré d'impressionnantes capacités d'apprentissage instantané (zero-shot learning) [Radford et al., 2021]. Ce modèle a récemment été testé sur des tâches pour lesquelles il n'était pas initialement entraîné, comme l'apprentissage par transfert (tranfer learning) et l'apprentissage avec peu d'exemples (few-shot learning) sur des ensembles de données unimodaux et multimodaux, ou la description d'images, établissant de nouveaux résultats SOTA sur certaines tâches [Bielawski et al., 2022, Mokady et al., 2021].

Dans le cas spécifique de la description d'images, de nombreuses études utilisent des modèles préentraînés pour l'encodage des caractéristiques de l'image ainsi que pour la génération de texte. Cependant, une étape d'ajustement de bout en bout de l'image à la légende est généralement requise pour aligner les représentations visuelles et linguistiques de manière supervisée sur un jeu de données de légende et d'image appariées [Chen et al., 2021, Fang et al., 2021, Zhou et al., 2019]. Il existe une exception évidente à cette règle : lorsque le préapprentissage du modèle a déjà aligné les caractéristiques du texte et de l'image – comme dans le cas de CLIP. Par conséquent, nous viserons ici à tirer parti de cette propriété en implémentant une pipeline de description qui n'utilise pas de données appairées.

Nous formerons d'abord un "décodeur de texte CLIP" pour reconstruire des

légendes basées sur leurs représentation dans l'espace latent de CLIP (un objectif linguistique unimodal) ; ce décodeur de texte sera ensuite figé. Par conséquent, nous comparerons une pipeline de description directe – alimentant le décodeur de texte avec des caractéristiques (features) extraites par CLIP afin de générer une légende – avec une pipeline où un traducteur inspiré de CycleGAN [Zhu et al., 2017] – formé avec uniquement des fonctionnalités visuelles et textuelles non appariées – est utilisé pour convertir des caractéristiques extraites d'images en caractéristiques de texte avant de les transmettre au décodeur de texte. Même si l'espace latent de CLIP est déjà préentraîné avec une approche de force brute pour aligner ses représentations visuelles et linguistiques sur 400 millions de paires de légendes et d'images, nous démontrerons que notre modèle de conversion de caractéristiques formé à l'aide de la cohérence de cycle dans l'espace latent de CLIP améliore considérablement les performances de description par rapport à la méthode directe.

1.8 Bilan

Dans cette thèse, je présente plusieurs façons d'évaluer et d'exploiter efficacement les capacités des modèles multimodaux. Il semble que les types actuels d'entraînement multimodaux apportent des informations supplémentaires à la représentation de chaque modalité, mais cela est également préjudiciable à certains autres égards. Passons en revue quelques avantages et inconvénients de l'entraînement multimodal mis en évidence dans cette thèse.

La première étude (au chapitre 4) montre que l'entraînement multimodal n'apporte aucun avantage par rapport à un simple entraînement visuel en matière

de détection d'objets [Devillers et al., 2021]. Cela n'aide pas non plus pour la robustesse des modèles. C'est surprenant, car on pourrait attendre deux choses de la multimodalité :

- Qu'elle améliore la généralisation dans les tâches de vision grâce aux informations sémantiques incorporées dans le domaine visuel
- Qu'elle améliore la robustesse grâce à la segmentation sémantique (qui évite par exemple de confondre un chien et un camion sur une image car ils sont sémantiquement très dissemblables)

Ce qui n'est pas le cas pour les jeux de données visuels standard et ni les attaques ciblées ni non ciblées.

Un début d'explication vient de l'étude des espaces de représentation textuel, visuel et bimodal ; les modèles bimodaux ne se situent pas entre les représentations textuelles et visuelles. Ils constituent leur propre domaine, comme s'ils avaient été formés sur une troisième modalité qui n'est ni la vision ni le langage.

Par conséquent, cette modalité doit être étudiée et évaluée. Les modèles multimodaux sont sous-performants sur les tâches visuelles standard, mais offrent en fait des performances améliorées sur ce que nous avons appelé des tâches « centrées sur l'humain », du moins pour CLIP, dans des contextes textuels, visuels et bimodaux. Cela signifie que – probablement en partie étant donné que les ensembles de données multimodaux sont issus d'Internet, où se produisent des interactions interhumaines réelles – les modèles multimodaux tels que CLIP sont plus efficaces lorsque le monde humain est pris en compte et moins lorsque la tâche est orientée objet.

Comment tirer efficacement parti du préentraînement multimodal ?

Une formation multimodale nécessite beaucoup de données annotées pour être compétitive avec d'autres modèles unimodaux sur des tâches standards. Afin de générer des représentations correctes, CLIP a dû être entraîné sur 400 millions de paires image-légende. Cet énorme ensemble de données permet au modèle d'apprendre des fonctionnalités qui peuvent généraliser – parfois mieux, parfois moins bien que leur homologue unimodal – et que nous devrions pouvoir utiliser sans avoir à les réentraîner sur un autre ensemble de données annotées. C'est ce que nous avons essayé de réaliser avec notre CycleGAN Latent, en créant des pipelines de description qui fonctionnent sans avoir été formés sur des données correspondantes (en dehors de la préformation de CLIP) – une telle pipeline ne fonctionnait pas avec deux modèles unimodaux distincts, ce qui signifie que la multimodalité est essentielle dans ce contexte.

Cependant, nous restons loin du SOTA, ce qui signifie que l'espace latent de CLIP ne peut pas être exploité à un coût de calcul très faible avec des performances compétitives à l'heure actuelle. Inventer des méthodes d'entraînement non supervisées qui génèrent des performances SOTA en tirant parti de modèles préentraînés est un autre défi pour le champ de l'AI multimodale.

Limites

Les résultats de nos études se limitent à une sous-partie des modèles multimodaux et unimodaux. Nous nous sommes principalement concentrés sur CLIP du côté

multimodal, sur les ResNets et les transformers d'images entraînés pour la classification des images du côté visuel, et sur BERT pour le côté textuel.

Cependant, ces modèles représentent le SOTA actuel dans leurs domaines respectifs. Ils sont également standard en termes d'architecture, en termes d'objectif de formation et en termes d'ensemble de données de formation. Il n'est pas possible d'explorer de manière exhaustive toutes les architectures, méthodes et prétraitements de données qui existent. Nous avons sélectionné nos algorithmes par le fait qu'ils avaient l'utilisation la plus large et les meilleures performances dans les métriques standard, ce qui signifie qu'ils représentent l'état actuel du domaine. Notre objectif était d'explorer les possibilités des modèles SOTA, et la conclusion que nous tirons sur les modèles multimodaux par rapport aux modèles unimodaux ne vaut que pour les modèles actuels. À long terme, nous sommes d'accord avec [Bisk et al., 2020], qui dit essentiellement que les futurs modèles aura *besoin* d'être multimodaux – une sorte de multimodal qui n'existe pas encore – afin d'être compétitifs.

En ce qui concerne notre méthode non supervisée de description d'images à l'aide de CLIP, une critique évidente pourrait être que nous sommes loin d'atteindre les performances SOTA. Et en effet, nous ne pouvons pas nous attendre à rivaliser avec les méthodes d'entraînement supervisée, en particulier avec un petit ensemble de données comme COCO.

Notre objectif ici était de démontrer qu'un entraînement non supervisée pouvait conduire à de meilleurs résultats qu'une simple pipeline branchée sur un espace latent multimodal – une conclusion secondaire étant que l'espace latent de CLIP n'est en fait pas entièrement multimodal. Cela pourrait également signifier qu'avec la prochaine génération d'algorithmes multimodaux, la pipeline simple pourrait

gagner en efficacité. Cela pourrait également signifier que notre méthode non supervisée pourrait se rapprocher du SOTA en l'appliquant simplement à un espace multimodal futur.

Néanmoins, nous avons présenté une preuve de concept pour un CycleGAN Latent multimodal. On peut maintenant essayer d'entraîner un tel algorithme sur des données encore plus distinctes, éventuellement de distribution différente – notre ensemble de données d'entraînement étant la version non-appairée des légendes et des images de COCO.

Travaux futurs

Comme indiqué ci-dessus, la première chose à faire avec le CycleGAN Latent est de l'entraîner sur un autre jeu de données multimodal non-appairé, tel que Conceptual Caption par exemple. Nous n'avons pas eu le temps de le faire, mais cela donnerait probablement des résultats légèrement meilleurs que les nôtres. Surtout étant donné qu'une limitation de l'algorithme pourrait être les capacités de généralisation du décodeur de texte, qui est uniquement formé sur les légendes de l'ensemble de données – et il y a plus de données textuelles dans Conceptual Caption (413 915 légendes dans COCO contre 3,3 millions dans Conceptual Caption).

Bien sûr, avoir plus de données des deux modalités conduira sûrement à de meilleurs résultats pour le CycleGAN lui-même.

Mais ce n'est qu'une première étape dans l'utilisation du CycleGAN Latent dans un contexte différent. Tout d'abord, on peut l'utiliser sur un jeu de données multimodal (non-appairé) où le texte n'est pas (seulement) composé des légendes

des images, comme WIT [Srinivasan et al., 2021], où les images sont extraites des pages Wikipédia, accompagnées de leur description, du titre de la page et de son paragraphe d'introduction. Deuxièmement, on pourrait essayer d'aller encore plus loin et de rassembler autant d'images que possible, autant d'échantillons de texte que possible et essayer le CycleGAN Latent sur un tel ensemble de données, où les distributions des images et du texte n'ont aucun rapport. Avec suffisamment de données, le CycleGAN Latent pourrait donner des résultats intéressants.

En ce qui concerne l'évaluation des capacités de généralisation des modèles multimodaux, il y a deux perspectives que j'aurais explorées si j'avais eu le temps – et peut-être l'aurai-je.

La première est la robustesse des modèles multimodaux. Elle a été explorée dans le premier article présenté ici pour le domaine visuel, et il montre que, contrairement à ce que l'on pouvait attendre, l'ancrage sémantique n'apporte pas de robustesse supplémentaire. Cependant, il serait intéressant de regarder les attaques ciblées – où le but de l'attaque est de tromper le réseau en le forçant à déduire une classe spécifique, qui n'est évidemment pas celle de l'image qui est présentée. Si l'ancrage sémantique était en quelque sorte efficace, il devrait être plus difficile de confondre un chien et un avion plutôt qu'un chien et un chat, en raison de la distance sémantique entre les concepts. Cette hypothèse peut être testée avec les résultats des attaques présentés dans [Devillers et al., 2021] avec un peu d'analyse. Le taux de réussite des attaques ciblées devrait diminuer avec la distance sémantique entre la vraie classe et la mauvaise.

La deuxième perspective serait une comparaison similaire à celle du premier article, mais du côté textuel, avec des tâches textuelles standard. En effet, notre conclusion concerne principalement le domaine visuel, car la modalité textuelle

n'a été introduite qu'avec les tâches centrées sur l'humain du deuxième article. Il n'est pas possible d'affirmer que le côté langage de CLIP ne dépasserait pas le modèle de texte unimodal sur les tâches standard. C'est même probable, car les tâches textuelles standard ne peuvent pas vraiment être considérées comme orientées objet, contrairement aux tâches visuelles standard.

Conclusion

La multimodalité dans l'Artificial Intelligence n'en est qu'à ses débuts. Le domaine doit maintenant inventer de nouvelles méthodes d'apprentissage qui prennent en compte ce que les modèles unimodaux peuvent faire et comment ces performances sont obtenues, pour créer des modèles qui surpassent les modèles unimodaux dans tous les domaines. Une fois ces modèles conçus, une autre tâche consiste à ajuster l'espace latent qu'ils génèrent afin qu'ils puissent être utilisés efficacement pour diverses tâches, afin de maintenir les coûts de calcul et de données en aval à un faible niveau.

Chapter 2

Introduction

2.1 Overview

As larger multimodal datasets are becoming available on the web, the possibility for better, more human-like multimodal models grows. My research goal is to evaluate what multimodality brings to machine representation of data, especially when it comes to generalizing in one or two modalities (image and/or text), as well as to find ways of improving the quality of the latent space of multimodal algorithms. Bigger datasets and larger computational power enable better algorithms to be developed, but in this project, I aim at using as little data as possible, with as few annotations as possible, to improve the multimodal representation of pretrained algorithms.

There has been great progress in multimodal dataset availability, mostly due to the possibility of extracting information from big unstructured data on the web. The attention networks, originally designed for text only, have proven successful in their capacity for merging data. Most recently, the contrastive learning objective

applied on hundreds of millions of annotated images has provided State-of-the-Art (SOTA) results. However, the standard methods and evaluations in the multimodal field have two shortcomings:

- The generalisation abilities of models trained multimodally are yet to be determined
- There is no computationally cheap way, both in terms of data and power, to improve or leverage the latent space abilities of these cost-expensive algorithm on a tasks such as image captioning

In this thesis, the first shortcoming is addressed by our evaluation tasks, that can be applied to other networks in order to compare the generalisation ability of any image and/or text model. Part of the second issue is dealt with using our Latent CycleGAN, which is very cost-effective, and which improves a more straightfoward captioning pipeline with unmatched multimodal data.

2.2 Contribution of this thesis

Most models in multimodal deep learning are designed without considering what was actually "learned" by the model. Indeed, when a model has been trained on hundreds of millions of samples, it is evaluated on some standard tasks, and if it performs well, then the model is said to be satisfying. It is one legitimate way of assessing the quality of a representation, but in this thesis, we want to dive more deeply into the mind of machines. That is to say that we are going to evaluate in a more fine-grained manner the generalisation abilities of multimodal (versus

unimodal models). Furthermore, when a model is used for a downstream task, the quantity of data and annotation required for fine-tuning is often considered to be irrelevant, as long as the task can be performed with the best score possible. We also want in this thesis to maximize the cost-effectiveness when leveraging the properties of the latent space obtained with an already computationally expensive training, by developing methods that require little data and little annotation. For both these two objectives, we are going to use a bimodal algorithm that became an instant classic: CLIP.

Chapter 3

Multimodality

3.1 From Unimodal to Multimodal Models

Human experience is essentially multimodal. Yet, most current algorithm can only process one type of data. This gives the different fields of Artificial Intelligence (AI): Computer Vision (CV), Natural Language Processing (NLP)... In these fields, algorithms are trained on unimodal datasets (MNIST [Deng, 2012], ImageNet [Deng et al., 2009], CIFAR [Krizhevsky et al.,] for CV ; Wikipedia, Common Crawl, Book Corpus for NLP), usually with a training for a single task. Sometimes their understanding is specialized for this one task, and cannot be transferred to another – or with very poor results. Most SOTA algorithm nowadays are actually pretrained on a big dataset (several million samples) and on a general task (object recognition or segmentation for vision [Kolesnikov et al., 2019, He et al., 2015], inferring the missing word in a sentence, or assessing whether two sentences follow each other in a text for language [Devlin et al., 2018]) that allow them to understand many aspects of the modality. After this pretraining, algorithms are

then fine-tuned, to specialize their representation, for other more specific task. Big unimodal datasets have been available for years now, allowing scientists to develop models with robust features, that can generalise well to many tasks within their modality.

If we want to go one step further in creating algorithms that can learn features that adapt to various conditions (in terms of inputs and in terms of tasks) – just as human can do – we need to introduce multimodality. This is only a step in the way of creating intelligent machines such as robots. [Bisk et al., 2020] lays out the past and future steps towards creating a human-like intelligence, starting from a Natural Language Processing perspective. There are five of them, which are called World Scopes. Each one is an extension of the components of the world an algorithm can access and process along with the other. They are the following:

- WS1: Corpus *(Our Past)*
- WS2: Internet *(Most of current NLP)*
- WS3: Perception *(Multimodal NLP)*
- WS4: Embodiment
- WS5: Social

The first two World Scope are in the field of unimodal NLP. The current NLP SOTA models have a World Scope that stops at step 2, which means that they are trained with large textual dataset extracted from the Web.

The field is now evolving into multimodality. Completing this step will result in models able to ground their language representation in other perceptual domains, leading to a richer, more subtle and robust understanding of the world.

Beyond this point, we enter science-fiction. The embodiment scope relates to robotics, which is now mostly a separate field, with little, but growing, interactions with multimodal AI [Mehlmann et al., 2014]. A physical body will make AI more human-like or animal-like, with proprioception and maybe positive and negative sensation like pain and pleasure. The social scope is when the model will learn to directly interact with other social beings, with notions of emotional reactions, empathy, social hierarchy, and to develop social behaviors such as the social gaze, and to accompany its speech with appropriate gestures [Bee et al., 2010, Cassell et al., 1994, Cassell and Thórisson, 1999].

The availability of large multimodal dataset being very recent, it is only now that we can create multimodal models comparable, in term of number of training samples [Lai, , Miech et al., 2019], to unimodal ones (see Chapters 4 and 5). Before that, the training set of multimodal algorithm was very restrained (COCO [Lin et al., 2014], Conceptual Caption [Sharma et al., 2018a], LIRIS-ACCEDE [Baveye et al., 2015]) and although models can achieve good results on some tasks, the multimodal features learned by them were often not usable for other tasks [Devillers et al., 2021]. This type of models includes: Virtex [Desai and Johnson, 2020], SimVLM [Wang et al., 2021], Frozen [Tsimpoukelli et al., 2021], BUTD [Anderson et al., 2017] and Dall-E 2 [Ramesh et al., 2022] among others, which are described in the section.

3.2 Unimodal Pipelines

In order to present some interesting multimodal models, we first need to introduce some of their inner pipelines, which were originally designed for unimodal tasks.

Word Embedding

A word embedding is a vector that represent a word. The principle was popularized by Word2Vec [Mikolov et al., 2013b], an algorithm trained to infer missing words in a sentence. After the training, each word known by the network is associated with a vector, and relations in the vectorial space can be translated into relations between the properties of the corresponding words. The standard example is: take the vector for "king", add the vector for "woman" and subtract the vector associated with "man" and you will get a vector resembling the one corresponding to "queen".

Nowadays, pretrained word embeddings are used in more complex networks, such as transformers, in order to be enriched by the context (i.e. the sentence or a larger piece of text) in which the words appear.

An example of a 2D projection of a word embedding space can be seen in Figure 3.1.

Long Short-Term Memory (LSTM)

LSTM are now used in a variety of tasks, but were originally mostly used to process text [Hochreiter and Schmidhuber, 1997]. This is a classic subcase of Recurrent Neural Network (RNN) (see Figure 3.2). The specificity of the LSTM

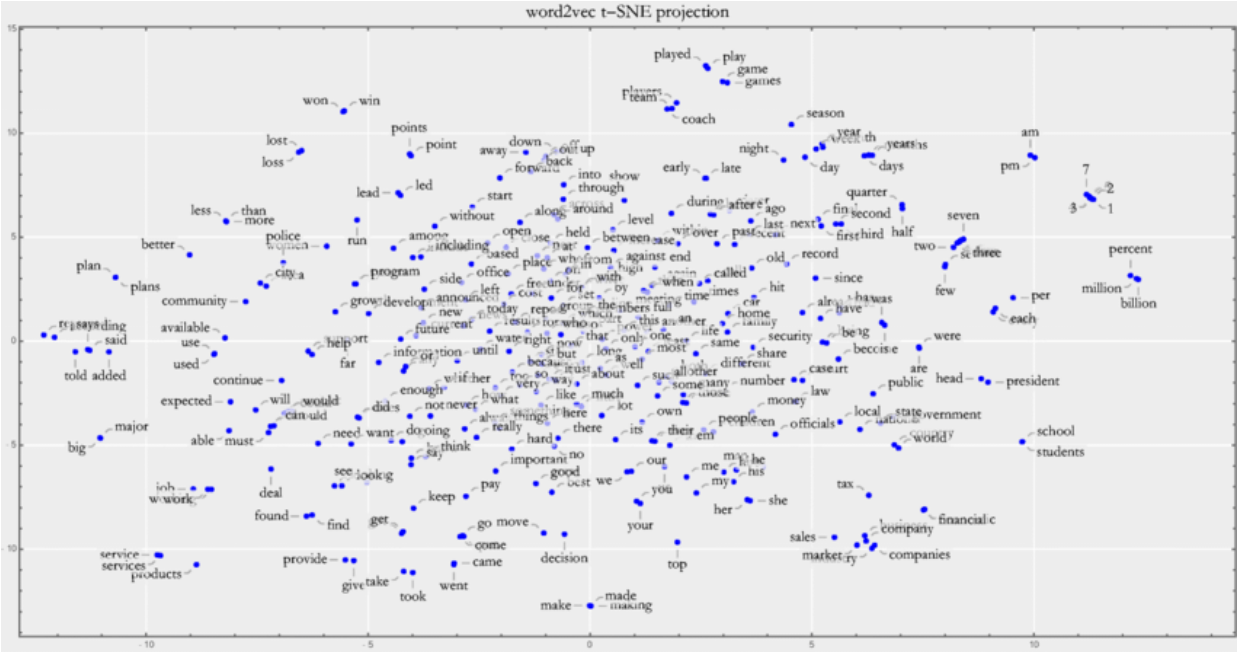

Figure 3.1: t-SNE 2D projection of the word2vec vectors representing a selection of the most frequent words in its training corpus. Source: [Gastaldi, 2021], under the Terms and Conditions of Springer Nature journal for academic use.

is that it has two states, linked to each other by some gates detailed in Figure 3.3. These states are called hidden state and cell state. The cell state is used to keep longer term information and the hidden state shorter term ones. LSTM can be used as encoder or decoder. In a decoder setting, the LSTM is simply used in an autoregressive fashion. As an encoder, the last state of the LSTM can be used a feature vector for a whole sentence.

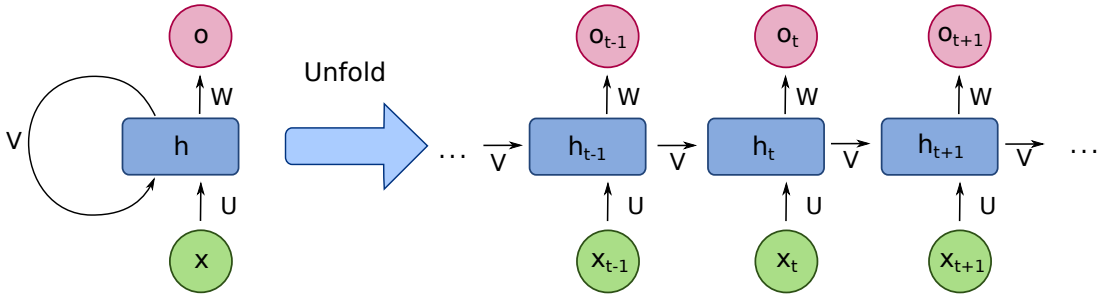

Figure 3.2: A RNN has several timestep, where the network receives a new input and the hidden state from the previous timestep. At each timestep, the network can output something (but this is not mandatory). Source: wikipedia.com, under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Attentional LSTM

Attention was first introduced to improve LSTMs in an encoder/decoder setting [Bahdanau et al., 2014]. Normally, a LSTM encodes the sentence in vector, and then the decoder, which is also a LSTM, uses this fixed-size vector to generate the output. In this configuration, the decoder, at each timestep, has access to one cell state, the one from its precedent timestep (or, for the first timestep, the encoded sentence vector). Attention was made so that the decoder instead looks all the states of the encoder, by using a weighted sum of them in replacement for its cell state. The weights are provided by another neural network trained along with the LSTMs. Figure 3.4 shows the attentional LSTM.

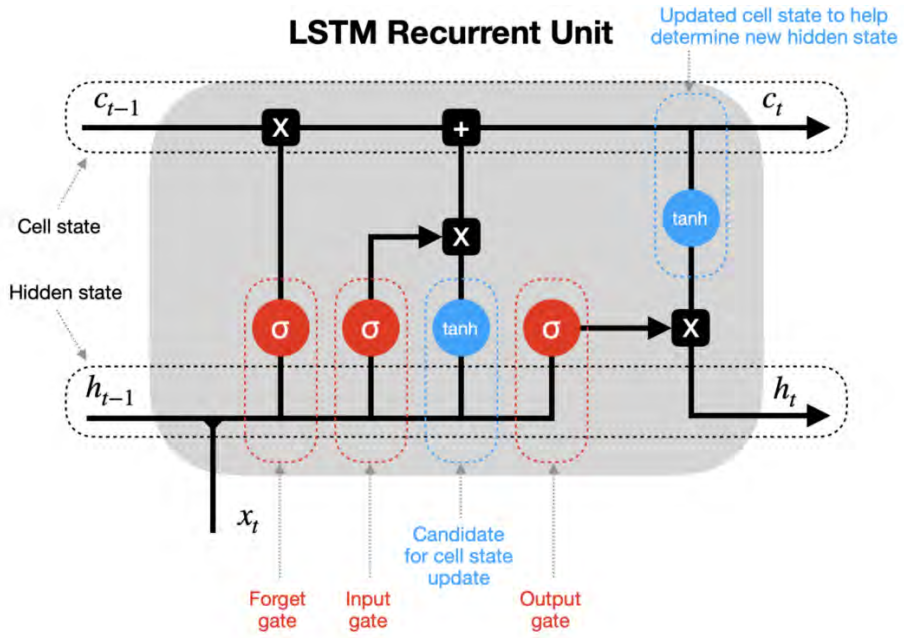

Figure 3.3: The LSTM has two states: the hidden state and the cell state. Different operations are computed to generate the new states (the red bubble is a sigmoid activation, the blue one a hyperbolic tangent, and some concatenation, multiplication and addition of vectors are computed). Source: https://towardsdatascience.com/, I do not own this content, credits to Michael Phi.

Transformer

The transformer paper [Vaswani et al., 2017] was a breakthrough in the NLP field, and has now been transferred to the CV (see the next subsection) and multimodal field. It discarded the recurrency of the LSTM and only kept the idea of Attention ("Attention is all you need"). The original transformer is used for Machine Translation as an end-to-end encoder-decoder.

The general principle is to enrich the encoding of an element of a sequence by

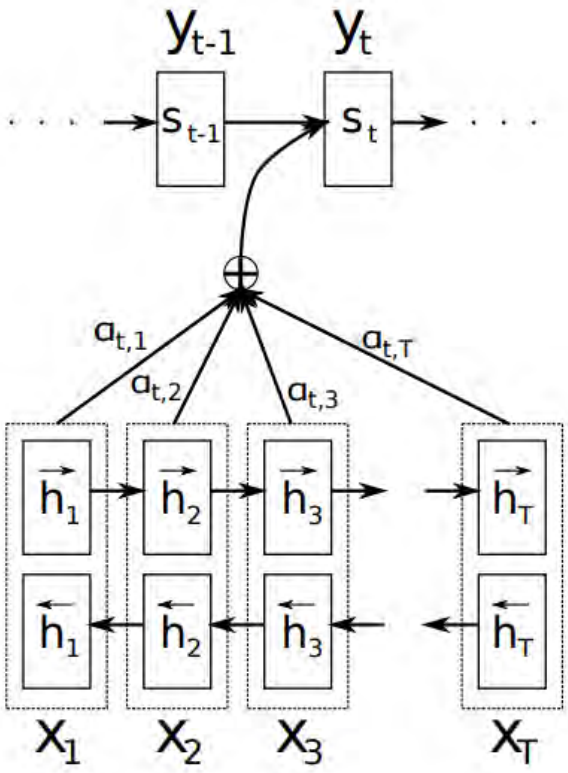

Figure 3.4: The bidirectional encoder produces hidden states for each timestep. At each timestep of the decoder, a weighted sum of these encoder hidden states is computed to produce an input for the next timestep. Source: [Bahdanau et al., 2014], I do not own this content, all credits go to its rightful owner.

the other elements of this sequence. This creates an contextualized representation of the element. The transformer networks do that by computing, for each element, three vectors: the query, the key and the value. These vector are then used to calculate (see Figure 3.5) how much each of the elements is going to enrich the representation of the others. This is the encoding part, which can be used as an independent network, the most famous being BERT [Devlin et al., 2018].

The calculation of the enriched embedding is position agnostic. To take in account the position of each word in the sentence, a position embedding is added to each word embedding before the keys, queries and values are calculated.

For the decoding part, a similar principle is applied, but the keys and the values are based on the outputs of the encoder. Only the queries come from the autoregressively generated words.

The decoder can also be used as such, given some modifications that remove the dependency on the encoder outputs [Radford et al., 2018a, Radford et al., 2018b, Brown et al., 2020].

BERT

BERT is a transformer encoder with three special tokens: the [MASK] token, the [CLS] token and the [SEP] token. It has been trained on two tasks, one is to encode two sentences separated by the [SEP] token and to use a classifier on top of the enriched [CLS] token to decide whether the two sentences follow each other in a larger text or not (which means the [CLS] token can be seen as a summary embedding of the sentences). The other task is to demask some tokens of a sentences that have been replaced by the [MASK] token – i.e. to guess which token is supposed to be in the sentences at the masked positions.

Visual Transformer

The visual transformer takes the transformer architecture and transfers it into the visual domain. As the transformer works with sequences, an image first needs to be turned into a series of patches to be processed by the vision transformer.

Each image token (patch) is then contextualized using the transformer pipeline. Just as in BERT, a [CLS] token is added to the sequence and sums up all the patches (which makes it represent the whole image). This token is then used for the classification task. See Figure 3.6.

Convolutional Neural Network (CNN)

Convolution Neural Networks [Lecun et al., 1998] are network based on the convolution operation (a sliding matrix of operation that calculates a new matrix based on the input one).

One of the most commonly used CNN is the VGG16 [Simonyan and Zisserman, 2014], one example of which is shown in Figure 3.7.

CNN are most often used when processing an image, as the convolution operation takes naturally in account the spatial features of the input, mimicking the convolution happening in the early visual cortex [Lindsay, 2021]. They produce feature maps, that are usually fed to a classifier, after some sort of pooling, and typically for object detection or recognition.

Region Proposal Network (RPN)

RPN are used for extracting bounding boxes for objects (or Regions of Interest) in an image. The classic RPN is R-CNN [Girshick et al., 2013] which was soon replaced by the more efficient Faster-RCNN [Ren et al., 2015], that is detailed in Figures 3.8 and 3.9.

Faster-RCNN is based on a CNN, on top of which a classifier and a regressor are placed. They are using the feature maps from the CNN to infer, for each pixel, the probability that it is the center of a given object (this is the job of the classifier), whose bounding box is determined by the regressor.

RPN can be used for the unimodal task of object segmentation, but also as an intermediary step towards a multimodal task that involve a fine-grained understanding of the spatial structure of images, such as Image Captioning or VQA [Anderson et al., 2017].

GAN

The Generative Adversarial Network (GAN) (introduced by [Goodfellow et al., 2014]) are algorithms composed of two parts. The goal of such a network is to generate a specific kind of data (images, text, vectors etc.). To train, the generator needs what is called a disciminator. This discriminator is fed with samples generated by the generator and real samples (the one the generator is trying to generate with high fidelity). The goal of the discriminator is to judge whether a sample is a real one or a fake (generated) one; reversely, the goal of the generator is to fool the discriminator so that it believes the generated samples are real. A GAN is usually trained unsupervised. Figure 3.10 details a standard GAN.

3.3 Multimodal Fusion

The unimodal pipelines presented above are going to be used in the multimodal models that we are introducing in the following sections. The current one presents

how to merge data from several modality and how to represent them in a common (or two comparable) space(s). The next one will describe how to translate between modalities without having to share a representational space.

For a model to be able to represent multimodal data, it needs some sort of fusion between (at least) two modalities. This can be done with several methods described in Figure 3.11.

The method of early fusion consists in merging the data from different input before feeding it to the model (or right after a simple encoding pipeline, e.g. convolution on images or word embedding for text). Thus, the model processes multimodal data from the start, and is therefore multimodal all the way down to its deeper representations. In this case, a joint representation is often created [Baltrusaitis et al., 2017].

The method of model-level consists in fusing the vectorial representations of separate streams, each processing only one type of data. By doing so, the multimodality comes deeper in the architecture, when data has already been turned into features.

The name late fusion is sometimes used in the sense of model-level fusion (usually, when the two unimodal streams have been trained separately), but here we make a distinction: late fusion happens after each of the pipeline has already made a prediction.

When the two streams are trained together, and coordinated through a similarity measure (like in CLIP [Radford et al., 2021]) or by a structure constraint, the model is called a coordinated representation model [Baltrusaitis et al., 2017]. It is not exactly fusion, therefore this will be presented in a separate section (3.4).

Other fusion methods, usually mixing the ones presented above, exist as well.

The section below will present some multimodal models. It will highlight some of their advantages and inconvenients and provide an overview of the State-of-the-art multimodal models in which we will pick the best suited ones for assessing and leveraging their generalisation abilities.

Early Fusion

Early fusion has the advantage to merge data from the beginning, which creates fully multimodal representations. This kind of algorithm, however, works in suboptimal condition when representing unimodal data. Nevertheless, early fusion models are the most common multimodal models, especially since the attentional models, have become very powerful; they can, with nothing more than a separation token between modalities, learn to distribute their attention differently between vision and text, and between different tokens within a modality.

VisualBERT

This model can be considered as the standard transformer-based multimodal encoder. It is simply a BERT [Devlin et al., 2018] model, but instead of encoding text only, it encodes text tokens and image tokens separated by a [SEP] token. VisualBERT [Li et al., 2019] is detailed in Figure 3.12. The network is fed with both text and image embeddings (images are turned to sequences just as in ViT [Dosovitskiy et al., 2020]). The special token [SEP] is used to separate between text and images. The algorithm is trained on a dataset of images and captions with two objectives, one where part of the caption is masked and the missing tokens have to be guessed (demasking the [MASK] tokens), the other with two sentences

and an image as input, where the network has to decided if one of the sentences does not describe the image, using the [CLS] token, which is therefore trained to sum up sentences in one enriched embedding.

SimVLM

If VisualBERT is the standard multimodal transformer encoder, SimVLM [Wang et al., 2021] is the standard multimodal full transformer. It is composed of a bimodal encoder and a unimodal decoder (see Figure 3.13). The bimodal encoder takes image features (generated by a ResNet applied on image patches – similarly to ViT) and word embeddings to encode them jointly through a multimodal transformer, similarly to VisualBERT. The vectors resulting from the encoding are then used to calculate keys and values for the autoregressive transformer decoder, while the query are computed based on the already generated words (like in Figure 3.5). This enables the decoder, given the beginning of a caption (or just image features), to generate the missing words one after another. SimVLM can be used for Visual Question Answering (VQA) and for image captioning. Some results are displayed in Figure 3.14.

OSCAR

OSCAR [Li et al., 2020] stands for **O**bject-**S**emantics **A**ligned P**r**e-training. It is an algorithm trained with a special kind of early-fusion, as a modality is added to the vision and the language one. OSCAR takes as input an image, some object tags that represent the objects that are present in the image, and a caption (see Figure 3.15). The input image is passed through a RPN, and the features from the

salient regions are extracted, while the rest of the image is discarded (see BUTD in 3.5 for more details on how that can be done). The caption is inputted as a normal textual input. The object tags however constitute an early form of multimodality. Indeed, they correspond to *visual* data, but take the form of word embeddings, therefore have the form of *textual* data. The multimodality in OSCAR comes from merging visual and textual data, a fusion that is enhanced through these object tags, which help the alignment of both domains.

OSCAR is trained to demask some tokens of the input descriptions (just as in BERT), which somehow makes it the advanced version of VisualBERT, trained on a more heavily annotated database, with an additional visio-linguistic modality. It can be fine-tuned and used for image captioning. It is a standard SOTA baseline, and some of its performances can be seen in Figure 3.23 and 3.14.

Model-Level Fusion

Model-level fusion algorithms have several unimodal streams of data that merge deep into their architectures. Thus, as some of their pipelines are unimodal, the interpretability of what has been learned thanks to each domain is higher. They also allow for a higher level fusion, in spaces that can be geometrically more similar between domains (two 2048-dimensional feature spaces for instance), and without having to artificially preprocess one modality, contrary to what we have seen above, for example when images are separated into sequences of token in order to be processed by a tranformer.

Frozen

Frozen [Tsimpoukelli et al., 2021] is a model-level fusion level that uses a frozen unimodal text encoder, to train an image encoder in a multimodal setting. First, the image and part of the caption are encoded through two separate encoders, then on top of the concatenated (therefore multimodal) representation of the image-text pair, a frozen transformer-like architecture (pretrained with unimodal text data) is used to predict the end of the caption. Figure 3.16 illustrates it. The particularity of this model is that the multimodal transformer is also frozen. It is simply a unimodal transformer, which will use the features from the image encoder as if it was text features. The multimodality comes from the vision side, that will learn to produce text-like embeddings for the frozen transformer. Thus Frozen learns to embed images in a textual representational space, so that the features can be understood by a unimodally trained transformer, which means it could also be seen as a coordinated representation model (see 3.4).

Book Cover and Title Fusion Model

In [Lucieri et al., 2020], the authors introduce several ways of fusing image and text data to classify the genre of a book given its title and cover. Figure 3.17 shows 3 fusions in one graph, what they call early, late and dual fusion. The late-fusion (which is actually what we call here model-level fusion) is the best performing one. In this setting, images are passed through a unimodal Inception-ResNet [Szegedy et al., 2016], text is passed through a unimodal FastText [Bojanowski et al., 2016, Joulin et al., 2016a]. The unimodal vectors resulting from these pipelines are then concatenated and fed to a (multimodal) classifier,

which is then trained on the genre classification task. Chapter 5 compares several model-level fusion model (including the fusion of vectors generated by multimodal algorithm) with this previous SOTA on this dataset.

3.4 Coordinated Representation

Coordinated representation models have the advantage of being able to represent each modality independently. This enables us, when assessing their generalization capability, to use them in a unimodal or a multimodal setting (for a multimodal model-level fusion, we just have to concatenate the two vectors produced by each of the pipelines). The representation learned by each multimodal pipeline has implicitly incorporated information specific to the other one, but doesn't need a piece of data from the other domain to work in an optimal setting, while for the other models above, masking one modality is detrimental to the representation of data. This is why we are mostly going to focus on CLIP (see just below) to work in both unimodal and multimodal conditions.

CLIP

CLIP [Radford et al., 2021] stands for Contrastive Language-Image Pre-training. It is a network trained with image-caption batches, where each caption corresponds to an image in the batch (see Figure 3.18). The goal of the network is to create representations for image and text in two coordinated vectorial spaces (which can be considered as one multimodal latent space), with the constraint that the vector for an image and its caption should be as similar as possible, while the vector for

an image and the other captions in the batch should be as far as possible. This type of training is called contrastive training. CLIP has been trained on a dataset of 400 millions image-caption pairs with batches of size 32,768. Unfortunately, its dataset is not public, which forces us to use tricks to make sure that the generalisation abilities that we are going to evaluate in the next chapters are really performed on unseen data (see Chapter 4). CLIP exists in different version (5 ResNet [He et al., 2015] versions and 3 Vision-transformer [Dosovitskiy et al., 2020]), because it can use different image encoder architectures.

CLIP can be used in many different tasks. The most impressive is zero-shot image classification, which can be done with a simple method described in Figure 3.19. Basically, by using a standard sentence structure and by changing only a word in this sentence (for instance, with the sentence: "This is a photo of a [object]", [object] being replaced by each class of the task at hand), we can create a batch of descriptions, that we can then compare with an image in CLIP's latent space. The caption with the highest similarity (which correspond to a specific category of objects) will determine the class attributed to the image by this zero-shot pipeline.

Its classification abilities can be compared to a standard unimodal network's like a pretrained ResNet with a trained linear probe on top, which are shown in Figure 3.20. CLIP is often outperforming the ResNet, notably on the standard ImageNet dataset – which gives the hint that multimodal training can sometimes improve unimodal perfomances, a hypothesis that we will explore in this thesis.

3.5 Multimodal translation

If the model doesn't represent both image and text in a latent space, but only, for instance, generates text conditioned on images (or the reverse), then fusion is not mandatory. Usually, a unimodal representational space is used and its features are fed to a decoder of the other modality. These model will be referred to as multimodal translation models. They are often trained and used end-to-end, which yields very good results but requires fully annotated datasets. We provide some classic examples and some current SOTA, but note that in our model (see 6), we will use unsupervised training on unmatched multimodal data, and we will leverage the property of a pretrained latent representation in order to keep our computational cost very low.

Image captioning models

BUTD

BUTD [Anderson et al., 2017] stands for Bottom-Up Top-Down. It is a classic captioning model that uses both bottom-up and top-down attention to generate the right caption given an image. First, the model uses a Faster-RCNN 3.8 with the convolutional network Resnet-101 [He et al., 2015] to determine the salient region of the image. Instead of splitting the image in a fixed-size grid, the model uses the bounding boxes of detected objects as region of interest and uses the features from these regions (see Figure 3.21). This means that the irrelevant parts of the image are discarded with this "hard" attention mechanism.

Then the network can perform its top-down attention mechanism. Two LSTM

are stacked on top of the image features extractor. One uses already generated input and the image features to produce weights that are gonna be used to compute the input for the second LSTM, that generates text conditioned on previously generated words (see Figure 3.21). BUTD can be trained on images and their descriptions, or on a VQA dataset. In this case, an additional top-down mechanism is used to compute the weights calculated by the first LSTM, which takes in consideration the question being asked about the picture.

Some results for image captioning have been displayed in Figure 3.14.

ClipClap

ClipClap [Mokady et al., 2021] is a captioning model that uses CLIP [Radford et al., 2021] as an image encoder, and that uses its high level representation to condition prefixes for the generative language model GPT2 [Radford et al., 2018b]. GPT2 is autoregressive, which means its generation is conditioned on what it has already generated. The prefixes simulate the beginnning of a generation so that GPT2 outputs the right caption. GPT2 can be fine-tuned or not, depending on the way the prefixes are learnt. ClipClap is thus trained end-to-end with a frozen CLIP to generate a caption for each image of the dataset. It has be trained on COCO [Lin et al., 2014] or Conceptual Captions [Sharma et al., 2018b]. See Figure 3.22 for the architecture, Figure 3.23 and 3.24 for some captioning results.

ClipClap will be an inspiration for our captioning model, but instead of training the model with matched data, we will use CLIP as text encoder, and train prefixes for GPT2 based on caption embeddings, which is a way of creating a text encoder-decoder pipeline. The multimodality will be learnt differently without matched

data, using cycle-consistency (see Section 3.6) between modalities.

Text-to-image translation

DALL-E 2

DALL-E 2 [Ramesh et al., 2022] is an image generator based on CLIP's latent space. It is composed of two parts, one of them being CLIP's frozen text encoder. After embedding a sentence with CLIP, DALL-E 2 uses unCLIP (shown in Figure 3.25), a pipeline that allows it to turn the text encoding into a corresponding image encoding in the same latent space, and then to generate an image based on this "fake" image vector using a diffusion model [Sohl-Dickstein et al., 2015].

The fact that text vector need to be translated into image vector indicates that in CLIP's latent space, text and image features are different, i.e. that their multimodality is only partial. This info hints what we will see in Chapter 6: a text decoder trained on CLIP's text features cannot generate proper descriptions when fed with image features. We will thus create a translation algorithm. Contrary to DALL-E 2 however, which is trained on 400M image-caption pairs, we won't need supervision nor a huge dataset to train our translation model.

Some example of generation using DALL-E 2 can be seen in Figure 3.26.

MirrorGAN

MirrorGAN [Qiao et al., 2019] is a multimodal GAN that uses the cycle-consistency principle (see section 3.6). First, the description of the image is passed through a textual module (a RNN) that creates a sentence embedding

and embeddings for each word in the sentence. Then these embeddings are fed to stacked image feature generators. Each generator uses attention (with a transformer-like pipeline) conditioned on the embeddings and the generated image features from the preceding generator in order to generate the next visual feature. At each stage of the feature generation, an image generator (that is part of a GAN) creates a corresponding image. Then, when the last image is generated (the purpose of MirrorGAN is this very image, that is supposed to correspond to the input description), a recaptioning pipeline is used to better align text and image. A CNN turns the image into features, that are then used by a RNN to regenerate the input caption, which is then compared with the true caption with the goal of being as similar as possible. The MirrorGAN algorithm is trained end-to-end with several objectives:

- For the image generator to fool the image discriminator
- For the discriminator to recognize fake and real images
- For the caption RNN to generate the right caption (with a Cross-Entropy-based loss on the word tokens)

Figure 3.27 details the different modules of MirrorGAN and Figure 3.28 shows images generated by MirrorGAN and the original caption versus other model and the ground truth.

3.6 Cycle-consistency

Cycle-consistency will be used in our multimodal translation model. It allows a model to learn without annotated data, which will be very useful when leveraging a fully supervised pretraining.

The cycle-consistency principle states that when translating a piece of data from one domain to another, and back, the resulting piece of data should be the same as the original one. For instance, let's say you are translating a sentence from English to French, and then back to English, then the resulting sentence should be unchanged. The cycle-consistency principle can be applied as an objective in order to learn translation without matching data. It has been notoriously applied in the CycleGAN paper [Zhu et al., 2017].

CycleGAN

CycleGAN is an image-to-image GAN that, with unmatched data from two distribution, learns to translate between them. For instance, it can translate pictures of a landscapes in a given season to pictures of that same landscape in another given season (in this case it has "seen" images of landscapes in the first season and images of *different* landscapes in the second season. See Figure 3.29) or can change pictures of horses into pictures of zebras, and zebras' into horses'.

CycleGAN [Zhu et al., 2017] is composed of two GANs, one that is trained to translate from modality X to Y (which means its generation is conditioned on an image input) and the other from modality Y to X.

CycleGAN has several objectives:

- The Generators objective to fool their corresponding Discriminator
- The Discriminators objective to not be fooled by the Generators
- The cycle-consistency objective: when translating from modality X to Y and then from Y back to X, the resulting piece of data should be as close as possible to the input
- An optional objective: when a Generator is fed with input from its target modality (i.e. the Generator from X to Y is fed with data from Y), the result of its generation should be identical to the input.

Figure 3.30 shows the cycle consistent operations and Figure 3.31 displays some example of translation using CycleGAN.

We will use a version of CycleGAN applied to a multimodal latent space (the Latent CycleGAN) to design our unsupervised captioning pipeline in Chapter 6.

3.7 Summary

When assessing the generalisation abilities of multimodal models, one should consider on which task and in which setting. The comparison we want to perform, as we consider it the basic one, is between unimodal and multimodal models. To do so, we want multimodal models that are able to represent unimodal data with minimal damages to the feature extraction. This guides us towards coordinated representation models, where two unimodal pipeline inform each other during training, but where at inference, each pipeline is actually independent (We can

use these pipelines in a multimodal, model-level fusion setting, where the features extracted by two unimodal encoders of different modalities are concatenated in order to create a multimodal vector). This is why CLIP will be so central in our thesis.

Furthermore, we have seen with MirrorGAN that cycle-consistency can be used in a multimodal setting. Even though in MirrorGAN's case the data are matched, we have seen in the CycleGAN paper (which is designed for unimodal translation between two image distributions) that actually, matched data are unnecessary and that the cycle-consistency principle works in itself. This two models will help us to design a multimodal translation model in CLIP's latent space, that will enable us to perform image captioning with unsupervised training, by leveraging the multimodality that is already present in CLIP's latent representations.

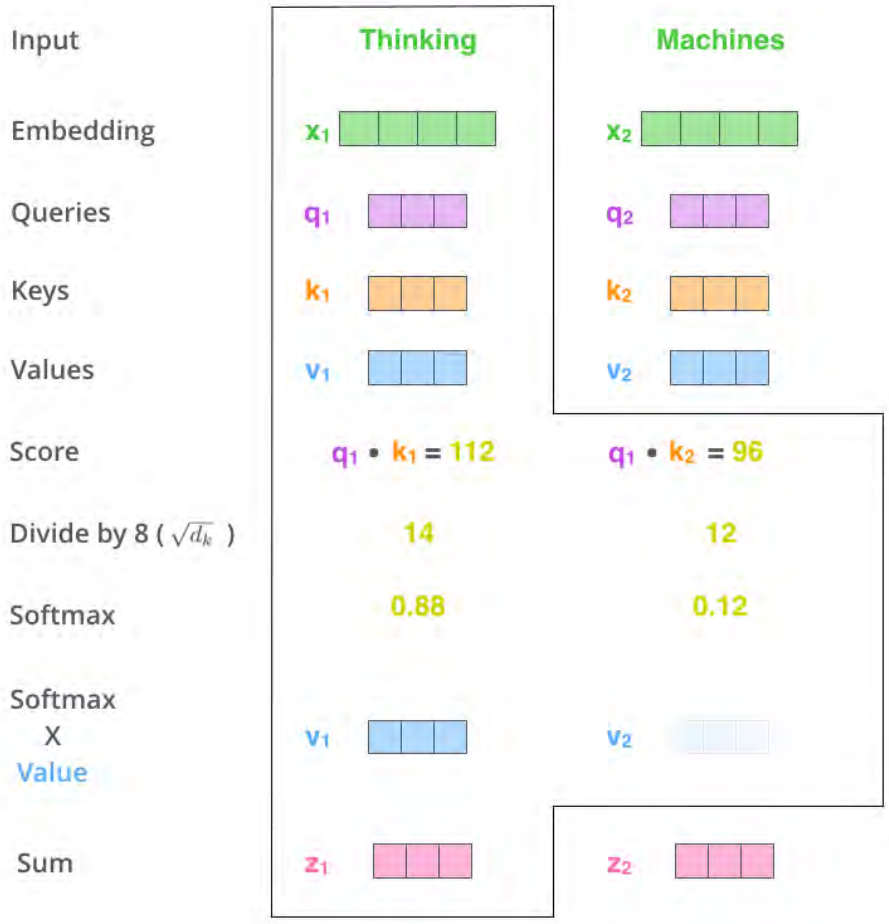

Figure 3.5: Example of self-attention computed for the word "Thinking" in the sentence "Thinking Machines". The query (q_1) for the word "Thinking" multiplied by the key (k_1 and k_2) of each word gives different scores that are transformed into weights that will be used for a weighted sum of the values (v_1 and v_2). This sum is the new embedding for the word "Thinking" (z_1). Source: jalammar.github.io, under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

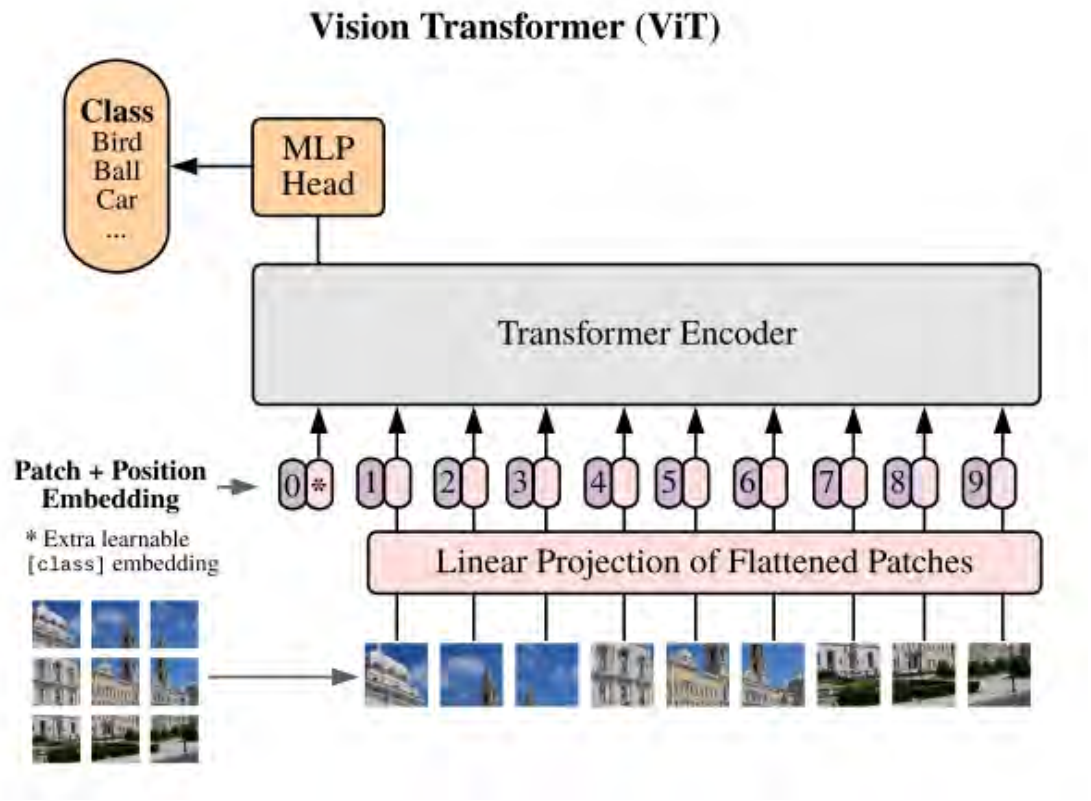

Figure 3.6: The image is divided into several patches (that are going to be flattened into 1-dimensional vectors) to form a sequence-like input. Then, a standard transformer encoder takes over. The classification is made using only the [CLS] token introduced in [Devlin et al., 2018]. Source: [Dosovitskiy et al., 2020], I do not own this content, all credits go to its rightful owner.

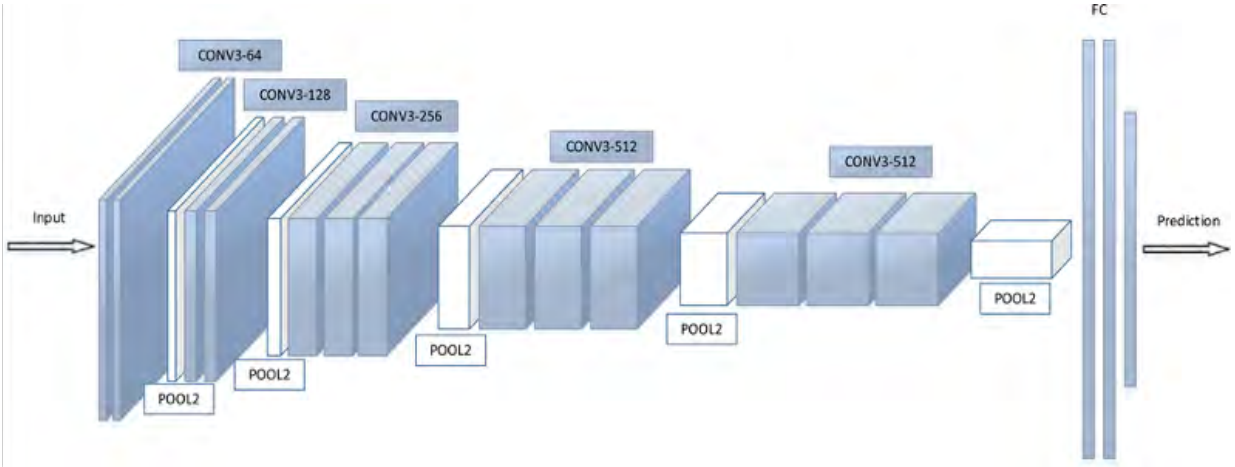

Figure 3.7: VGG16 is made of stacked convolutional operations, with layers of pooling in between. At the end of the last pooling, fully connected layer are stacked before the prediction (e.g. image classification) can be made. Source: [Eminaga et al., 2018], under the Attribution-NonCommercial-NoDerivs License

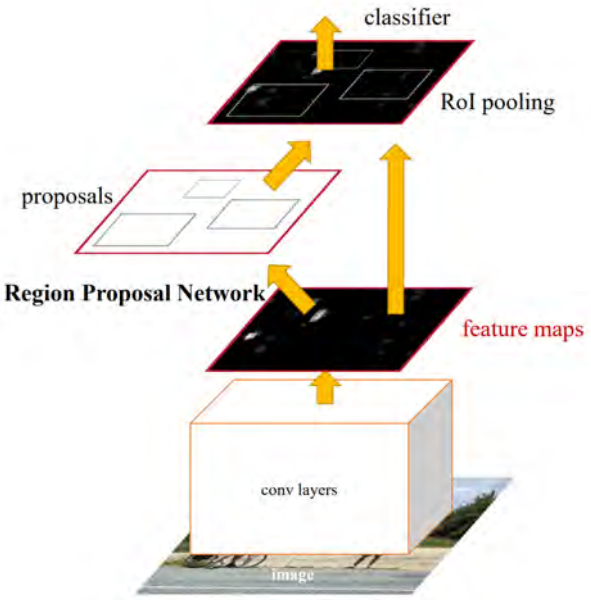

Figure 3.8: Faster-RCNN is composed of a VGG, on top of which the classifier and the regressor are place to propose region in which a type of object could be present with a certain probability (score). Source: [Ren et al., 2015], I do not own this content, all credits go to its rightful owner.

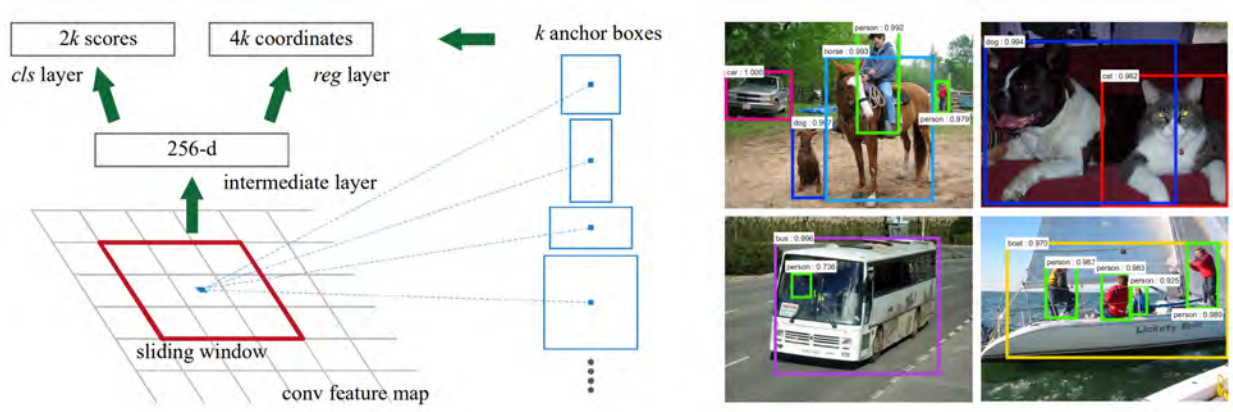

Figure 3.9: Left: The classifier and the regressor. There is for each sliding window k possible boxes. The classifier output two probabilities for each box: the probability that the box contains an object and that it doesn't (which gives 2k scores). The regressor outputs the coordinates of the center of the box, its width and its height (which gives 4k outputs per window). Right: Examples of detection using Faster-RCNN. Source: [Ren et al., 2015], I do not own this content, all credits go to its rightful owner.

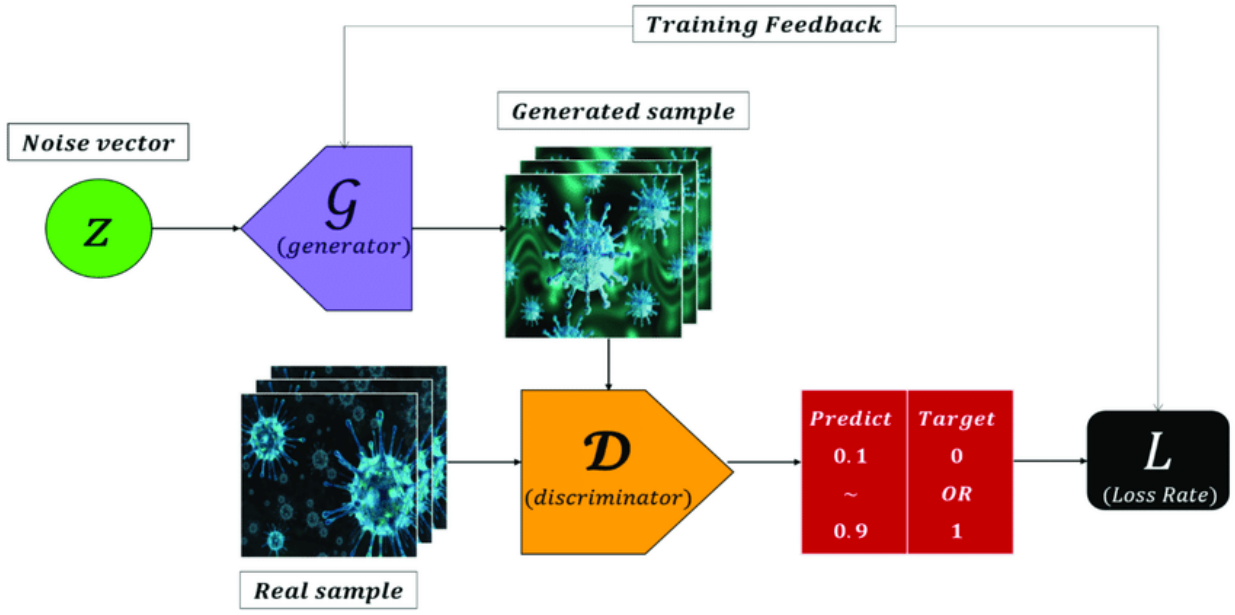

Figure 3.10: An example of a GAN where the generator is trained to output images that look like the real ones given to the Discriminator. The GAN needs a latent sample to not always generate the same piece of data. As the GAN is conditioned on this vector, some GANs (called Conditional GANs) can choose the sample to guide the generation, for instance to generate a specific category of image instead of a random one. Source: [Park et al., 2021], under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License

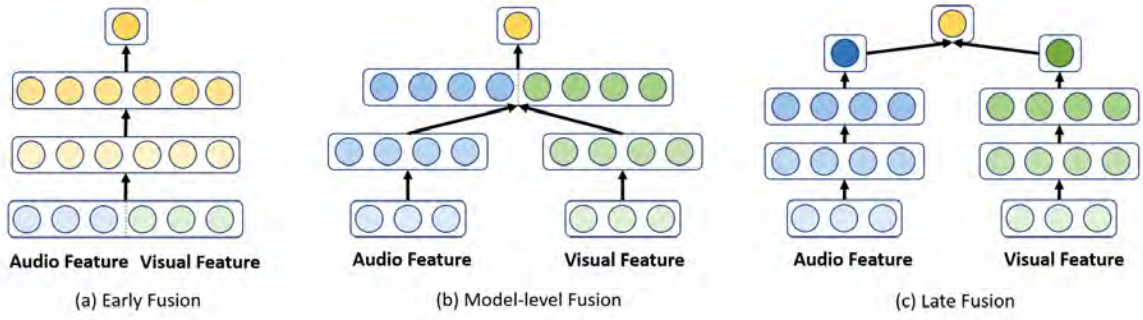

Figure 3.11: Different example of fusion with audio and visual modalities. Fusion can happen directly when modalities are inputed to the model (a), or it can happen after separate unimodal models have processed it, with features at a higher level (b). The prediction of the model can also be made with regards to the predictions of the unimodal ones (c). Of course, these method can be combined, and fusion can happen at any depth within the model, and sometimes different pipelines can fuse multiple times during the process (this is often called slow fusion). Source: [Chen and Jin, 2017], I do not own this content, all credits go to its rightful owner.

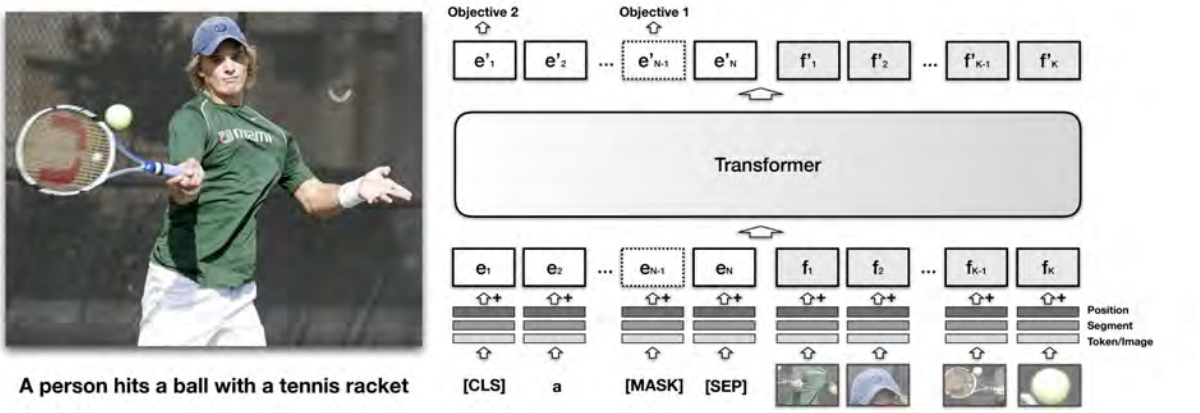

Figure 3.12: Left: An example from VisualBERT's training dataset. Right: The general architecture and training of VisualBert. The objectives here are demasking hidden tokens given the other textual tokens and the image patches (objective 2), and classifying the sentences with the [CLS] token to guess whether one sentence does not describe the input image (objective 1). Source: [Li et al., 2019], under the CC BY-SA License

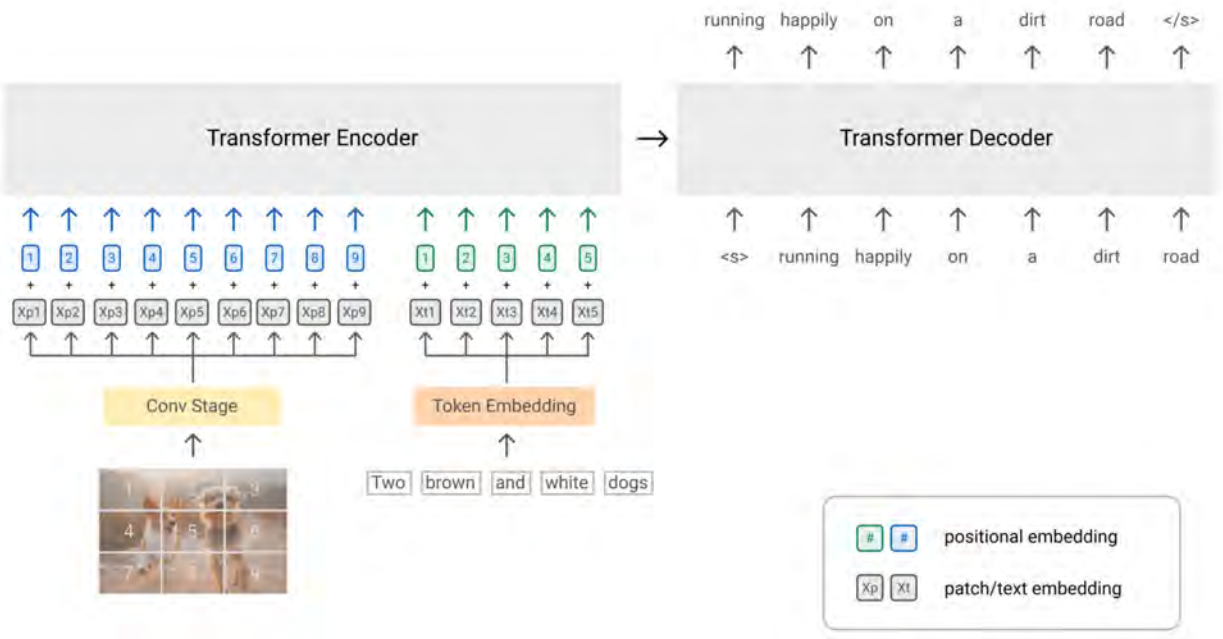

Figure 3.13: SimVLM is trained to generate the caption of an image or to complete an incomplete one, given the image and part of the caption. During the training, text-only data are also shown to the model and it has to finish the sentence. Source: [Wang et al., 2021], under the CC BY-SA License

	Setup	CoCo Caption				NoCaps			
		B@4	M	C	S	In	Near	Out	Overall
BUTD[a†]	supervised	36.3	27.7	120.1	21.4	-	-	-	-
AoANet[b†]		39.5	29.3	129.3	23.2	-	-	-	-
M2 Transformer[c†]		39.1	29.2	131.2	22.6	81.2	-	69.4	75.0
$\text{SimVLM}_{\text{base}}$	zero-shot	9.5	11.5	24.0	7.5	83.2	84.1	82.5	83.5
$\text{SimVLM}_{\text{large}}$		10.5	12.0	24.9	8.3	97.6	96.5	96.3	96.6
$\text{SimVLM}_{\text{huge}}$		11.2	14.7	32.2	8.5	101.2	100.4	102.3	101.4
$\text{SimVLM}_{\text{base}}$	few-shot	34.7	29.2	118.7	21.9	95.0	91.9	98.5	93.7
$\text{SimVLM}_{\text{large}}$		35.4	30.2	124.1	22.7	102.5	100.9	106.0	102.2
$\text{SimVLM}_{\text{huge}}$		36.8	31.5	131.3	24.0	111.8	110.6	111.0	110.4
OSCAR[†]	pretrain-finetune	**41.7**	30.6	140.0	24.5	85.4	84.0	80.3	83.4
VinVL[†]		41.0	31.1	140.9	25.2	103.7	95.6	83.8	94.3
$\text{SimVLM}_{\text{huge}}$		40.6	**33.7**	**143.3**	**25.4**	**113.7**	**110.9**	**115.2**	**112.2**

Figure 3.14: Performances for image captioning for SimVLM in several settings, on two different splits of the COCO caption dataset (Karpathy [Karpathy and Fei-Fei, 2014] test split and NoCaps [Agrawal et al., 2018] validation split). SimVLM outperforms other SOTA models, as well as BUTD (presented in section 3.5). For COCO captions, the results in four metrics are displayed (B@4: BLEU@4, M: METEOR, C: CIDEr, S: SPICE). For NoCaps, {In, Near, Out} refer to in-domain, near-domain and out-of-domain respectively. Source: [Wang et al., 2021], under the CC BY-SA License

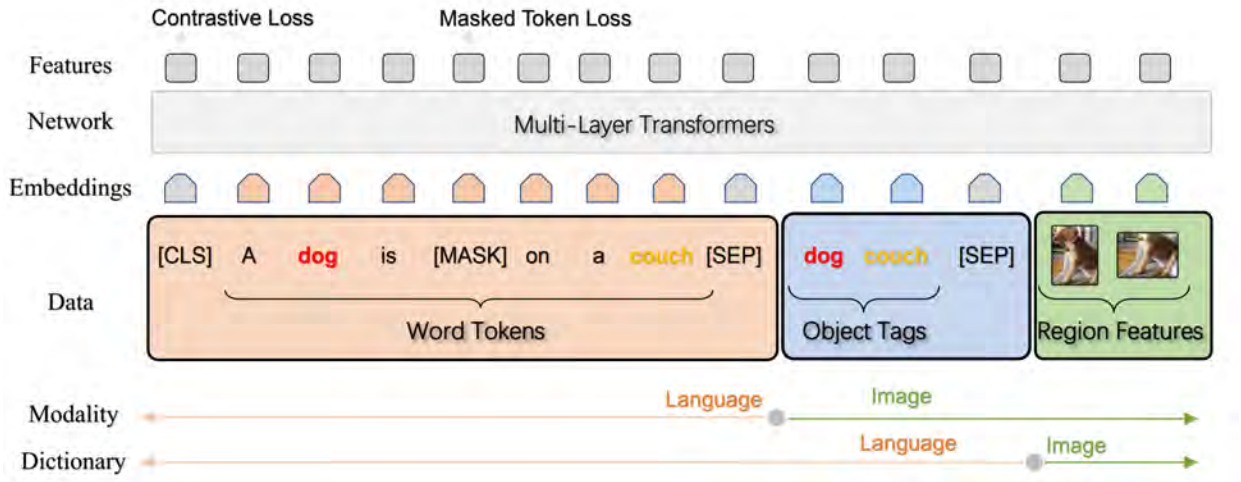

Figure 3.15: Illustration of Oscar. The image-text pair is turned into a triplet of [word tokens , object tags , region features], where the object tags (e.g., “dog” or “couch”), that allows a better visio-semantic alignement. The input triplet can be understood from two perspectives: a modality view (and thus object tags come from the visual domain) and a dictionary view (object tags are represented as word tokens). The tags constitute a modality in-between image and text. Source: [Li et al., 2020], under the CC BY-SA License

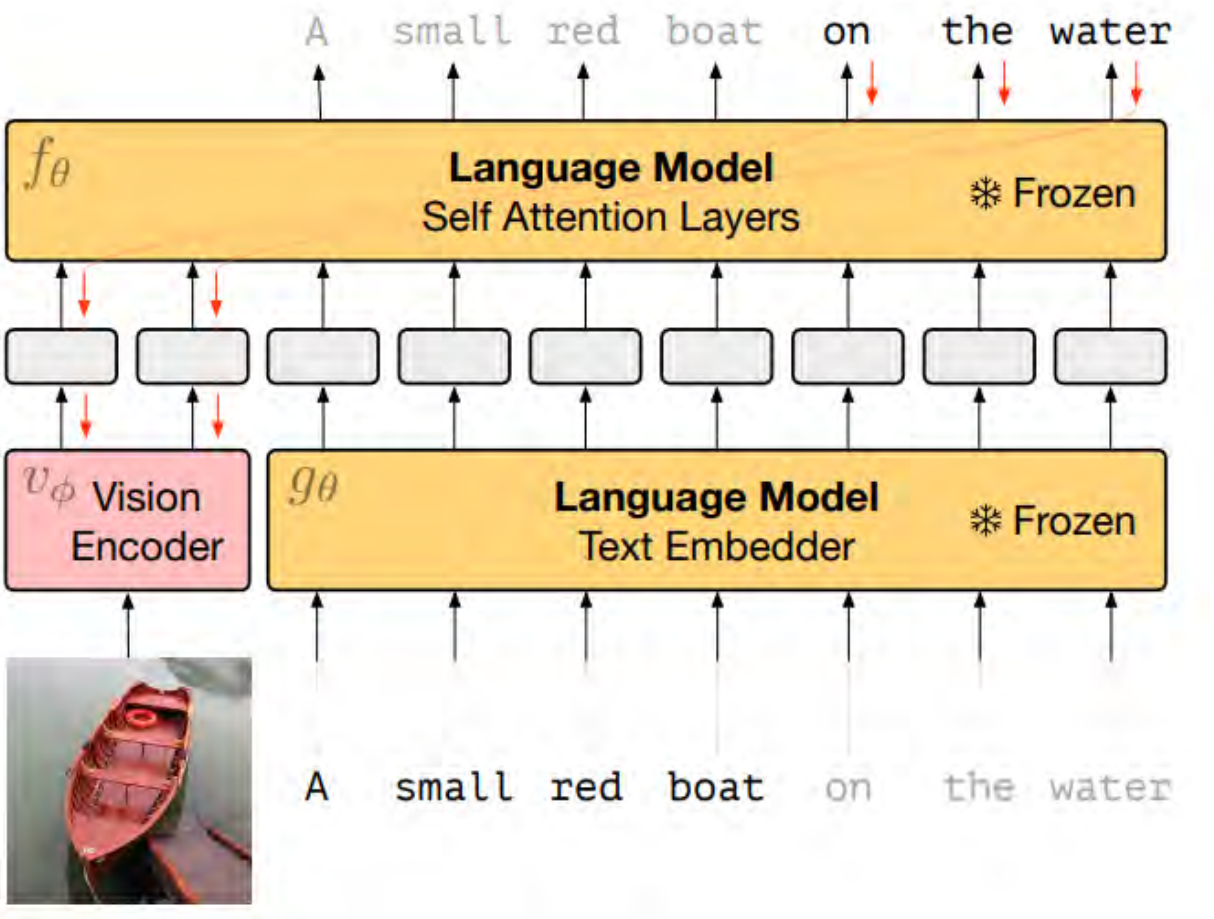

Figure 3.16: In Frozen, the text embedder and the transformer model are not trained. Therefore, the multimodality is incorporated within the vision encoder, that learns to produce features that can then be properly used by the frozen self attention model. This one uses continuous embeddings (originally produced by a language model) as input, which the vision encoder will "imitate" given a visual input. Source: [Tsimpoukelli et al., 2021], I do not own this content, all credits go to its rightful owner.

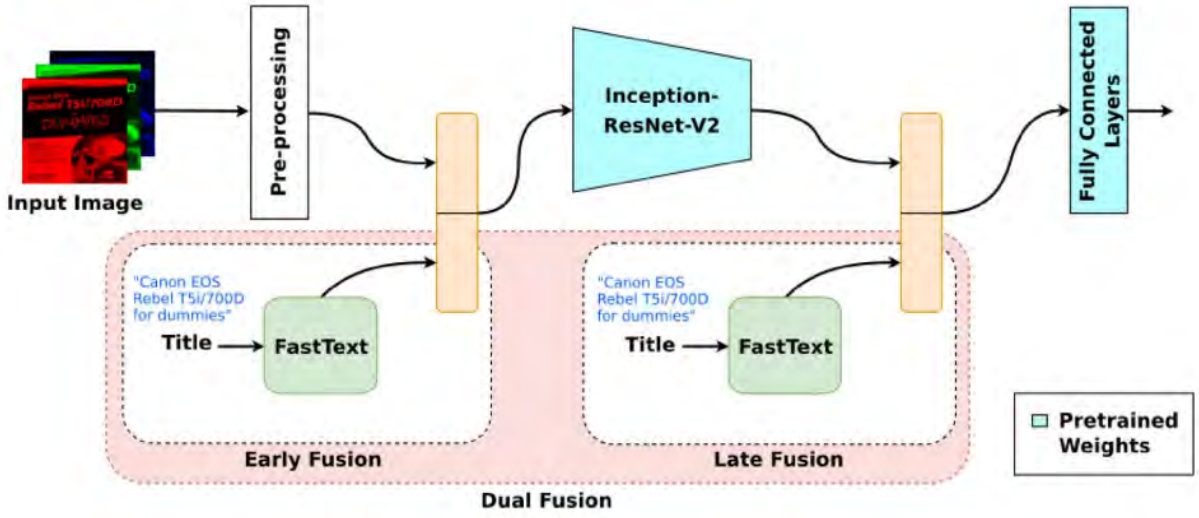

Figure 3.17: Here, 3 fusion models are represented. The best performing one is the "late-fusion" one (actually model-level fusion according to our definitions). Unimodally produced representation are concatenated and fed to a classifier that learns to use both modalities to predict the genre of the book. Source: [Lucieri et al., 2020], I do not own this content, all credits go to its rightful owner.

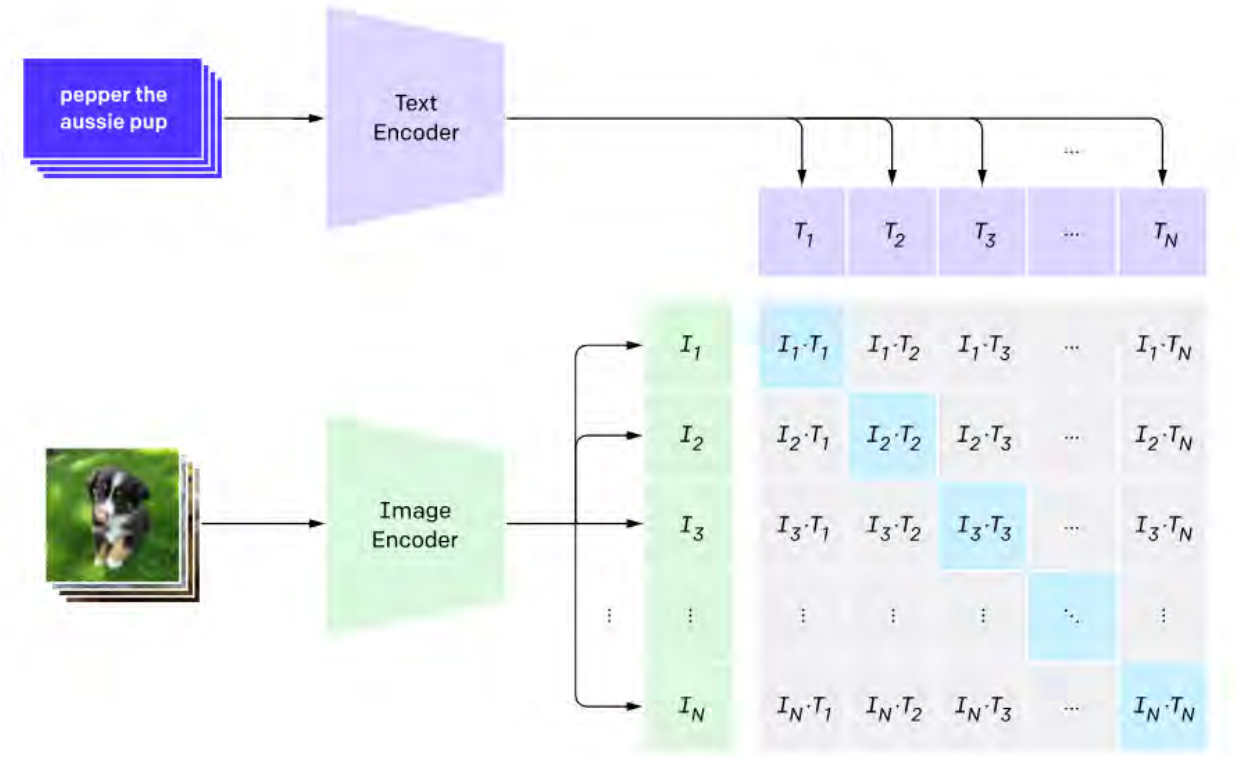

Figure 3.18: CLIP represents images and text in vectorial spaces with the same dimensionality. Its goal is to have the highest possible cosine similarity between an image and its caption, and the lowest between an image and all the other captions (each latent space is therefore coordinated with the other one during training, which allows us to consider that both are actually a single multimodal latent space). In other words, CLIP has to maximize the value in the diagonal of the matrix in the figure and to minimize the other ones. Source: [Radford et al., 2021], I do not own this content, all credits go to its rightful owner.

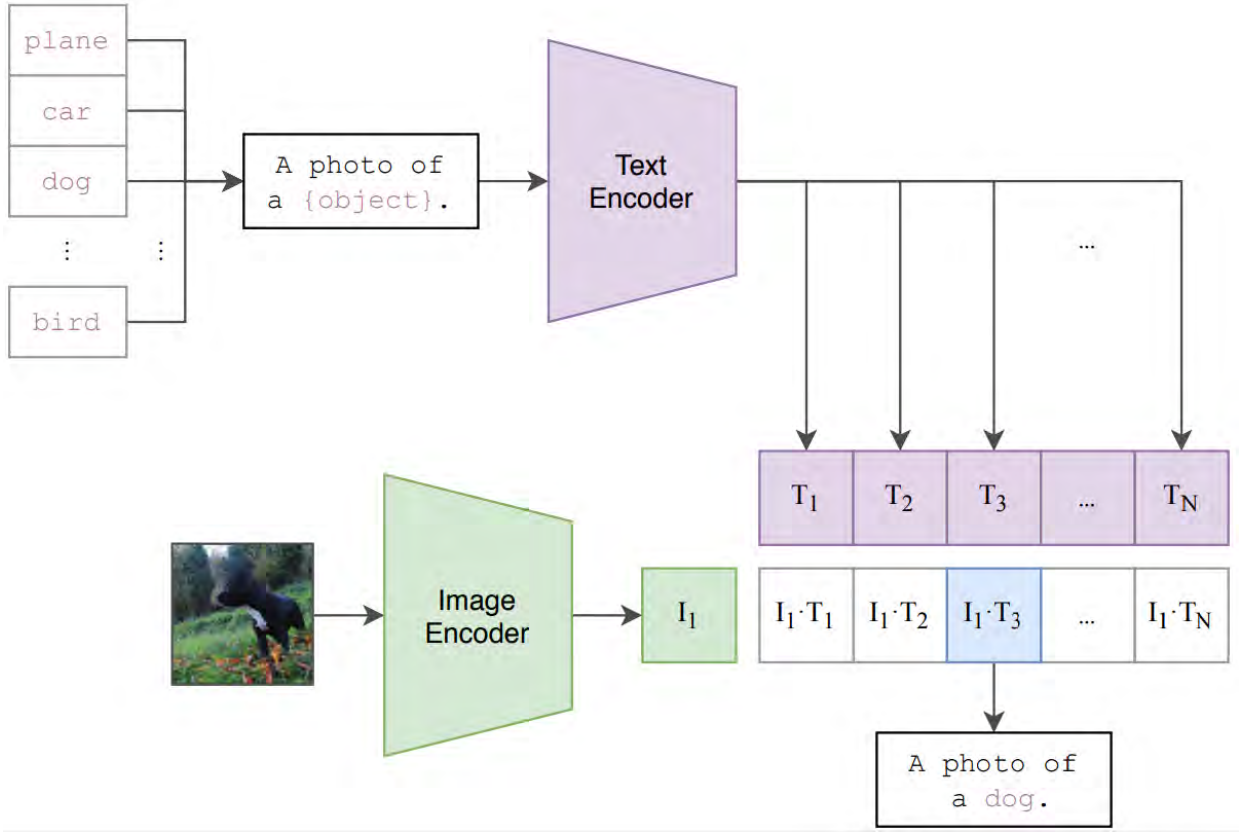

Figure 3.19: CLIP can be used for zero-shot image classification. It simply requires to find the most similar caption in CLIP's latent space, where the caption is of a standard form in which only the class name varies. Source: [Radford et al., 2021], I do not own this content, all credits go to its rightful owner.

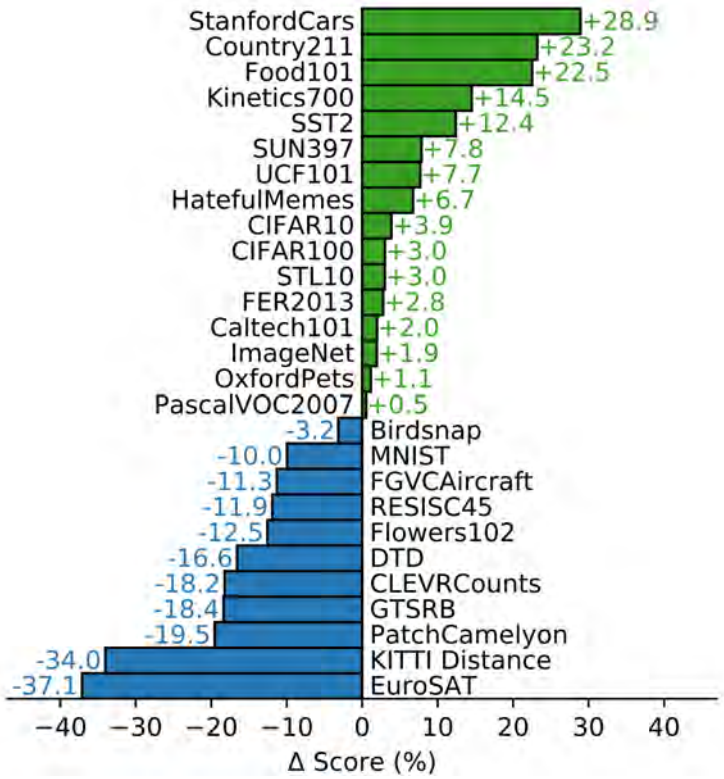

Figure 3.20: CLIP in a zero-shot setting can be compared with a fully supervised ResNet-based classifier. It is often the best performing model of the two. Source: [Radford et al., 2021], I do not own this content, all credits go to its rightful owner.

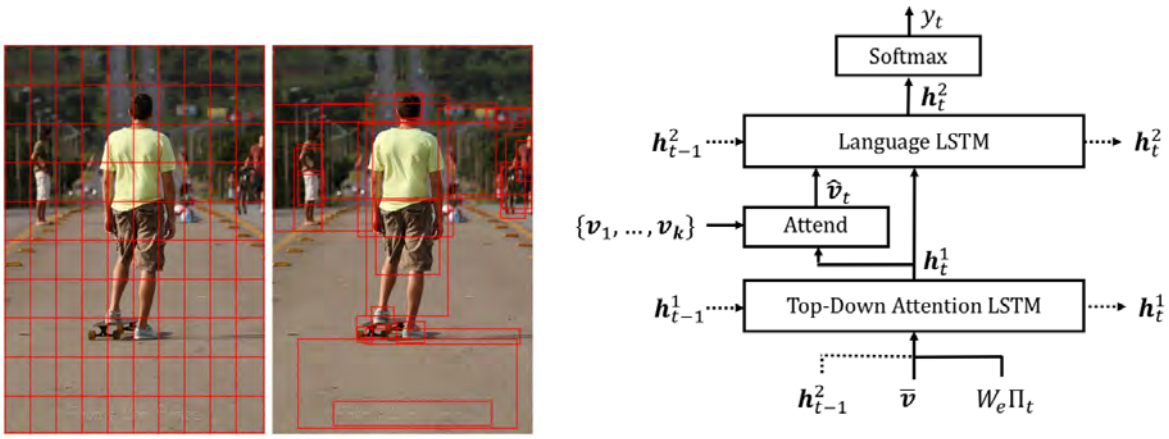

Figure 3.21: Left: Contrary to a standard convolutional neural network, BUTD uses a Faster-RCNN to extract the bounding boxes of salient objects. The features from these regions are used instead of the one that could be obtained with a fixed-size grid. Right: After the bottom-up attention mechanism has provided image features, two stacked LSTM are fed with them. The first one uses already generated text (if there is) and the image features to produce weights. They are then used to compute a weighted sum of the image feature, that the second LSTM will take as input along with its previous hidden state to generate the next word of the caption. Source: [Anderson et al., 2017], I do not own this content, all credits go to its rightful owner.

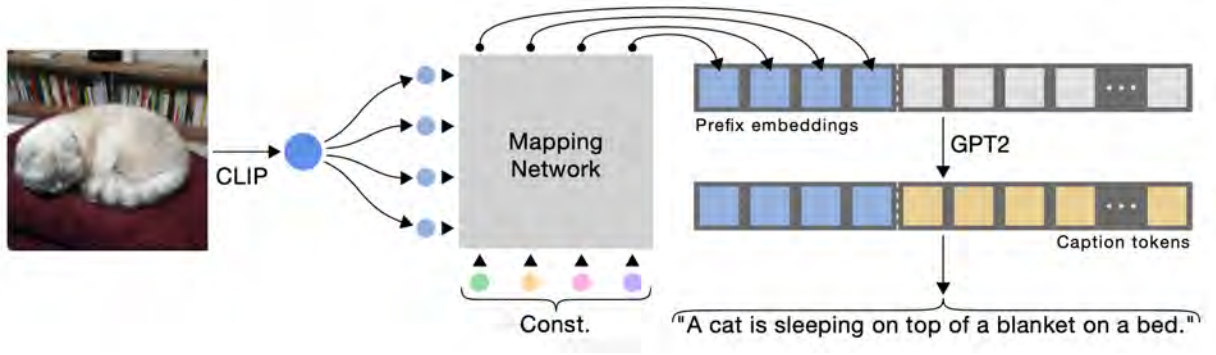

Figure 3.22: In ClipClap, a mapping network from the image feature space of CLIP is trained to learn the right prefixes for GPT2 so that it generates the correct caption. GPT2 can be fine-tuned or not in that process. Source: [Mokady et al., 2021], under CC-BY

Conceptual Captions

Model	ROUGE-L ↑	CIDEr ↑	SPICE ↑	#Params (M) ↓	Training Time ↓
VLP	24.35	77.57	16.59	115	1200h (V100)
Ours; MLP + GPT2 tuning	**26.71**	**87.26**	**18.5**	156	80h (GTX1080)
Ours; Transformer	25.12	71.82	16.07	**43**	**72h** (GTX1080)

COCO

Model	B@4 ↑	METEOR ↑	CIDEr ↑	SPICE ↑	#Params (M) ↓	Training Time ↓
BUTD [4]	36.2	27.0	113.5	20.3	52	960h (M40)
VLP [47]	36.5	28.4	117.7	21.3	115	48h (V100)
Oscar [19]	**36.58**	**30.4**	**124.12**	**23.17**	135	74h (V100)
Ours; Transformer	33.53	27.45	113.08	21.05	**43**	**6h** (GTX1080)
Ours; MLP + GPT2 tuning	32.15	27.1	108.35	20.12	156	7h (GTX1080)

Figure 3.23: *Ours* refers to the ClipClap model. ClipClap is often competitive with other SOTA models, and sometimes the best of all. It also has very efficient training, as it leverages the properties of the already pretrained latent space of CLIP. Source: [Mokady et al., 2021], under CC-BY





GPT-2 tuning	Caption	a motorcycle is on display in a showroom.	a group of people sitting around a table.	a living room filled with furniture and a book shelf filled with books.	a fire hydrant is in the middle of a street.	display case filled with lots of different types of donuts.
	Prefix	com showcase motorcycle A ray motorcycle-posed what polished Ink	blond vegetarian dishes dining expects smiling friendships group almost	tt sofa gest chair Bart books modern doorway bedroom	neon street Da alley putistan colorful nighttime	glass bakery dough displays sandwiches2 boxes Prin ten
w/o tuning	Caption	motorcycle that is on display at a show.	a a group of people sitting at a table together.	a living room with a couch and bookshelves.	a fire hydrant in front of a city street.	a display case full of different types of doughnuts.
	Prefix	oover eleph SniperÃÂÃÂ motorcycle synergy undeniably achieving\n	amic Delicious eleph SukActionCode photographers interchangeable undeniably achieving	orianclassic eleph CameroonÃÂÃÂÃÂroom synergy strikingly achieving\n	ockets Pier eleph SniperÃÂÃÂ bicycl synergy undeniably achieving\n	peanuts desserts elephbmÃÂÃÂÃÂ cooking nodd strikingly achieving\n

Figure 3.24: Some captions generated by ClipClap, with and without fine-tuning GPT2. Source: [Mokady et al., 2021], under CC-BY

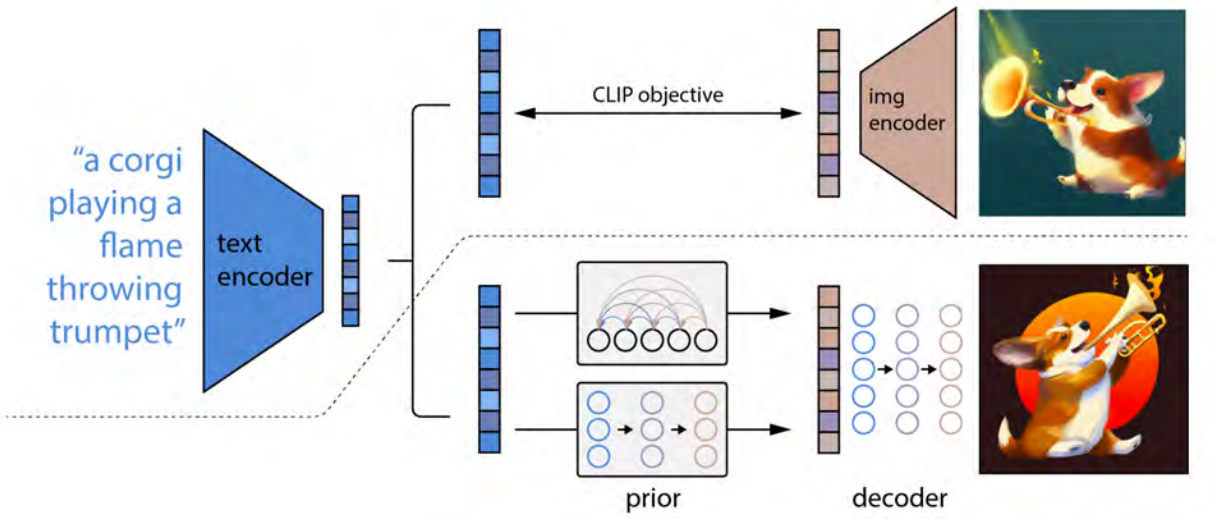

Figure 3.25: The text is first encoded through CLIP's textual pipeline (which, as shown at the top, has been trained to make the encoding of an image and its caption as close as possible in its latent space), then a latent translator turns the sentence embedding into a "fake" image feature, that is then used by a diffusion model [Sohl-Dickstein et al., 2015] to create an image. Source: [Ramesh et al., 2022], under CC-BY-SA.

Figure 3.26: Images generated by DALL-E 2 and below them, the caption that guided the generation. Source: [Ramesh et al., 2022], under CC-BY-SA.

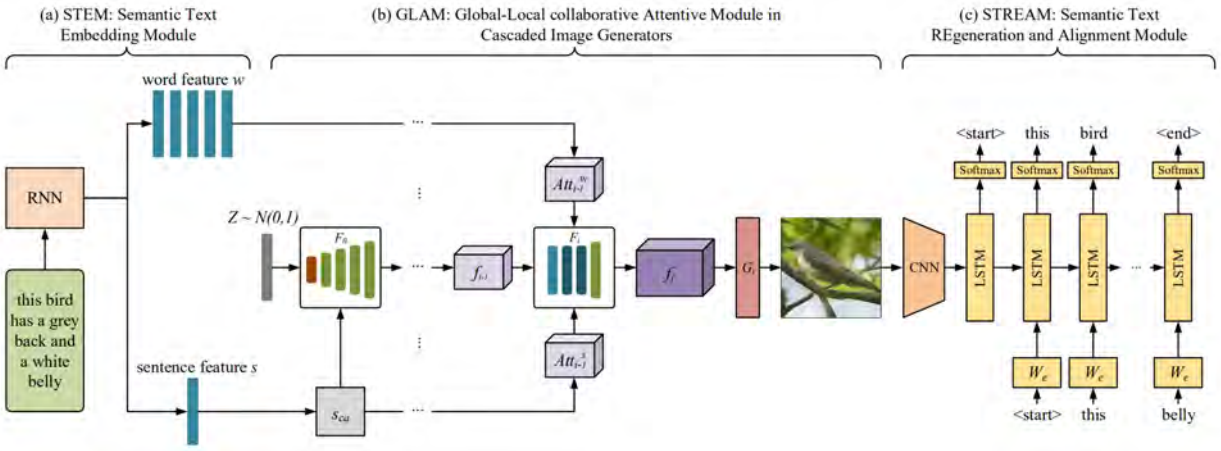

Figure 3.27: The three modules of MirrorGAN. The first one (STEM) encodes text-only, the second (GLAM) uses bimodal attention between semantic embeddings and generated image features, the third one (STREAM) is a CNN that generates features based on the generated image, that is followed by a RNN that generates a caption. Source: [Qiao et al., 2019], under CC-BY-SA.

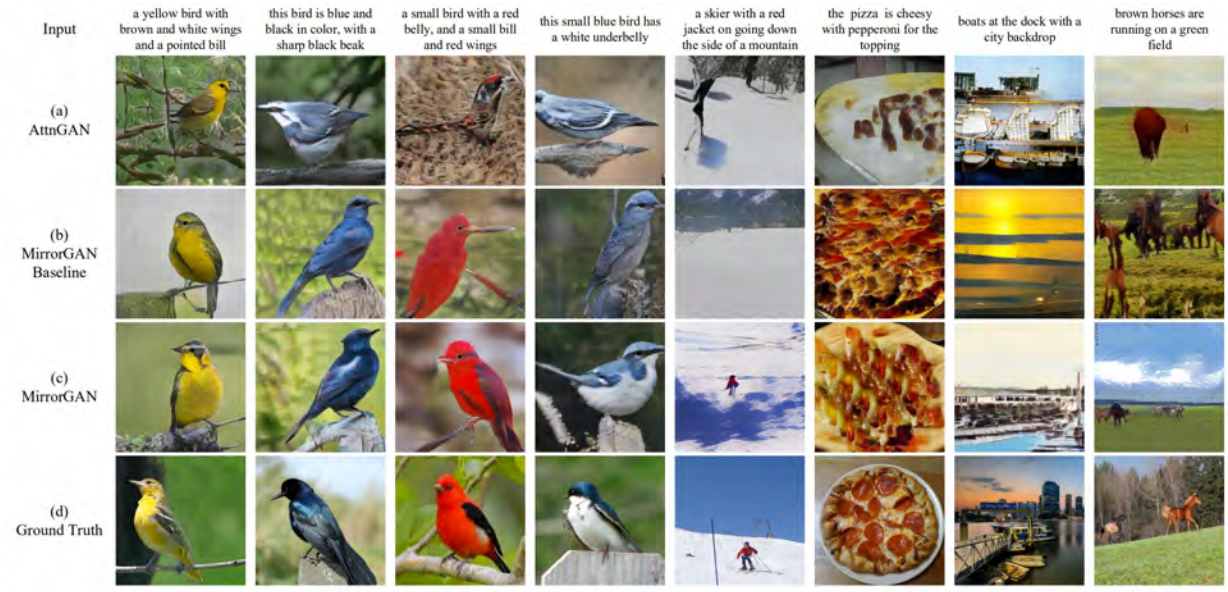

Figure 3.28: MirrorGAN is compared with AttnGAN [Xu et al., 2017], a version where only word-level attention is used for the image generation (and not the sentence level one) called MirrorGAN Baseline and the ground truth corresponding to the caption on top. Source: [Qiao et al., 2019], under CC-BY-SA.

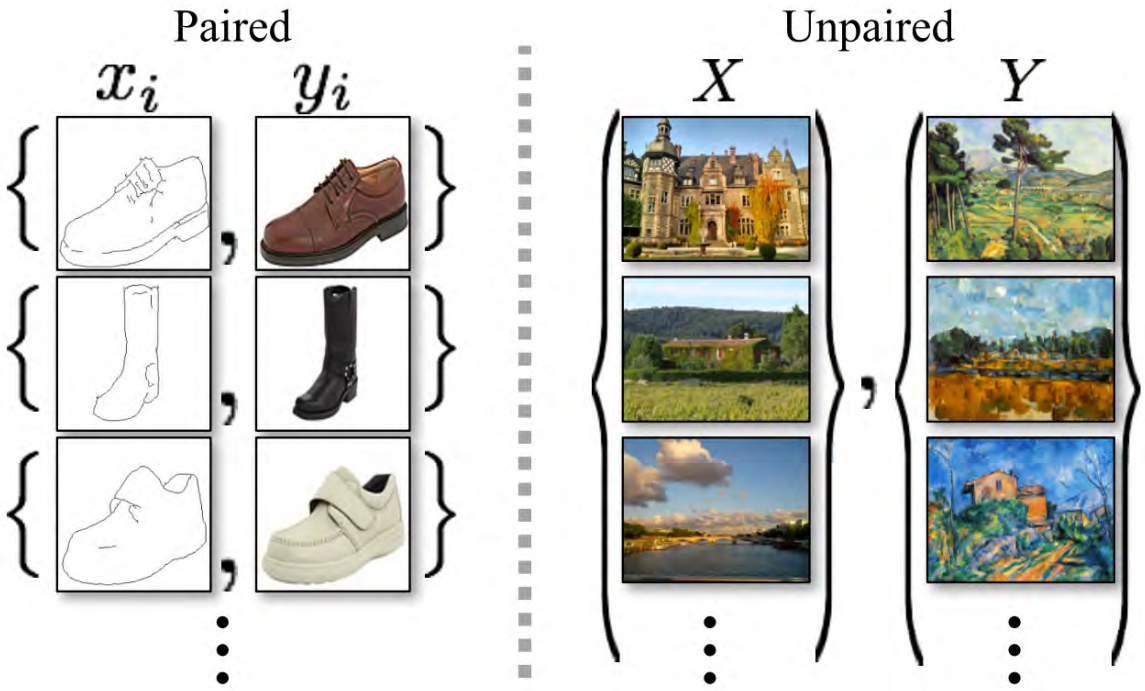

Figure 3.29: Paired vs unpaired data. CycleGAN has been trained with unpaired set of data from different distribution. In this example, it will learn to turn pictures into Cézanne-style painting and Cézanne paintings into pictures (see Figure 3.31 for an example of translation). Source: [Zhu et al., 2017], under CC-BY-SA.

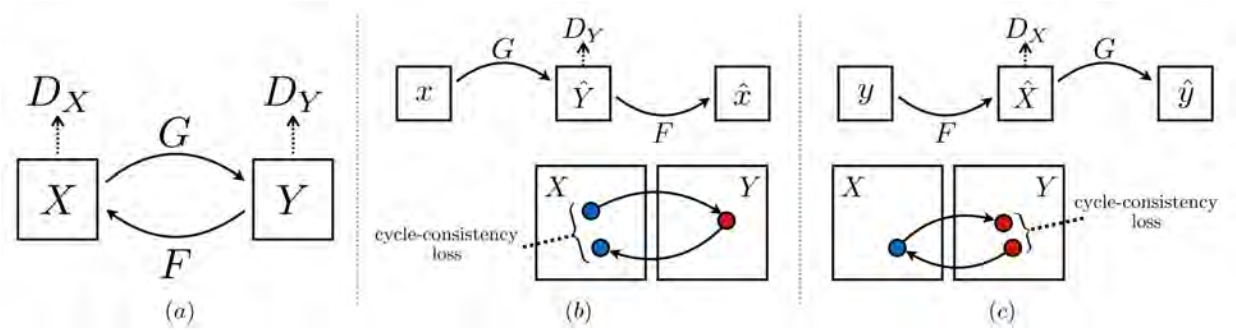

Figure 3.30: (a) G and F, the two generators, are trained to fool respectively D_X and D_Y, their corresponding discriminators. (b) and (c) show the two ways of computing the cycle-consistency loss, starting from each of the two distributions. Source: [Zhu et al., 2017], under CC-BY-SA.

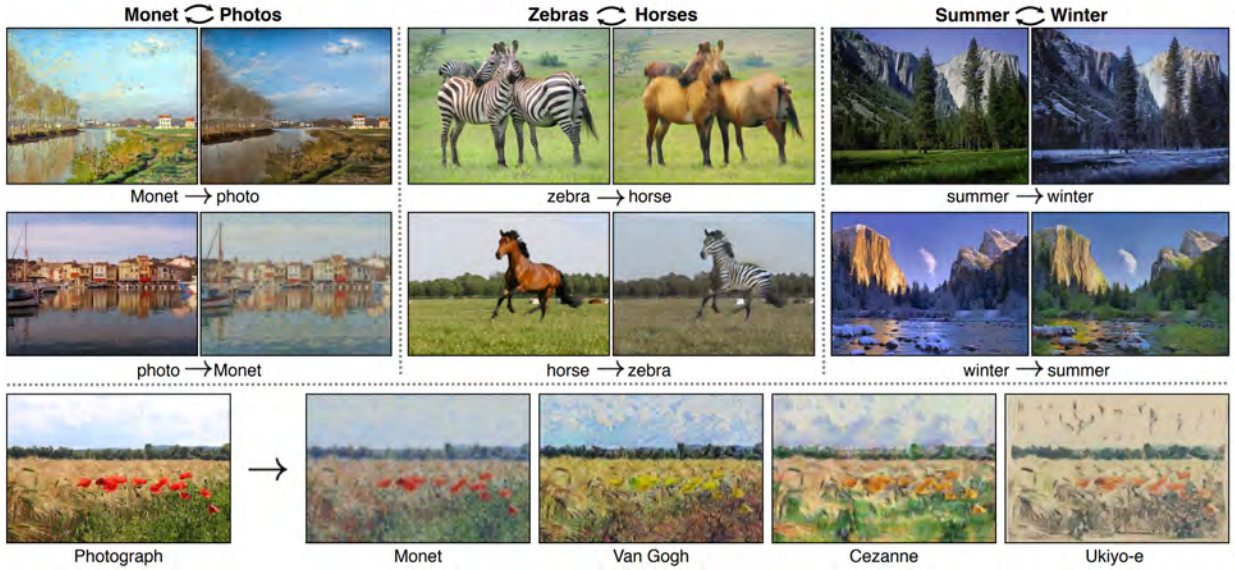

Figure 3.31: Several examples of what different CycleGANs, trained on different pairs of distribution, can produce. Source: [Zhu et al., 2017], under CC-BY-SA.

Chapter 4

Generalisation abilities of multimodal models on standard visual tasks

4.1 Preamble

In this chapter, we will mostly focus on the paper by [Devillers et al., 2021], to which I contributed, and that precedes the work described in chapter 5. I am not first author on the paper, however I participated in its elaboration and redaction. I mostly participated in the writing of the representational analysis and the linguistic parts and in the analysis of the results of these sections. Along with the other authors, I elaborated the global interpretation of all the results, and wrote parts of the introduction and of the conclusion. This paper is displayed here because the work of my first published paper strongly relates to this one and complete its result, in order to produce a more general statement on the generalization abilities of CLIP.

This paper aims mostly at evaluating the generalization abilities of the visual

side of multimodal models and to compare them with visual-only models. It focuses on standard visual tasks (object recognition), and on similarity between visually constrained word representations.

4.2 Comparing multimodal and unimodal models on standard visual tasks

Does language help generalization in vision models?

Benjamin Devillers, Bhavin Choksi, Romain Bielawski & Rufin VanRullen

Abstract

Vision models trained on multimodal datasets can benefit from the wide availability of large image-caption datasets. A recent model (CLIP) was found to generalize well in zero-shot and transfer learning settings. This could imply that linguistic or "semantic grounding" confers additional generalization abilities to the visual feature space. Here, we systematically evaluate various multimodal architectures and vision-only models in terms of unsupervised clustering, few-shot learning, transfer learning and adversarial robustness. In each setting, multimodal training produced no additional generalization capability compared to standard supervised visual training. We conclude that work is still required for semantic grounding to help improve vision models.

Introduction

Learning vision models using language supervision has gained popularity [Quattoni et al., 2007, Srivastava et al., 2012, Frome et al., 2013, Joulin et al., 2016b, Pham et al., 2019, Desai and Johnson, 2020, Hu and Singh, 2021, Radford et al., 2021, Sariyildiz et al., 2020a] for two main reasons: firstly, vision-language training

allows to build massive training datasets from readily available online data, without manual annotation; secondly, language provides additional semantic information that cannot be inferred from vision-only datasets, and this could help with semantic grounding of visual features.

Recently [Radford et al., 2021] introduced CLIP, a language and vision model that shows outstanding zero-shot learning capabilities on many tasks, and compelling transfer-learning abilities. A recent report [Goh et al., 2021] showed that CLIP produces neural selectivity patterns comparable to "multimodal" concept cells observed in the human brain [Quiroga et al., 2005, Reddy and Thorpe, 2014]. From these results, it is tempting to assume that CLIP's generalization properties stem from semantic grounding provided by the joint vision-language training.

Here, we show that CLIP and other vision-language models do not perform better than vision-only, fully supervised models on a number of generalization settings and datasets. Representation similarity [Kriegeskorte et al., 2008] analysis reveals that the multimodal representations that emerge through vision-language training are different from *both* linguistic and visual representations–and thus possibly unsuitable for transfer-learning to new visual tasks. In conclusion, additional work on linguistic grounding is still needed, if it is to improve generalization capabilities of vision models.

We provide our code for reproducibility[1].

[1] `https://github.com/bdvllrs/generalization-vision`

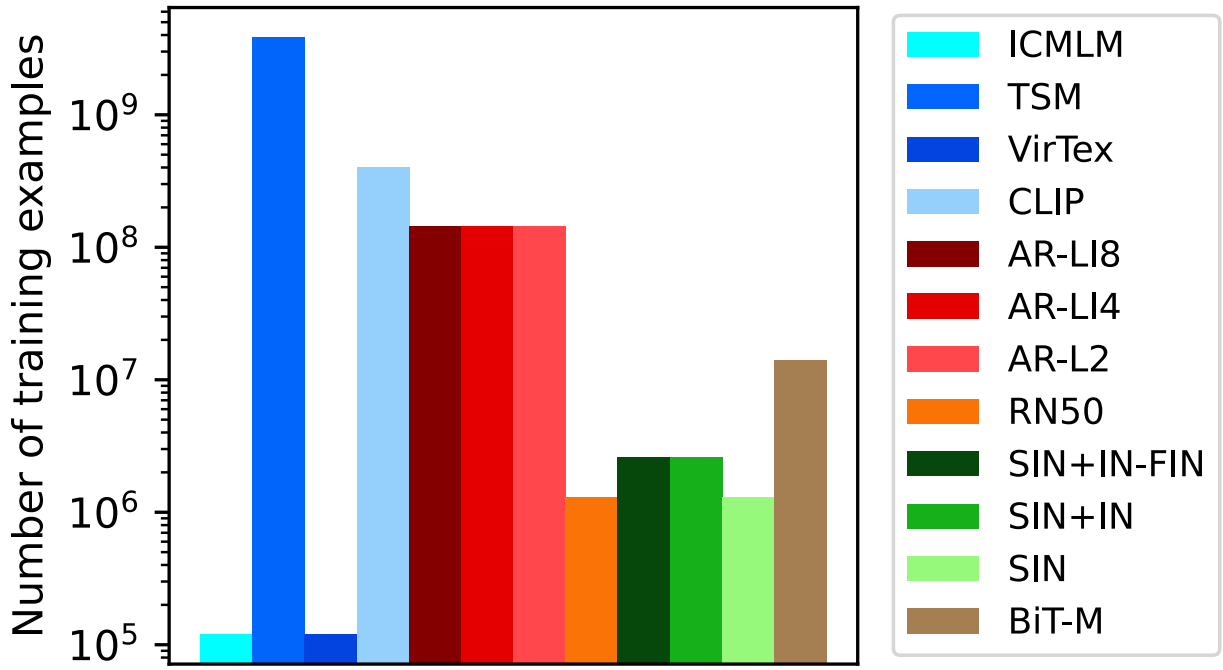

Figure 4.1: Size of the training dataset used by the models (number of images, in log scale). ICMLM and VirTex are trained on COCO, TSM on HowTo100M, CLIP on a (not publicly available) scrape of the internet, RN50 is trained on ImageNet-1k, the AR models and SIN models are trained on augmented versions of ImageNet-1k and BiT-M is trained on ImageNet-21k.

Models

We use a number of publicly available vision, text or multimodal pretrained models, and compare their representations and generalization abilities. To facilitate interpretation and comparisons between the models, Figure 4.1 reports the training dataset size for each of the visual models (including the vision-language models). They are all based on the same backbone (a ResNet50 architecture).

In CLIP, the authors train the joint embedding space of a visual network (hereafter called simply CLIP) and a language network (hereafter called CLIP-T) using contrastive learning on 400M image-caption pairs. Note that in the present paper, the visual backbone of CLIP is a ResNet50, even though the visual-transformer-based CLIP model could reach higher performance; this choice allows for a fair comparison with the other visual models that are all based on the ResNet50 architecture. In addition, we also consider TSM [Sariyildiz et al., 2020b], another multimodal network trained with a contrastive loss on video, audio and text inputs from the HowTo100M dataset [Miech et al., 2019] (containing more than 136M video clips with captions. For training, the authors effectively used 120M video clips of 3.2s sampled at 10 fps). The effects of CLIP's and TSM's contrastive training paradigm can be compared with VirTex and ICMLM—two other recent multimodal networks. In VirTex, the visual feature representations are optimized for an image captioning task [Desai and Johnson, 2020], and for a text-unmasking task in ICMLM [Sariyildiz et al., 2020a]. Such text-based objectives aim to provide a form of linguistic grounding using significantly fewer images than CLIP (VirTex and ICMLM models are trained on the COCO dataset [Lin et al., 2014] with approximately 120K captioned images).

To understand the potential effects of linguistic training, we compare the multimodal networks to vision-only networks. We include a baseline architecture (ResNet50) trained on ImageNet-1K [He et al., 2015] (1.3M labelled images). Second, we consider a similar architecture (ResNet50 backbone) called BiT-M [Kolesnikov et al., 2019], trained on ImageNet-21K, a much larger dataset (14M labelled images).

While generalization and robustness properties can often be derived from ac-

cess to large labelled image datasets (as in BiT-M), obtaining such labels is costly. An alternative is to train models with additional datapoints based on assumptions about the real-life data distribution–as done, e.g., with adversarial training. In this study, we use the Adversarially Robust (AR) ResNet50 models provided by [Engstrom et al., 2019b], trained on the 1.3M ImageNet training set plus 110 adversarial attacks of each image (i.e. more than 140M images overall). The different model variants (AR-L2, AR-LI4, AR-LI8) correspond to distinct adversarial attacks (refer to [Engstrom et al., 2019b] for more details). This adversarial training was found to produce more perceptually aligned features and to improve generalization (e.g. transfer learning) in some settings [Salman et al., 2020]. Another such technique was used for StylizedImageNet (SIN) models [Geirhos et al., 2019], where a variant of the ImageNet dataset (1.3M images) was designed via style-transfer to specifically reduce the network's reliance on texture information. The authors provide weights for models that are (i) only pretrained for SIN images (SIN), (ii) trained on SIN and ImageNet (SIN+IN) combined, or where (iii) a SIN+IN model is finetuned on ImageNet (SIN+IN-FIN).

For the vanilla ResNet50, SIN, AR and BiT-M models, we use activations after the final average pooling operation as feature representations. Although all these models share a ResNet50 backbone, there are minor differences in their implementations. We assume that such small architectural differences would not dramatically affect the feature spaces learned by these models.

Finally, we also use two text-only language models, GPT-2 [Radford et al., 2018b] and BERT [Devlin et al., 2018], in our feature-space comparisons. As these models are not designed to process visual inputs, they cannot be tested on visual generalization; but we can use their representations of class *labels* (or sentence

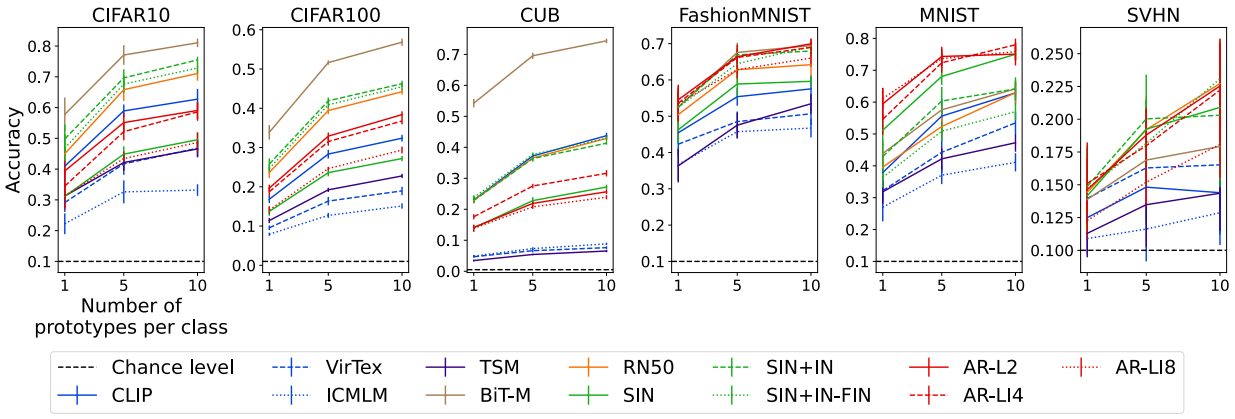

Figure 4.2: 1-shot, 5-shot and 10-shot accuracy over our evaluation datasets. Multimodal networks (ICMLM, VirTex, CLIP, TSM, in blue) have typically worse performance than the other models for all datasets.

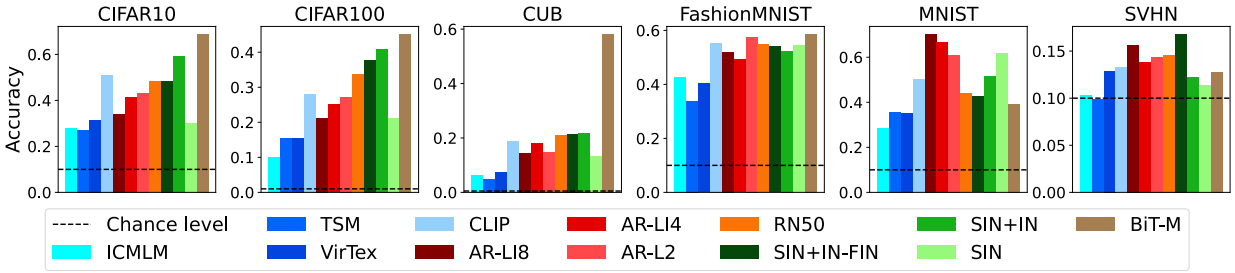

Figure 4.3: Unsupervised clustering accuracy over our evaluation datasets. Clustering is obtained using Scikit-learn Spectral Clustering algorithm. Multimodal networks (ICMLM, VirTex, CLIP, TSM, in blue) are worse than vision-only models (in various colors) on average.

captions) as a basis for comparison with visual or multimodal network representations. In a similar way, the language stream of the CLIP model (CLIP-T) can be treated as a third language model for our comparisons.

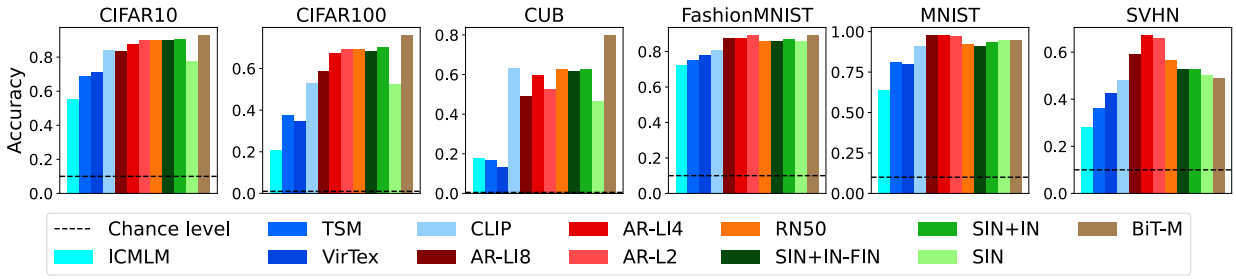

Figure 4.4: Transfer learning accuracy over our evaluation datasets. For each dataset and model, we train a linear layer to classify the models' visual features. Multimodal networks (ICMLM, VirTex, CLIP, TSM, in blue) have worse performance accuracy than vision-only models (in various colors).

Generalization tasks

In [Radford et al., 2021], CLIP was systematically tested in a zero-shot setting. However, this requires a language stream to describe the different possible targets, which is not available in standard vision models. To compare the generalization capabilities of multimodal and vision-only models, we thus focus on few-shot, transfer and unsupervised learning. In each case, we evaluate performance on MNIST [Lecun et al., 1998], CIFAR10, CIFAR100 [Krizhevsky et al.,], Fashion-MNIST [Xiao et al., 2017], CUB-200-2011 (CUB) [Wah et al., 2011] and SVHN [Netzer et al., 2011][2]. These datasets test generalization capabilities for natural images of various classes.

Few-shot learning

As a first generalization experiment, we compare few-shot learning accuracy. For this experiment, we directly pass N randomly selected samples for each class

[2]For more details on these datasets, see appendix **??**.

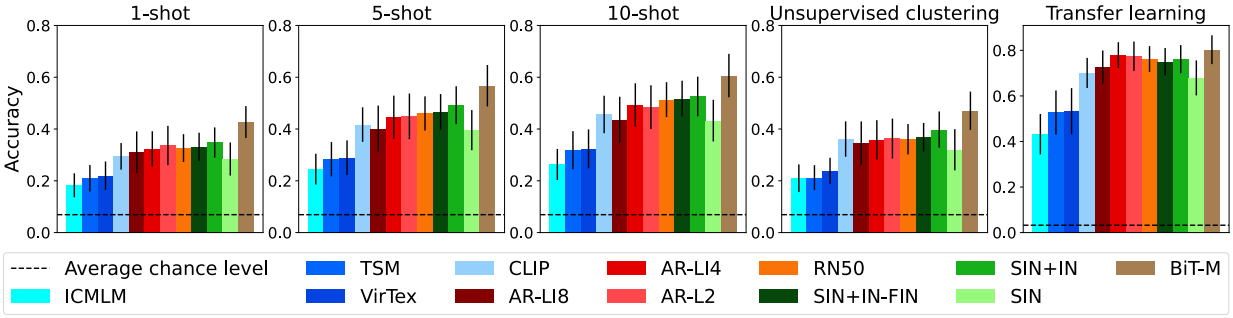

Figure 4.5: Average performance of the models across datasets, with standard error of the mean, for the various generalization tasks (few-shot learning, unsupervised clustering, transfer learning). Multimodal networks (ICMLM, VirTex, CLIP, TSM in blue) have worse generalization accuracy across all tasks.

(N-shot learning) through the pretrained models to obtain a feature representation for each sample. Then, we define a class prototype by averaging the feature representations of all the samples in each class. We measure the performance of vision-only and text-vision models for $N = 1$, 5 and 10. Each time, the reported performance is averaged over 10 trials with different class prototypes (i.e., different random selection of samples). Figure 4.2 shows the performance of each model on each dataset. For CIFAR10, CIFAR100 and CUB (all the natural images datasets), BiT-M has the best accuracy. On the other hand, ICMLM, VirTex, CLIP and TSM do not perform better than the vision-only models.

Figure 4.5 shows the average performance of each model across datasets, in the leftmost 3 panels.

Unsupervised clustering

Our second generalization test is an unsupervised clustering task over the same datasets. For this, we apply an out-of-the-box spectral clustering algorithm [Pedregosa et al., 2011] using the cosine of two feature vectors as a metric. We provide the number of required clusters (number of classes) to the clustering algorithm: this ensures that all classes are represented by a cluster. The clusters are computed only on the test-sets.

To compute the accuracy on the prediction, we need to assign labels to each cluster. To do so, we use a greedy algorithm: we first choose the cluster containing the most elements in common with a given class and assign it the corresponding label. We then continue with the second cluster that has the most elements in common with another class, and so on until all clusters have been labelled.

Figure 4.3 shows the unsupervised clustering performance on individual datasets. It shows a similar ranking to the few-shot learning task where BiT has the best performance overall and the visio-linguistic models lag behind the vision-only models. Figure 4.5 panel 4 (from left) shows the performance of the unsupervised clustering algorithm averaged over all datasets.

Transfer learning

To further evaluate the models' generalization properties, we use a transfer learning setting as described in [Salman et al., 2020]. We use the same datasets as in the other tasks, each time training a linear probe using the Adam optimizer. We train each linear probe for 20 epochs with a learning rate of 1e-3 and a weight decay of 5e-4.

Fig 4.4 shows the performance of the models on this task, separately for each dataset, and Fig 4.5 (rightmost panel) reports the average across datasets. Multimodal networks fail again to improve generalization.

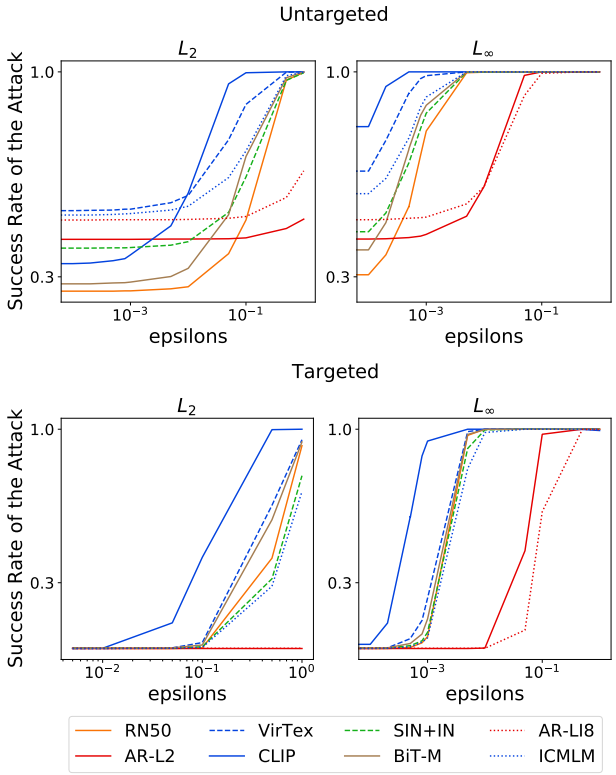

Figure 4.6: Robustness of some of the models to untargeted (top) and targeted (bottom) random projected gradient descent (RPGD) attacks for varying epsilons, with L_2 (left) or L_∞ norm (right). AR models are robust by design. Multimodal networks (CLIP, VirTex) are less robust than vision-only models (RN50, SIN+IN, BiT-M).

Robustness to adversarial attacks

Another important test for generalization is the robustness to input perturbations (a form of out-of-distribution generalization). Here, we compare the adversarial robustness of different models against untargeted and targeted random projected gradient descent (RPGD) attacks [Madry et al., 2017]. We use L_2 and L_∞ norms to distinguish any norm-specific effects. Figure 4.6 shows the success rate of the 100-step RPGD attacks on 1000 images taken from the ImageNet validation set. We use the foolbox API [Rauber et al., 2017] to perform all the attacks with configurations provided by [Engstrom et al., 2019a].

Summary

Overall, models trained with multimodal information (CLIP, VirTex, ICMLM, TSM) do not achieve better performance than the visual-only ResNet-based models. This systematic observation across multiple image datasets and generalization tasks (including few-shot, transfer and unsupervised learning, as well as adversarial robustness) goes against the assumption that linguistic grounding should help generalization in vision models.

Among the multimodal networks, CLIP does indeed appear to be more generalization-efficient than VirTex, ICMLM and TSM. As mentioned in [Radford et al., 2018b], directly predicting highly variable text captions (as done in VirTex or ICMLM) is a difficult task that does not scale well. CLIP (and TSM) avoid generating text, relying instead on a contrastive loss between visual and linguistic embeddings. However, even with the potential benefits provided by this contrastive loss, CLIP (and TSM) do not outperform the vision models.

Finally, BiT-M, a simple vision-only model trained on a very large labelled dataset, turns out to be the overall best performing model for few-shot learning, unsupervised clustering and transfer learning, and on par with the standard ResNet50 for adversarial robustness.

Although these results are fairly consistent across datasets, there are still some differences.

For the CUB dataset, BiT-M largely outperformed the other models. This result is to be expected as the bird species in CUB are also part of ImageNet-21K labels. Then, among visio-linguistic models, CLIP is the only one competitive with the remaining visual models on this dataset.

MNIST and SVHN require classification of digits. According to [Radford et al., 2018b], CLIP should be able to generalize to this task, as its training set included numerous images with text and digits. Indeed we observe that CLIP can perform as well as some of the vision models for these datasets. However, SIN and AR models perform generally better than other models.

For datasets with more natural images (CIFAR, FashionMNIST, CUB), vision models are generally better than their visio-linguistic counterparts.

Model comparison

To better understand the similarities and differences between the feature spaces learned by the various models, we now compare them using RSA [Kriegeskorte et al., 2008].

Method RSA is a comparison method originally used to compare fMRI data. It allows us to compare different models (with different latent space dimensions,

norms, ...) which share the same structure.

This works by comparing the models' *()*. RDMs are obtained by computing the 2 by 2 distances for each class of the latent representations (see figure 4.7). More specifically, for each visual model, we define for each class the set $\mathcal{F}_c$ containing the feature vectors of all the images of class c, its average $\bar{f}_c$ and its standard deviation σ_c. The RDM matrix is then defined as $[RDM_{i,j}]$ where

$$RDM_{i,j} = \left\| \frac{\bar{f}_i - \bar{f}_j}{\sqrt{\frac{\sigma_i^2}{|\mathcal{F}_i|} + \frac{\sigma_j^2}{|\mathcal{F}_j|}}} \right\|_2 \tag{4.1}$$

for each pair of class (i, j).

We use the norm of the unequal variance t-test [Welch, 1947] as our distance metric between the latent representations, because it allows to normalize the distances between class centroids with respect to their variances. Indeed, each class is represented by a cluster of latent vectors of different sizes.

In the case of language models (all transformer-based), we use as latent representations, the encoding of the sentence "a photo of x." where we replace "x" by the corresponding label. We then use the contextualization of the label as the text feature vector. Compared to the vision models, there is only one representation per class (only one sentence per class) hence a lack of variance associated with the feature vector of each class. As a result, the distance used in the RDM matrix becomes an L_2 norm.

The RDM matrix obtained with this method contains the respective distances between pre-defined concepts (in our case the 1000 classes of ImageNet). RDMs can therefore be considered as a standardized representation of latent spaces. This

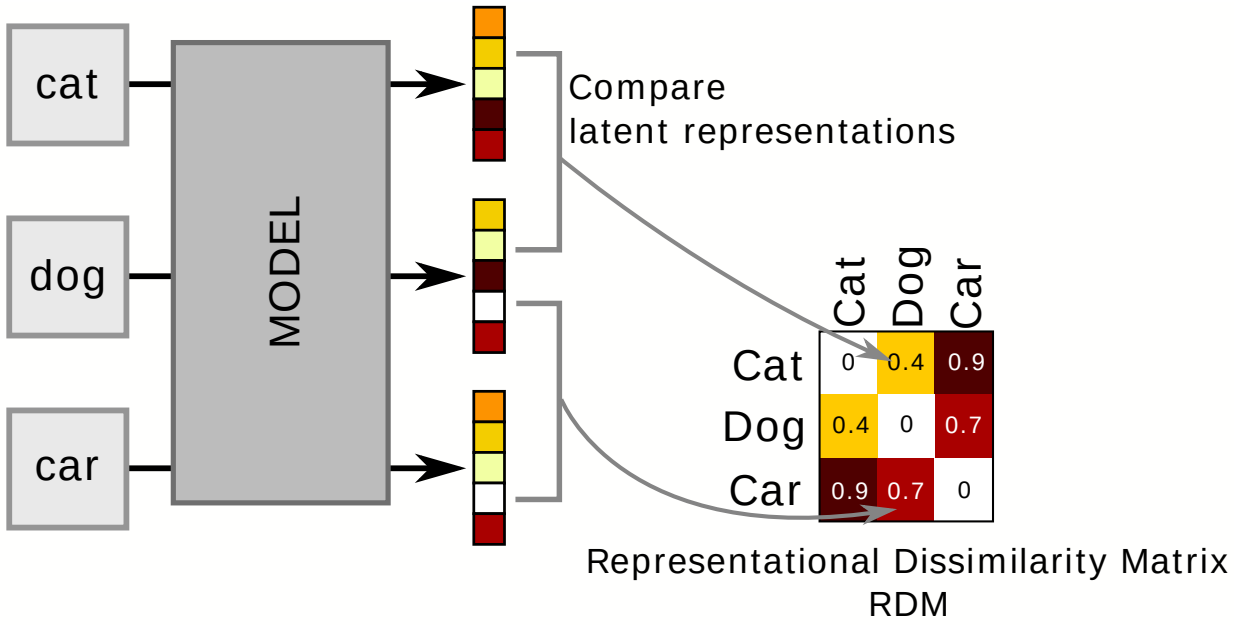

Figure 4.7: How to compute representational Dissimilarity Matrices (RDMs). RDMs are built from the model's embedding space. The RDMs can then be used for a Representational Similarity Analysis by comparing them using a Pearson Correlation.

means that we can compare our models' representations by computing the Pearson correlation between their respective RDMs. The corresponding comparison matrix, for all pairs of models, is illustrated in Fig 4.8.

Results Figure 4.9 shows the results of a hierarchical clustering (a) or t-SNE [**?**] embedding (b) of the RDMs using Pearson correlation as a distance. Looking at the dendrogram, all the vision-only models are very close to one another with a maximum distance <0.2. Then, multimodal models stand a bit further (CLIP, TSM, VirTex, ICMLM); and finally, CLIP-T and the language models

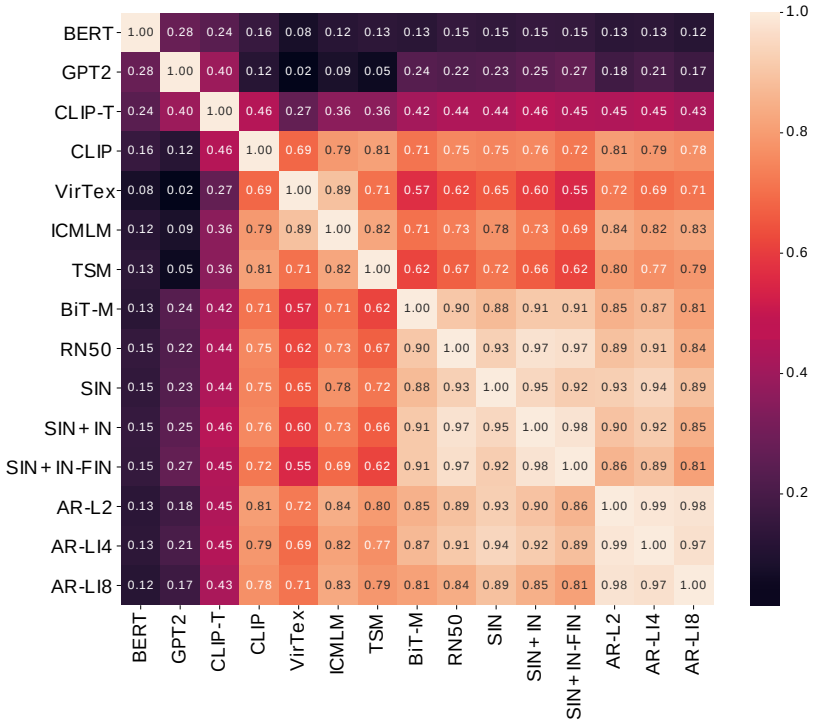

Figure 4.8: Correlations of the RDMs of our evaluation models. The RDMs are computed as explained in Fig 4.7 using the ImageNet dataset.

(BERT, GPT2) are the furthest away. This indicates that the language supervision (contrastive embedding, text-generation or text-unmasking objectives) has changed the structure of the ResNet latent space for CLIP, TSM, VirTex and ICMLM models (respectively). Yet these multimodal models are not truly linguistic either, as they are very distant also from the standard language models.

This conclusion is also supported by the t-SNE plot, showing a cluster of BiT-M, RN50 and SIN vision models, a second cluster with the AR models, and further along the same direction, the multimodal networks (CLIP, VirTex, ICMLM, TSM). Note that, although this arrangement might suggest that multimodal net-

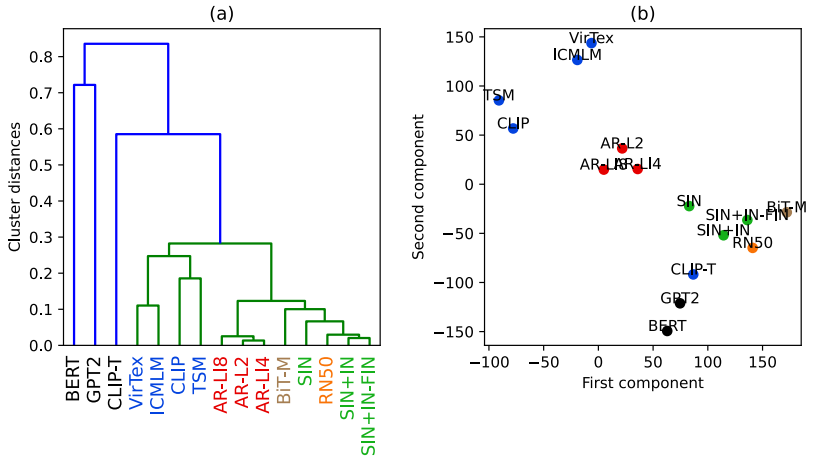

Figure 4.9: (a) Dendrogram of a hierarchical clustering of the RDMs. (b) t-SNE of the RDMs.

works possess adversarial robustness properties in common with AR models, this suggestion was not supported by our tests using actual adversarial attacks (Fig 4.6). Finally, the language models (BERT, GPT2 and CLIP-T) are separated from the rest, along a distinct direction. Overall, the analysis suggests that multimodal representations are neither visual nor linguistic, but surprisingly, *not really in-between either*[3]. This is surprising as we should expect that representations trained with access to both vision and language would derive information from both modalities, and consequently end up somewhere in-between purely visual and purely textual representations.

[3]Of course, we describe multimodal networks as *neither visual nor linguistic*, but this is to be understood in relative terms–they are *relatively* far from both visual models and linguistic models. In absolute terms, there is always a reasonable amount of similarity between multimodal networks and certain visual or linguistic models.

Performance on linguistic tasks

This suggestion might be further supported by evaluating the usefulness of the learned visual representations on *linguistic* tasks. According to the above findings, visual representations obtained via multimodal training may fare no better than vision-only representations. To test this, for each vision model, we collect the ImageNet features for each image class, and train a standard word embedding (Skip-Gram method) while constraining the class label words to these visual feature vectors. The resulting linguistic space will thus capture some of the structure of the vision model's latent space.

Method

Architecture We train Skip-Gram models [Mikolov et al., 2013a] on Wikipedia using the Gensim library [Řehůřek and Sojka, 2010]. Before training, some of the embedding vectors (corresponding to the ImageNet class labels) are set to the latent representations of a vision model, and frozen during training. This training procedure forces the word embedding space to adopt a similar structure to the vision model's latent space (at least for the frozen words, i.e. the class labels).

Visual words We denote 'visual word embeddings' (resp. visual words) as the word embeddings (equivalent to the visual feature vectors) obtained from the vision models (resp. the associated word token) on ImageNet classes. Some of the classes are composed of multiple words (e.g. "great white shark"). We leverage the WordNet [Miller, 1998] structure of ImageNet classes to only keep the hypernym of the class that contains only one word (e.g. "great white shark" becomes "shark").

All of the ImageNet categories that have the same one-word hypernym are grouped together into one unique hyperclass. For instance, the "shark" hyperclass contains the classes "great white shark" and "tiger shark". Finally, to obtain the visual word embeddings, we average the visual representation of all the images of each hyperclass from the ImageNet validation set. This gives a total of 824 visual words.

Besides, we choose a vocabulary of 20,000 words (taken from the most frequent tokens in Wikipedia). Only 368 visual words are among the 20,000 most frequent words, so we extend our vocabulary to also contain the 456 other visual words, resulting in a total vocabulary of 20,456 words.

Embedding dimension Since the vision models do not all share the same feature dimensions, in order to compare all Skip-Gram models, we reduce the dimensionality of the feature spaces of all vision models to 300 dimensions using a PCA. The PCA is computed using the visual features of all images in the ImageNet validation set. Consequently, the Skip-Gram word embeddings are trained with 300 dimensions.

Training We train the Skip-Gram models for 5 epochs, using the standard negative sampling strategy. We use window sizes of 5 words and a learning rate of 1e-3. We use the "`vectors_lockf`" feature of the Gensim library to freeze certain word embeddings during training.

For the dataset, we use a recent dump of Wikipedia and we split it into two sets containing 80% and 20% of the articles for the training and validation sets.

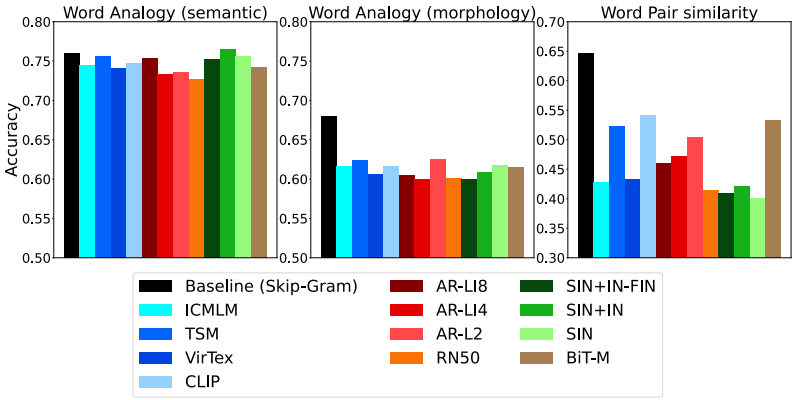

Figure 4.10: Semantic Word Analogy (such as "son", "daughter", "boy", "girl"), Morphology Word Analogy (such as "write", "writes", "work", "works") and word pair similarity results for the visually constrained Skip-Grams. The Baseline is a vanilla Skip-Gram model (300 dimensions) where all 20,456 word embeddings are free to be learned.

Evaluation

We evaluate our Skip-Gram embeddings on two tasks: word analogies and word pair similarities.

Word Analogy This standard task [Mikolov et al., 2013b] for evaluating the quality of word embeddings consists of quadruplets $\{A, B, C, D\}$ (e.g. "man", "king", "woman","queen") supporting the relation "A is to B as C is to D". The task consists in finding the 4th one given the first three, by solving the equation in the latent space: $D = B - A + C$. The more accurate the model, the better its representation. We evaluate the word embeddings on the full dataset provided by [Mikolov et al., 2013b] that we split in two different sets: *morphology analogies* (such as "write", "writes", "work", "works"), and *semantic analogies* (such as

"son", "daughter", "boy", "girl"). If vision-language training helps "ground" the visually-derived word embeddings, we expect this grounding to be more helpful in the resolution of semantic, rather than morphology analogies.

Word Pair similarity Another task for evaluating the quality of word embeddings is to ask humans to rate the semantic similarity of pairs of words (e.g. on a scale of 0 to 10, how close is "queen" to "king"? How close is "queen" to "woman"? etc.) [Finkelstein et al., 2001] and then compute the same similarity evaluations in the latent spaces of the models. The higher the (Pearson) correlation between the pairwise similarities of a model and human pairwise similarity judgments, the better the representation of the model.

Results

The baseline Skip-Gram produces the best word embeddings overall (black bars in Fig 4.10). This is to be expected since the embeddings are learned freely, without any additional constraint during training. Interestingly, this baseline advantage is weakest in the case of the semantic analogy task (Fig 4.10, leftmost panel), where some of the vision and visio-linguistic models are on par with the baseline. This shows that the frozen vectors do not necessarily impede the performance when the analogies are defined semantically (and might thus be presumed to contain some visual component). However, even for these semantic analogies, vision or vision-language word embeddings never significantly surpassed the baseline performance.

In the word pair similarity task, networks show variable performance levels,

but without a clear distinction between vision-only and vision-language models. Among the visio-linguistic networks, CLIP and TSM, which are trained contrastively on a large amount of data (see Figure 4.1) have embeddings that correlate well with human word similarity judgements. However, when compared with the vision-only models, we do not observe a clear-cut performance improvement. Indeed, the best vision-only model (BiT-M) is on par with CLIP and TSM. Interestingly, by comparing the results from the Fig 4.10 rightmost panel to the data plotted in Fig 4.1, we observe that among our twelve models, the top six for the word pair similarity task (TSM, CLIP, BiT, and the three AR models) correspond to those models that were trained on the largest datasets.

For the analogy tasks (semantic and morphology), there is no particular trend. However in both cases, the best performing model (excluding the baseline) is a visual one: SIN+IN in the semantic case, and AR-L2 in the morphology case.

In summary, we find that multimodal training of visual features does not improve their usefulness for language tasks either, and we suggest that the amount of training data may be a more important factor for generalization.

Legitimacy of the visual word embeddings

In the previous results, for training the visually-guided word embedding models, we averaged the visual feature vectors over many examples for each class. This averaging can potentially alter the quality of the embeddings, e.g. by discarding important information about the feature distributions. Thus, we check the validity of these averaged feature vectors[4], by verifying that they remain useful in a vision

[4]We here test the 300d vectors after the PCA dimensionality reduction.

context. We use these visual feature vectors as class prototypes and evaluate the corresponding nearest-neighbor classification accuracy on the ImageNet validation set[5] with a method similar to section 4.2. For all models considered, classification accuracy was well above chance ($p<0.01$): this means that the class-specific vectors indeed remain useful as visual representations of their category.

Furthermore, we computed the correlation between this visual classification accuracy of the word embedding, and the corresponding word analogy or word-pair similarity accuracy for each model. The resulting Pearson correlation coefficient was r=-0.0821 with the semantic Word Analogy performance, r=0.301 with the morphology Word Analogy performance, and r=0.797 with the Word Pair Similarity.

The significant high correlation of visual classification with word-pair similarity performance might be caused by the visual component of the word similarity judgments performed by human subjects. Indeed, many "similar words" also entail similar visual features (tiger, jaguar, cat, feline), and so the word-pair similarity task may not be a pure language task.

Discussion and Conclusion

It is a highly appealing notion that semantic grounding could improve vision models, by introducing meaningful linguistic structure into their latent space, and thereby increasing their robustness and generalization properties. Unfortunately, our experiments reveal that current vision-language training methods do not achieve this objective: the resulting multimodal networks are not better than

[5]With the images regrouped into our 824 classes.

vision-only models, neither for few-shot learning, transfer learning or unsupervised clustering, nor for adversarial robustness. In addition, compared to vision-only models, the multimodal networks' visual representations do not appear to provide additional semantic information that could serve as a useful constraint for a word-embedding linguistic space.

The present inability of linguistic grounding methods to deliver their full promise does not imply that this cannot happen in the future. In fact, we believe that exploring the current models' performance and representations, as we do here, can help us understand their limitations and adjust our methods accordingly. Specifically, we found that multimodal representations are neither visual nor linguistic, but are not really in-between either (Fig 4.9). In CLIP and TSM, for instance, the contrastive learning objective encourages the visual and language streams to agree on a joint embedding of images and corresponding captions. However, such agreement, by itself, does not constrain either latent space to remain faithful to its initial domain. As a result, CLIP's (and TSM's) visual representations may discard information that could prove critical for transfer-learning to other visual tasks. If this is true, we predict that adding domain-specific terms to the multimodal loss function (e.g. self-supervision) could be a way to improve visual generalization, while retaining the advantages of multimodal training—possibly including semantic grounding.

4.3 Afterword

This first collaborative work showed how non trivial the generalisation abilities of multimodal models were. Counter intuitively, multimodality didn't help for unimodal tasks. Actually, while it would have been expected that incorporating semantic knowledge in the visual domain would have made it more robust to adversarial attacks, or more prone to differentiate between objects, the contrary happened.

However, this study only investigates one aspect of the visual domain, object recognition – and one aspect of the textual domain: similarity between insulated, visually constrained word vectors. This, by far, does not cover all there is to perform in these modalities. This is why the next study will focus on another domain, more symbolic and cultural. It will thus complement and broaden the conclusion of the current one.

Chapter 5

Generalisation in human-centric datasets

5.1 Preamble

After training a multimodal model on a large dataset and on one or several general tasks, it has learned representational features that can be used for other, more specific downstream tasks. For image and text classification, a pretrained algorithm doesn't need to be retrained (which means its data representation remains unaltered) if a classifier is plugged on top of a latent space and trained. Given a latent space, one can thus evaluate the quality of the feature generated by an algorithm by training similar classifiers on top of several latent spaces, by comparing the accuracy of the prediction given by each of them.

This kind of benchmark can be done for image (as in the previous chapter) and/or for text representations. Here, we want to compare CLIP's image representation with other standard unimodal image encoder, and CLIP's text representation against unimodal text encoders, as well as how well these representation can be concatenated to represent multimodal data, for specific, non-standard tasks.

This work follows the work of [Devillers et al., 2021], presented above, which found that on standard image classification dataset, CLIP's image representation gave poorer results than other unimodal image encoder in transfer learning, few-shot learning and unsupervised learning. This was even truer for other multimodal model's latent space.

My idea was that these standard image classification dataset (CIFAR, CUB, SVHN, MNIST, FashionMNIST) actually evaluate only a specific part of the whole visual domain: object recognition. This is why I crafted the Plotster dataset, as well as found other already benchmarked more "human-centric" datasets. On these dataset, the unimodal algorithms' features gave poorer results than the one of CLIP's, showing that multimodallity can improve generalization in each modality.

In this chapter, contrary to the one above, we are not only gonna compare results on the visual domain, but also on the textual domain, both separately and jointly.

5.2 Comparing multimodal and unimodal models on human-centric tasks

When does CLIP generalize better than unimodal models? When judging human-centric concepts

Romain Bielawski, Benjamin Devillers & Rufin VanRullen

Abstract

CLIP, a vision-language network trained with a multimodal contrastive learning objective on a large dataset of images and captions, has demonstrated impressive zero-shot ability in various tasks. However, recent work showed that in comparison to unimodal (visual) networks, CLIP's multimodal training does not benefit generalization (e.g. few-shot or transfer learning) for standard visual classification tasks such as object, street numbers or animal recognition. Here, we hypothesize that CLIP's improved unimodal generalization abilities may be most prominent in domains that involve human-centric concepts (cultural, social, aesthetic, affective...); this is because CLIP's training dataset is mainly composed of image annotations made by humans for other humans. To evaluate this, we use 3 tasks that require judging human-centric concepts: sentiment analysis on tweets, genre classification on books or movies. We introduce and publicly release a new multimodal dataset for movie genre classification. We compare CLIP's visual stream against two visually trained networks and CLIP's textual stream against two linguistically trained networks, as well as multimodal combinations of these networks. We show that CLIP generally outperforms other networks, whether using one or two modalities. We conclude that CLIP's multimodal training is beneficial for both unimodal and multimodal tasks that require classification of human-centric concepts.

Introduction

Vision-language pretraining in neural networks is gaining popularity due to the growing interest in multimodal tasks such a Visual Question Answering or Image Captioning [Anderson et al., 2017, Lu et al., 2019, Li et al., 2019, Singh et al., 2019], but also to the availability of online resources that allow to build large-scale training datasets without manual annotations [Radford et al., 2021, Jia et al., 2021]. In theory, training a model on multimodal data should help improve its representation of data from each of the modalities. For an image-text model, for instance, the image features could be enriched by the abstraction of the linguistic data –the semantic grounding property–, and inversely, the linguistic features could gain informativeness through visual grounding [Harnad, 1990].

Unfortunately, this does not always happen in practice.

Recently, [Devillers et al., 2021] evaluated the visual generalization abilities of CLIP [Radford et al., 2021], a popular network trained with a contrastive learning objective on more than 400M image-caption pairs scraped from the web, and other multimodal models [Sariyildiz et al., 2020b, Desai and Johnson, 2020]. They showed that for standard object classification tasks (e.g. digit, fashion item or natural image classification), multimodal networks like CLIP underperformed compared to other unimodal (vision-only) models like BiT-M [Kolesnikov et al., 2019] in transfer learning, few-shot learning and unsupervised learning settings. Here, we revisit this question using datasets focusing on more "human-centric" concepts.

Human learning generally involves interacting with multimodal data. Thus, one could expect that CLIP's representations of images and text should be somewhat closer to human representations than those learned by unimodal models. Moreover,

given that CLIP was trained on image-caption pairs from a variety of sources from the Internet (including social networks), we can assume that an important part of its training captions was written by humans for other humans. This is different from standard vision datasets, in which labels or annotations are sometimes human-generated (e.g. through Amazon's Mechanical Turk), but always produced for machine-learning purposes. Again, this difference should bring CLIP's representations closer to human ones when compared to unimodal models. Thus, there should exist at least *some specific tasks* for which CLIP's multimodal training provides advantages over unimodal models. As an example, consider the task of assigning a genre to a movie based on its poster and title. This requires retrieving fine-grained information about, among other things, the artistic, emotional or stylistic aspect of an image or a piece of text (or both). This can only be properly achieved if the model's training offered appropriate exposure to such human-centric concepts. Here, we use the term *human-centric* whenever a concept refers to cultural, social, aesthetic and/or affective components of the world.

We thus make the hypothesis that CLIP should perform better than unimodal models in generalization tasks where human-centric concepts are involved. We evaluate this hypothesis on three tasks involving such human-centric concepts: sentiment analysis on tweets; genre classification of books; genre classification of movies. All tasks can be performed based on visual data (images), text data (tweet, book or movie title, movie plot summary), or both. For the movie genre classification, we introduce a new, large-scale multimodal dataset obtained by a crawling on The Movie Database (TMDb). As detailed below, we find that CLIP outperforms unimodal models in both vision and text-based classification, as well as pairwise combinations of these unimodal models in the case of multimodal

(image+text) classification. Consequently, CLIP establishes a new SOTA on these tasks.

We provide our code for reproducibility[1].

Models

We compare CLIP (trained contrastively on both images and text) against several unimodal models. For fairer comparisons, all the vision models are ResNet50 [He et al., 2015] based architectures and all the text models are transformer encoders.

CLIP was trained using a contrastive loss on a large (400M) set of image-text pairs. The training of CLIP consists in creating a joint (multimodal) embedding space. For one batch of image-text pairs, the objective of the network is that the embedding of an image (through a ResNet50 backbone, here simply referred to as CLIP) and the embedding of its text description (through a transformer backbone, here referred to as CLIP-T) are as close as possible, while the embedding of an image and the embeddings of text descriptions of other images in the batch are as far as possible. After training, the text encoder and the image encoder can be used as single-modality encoders.

For unimodally trained vision networks, we use two pretrained ResNet50-based models: the standard ResNet50 that was trained for classification on ImageNet-1K (here referred to as RN50), and BiT-M that was trained on ImageNet-22K [Deng et al., 2009].

[1] https://github.com/Bila12/CLIP-judging-human-centric-concepts

For unimodal text embeddings, we test two standard text encoders against CLIP's: Bert-large and Bert-base [Devlin et al., 2018]. We use the Bert sentence transformer version [Reimers and Gurevych, 2019], based on Bert's [CLS] token and fine-tuned on SNLI [Bowman et al., 2015] and MultiNLI [Williams et al., 2018]. Among the transformer encoders provided in the HuggingFace [Hug,] repository at the time our experiments were conducted, these were the two best-performing across several text classification tasks, and are now still close to SOTA. These versions of Bert-large and Bert-base are fine-tuned on downstream text classification tasks, but we refer to them in this paper simply as Bert-large and Bert-base.

Although all 3 text encoders are transformer encoders [Vaswani et al., 2017], they do not have the same number of parameters. Bert-large has 300M, Bert-base has 110M, and CLIP-T has 80M parameters. This gives a structural disadvantage to CLIP-T, which only strengthens our conclusions, as we found CLIP-T to be the overall best-performing text model.

We consider both unimodal tasks (classification of images or text), as well as multimodal tasks (classification of image-text pairs). When performing a unimodal task, the encoding of the image (resp. the text) is used directly by the corresponding classifier. When performing a multimodal task (image-text based classification), the encoding of an image by a visual model and the encoding of the corresponding text by a textual model are simply concatenated to create the multimodal vector that is used for the classification.

For BiT-M and RN50, we use the last layer output before the classification head used for their training, which counts 2048 dimensions. For CLIP, we use the latent vector in the multimodal space generated by the visual pipeline, counting 1024 dimensions; for CLIP-T, the one generated by the textual pipeline (1024

dimensions); and for the two Bert models, we use the vectors directly provided by the Sentence Transformer pipelines (1024-dimensional for Bert-large and 768-dimensional for Bert-small).

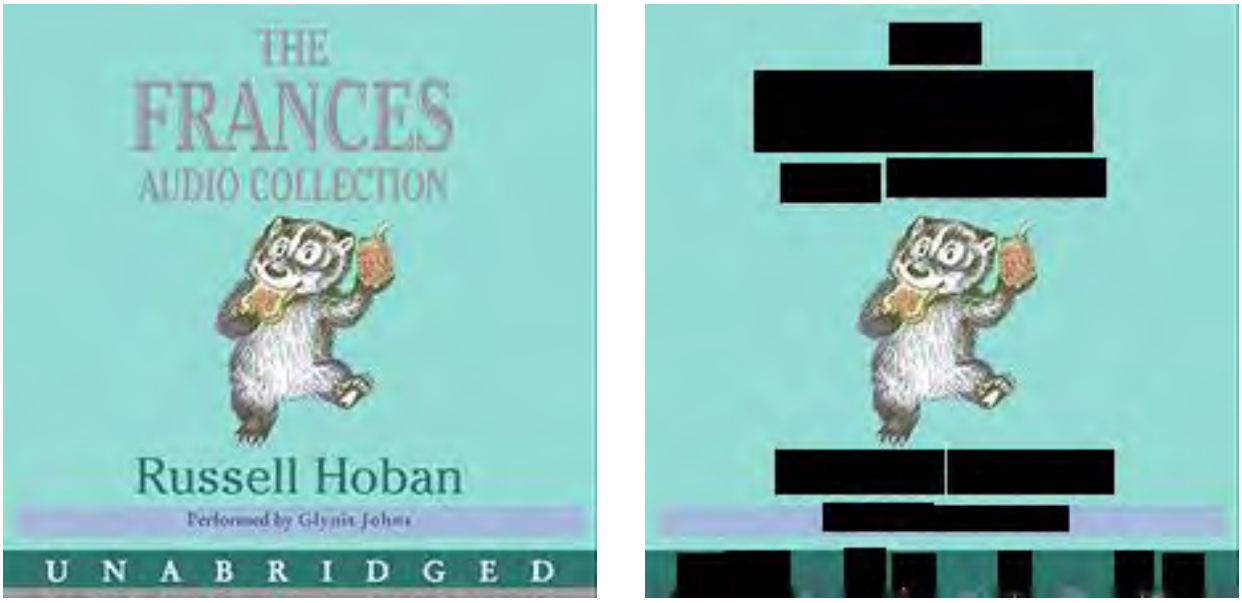

Figure 5.1: An original cover from the Book Cover dataset (left) and the associated masked cover (right). The title, the name of the author and parts of the text have been blacked out by the EAST algorithm, while the white text was incompletely detected, but subsequently blurred by the second algorithm. This sample belongs to the "Children's books" genre. Its title is: "Frances Audio Collection CD (I Can Read Level 2)". This image is copyright from Amazon.com, Inc. and used here for academic purpose only.

Datasets

We evaluate the models on three datasets composed of labelled image and text data, that can be inputted as pairs for multimodal classification tasks, or used as single

inputs for unimodal classification tasks. The language part of all these datasets is in English.

MVSA

MVSA or "Multi-View Sentiment Analysis" [Niu et al., 2016] is a dataset of pairs of images and associated text from Twitter, labelled with three possible sentiments (Positive, Neutral or Negative). Each image and each piece of text has three labels given by three different users, adding up to 6 labels for each image-text pair. We assign a score for each label (Positive: 2, Neutral: 1, Negative: 0) and we compute the rounded average score for each pair. By doing so, we get only one label per image-text pair that we can then use for single-label classification across modalities.

Book covers

The Book Covers dataset was introduced by [Iwana et al., 2017]. It consists of 57k images of book covers scraped from the Amazon website, with their title as text information. Each pair of cover+title is labelled with one genre among 30 possibilities. A cleaner version of the dataset, removing one genre and grouping two similar ones, with only 28 classes and 55.1k images, was later introduced by [Lucieri et al., 2020]. This is the dataset we use for our experiments.

Plotster and TMDb

We introduce and publicly release the *Plotster*[2] dataset, obtained by crawling TMDb (www.themoviedb.org) using their provided API. It consists of 207,902 triplets of {poster, title, plot} (split in 189,185 train samples and 18,717 test samples), with each having several potential labels among 19 genres. A representative sample from this dataset is shown in Figure 5.2. Typically, each movie has between 1 and 6 genres, with an average of 1.7. Each poster is an RGB image of 900×600 pixels (height$\times$width). Plots have an average length of 310.8 characters, and titles an average length of 18.6 characters. For text input, in unimodal or multimodal settings, we can choose either plot or title. The results of both configurations were computed and are displayed in this paper.

A previous crawling on TMDb had been made by [Mangolin et al., 2020]. It contained only 10,594 movies, as the authors aimed to retrieve other pieces of data such as trailer video clips and subtitles. They had not included titles in their dataset. From these movies, 10,554 (i.e., 99.6%) can also be found in *Plotster*. For comparison, we isolated the posters and plots from this dataset, and verified that our results obtained on the full *Plotster* were still valid on this subset.

In another control experiment, we verified that CLIP's improved performance on the *Plotster* dataset was not a result of specific movie posters, plots and titles from TMDb having been included in CLIP's training (as the training set is not public, there is no direct way to determine this). For our control experiment, we crawled TMDb again, looking for movies with a release date later than January 5, 2021, date of the OpenAI blog post introducing CLIP. We thus assume that

[2]https://github.com/Bila12/Plotster_dataset

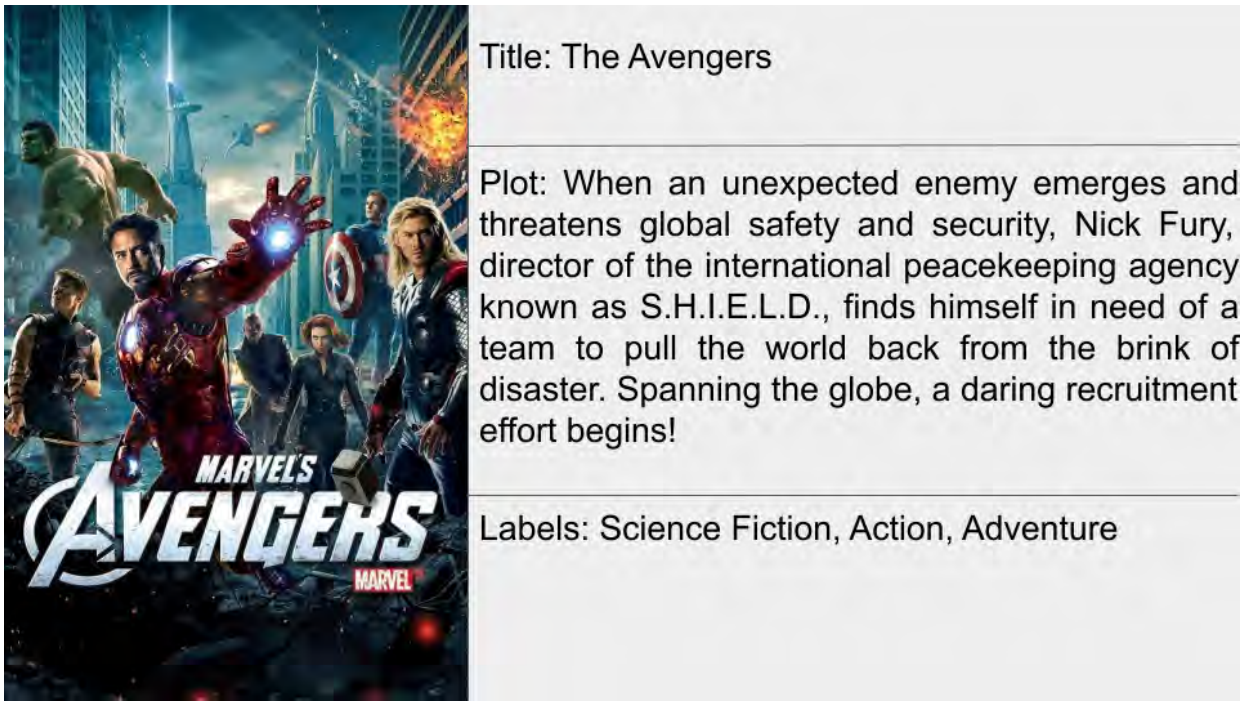

Figure 5.2: A data sample from *Plotster*. The image displayed here is property of The Walt Disney Company / Marvel Entertainment and under the CC BY-SA 2.0 license.

most of this data could not have been included in CLIP's training dataset. The new crawl resulted in 20,280 movies, only 93 of which had been present in the original *Plotster* dataset. We tested on these 20,280 new samples the classifiers trained on *Plotster* (only in unimodal settings), and report the corresponding results.

Masking

CLIP has been found to have an ability to "read" text inside images [Goh et al., 2021]. As most of the images in the Book Cover dataset and in *Plotster* have text on them, and as this text could be informative about the genre of the book or movie, we

Vision \ Text	None	Bert-base	Bert-large	CLIP-T
None	$\emptyset$	63.33 ± 0.18	64.02 ± 0.74	$\underline{64.60} \pm 0.30$
RN50	55.17 ± 0.37	63.93 ± 0.36	63.92 ± 0.55	64.13 ± 0.37
BiT-M	60.0 ± 1.46	61.93 ± 2.05	63.16 ± 2.82	62.77 ± 0.72
CLIP	$\mathbf{63.07} \pm 0.23$	$\underline{\mathbf{66.03}} \pm 0.15$	$\underline{\mathbf{66.03}} \pm 0.6$	65.58 ± 0.38

Table 5.1: Accuracies for the MVSA dataset. CLIP is the best vision model, CLIP-T the best text model. All text models perform similarly in both unimodal and multimodal setting, except when paired with CLIP (which yields the best performance of each column).

Vision \ Text		None	Bert-base	Bert-large	CLIP-T
None		$\emptyset$	54.70 ± 0.25	54.92 ± 0.43	$\underline{57.28} \pm 0.27$
Standard	RN50	10.04 ± 4.33	54.85 ± 0.52	55.53 ± 0.39	57.20 ± 0.49
	BiT-M	29.33 ± 0.92	50.11 ± 0.57	50.49 ± 0.59	52.60 ± 0.46
	CLIP	$\mathbf{53.75} \pm 0.23$	60.38 ± 0.34	60.62 ± 0.27	$\underline{\mathbf{60.66}} \pm 0.26$
Masked	RN50	10.41 ± 2.43	54.26 ± 0.25	55.11 ± 0.17	57.26 ± 0.29
	BiT-M	24.87 ± 0.99	48.93 ± 0.77	50.09 ± 0.71	52.08 ± 0.59
	CLIP	$\mathbf{33.04} \pm 0.21$	57.86 ± 0.45	58.47 ± 0.40	$\underline{\mathbf{59.54}} \pm 0.28$

Table 5.2: Accuracies for the Book Cover dataset (standard images on top, masked images on the bottom). CLIP and CLIP-T are the best performing models of each unimodal test, and together provide the best multimodal combination for both standard and masked images. Masks diminish the performance of all models (and their combinations), but the advantage for CLIP (and CLIP-T) remains.

worried that this ability could give CLIP an unfair advantage over other vision models. To minimize this possibility, we created alternative versions of these two datasets by applying a masking procedure on the images (see Figure 5.1). We used

Vision \ Text		None	Title			Plot		
			Bert-base	Bert-large	CLIP-T	Bert-base	Bert-large	CLIP-T
None		$\emptyset$	.314 ± .01	.323 ± .01	.397 ± .00	.582 ± .00	.599 ± .01	.612 ± .00
Standard	RN50	.090 ± .01	.338 ± .01	.363 ± .01	.393 ± .02	.578 ± .01	.599 ± .01	.599 ± .01
	BiT-M	.415 ± .01	.490 ± .01	.499 ± .01	.507 ± .01	.625 ± .01	.637 ± .01	.631 ± .01
	CLIP	**.526** ± .01	.559 ± .01	.558 ± .01	**.593** ± .01	.672 ± .00	.683 ± .00	**.687** ± .00
Masked	RN50	.070 ± .01	.335 ± .02	.352 ± .01	.383 ± .02	.576 ± .01	.597 ± .01	.596 ± .01
	BiT-M	.372 ± .00	.457 ± .02	.480 ± .01	.490 ± .01	.617 ± .01	.631 ± .01	.621 ± .01
	CLIP	**.449** ± .01	.525 ± .01	.534 ± .01	**.564** ± .00	.658 ± .00	.667 ± .00	**.676** ± .00

Table 5.3: f1-scores for the *Plotster* dataset. CLIP is the best model in vision, CLIP-T the best in text whether titles or plots are given as input, and CLIP+CLIP-T is the best multimodal combination in all cases. The masking doesn't affect the advantage for CLIP.

the EAST algorithm [Zhou et al., 2017] to generate bounding boxes around text; if the score given to a text detection reached a certain threshold, a black rectangle was applied over the corresponding bounding box. On top of that, a second algorithm detects the remaining small white text using a thresholding method, a saturation filter and a size filter, and then does a Telea inpainting [Telea, 2004] to remove it.

The results on the datasets with masks are reported along with those of the originals.

Results

To compare the generalization capabilities of our text, vision, and multimodal models, we focus on transfer learning and few-shot learning settings.

Transfer learning

Our first experiment is transfer learning. We use the pretrained networks (see Section 5.2) with frozen weights as encoders, and train a new classification head for each of our datasets in unimodal or multimodal settings.

For transfer learning in single-label classification (sentiment on MVSA, book genre), we plug on top of the frozen feature vector encoder one dense layer (ReLu activations) bringing the dimensions down to 256, and then another dense layer (softmax activation) for the classification. We then train only the weights of these 2 layers on the classification task with a Cross-Entropy Loss; therefore the network learns to output a probability density over the classes.

For multi-label classification (movie genres) the loss is a Binary Cross-Entropy Loss, and therefore the second dense layer outputs a number between 0 and 1 for each class. As the ground-truth label vector for one sample is a 19-dimensional one-hot vector, we round the 19-dimensional prediction of the network to get a binary predicted label vector. A f1-score [Pedregosa et al., 2011] comparing the predicted label vector to the ground-truth vector is reported, as raw accuracy is not a reliable measurement for multi-label classification. The f1-score takes into account the number of True Positives (TP), False Positives (FP) and False Negatives (FN) according to the following formula:

$$f_1 = \frac{\text{TP}}{\text{TP} + \frac{1}{2} \times (\text{FN} + \text{FP})}$$

The f1-score is computed for each movie, and subsequently averaged over the test set of each dataset. For f1-scores, as for accuracy, the higher the better.

Tables 5.1 and 5.2 show the results on the single-label datasets: MVSA and

Book Cover. The first column corresponds to the result of the vision-only experiment, the first line to those of the text-only experiments, and the other cells display the results of the multimodal ones. Table 5.3 shows the results for the multi-label dataset (Plotster). In all tables, the best vision-only performance is highlighted in **bold**, the best text-only is underlined and the best multimodal one is **both underlined and bold**. The standard deviation is calculated over five experiments with different random seeds and random initialization of the weights of the classifiers.

On MVSA (Table 5.1), CLIP is the best performing vision-only model and CLIP-T the best text-only model. The best multimodal combinations are CLIP+Bert-base and CLIP+Bert-large, with CLIP+CLIP-T near the same level (less than 0.5 percentage point behind). This is not unexpected, as CLIP-T counts much fewer parameters than Bert-base or Bert-large (see section 5.2).

For the Book Cover dataset (Table 5.2), CLIP is by far the best performing vision model, both with the standard covers and with the masked covers as input. The difference between CLIP's accuracy (53.8%) and the other two (RN50: 10.0%; BiT-M: 29.3%) remains high in the masked configuration (with CLIP at 33.0% and the other two below 25%), even though CLIP has lost the ability to read the text on the covers. This indicates that CLIP's reading ability is not the sole explanation for its advantage over vision-only models. CLIP-T is again the best text-only model. Here, the best multimodal combination is CLIP+CLIP-T for both standard and masked configurations. Finally, compared to previously established SOTA performance on the Book Cover dataset by [Lucieri et al., 2020], CLIP easily beats the previous visual SOTA (27.8 % accuracy), CLIP-T the previous textual SOTA (55.6%), and CLIP+CLIP-T the previous bimodal SOTA (55.7%).

Concerning our new *Plotster* dataset (Table 5.3), similar conclusions emerge. In vision-only conditions, RN50 performs relatively poorly; in the standard dataset, CLIP largely outperforms BiT, and this difference decreases but remains in the masked dataset. In text-only conditions, CLIP-T is the best model, both with titles and plots as input. Finally, in the multimodal settings, CLIP+CLIP-T is always the best-performing combination, whether using standard or masked images, title or plot as textual inputs. As before, the prevalence of CLIP in all task settings, even when text has been removed from the movie posters, indicates that its superiority in our movie genre transfer learning task is not solely due to its reading ability. We surmise that this advantage reflects a form of semantic grounding resulting from CLIP's multimodal training.

We also tested CLIP, CLIP-T and their combination on a subset of *Plotster* corresponding to the dataset of [Mangolin et al., 2020], in order to compare with previous SOTA values. We found that CLIP beats the previously established visual SOTA (f1-score of 0.603 against 0.409), CLIP-T the textual SOTA (f1-score of 0.589 against 0.488) and CLIP+CLIP-T the bimodal SOTA (0.670 against 0.628).

In a separate control experiment, we tested all our models (trained on the entire *Plotster* training set) on a new set of movies, all released after OpenAI's initial blogpost introducing the CLIP model. On this new test set, CLIP's f1-score changes from 0.526 to 0.439, BiT's goes from 0.415 to 0.318 and RN50's from 0.090 to 0.020. CLIP-T's (with title as text input) goes from 0.397 to 0.276, Bert-large from 0.323 to 0.237 and Bert-base from 0.314 to 0.229. The general diminution of the f1-score across all networks is probably due to the fact that features trained to classify older movies do not work equally well when they are applied to more recent movies. Nevertheless, CLIP and CLIP-T remain the top-

performing models; as it is unlikely that these recent movie posters and captions had been included in CLIP's training dataset, we conclude that CLIP's high transfer-learning performance on *Plotster* is not a consequence of prior exposure to these stimuli, but a true form of generalization.

In general, we see that in all the unimodal settings, CLIP outperforms the other vision models, and CLIP-T the other text models. This is true, even though CLIP has roughly the same number of parameters than RN50 or BiT-M, and fewer dimensions in its latent space (and thus, less parameters in its classifier head). Similarly, CLIP-T counts much fewer parameters than Bert-base or Bert-large (although it has a higher-dimensional latent space than Bert-small). In most of the multimodal settings, changing from one visual model to CLIP or from one textual model to CLIP-T improves performance (the only exceptions are for CLIP-T on MVSA and on *Plotster* with plots as text inputs). The best multimodal models always involve CLIP, and also involve CLIP-T in all cases except MVSA. This makes the CLIP + CLIP-T combination the best overall multimodal model in our experiments.

Few-shot learning

The second experiment we conduct is a visual few-shot learning task: we measure test classification accuracy based on exposure to a small number of randomly chosen training samples (or "prototypes") from each class. We can thus compare the results for our datasets with those of [Devillers et al., 2021], who also measured visual few-shot learning performance.

In their paper, [Devillers et al., 2021] used a single prototype vector for each

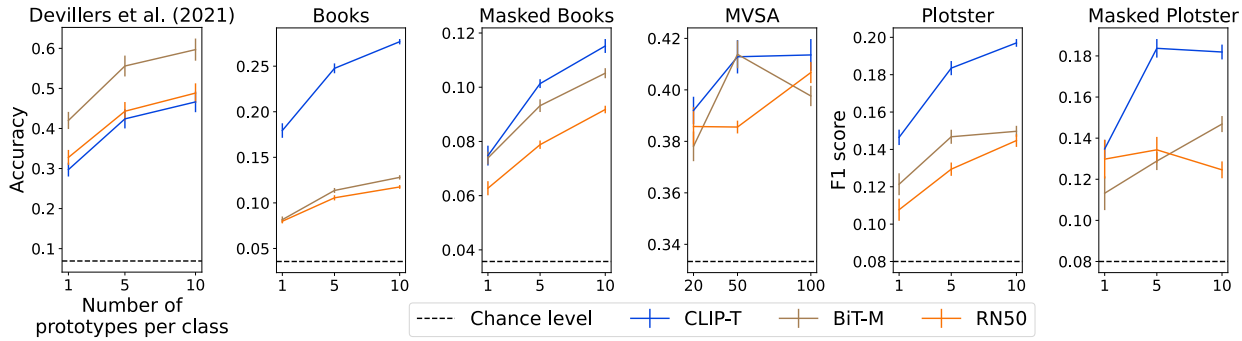

Figure 5.3: Few-shot learning accuracy (vision-only) over single label datasets (Book Covers, MVSA) and f1-score over the multilabel *Plotster* datasets. The leftmost panel reports average accuracy on 6 standard visual datasets used in [Devillers et al., 2021] – namely CIFAR10, CIFAR100, CUB, FashionMNIST, MNIST and SVHN. Accuracy was recomputed using the same method as for our datasets; the conclusions are identical to those of [Devillers et al., 2021]: CLIP does not perform better than RN50 or BiT in this few-shot learning setting. On the contrary, for our datasets CLIP outperforms the two other vision models. The advantage is reduced but still present when masks are applied.

class, obtained by averaging the latent representation of the N randomly drawn training samples for that class. Here, we prefer to retain all N individual samples as prototypes, and use a 1-nearest-neighbor (1-NN) classifier [Pedregosa et al., 2011] to classify the new vectors. We verified that this method, when applied to the same datasets as in [Devillers et al., 2021], does not alter their conclusion (see the first plot of Figure 5.3). To select the class prototypes of *Plotster* (which is a multiclass dataset), we randomly choose movies with a given class label. For example, a movie with genres “adventure” and “action” could be randomly chosen as a prototype of either genre. Moreover, when predicting the genres of a movie using the 1-NN classifier, we predict all the genres of the closest prototype.

Figure 5.3 reports the few-shot accuracy on the Book Covers and MVSA datasets as well as the f1 score for the *Plotster* datasets. Contrary to the conclusion of [Devillers et al., 2021] using standard visual datasets (see Figure 5.3, left), our results show a clear advantage to CLIP in our more "human-centric" visual tasks, even when masks are applied. For MVSA, the networks required more samples (between 20 and 100) to reach above-chance accuracy than for the other datasets (that use 1 to 10 samples). In that specific case, the three models are more difficult to distinguish, but CLIP still appears better than the other two visual models.

Summary

In the visual domain, CLIP systematically outperforms the unimodal vision models in transfer learning (Tables 5.1-5.3) and in visual few-shot learning (Figure 5.3), despite having a smaller embedding space than the other two ResNet50-based models. Part of CLIP's superiority may be due to its ability to read, but the advantage remains when text is removed from the images. This conclusion goes against the observations of [Devillers et al., 2021] using standard visual datasets (including SVHN, a digit reading dataset), where CLIP was never better (and often slightly worse) than other ResNet50 based models, including RN50 and BiT-M. We explain this difference by the nature of the classification performed: our tasks involve human-centric concepts, as defined earlier.

In the text domain, CLIP-T, despite having been trained with fewer parameters than the other two transformers (Bert-small and Bert-large), is systematically the best performing model in transfer learning.

Across seven multimodal settings (MVSA dataset; Book Covers dataset

[with / without masks]; *Plotster* with [titles / plots] $\times$ [with / without masks]), CLIP+CLIP-T was the best multimodal combination in six cases. In the remaining case (MVSA), it was a tie between CLIP+Bert-large and CLIP+Bert-small (two language models that count many more parameters than CLIP-T).

We think that the semantic grounding provided by linguistic inputs when training CLIP's visual stream, and respectively, the visual grounding provided by image features when training the CLIP-T language model, shaped their latent space in a way that makes it possible to better grasp the human-centric components of an image or a text.

Discussion and conclusion

CLIP's generalization abilities were originally described in the context of zero-shot learning [Radford et al., 2021], but they may also extend to other settings, including transfer learning and few-shot learning. Past work has revealed that this is not always the case [Devillers et al., 2021]. Considering the latent representations learned by CLIP may help us better understand when multimodal training does or does not benefit generalization abilities, continuing the work of [Hossain et al., 2019]. In our case, it appears that one of the domains where the improvement is most significant is when human-centric concepts are being judged.

During their joint contrastive training, CLIP and CLIP-T have learned to extract common information between image and text modalities, so that the two streams would result in similar embedding vectors. This means that the representation of text in CLIP-T has been enriched with visual data, and symmetrically, that the representation of images in CLIP has been improved by semantic or

linguistic enrichment. This is what is collectively referred to as the "semantic grounding" property [Harnad, 1990, Bender and Koller, 2020]. However, another consequence of this multimodal contrastive training is that when learning a common ground between modalities, some relevant information could be lost. For text, what cannot be directly linked to images (including grammatical or syntactic properties); and for images, what is not directly relevant to the text description (including fine-grained visual details that are rarely mentioned in the corresponding caption). This information loss might be the reason why CLIP was found to perform worse than standard vision-only models in a unimodal setting with standard visual datasets [Devillers et al., 2021]. For the same reason, one could actually expect that in a multimodal setting, the combination of CLIP's vision and text streams (CLIP+CLIP-T) could lead to worse performance than other combinations (e.g. RN50+Bert). The unimodal networks are trained to capture the relevant features of their modality, and when combined, could cover the multimodal feature space more fully than CLIP, a network trained to discard information that is not redundant across modalities. Our results show that, at least in our human-centric classification tasks, this limitation was not consequential: CLIP, CLIP-T and their combination often performed optimally. This may be because human-centric information is particularly well captured by features expressed in *both* images and text, rather than in each modality independently. On the other hand, this same reasoning could explain why CLIP+Bert combinations performed slightly better than CLIP+CLIP-T on MVSA: Bert may have provided additional information not captured by CLIP, which was lacking in CLIP-T because of their redundant embeddings (or, this might simply be due to the fact that Bert has many more parameters than CLIP-T).

Our suggestion that CLIP (and CLIP-T) perform particularly well when judging human-centric concepts resonates with recent findings relating CLIP's representations to human brain representations. [Goh et al., 2021] reported that some artificial neurons in CLIP's visual stream (but not in standard visual models like Inception or ResNet) are systematically activated by specific "concepts" such as a particular person, emotion, country, religion, etc. Furthermore, these neurons could be equally activated by visual features (e.g., a photograph or drawing of the person's face) or by written text (e.g., the person's name). The authors related this multimodal invariance to properties of specific biological neurons found in the human hippocampus and temporal medial lobe, called "concept cells": these cells would also systematically activate when presented with a picture, drawing or written word representing a specific concept, such as a photograph of the actress Jennifer Aniston or her written name [Quiroga et al., 2005, Reddy and Thorpe, 2014]. Indeed, more recently [Choksi et al., 2021] compared brain fMRI representations in the human hippocampus with the patterns of representations measured in various vision models. They found that CLIP and other networks trained with multimodal objectives were more similar to human hippocampus representations than standard vision models (including RN50 and BiT-M). This could explain why a multimodal network like CLIP performs better when judging "human-centric concepts".

To conclude, we think that it is crucial to investigate the specific domains in which a multimodal training such as CLIP's can (or cannot) improve generalization. Our work indicates that *multimodality* will be key for developing algorithms designed for human-centric tasks (even for *unimodal* tasks) such as detecting emotions, analyzing personality, conducting a conversation or, more generally, when human-machine interactions are involved.

5.3 Afterword

This section completes the one before it. The visual modality is composed of many domains. The understanding of some of them is improved when training in a multimodal setting, while some other might not benefit from the current multimodal methods. This shows that training on multimodal data is not sufficient to improve the overall understanding of each domain. The method of training is key, and still need to be improved in order to complete the third step (WS3) described by [Bisk et al., 2020].

To sum-up what has been shown so far: when evaluating generalization abilities, one cannot only focus on standard tasks and datasets. A modality's representation covers many subdomains, and multimodallity, especially in the case of CLIP, has improved the representation of both text and images when it comes to human-centric concepts, but this was somehow detrimental to more object-centered representations.

Nonetheless, this is encouraging, as AI tends to become more and more human compatible – interactions between human and AI driven robots being one of the key objective the field is aiming at.

In the next section, we will take on a standard multimodal task and see how we can very efficiently obtain some interesting results by leveraging CLIP's latent space properties. This will highlight another aspect of this latent space: its incomplete multimodality and at the same time the impressive correspondence between textual and visual vectors – which enables us to easily, at a very low computational cost, translate between them by using cycle-consistent training.

Chapter 6

Improving image captioning with the Latent CycleGAN

6.1 Preamble

The generalisation abilities of multimodal models can be assessed and leveraged at the same time. This is what has been done previously, with transfer learning and few-shot learning.

In this study, the task at hand is image captioning. So, contrary to the previous experiment, where no translations were involved, and where the classification required supervision and annotated data, we are here going to put in place an image-to-text pipeline trained without supervision.

This is done in order to lower the computational cost of the task at hand, as big pretrained multimodal models already exist, and we want to exploit the properties of their rich latent space.

We will show that an unsupervised training method in the latent space of CLIP

can yield better results than a direct text-decoding pipeline, which will prove that the cycle-consistency principle alone, as applied in CycleGAN [Zhu et al., 2017], can be used to train a GAN in a latent space without having to explicitly generate pieces of data (such as text or images). A side conclusion of this paper, that establishes a proof of concept for a Latent CycleGAN, is that CLIP's latent space is not fully multimodal; text and image features are distinguishable and need to be translated in one another to perform a multimodal translation task such as image captioning.

6.2 The Latent CycleGAN

CLIP-based image captioning via unsupervised cycle-consistency in the latent space

Romain Bielawski & Rufin VanRullen

Abstract

Image captioning typically involves an image encoder to extract meaningful image features, and a text decoder to generate appropriate sentences. Powerful pretrained models can be used for both image encoding and text decoding; but in this case, a separate multimodal translation stage between image-encoder output features and text-decoder input features must be learned. One exception is when image and text features are already aligned by construction, as in the CLIP model (Contrastive Language and Image Pretraining – a bimodal network pretrained on 400M image-text pairs). Pretrained CLIP-image features can be directly fed to a text-decoder trained to reconstruct captions from their pretrained CLIP-text features. Here we show that this direct captioning method is in fact sub-optimal. Instead, we propose an alternative method to translate CLIP-image features into CLIP-text features in a strictly unsupervised way, using the CycleGAN architecture – originally designed for unpaired image-to-image translation. Our Latent CycleGAN, optimized solely for an unsupervised cycle-consistency objective, generates CLIP-text latent features conditioned on CLIP-image latent features and vice-versa. Using these CLIP-text latent features as input to the text decoder, our method largely outperforms the direct captioning method that uses CLIP-image features – despite the fact that CLIP's large-scale pretraining should have already aligned the two feature spaces. This implies that cycle-consistency on unmatched multimodal data can be efficiently implemented in a bimodal latent space, and that CLIP-based image captioning can be improved without additional supervised training.

Introduction

Multimodality is gaining popularity due to the recently available online resources that make the creation of huge visio-linguistic datasets possible [Jia et al., 2021]. Many models have been created to perform specific bimodal tasks such as Visual Question Answering or Image Captioning [Anderson et al., 2017, Lu et al., 2019, Li et al., 2019, Singh et al., 2019], but some have been designed with a more general objective: producing a multimodal latent vectorial space where images and text can be represented and compared. Among these models, CLIP – an algorithm trained with a multimodal contrastive objective on a large dataset (400M samples) of image-caption pairs – has shown impressive zero-shot learning abilities [Radford et al., 2021]. This model has recently been tested on tasks for which it was not initially trained, such as transfer learning and few-shot learning on unimodal and multimodal datasets, or image captioning, establishing new SOTA results on some tasks [Bielawski et al., 2022, Mokady et al., 2021].

In the specific case of image captioning, many studies use pretrained models for image feature encoding as well as for text generation. An end-to-end image-to-caption fine-tuning stage is typically required, however, to align visual and linguistic representations in a supervised way on a matched image-caption dataset [Chen et al., 2021, Fang et al., 2021, Zhou et al., 2019]. There is an obvious exception to this rule: when the pretraining of the model already aligned text and image features – as in the case of CLIP. Therefore, here we aim at leveraging this property by implementing a captioning pipeline that does not use matched image-caption data.

We first train a “CLIP-text decoder” to reconstruct captions based on their

textual features in CLIP's latent space (a unimodal, linguistic objective); this text decoder is subsequently frozen. Hence, we compare a direct captioning pipeline – feeding the text-decoder with CLIP image features in order to generate a caption – with a pipeline where a CycleGAN [Zhu et al., 2017] inspired translator – trained with only unpaired visual and textual features – is used to convert image features into text features before feeding them to the text-decoder. Even though CLIP's latent space was already pretrained with a brute-force approach to align its visual and linguistic representations on 400M image-caption pairs, we demonstrate that our feature conversion model trained using cycle-consistency in the CLIP latent space significantly improves captioning performance, compared with the direct method.

Dataset

To train our algorithms, we use the COCO [Lin et al., 2014] train 2014 dataset, composed of images representing complex scenes, along with their descriptions. We simply use the captions and the images independently, as two sets of unpaired unimodal data from each modality.

For the evaluation, we use the COCO validation 2014 dataset.

Models

Pretrained models

We use CLIP ViT-B/32, a pretrained Vision-Transformer-based [Dosovitskiy et al., 2020] CLIP checkpoint, as image and text encoder. CLIP's vision encoder will be ther-

after referred to as just CLIP, and CLIP's text encoder will be called CLIP-T.

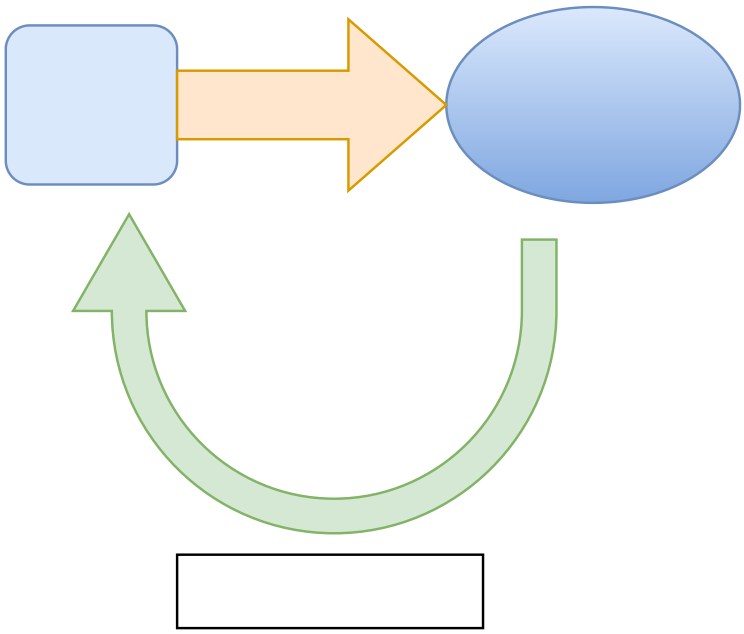

Figure 6.1: The text decoder is trained to reconstruct COCO train captions from their textual embedding in CLIP's latent space. It learns a mapping from CLIP-T features to prefixes that condition the generation of text with a pretrained (frozen) GPT-2. Note that the text-decoder is trained only with (unimodal) linguistic data.

In order to create our CLIP-T decoder (see Figure 6.1) we rely on the code provided by [Mokady et al., 2021], inspired from [Li and Liang, 2021]. Their decoder was originally trained on the CLIP image features of COCO images, with the objective to reconstruct their corresponding captions (therefore using paired vision-language data to align the text decoder training with pretrained image features). Instead, our text decoder is trained in a unimodal setting on the CLIP-T textual features of captions from the COCO train set (414K captions), with the objective of regenerating the original text. This decoder uses GPT-

2 [Radford et al., 2019] as a frozen language generator, and learns to produce prefixes that condition the generation of text. The parameters of the text decoder are shown in Table A.1 (Appendix **??**). Once trained, our text decoder is frozen and used as such in the two captioning pipelines that we compare.

Architecture

The architecture and training procedure of the Latent CycleGAN are shown in Figure 6.2 to 6.4. It is trained as a CycleGAN on unpaired data from the image and text modalities of the COCO train dataset (83K images and 414K captions). The training takes two generators – one of text features, one of image features – and two discriminators – to discriminate between real image (resp. text) features and fake/generated ones (Figure 6.2).

Just as in any GAN, the objective of each generator is to fool the corresponding discriminator. This is done by generating a fake latent vector in one modality, given a real latent vector from the other (this source vector can thus be considered as the noise that conditions the generation). The discriminator's objective is to guess whether any latent feature vector is real or generated (Figure 6.3). The generators of a CycleGAN [Zhu et al., 2017] also have specific extra objectives. The cycle consistency objective (Figure 6.4) minimizes the L1 loss between a feature vector and its reconstruction when passed successively through the two generators (e.g. an image feature vector is passed through the text feature generator, then this vector is passed though the image feature generator: the result of this operation is a reconstructed image feature vector). The identity objective aims at learning the identity function when the image (resp. the text) generator is fed with an image

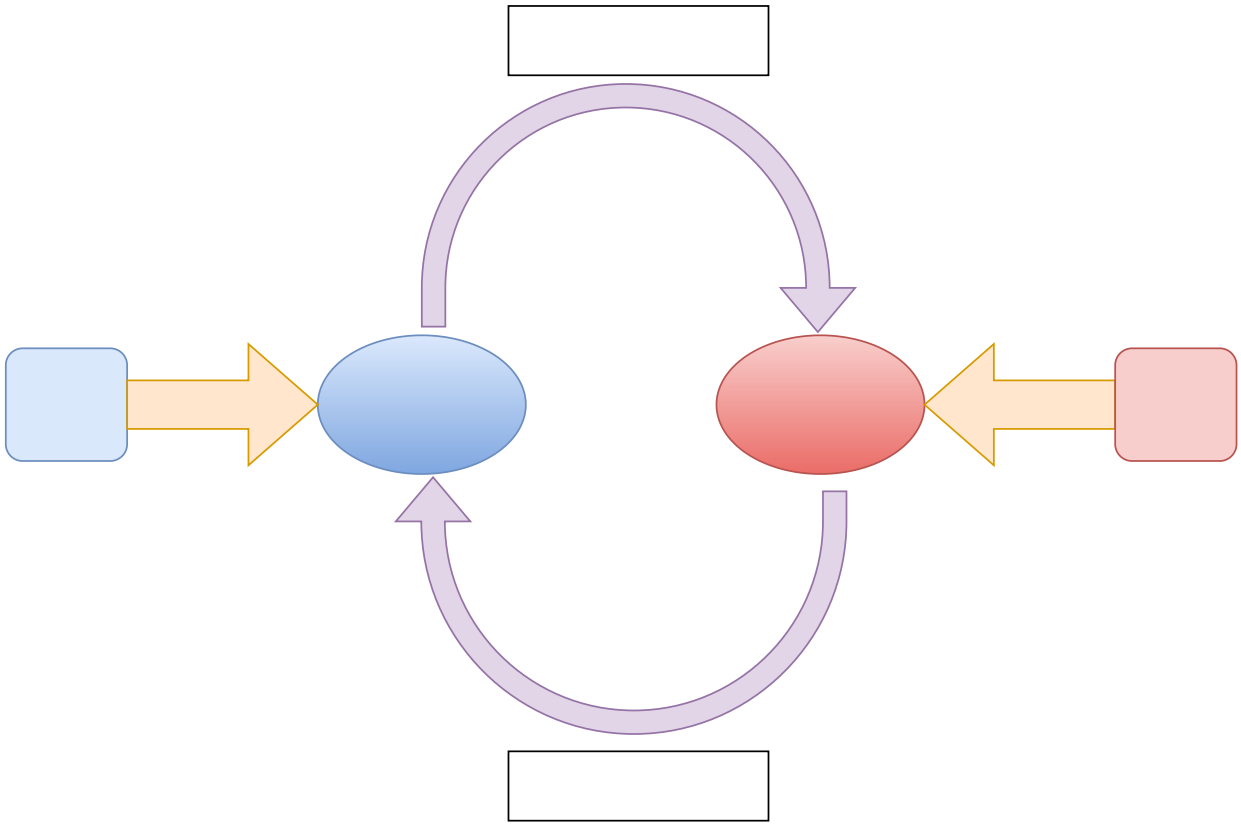

Figure 6.2: The full architecture of the latent CycleGAN. The generators (purple arrows) are trained with unmatched multimodal data from the COCO dataset. One is trained to generate latent image features given a CLIP-T embedding, the other is trained to produce latent text features given the image features, i.e. to "textualize" them. Discriminators are not shown here.

(resp. text) feature vector.

Each generator is composed of 4 dense layers of dimension 512x512 with Tanh activation; the discriminators are composed of two dense layers, one of dimension 512x256, the other of 256x1, with LeakyReLU activation.

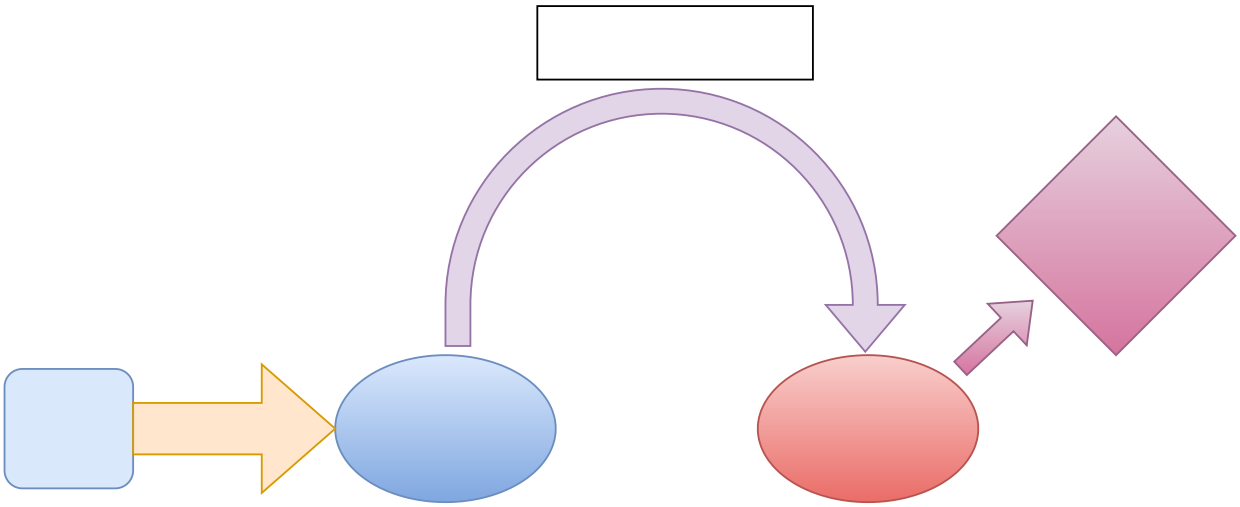

Figure 6.3: The GAN objective for Image feature generation: the generator must fool the discriminator, which must distinguish between real and fake inputs (here, between real image features and those translated from text features). A similar training objective and discriminator network exists for the other "textualisation" generator (not shown here).

After having trained the Latent Cycle-GAN to convergence, we can then compare the two captioning pipelines illustrated in Figure 6.5.

The first one uses the fact that in CLIP's latent space, the features extracted from an image are intended to be as close as possible to the features computed by CLIP-T for a matching caption. This similarity was enforced by extensive contrastive training over 400M paired image-captions. Therefore, we may simply feed our CLIP-T decoder (trained on text features) with image features, and generate a corresponding caption.

The second pipeline uses the image-to-text-feature generator (the rest of the Latent CycleGAN was only required during training, i.e. to compute and optimize cycle-consistency). The image-to-text-feature generator is used for what we call

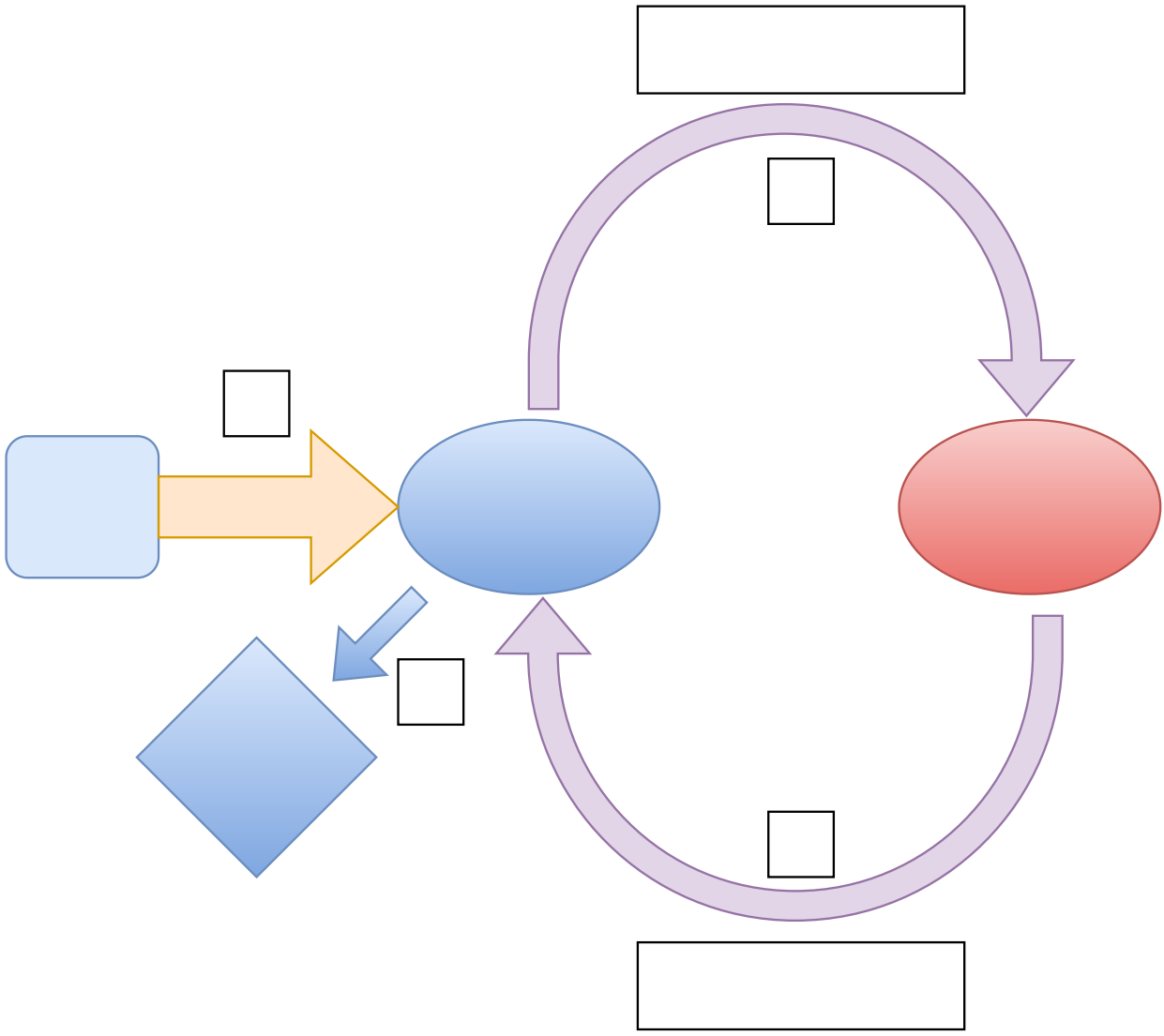

Figure 6.4: The cycle consistency objective consists in minimizing the L1 loss between a feature vector and its reconstruction after passing successively through both generators (here the translation of text features to and back from image features). The same cycle-consistency objective is also applied with cycles starting from the other (image) modality (not shown here).

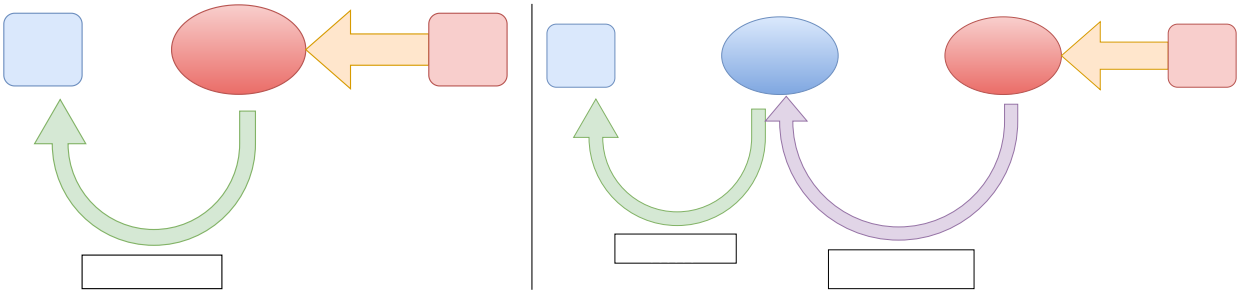

Figure 6.5: The two pipelines compared here for generating a caption. Our baseline (left) relies on the fact that CLIP was trained to project an image and its caption as close as possible in the latent multimodal space: the text decoder can thus generate a caption when given image features. The second one uses our generator, trained in an unsupervised way with unpaired multimodal data, to textualise the image features before feeding them to the text decoder.

here "textualisation", i.e. it generates a text feature vector conditioned on an input image feature vector. After the textualisation of an image vector, the textualised vector is fed to the CLIP-T decoder to generate the caption.

Task

Given an image from the COCO dataset, the model's task is to reconstruct one of the corresponding captions. Several scores can be used to evaluate the quality of the reconstruction. Here we dispay the BLEU-1 to BLEU-4 [Papineni et al., 2002] (BLEU-n counts matching n-grams in the model output to n-grams in the reference text), the ROUGE_L [Lin, 2004] (measuring the longest common subsequence between the model output and the reference), the CIDEr [Vedantam et al., 2014] (computing the average n-gram cosine similarity between the model output and several descriptions of reference and several n) and the

Scoring method	Image features as input	Textualised image features as input
BLEU-1	0.281	0.407
BLEU-2	0.120	0.231
BLEU-3	0.047	0.121
BLEU-4	0.020	0.062
ROUGE_L	0.239	0.341
METEOR	0.098	0.161
CIDEr	0.064	0.247

Table 6.1: Scores for image captioning on the COCO validation set, for the two pipelines displayed in Figure 6.5. Higher scores indicate better captioning. The captioning pipeline with image features as input to the text decoder underperforms, compared to the one with features textualised using the text feature generator.

METEOR [Denkowski and Lavie, 2014] (a variation of BLEU that aligns the reference and the output differently by incorporating semantic knowledge) scores.

Results

Results for the captioning task are displayed in Table 6.1. Despite the fact that CLIP's latent space was specifically designed and trained so that the encoding of an image and its description are as similar as possible, the strategy of directly using image latent features as input to the CLIP-T decoder does not perform well (for all scoring methods).

By simply training a Latent CycleGAN on unmatched COCO images and description (i.e. training in an unsupervised way on approximately 82K images, compared to the 400M image-text pairs of CLIP's initial training set) the improvement in score can go up to more than a factor 3.

Some uncurated examples of images and output captions from the COCO validation set can be seen in Appendix **??**.

Discussion and conclusion

CLIP's bimodal alignment can allow image captioning at a SOTA level, but this requires a fine-tuning with paired image-caption data [Mokady et al., 2021]. Since image and text are projected in the same latent space, it is also possible to use a direct captioning method with a trained CLIP-T decoder, without requiring any bimodal training; however, as we show here, this method is sub-optimal. We show how, using only unpaired images and captions, it is possible to significantly improve performance, while still taking advantage of CLIP's latent space multimodal alignment. Nonetheless, the results of the unpaired translation method implemented here remain far from the SOTA reached with supervised image captioning. Moreover, in our experiment, each caption implicitly matched an image from the training dataset, even though the matching was not given to the model. Finally, the translation module that has been designed here can be used as such only for modalities that are not temporal or sequential by nature (like sound or video), unless a fixed size vector can be extracted to represent these modalities. In future work, one might try enlarging the training domain of each modality, and incorporating data from separate, potentially larger unimodal datasets.

Our work suggests that the geometries of the representation of the vision and language modalities differ in CLIP's latent space. That is, CLIP's training has not properly brought together the two modalities. If it had, the image features would be directly usable by the text decoder, and our unsupervised "textualisation" training would not help the caption generation. This means that CLIP can represent vision and language in the same space, but vectors extracted from one domain are not fully multimodal in the sense that they are not indistinguishable from vectors from the other domain – in other words, modality-specific information appears to interfere with full multimodality. Our Latent CycleGAN helps bridge the gap between the two latent representations, by enabling a unimodal text decoder to better understand image features, once they have been "textualised" by the text feature generator.

The recently proposed DALL-E2 [Ramesh et al., 2022] model, which uses a diffusion process to generate images from a caption, appears to have been based on a similar realization. Their diffusion image generator was trained to reconstruct an image given its CLIP image feature vector; however, for text-to-image generation, they did not directly feed the CLIP-T embedding into the diffusion generator, but first "translated" it into a suitable image-feature latent vector, exactly as we propose here.

6.3 Afterword

The latent space of CLIP is not fully multimodal, despite its multimodal training. The features extracted from an image are widely different from the features

extracted from their corresponding caption. It was necessary to design a translator. Fortunately, the feature are homeomorphic enough (just as pictures of horses and pictures of zebras are similar and can be translated one into another).

An experiment which I conducted but that hasn't been mentioned in this paper, with Bert [Devlin et al., 2018] and ViT [Dosovitskiy et al., 2020] features, showed that the same method for translation cannot operate between two unimodal latent space, even though they share the same dimensionality – the performance of a Latent CycleGAN in these spaces were the same as the other unimodal pipeline. This means that CLIP's latent space has somehow benefitted from multimodality, partially but significantly enough so that an unsupervised translation algorithm actually gains performance compared to a simple, unimodally trained image captioning pipeline.

Chapter 7

General Discussion

In this dissertation I introduced several ways of assessing and leveraging efficiently the abilities of multimodal models. It appears that the current types of multimodal training bring some additional information to each domain's representation, however it is also detrimental in some other ways. Let's review some advantage and drawbacks of multimodal training highlighted in this thesis.

7.1 What does multimodality bring to representations?

The first study (in Chapter 4) shows that multimodal training does not bring any advantage compared to a simple visual training when it comes to object detection [Devillers et al., 2021]. It does not help either for adversarial robustness. This is surprising, as one could expect two things from multimodality :

- That it improves generalisation in vision tasks thanks to the semantic infor-

mation incorporated in the visual domain (as it has been done successfully in [Baveye et al., 2015])

- That it improves adversarial robustness thanks to semantic segmentation (e.g. that it prevents mistaking a dog and a truck on an image as they are semantically very dissimilar)

Which is not the case for the current multimodal SOTA models on standard visual dataset and neither targeted nor untargeted attacks.

A beginning of explanation comes from the study of textual, visual and bimodal representational spaces; bimodal models do not stand in-between textual-only and visual-only representations. They constitute their own domain, as if they had been trained on a third modality that is neither vision nor language.

Hence, this modality needs to be investigated and evaluated. Multimodal models are underperforming on standard visual tasks, but actually provide enhanced performances on what we have reffered to as "human-centric" tasks, at least for CLIP, in textual, visual and bimodal settings. This means that – probably partly given that multimodal dataset are crawled from the internet, where actual human-to-human interactions happen – mutlimodal models such as CLIP are more efficient when the human world is being considered and less when the task is object-oriented.

7.2 How can we efficiently leverage multimodal pretraining?

A multimodal training requires a lot of annotated data to be competitive with other unimodal models on standard tasks. In order to generate proper representations, CLIP had to be trained on 400M image-caption pairs. This huge dataset enables the model to learn feature that can generalize – sometimes better, sometimes worse than their unimodal counterpart – and that we should be able to use without having to retrain on another set of annotated data. That is what we have tried to achieve with our Latent CycleGAN, by creating a captioning pipelines that operates without having been trained on matched data (apart from CLIP's pretraining) – such a pipeline didn't work with two separate unimodal models, which means that multimodality is key in this context.

However, we remain far from SOTA, which means that CLIP's feature space cannot be leverage at very low computational cost with competitive performances so far. Inventing unuspervised training methods that yield SOTA performances by leveraging pretrained models is another challenge for the multimodal AI field.

7.3 Limits

The findings of our studies are limited to a subpart of the multimodal and unimodal models. We mostly focused on CLIP on the multimodal side, on ResNets and image transformers trained for image classification on the visual side, and on BERT for the textual one.

However, these models represent the current SOTA in all their respective domains. They are also standard in terms of architecture, in terms of training objective, and in terms of training dataset. It is not possible to exhaustively explore all the architectures, methods and data preprocessing that exist. We selected the algorithms by the fact that they had the broadest use and the best performances in standard metrics, which means that they represent the current state of the field. Our aim was to explore the possibilities of SOTA models, and the conclusion we draw on multimodal versus unimodal models stands only for the current ones. In the long run, we agree with [Bisk et al., 2020], which basically says that future models will *need* to be multimodal – a kind of multimodal that doesn't exist yet – in order to be competitive.

Concerning our choice of the word "human-centric", one might wonder whether it is the right one. Here we must admit that this term was used by lack of a better one. It might be too general. The human-centric world is complex and mostly unexplored by science. We think that, *at least*, we can say that emotion detection and genre classification involve culture-dependent, somehow arbitrarily determined concepts, which is not the case for the object oriented tasks studied in the first paper. There is a subjective element to what genre or emotions can feel like and be perceived, and that's what we wanted to highlight in contrast to other, more objective tasks.

Let's add our generalization results might not apply to other datasets or other modalities, but that they are probably not backbone dependent, as the only differences between models were their training datasets and objectives.

When it comes to our unsupervized method for image captioning using CLIP, one obvious critique might be that we are far from reaching SOTA performances.

And indeed, we can't expect to compete with supervized training methods, especially with a small dataset like COCO.

Our goal here was to demonstrate that unsupervized training could lead to better results than a simple pipeline plugged onto a multimodal latent space – a side conclusion being that CLIP's latent space is actually not fully multimodal. This could also mean that with the next generation of multimodal algorithm, the simple pipeline might gain some efficacy. It could also mean that our unsupervized method might get closer to SOTA by simply applying it to a yet-to-be-conceived multimodal space.

Nevertheless, we presented a proof of concept for a multimodal Latent CycleGAN. One can now try to train such an algorithm on more unmatched data, possibly from different distribution, our training dataset being the unmatched version of COCO captions and images. The Latent CycleGAN principle can also be applied in a unimodal setting (in an image feature space for example, between two image distributions), which, to our knowledge, hasn't been done yet.

More generally, all our work about assessing and leveraging the generalization abilities of multimodal models highlight the fact that modalities are not equivalent and cannot be fully translated one into another. One might think that each modality contains the same (amodal) information about the world, and that their only difference is the shape that they are embodied in. We think that this view is partly false. Only some subdomain of one modality can be translated to another one, and this subdomain can vary depending on the target modality. By trying to create a multimodal space, CLIP probably created a space where information that can be embodied through both language and text are well represented, instead of creating a space that represents all the information from both spaces. However, this fine

representation of this intrinsically bimodal data allowed it to better depict some parts of the human world – more specifically data that are used to communicate between humans. This might be due to the fact that as humans, we tend to create pieces of data that speak to several of our modalities at the same time in order to better communicate. We might tend to stay within the range of information that can be understood both through text and images, in order to use this redundancy either to be better understood or to convey our message in a more striking way.

7.4 Future works

As stated above, the first thing to do with the Latent CycleGAN is to train it on another unmatched multimodal dataset, such as Conceptual Caption for instance. We didn't have the time to do it, but that would probably yield slightly better results than ours. Especially given that one limitation of the algorithm might be the generalisation abilities of the text decoder, which is solely trained on the captions of the dataset – and there is more textual data in Conceptual Caption (413,915 captions in COCO versus 3.3 millions in Conceptual Caption).

Of course, having more data from both modalities will surely lead to better results for the CycleGAN itself.

But this is only a first step in using the Latent CycleGAN in different context. Firstly, one can use it on a multimodal dataset (unmatched) where the text is not (only) composed of the captions of the images, such as WIT [Srinivasan et al., 2021], where images are extracted from Wikipedia pages, along with their description, the title of the page and its introductory paragraph. Sec-

ondly, one might try to go even further and gather as many images as possible, as many text samples as possible and try the Latent CycleGAN on such a dataset, where the distributions of images and text are completely unrelated. Given enough data, the Latent CycleGAN might give some interesting results.

When it comes to the evaluation of the generalisation abilities of multimodal models, there are two perspective I would have explored if I had the time – and maybe will.

The first one is the adversarial robustness of multimodal models. It has been explored in the first paper presented here for the visual domain, and it showed that, contrary to what could be expected, semantic grounding didn't provide extra adversarial robustness. However, it would be interesting to look at targeted attacks – where the goal of the attack is to fool the network by forcing it to infer a specific class, which is obviously not the one of the image that is presented. If the semantic grounding was somehow efficient, it should be more difficult to misclassify a dog for a plane rather than a dog for a cat, because of the semantic distance between concepts. This hypothesis can be tested with the adversarial attacks results presented in [Devillers et al., 2021] with a little bit of analysis. The rate of success of targeted adversarial attacks should decrease with the semantic distance between the true class and the wrong one.

The second perpective would be a similar comparison as in the first paper, but on the textual side, with standard textual tasks. Indeed, our conclusion stand mostly for the visual domain, as the text modality was only introduced with the human-centric tasks of the second article. It is not possible to state that the language side of CLIP would not beat unimodal text model on standard tasks. It actually might, because standard textual tasks cannot really be considered as

object-oriented, contrary to standard visual ones.

7.5 Conclusive words

Multimodality in Artificial Intelligence is only at its beginning. The field now needs to invent new methods of learning that take in account what unimodal models can do and how these performances are obtained, to create models that outperform the unimodal ones in all domains. Once such models are designed, another task is how to engineer the feature space that they generate so that they can be efficiently used for various tasks, in order to keep the downstream computational and data cost low.

Bibliography

[Hug,] Huggingface sentence transformer repository. https://huggingface.co/sentence-transformers. Accessed: 2022-02-21.

[Lai,] Laion-400-million open dataset.

[Agrawal et al., 2018] Agrawal, H., Desai, K., Wang, Y., Chen, X., Jain, R., Johnson, M., Batra, D., Parikh, D., Lee, S., and Anderson, P. (2018). nocaps: novel object captioning at scale. *CoRR*, abs/1812.08658.

[Anderson et al., 2017] Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., and Zhang, L. (2017). Bottom-up and top-down attention for image captioning and VQA. *CoRR*, abs/1707.07998.

[Bahdanau et al., 2014] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate.

[Baltrusaitis et al., 2017] Baltrusaitis, T., Ahuja, C., and Morency, L.-P. (2017). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP.

[Baveye et al., 2015] Baveye, Y., Dellandréa, E., Chamaret, C., and Chen, L. (2015). Liris-accede: A video database for affective content analysis. *IEEE Transactions on Affective Computing*, 6(1):43–55.

[Bee et al., 2010] Bee, N., Pollock, C., Andre, E., and Walker, M. (2010). Bossy or wimpy: Expressing social dominance by combining gaze and linguistic behaviors. volume 6356, pages 265–271.

[Bender and Koller, 2020] Bender, E. M. and Koller, A. (2020). Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, Online. Association for Computational Linguistics.

[Bielawski et al., 2022] Bielawski, R., Devillers, B., Van De Cruys, T., and Vanrullen, R. (2022). When does CLIP generalize better than unimodal models? when judging human-centric concepts. In *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 29–38, Dublin, Ireland. Association for Computational Linguistics.

[Bisk et al., 2020] Bisk, Y., Holtzman, A., Thomason, J., Andreas, J., Bengio, Y., Chai, J., Lapata, M., Lazaridou, A., May, J., Nisnevich, A., Pinto, N., and Turian, J. (2020). Experience grounds language.

[Bojanowski et al., 2016] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *CoRR*, abs/1607.04606.

[Bowman et al., 2015] Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

[Brown et al., 2020] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners.

[Cassell et al., 1994] Cassell, J., Pelachaud, C., Badler, N., Steedman, M., Achorn, B., Becket, T., Douville, B., Prevost, S., and Stone, M. (1994). Animated conversation: Rule-based generation of facial expression, gesture spoken intonation for multiple conversational agents. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '94, page 413–420, New York, NY, USA. Association for Computing Machinery.

[Cassell and Thórisson, 1999] Cassell, J. and Thórisson, K. (1999). The power of a nod and a glance: Envelope vs. emotional feedback in animated conversational agents. *Applied Artificial Intelligence*, 13:519–538.

[Chen et al., 2021] Chen, J., Guo, H., Yi, K., Li, B., and Elhoseiny, M. (2021). Visualgpt: Data-efficient image captioning by balancing visual input and linguistic knowledge from pretraining. *CoRR*, abs/2102.10407.

[Chen and Jin, 2017] Chen, S. and Jin, Q. (2017). Multi-modal conditional attention fusion for dimensional emotion prediction. *CoRR*, abs/1709.02251.

[Choksi et al., 2021] Choksi, B., Mozafari, M., Vanrullen, R., and Reddy, L. (2021). Multimodal neural networks better explain multivoxel patterns in the hippocampus. In *Neural Information Processing Systems (NeurIPS) conference: 3rd Workshop on Shared Visual Representations in Human and Machine Intelligence (SVRHM 2021)*.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

[Deng, 2012] Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142.

[Denkowski and Lavie, 2014] Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*.

[Desai and Johnson, 2020] Desai, K. and Johnson, J. (2020). Virtex: Learning visual representations from textual annotations. *CoRR*, abs/2006.06666.

[Devillers et al., 2021] Devillers, B., Choksi, B., Bielawski, R., and VanRullen, R. (2021). Does language help generalization in vision models? In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 171–182, Online. Association for Computational Linguistics.

[Devlin et al., 2018] Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

[Dosovitskiy et al., 2020] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929.

[Eminaga et al., 2018] Eminaga, O., Eminaga, N., Semjonow, A., and Breil, B. (2018). Diagnostic classification of cystoscopic images using deep convolutional neural networks. *JCO Clinical Cancer Informatics*, (2):1–8. PMID: 30652604.

[Engstrom et al., 2019a] Engstrom, L., Ilyas, A., Salman, H., Santurkar, S., and Tsipras, D. (2019a). Robustness (python library).

[Engstrom et al., 2019b] Engstrom, L., Ilyas, A., Santurkar, S., Tsipras, D., Tran, B., and Madry, A. (2019b). Adversarial robustness as a prior for learned representations.

[Fang et al., 2021] Fang, Z., Wang, J., Hu, X., Liang, L., Gan, Z., Wang, L., Yang, Y., and Liu, Z. (2021). Injecting semantic concepts into end-to-end image captioning. *CoRR*, abs/2112.05230.

[Finkelstein et al., 2001] Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: The

concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414.

[Frome et al., 2013] Frome, A., Corrado, G., Shlens, J., Bengio, S., Dean, J., Ranzato, M., and Mikolov, T. (2013). Devise: A deep visual-semantic embedding model.

[Gastaldi, 2021] Gastaldi, J. (2021). Why can computers understand natural language?: The structuralist image of language behind word embeddings. *Philosophy Technology*, 34.

[Geirhos et al., 2019] Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. (2019). Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*.

[Girshick et al., 2013] Girshick, R. B., Donahue, J., Darrell, T., and Malik, J. (2013). Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2524.

[Goh et al., 2021] Goh, G., †, N. C., †, C. V., Carter, S., Petrov, M., Schubert, L., Radford, A., and Olah, C. (2021). Multimodal neurons in artificial neural networks. *Distill*. https://distill.pub/2021/multimodal-neurons.

[Goodfellow et al., 2014] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks.

[Harnad, 1990] Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1):335–346.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9:1735–80.

[Hossain et al., 2019] Hossain, M. Z., Sohel, F., Shiratuddin, M. F., and Laga, H. (2019). A comprehensive survey of deep learning for image captioning. *ACM Comput. Surv.*, 51(6).

[Hu and Singh, 2021] Hu, R. and Singh, A. (2021). Transformer is all you need: Multimodal multitask learning with a unified transformer.

[Iwana et al., 2017] Iwana, B. K., Rizvi, S. T. R., Ahmed, S., Dengel, A., and Uchida, S. (2017). Judging a book by its cover.

[Jia et al., 2021] Jia, C., Yang, Y., Xia, Y., Chen, Y., Parekh, Z., Pham, H., Le, Q. V., Sung, Y., Li, Z., and Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. *CoRR*, abs/2102.05918.

[Joulin et al., 2016a] Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016a). Bag of tricks for efficient text classification. *CoRR*, abs/1607.01759.

[Joulin et al., 2016b] Joulin, A., Van Der Maaten, L., Jabri, A., and Vasilache, N. (2016b). Learning visual features from large weakly supervised data. In *European Conference on Computer Vision*, pages 67–84. Springer.

[Karpathy and Fei-Fei, 2014] Karpathy, A. and Fei-Fei, L. (2014). Deep visual-semantic alignments for generating image descriptions. *CoRR*, abs/1412.2306.

[Kolesnikov et al., 2019] Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., and Houlsby, N. (2019). Large scale learning of general visual representations for transfer. *CoRR*, abs/1912.11370.

[Kriegeskorte et al., 2008] Kriegeskorte, N., Mur, M., and Bandettini, P. (2008). Representational similarity analysis - connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2:4.

[Krizhevsky et al.,] Krizhevsky, A., Nair, V., and Hinton, G. Cifar-10 (canadian institute for advanced research).

[Lecun et al., 1998] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278 – 2324.

[Li et al., 2019] Li, L. H., Yatskar, M., Yin, D., Hsieh, C., and Chang, K. (2019). Visualbert: A simple and performant baseline for vision and language. *CoRR*, abs/1908.03557.

[Li et al., 2020] Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., Choi, Y., and Gao, J. (2020). Oscar: Object-semantics aligned pre-training for vision-language tasks. *CoRR*, abs/2004.06165.

[Li and Liang, 2021] Li, X. L. and Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. *CoRR*, abs/2101.00190.

[Lin, 2004] Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

[Lin et al., 2014] Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

[Lindsay, 2021] Lindsay, G. W. (2021). Convolutional neural networks as a model of the visual system: Past, present, and future. *Journal of Cognitive Neuroscience*, 33(10):2017–2031.

[Lu et al., 2019] Lu, J., Batra, D., Parikh, D., and Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *CoRR*, abs/1908.02265.

[Lucieri et al., 2020] Lucieri, A., Sabir, H., Siddiqui, S. A., Rizvi, S. T. R., Iwana, B. K., Uchida, S., Dengel, A., and Ahmed, S. (2020). Benchmarking deep learning models for classification of book covers. *SN Computer Science*, 1(3):139.

[Madry et al., 2017] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

[Mangolin et al., 2020] Mangolin, R. B., Pereira, R. M., Jr., A. S. B., Jr., C. N. S., Feltrim, V. D., Bertolini, D., and Costa, Y. M. G. (2020). A multimodal approach for multi-label movie genre classification. *CoRR*, abs/2006.00654.

[Mehlmann et al., 2014] Mehlmann, G., Häring, M., Janowski, K., Baur, T., Gebhard, P., and Andre, E. (2014). Exploring a model of gaze for grounding in multimodal hri.

[Miech et al., 2019] Miech, A., Zhukov, D., Alayrac, J.-B., Tapaswi, M., Laptev, I., and Sivic, J. (2019). Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2630–2640.

[Mikolov et al., 2013a] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[Mikolov et al., 2013b] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26.

[Miller, 1998] Miller, G. A. (1998). *WordNet: An electronic lexical database*. MIT press.

[Mokady et al., 2021] Mokady, R., Hertz, A., and Bermano, A. H. (2021). Clipcap: CLIP prefix for image captioning. *CoRR*, abs/2111.09734.

[Netzer et al., 2011] Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning.

[Niu et al., 2016] Niu, T., Zhu, S., Pang, L., and El-Saddik, A. (2016). Sentiment analysis on multi-view social data. In *MultiMedia Modeling*, page 15–27.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. (2002). Bleu: a method for automatic evaluation of machine translation.

[Park et al., 2021] Park, S.-W., Ko, J.-S., Huh, J.-H., and Kim, J.-C. (2021). Review on generative adversarial networks: Focusing on computer vision and its applications. *Electronics*, 10:1216.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Édouard Duchesnay (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830.

[Pham et al., 2019] Pham, H., Liang, P. P., Manzini, T., Morency, L.-P., and Póczos, B. (2019). Found in translation: Learning robust joint representations by cyclic translations between modalities. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6892–6899.

[Qiao et al., 2019] Qiao, T., Zhang, J., Xu, D., and Tao, D. (2019). Mirrorgan: Learning text-to-image generation by redescription. *CoRR*, abs/1903.05854.

[Quattoni et al., 2007] Quattoni, A., Collins, M., and Darrell, T. (2007). Learning visual representations using images with captions. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.

[Quiroga et al., 2005] Quiroga, R. Q., Reddy, L., Kreiman, G., Koch, C., and Fried, I. (2005). Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107.

[Radford et al., 2021] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020.

[Radford et al., 2018a] Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018a). Improving language understanding by generative pre-training.

[Radford et al., 2018b] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2018b). Language models are unsupervised multitask learners.

[Radford et al., 2019] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners.

[Ramesh et al., 2022] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. (2022). Hierarchical text-conditional image generation with clip latents.

[Rauber et al., 2017] Rauber, J., Brendel, W., and Bethge, M. (2017). Foolbox: A python toolbox to benchmark the robustness of machine learning models. In *Reliable Machine Learning in the Wild Workshop, 34th International Conference on Machine Learning*.

[Reddy and Thorpe, 2014] Reddy, L. and Thorpe, S. J. (2014). Concept cells through associative learning of high-level representations. *Neuron*, 84(2):248–251.

[Řehůřek and Sojka, 2010] Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. `http://is.muni.cz/publication/884893/en`.

[Reimers and Gurevych, 2019] Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084.

[Ren et al., 2015] Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497.

[Salman et al., 2020] Salman, H., Ilyas, A., Engstrom, L., Kapoor, A., and Madry, A. (2020). Do adversarially robust imagenet models transfer better? In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3533–3545. Curran Associates, Inc.

[Sariyildiz et al., 2020a] Sariyildiz, M. B., Perez, J., and Larlus, D. (2020a). Learning visual representations with caption annotations. In *European Conference on Computer Vision (ECCV)*.

[Sariyildiz et al., 2020b] Sariyildiz, M. B., Perez, J., and Larlus, D. (2020b). Learning visual representations with caption annotations. *CoRR*, abs/2008.01392.

[Sharma et al., 2018a] Sharma, P., Ding, N., Goodman, S., and Soricut, R. (2018a). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia. Association for Computational Linguistics.

[Sharma et al., 2018b] Sharma, P., Ding, N., Goodman, S., and Soricut, R. (2018b). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition.

[Singh et al., 2019] Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., and Rohrbach, M. (2019). Towards VQA models that can read. *CoRR*, abs/1904.08920.

[Sohl-Dickstein et al., 2015] Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N., and Ganguli, S. (2015). Deep unsupervised learning using nonequilibrium thermodynamics. *CoRR*, abs/1503.03585.

[Srinivasan et al., 2021] Srinivasan, K., Raman, K., Chen, J., Bendersky, M., and Najork, M. (2021). Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning. *arXiv preprint arXiv:2103.01913*.

[Srivastava et al., 2012] Srivastava, N., Salakhutdinov, R., et al. (2012). Multimodal learning with deep boltzmann machines. In *NIPS*, volume 1, page 2. Citeseer.

[Szegedy et al., 2016] Szegedy, C., Ioffe, S., and Vanhoucke, V. (2016). Inception-v4, inception-resnet and the impact of residual connections on learning. *CoRR*, abs/1602.07261.

[Telea, 2004] Telea, A. (2004). An image inpainting technique based on the fast marching method. *Journal of Graphics Tools*, 9.

[Tsimpoukelli et al., 2021] Tsimpoukelli, M., Menick, J., Cabi, S., Eslami, S. M. A., Vinyals, O., and Hill, F. (2021). Multimodal few-shot learning with frozen language models. *CoRR*, abs/2106.13884.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

[Vedantam et al., 2014] Vedantam, R., Zitnick, C. L., and Parikh, D. (2014). Cider: Consensus-based image description evaluation. *CoRR*, abs/1411.5726.

[Wah et al., 2011] Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

[Wang et al., 2021] Wang, Z., Yu, J., Yu, A. W., Dai, Z., Tsvetkov, Y., and Cao, Y. (2021). Simvlm: Simple visual language model pretraining with weak supervision. *CoRR*, abs/2108.10904.

[Welch, 1947] Welch, B. L. (1947). The generalization ofstudent's' problem when several different population variances are involved. *Biometrika*, 34(1/2):28–35.

[Williams et al., 2018] Williams, A., Nangia, N., and Bowman, S. (2018). A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

[Xiao et al., 2017] Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.

[Xu et al., 2017] Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., and He, X. (2017). Attngan: Fine-grained text to image generation with attentional generative adversarial networks. *CoRR*, abs/1711.10485.

[Zhou et al., 2019] Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J. J., and Gao, J. (2019). Unified vision-language pre-training for image captioning and VQA. *CoRR*, abs/1909.11059.

[Zhou et al., 2017] Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., and Liang, J. (2017). EAST: an efficient and accurate scene text detector. *CoRR*, abs/1704.03155.

[Zhu et al., 2017] Zhu, J., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593.

Appendix A

Appendix: Text decoder parameters

Parameter	# of epochs	prefix length	CLIP prefix length	mapping type	batch size	fine-tune GPT-2	# of layers	CLIP version
Value	20	40	40	Transformer	64	False	8	ViT-B/32

Table A.1: Parameters used for the training of the text decoder. For details see clipclap.

Appendix B

Appendix: Uncurated captioning examples

	
Ground truth	Pair of commodes side by side in unfinished bathroom area. A torn apart bathroom with some toilets inside of it. A demolished bathroom with two toilets and a window The floor and wall of the bathroom are coming apart. A toilet and bidet sit in a bathroom that is under construction.
Direct method	Damaged CCTV image of restaurant staff posing as uncanny and uncanny people.
Textualised input	A view of a rough and dingy bathroom with many objects in it.

	
Ground truth	A group of people are standing on the sandy beach. Several people on the beach with their surf boards. Three men and three women posing on a beach in front of surf boards. A group of young people standing next to each other on a beach. A group of people pose for a picture near surfboards.
Direct method	A photograph of a young Irish kitty with a sun-dappled beach, and her friends at the bottom of the ocean.
Textualised input	The group of people posing and holding surfboards and a surf board.

	
Ground truth	A cheesy pizza with red peppers is in a box. A meal from japan or china on a tray. A cheesy casserole covered with toppings is depicted. A pizza with cheese and vegetables in a box. A large square shaped pizza covered in melted cheese and veggies.
Direct method	A quick chili sauce knife cut in the background, and green beans, muffin, and muffin
Textualised input	A bunch of cheese, ready to go and baked in a cheesy tortilla

	
Ground truth	A man stands beside his black and red motorcycle near a park. A man in black jacket next to a red motorcycle. An older man is standing beside a red motorcycle. A man standing by a motor cycle on a street. A man riding on the side of a red motorcycle.
Direct method	A Frank Miller Fun Road shot taken from the time I was born in 2006.
Textualised input	This person is showing on the road with some fresh motorcycle parts on the horizon.