



HAL
open science

Active learning for the detection of objects of operational interest in open-source multimedia content

Paul Guélorget

► **To cite this version:**

Paul Guélorget. Active learning for the detection of objects of operational interest in open-source multimedia content. Machine Learning [cs.LG]. Institut Polytechnique de Paris, 2022. English. NNT : 2022IPPAS018 . tel-03947344

HAL Id: tel-03947344

<https://theses.hal.science/tel-03947344>

Submitted on 19 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



INSTITUT
POLYTECHNIQUE
DE PARIS

NNT : 2022IPPAS018

Thèse de doctorat



Active learning for the detection of objects of operational interest in open-source multimedia content

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom SudParis

École doctorale n°626 École doctorale de l'Institut Polytechnique de Paris (EDIPP)
Spécialité de doctorat: Informatique

Thèse présentée et soutenue à Palaiseau, le 9 Décembre 2022, par

PAUL GUÉLORGET

Composition du Jury :

Azeddine Beghdadi Professeur des universités, Université Sorbonne Paris Nord (L2TI)	Président
Alexis Joly Directeur de recherche, Université de Montpellier (LIRMM)	Rapporteur
Jenny Benois-Pineau Professeure des universités, Université de Bordeaux (LaBRI)	Rapporteuse
Anne Verroust-Blondet Chargée de recherche, INRIA Paris (RITS)	Examinatrice
Titus Zaharia Professeur, Télécom SudParis (ARTEMIS)	Directeur de thèse
Bruno Grilheres Senior Expert, Airbus Defence and Space	Encadrant

Notice

This document and its content are property of AIRBUS DEFENCE AND SPACE SAS and Télécom SudParis and must not be copied or distributed without permission. Any use outside of the expressly intended purpose is prohibited.

It is strictly prohibited to reproduce, distribute and use the content of this document without preliminary authorization of the author. Counterfeiters will be held liable for the payment of damages.

Copyright © 2022 - AIRBUS DEFENCE AND SPACE SAS - Télécom SudParis - All rights reserved.

The AIRBUS logo is displayed in a bold, blue, sans-serif font.

To Mélissa, Abdul & Saturne, my lockdown companions

Acknowledgments

This CIFRE thesis is the result of a partnership between AIRBUS DEFENCE AND SPACE and TÉLÉCOM SUDPARIS and was supported by the ANRT, *l'association nationale de la recherche et de la technologie*. I am grateful for this opportunity that was offered to me, at the crossroads where industry and university meet. Not making a choice was a good choice.

From now on I will switch to French, it feels more genuine that way.

Mes remerciements vont tout d'abord à Titus, mon directeur de thèse, et à Bruno, mon encadrant à Airbus. J'aimerais remercier Titus et tous les professeurs de la voie d'approfondissement *High Tech Imaging* à Télécom SudParis pour m'avoir donné l'envie de m'engager dans cette aventure. HTI est un formidable tremplin après lequel arrêter mes études m'a semblé inconcevable. J'aimerais remercier Bruno et Sylvie pour m'avoir accueilli comme stagiaire dans cette belle équipe TECIB7 d'Élancourt, avec déjà la stimulante perspective d'une poursuite en thèse CIFRE. Mais surtout, j'aimerais remercier Titus et Bruno pour la confiance qu'ils m'ont accordée dès le début, pour leur implication, leur patience et leurs précieux conseils. Je tiens à remercier tout particulièrement Titus pour sa disponibilité et sa remarquable réactivité, et pour avoir sans relâche traqué les formulations hasardeuses qui jonchaient mes ébauches de rédactions. Merci à Bruno qui a tout mis en œuvre pour que je ne manque de rien : un encadrement efficace, du matériel performant et un projet coïncidant harmonieusement avec mon sujet de thèse. Je mesure à quel point tout cela fut précieux.

Je remercie l'ensemble des membres du jury, qui m'ont fait l'honneur de bien vouloir étudier avec attention mon travail : Jenny Benois-Pineau et Alexis Joly pour avoir accepté d'être rapporteurs de cette thèse, pour leur lecture approfondie de mes travaux et leurs remarques pertinentes ; Anne Verroust-Blondet pour avoir accepté d'examiner cette thèse ; et enfin Azeddine Beghdadi pour m'avoir fait l'honneur d'accepter de présider ce jury.

Merci à toute la joyeuse équipe d'Élancourt. Les sigles changent mais les souvenirs restent. Je tiens à remercier tout particulièrement les valeureux thésards de cette troupe. Merci à Kilian pour ses conversations méridiennes extrêmement pointues. Merci à Howard pour ses exposés passionnés et ses conseils avisés. J'adresse un remerciement chaleureux à Guillaume pour n'avoir jamais cessé de chercher à m'inclure dans ses projets, mon enthousiasme lui doit beaucoup. Merci à Jacques pour la pâte à crêpes. Enfin, merci à Jonathan pour son éloquence téléphonique et ses services experts.

Je tiens à très sincèrement remercier Benjamin, Paul, Souhir, Sylvain et Ghislain, mes co-auteurs le temps d'un article. Leur sérieux et leur exigence les honorent.

Impossible hélas de nommer tout le monde ici, remercions toutefois Florian et Florentin grâce à qui je ne suis officieusement plus stagiaire, Frédéric, Kadiatou et Guillaume pour leur cruciale chefferie de projet. Merci à Antoine et Nicolas pour le shred forcené, merci à la Commune, whatever.

J'aimerais remercier Vanessa, Tom et Laure, mes merveilleux voisins de terrasse pendant les confinements successifs. La thèse ne devrait pas se vivre enfermé chez soi et fort heureusement, ils étaient là.

Merci du fond du cœur à mes parents, leur soutien indéfectible et leurs encouragements ne m'ont jamais fait défaut, ils furent d'une aide précieuse.

Enfin, merci à ma partenaire de confinement et désormais fiancée, Mélissa, à mes côtés pendant tout ce temps.

Résumé en français

Une profusion de contenus, acteurs et interactions en source ouverte sont ciblées par les analystes à des fins commerciales, politiques ou de renseignement. En effet, le *World Wide Web* et les réseaux sociaux qu'il héberge se sont démocratisés au point qu'il est désormais possible d'y atteindre les individus par des campagnes de communication, d'influencer les communautés qui les rassemblent, de sonder les tendances qui les animent ou de consulter les contenus multimédias qu'ils s'y échangent. Analyser l'immensité des données mises en jeu requiert une assistance automatisée. Les propositions récentes en matière d'architectures de réseaux de neurones ont montré de fortes capacités de traitement des contenus multimédia image et texte, plus particulièrement celles reposant sur des réseaux convolutifs, récurrents et des mécanismes attentionnels. Les modèles proposés sont de plus en plus profonds et comprennent des millions de paramètres entraînaibles, cependant, leur entraînement nécessite des jeux de données massifs, instantanés pour la majorité des classes d'intérêt opérationnel qui sont susceptibles de changer au gré des événements. Pour résoudre ce problème, **l'apprentissage actif** tire parti de la grande quantité de documents non annotés – accessibles en source ouverte, par exemple – en sollicitant un *oracle* humain pour obtenir les labels des documents présumés les plus informatifs, afin d'améliorer la précision du modèle entraîné. Néanmoins, les justifications derrière les décisions du modèle sont opaques, et sans garantie d'être alignées sur les justifications de l'oracle. De plus, à cause de ses longues étapes successives, le déroulement de l'apprentissage actif nuit à ses performances en temps réel. Nos contributions dans cette thèse visent à analyser et résoudre ces problèmes, en quatre étapes. Premièrement, nous observons les justifications derrière les décisions prises par un réseau de neurones dédié à la classification de texte par l'extraction de cartes d'activation de classes. Les informations qu'elles transmettent sont précieuses pour déceler les groupes de mots les plus saillants d'après le modèle. Deuxièmement, nous mettons ces justifications en perspective avec celles élaborées par des humains en comparant le comportement de ce réseau de neurones (qui base ses décisions sur son jeu d'entraînement) avec un modèle par règles reposant sur le recensement de mots appartenant à un vocabulaire précis, ce qui ouvre des perspectives d'amélioration mutuelle pour ces deux modèles. Troisièmement, dans un contexte d'annotations peu abondantes, caractéristique des premières étapes de l'apprentissage actif, nous incitons un réseau de neurones à aligner ses justificatifs sur ceux d'un modèle professeur qui simule les justificatifs d'un oracle humain, et améliorons ainsi sa précision. Cela est notamment rendu possible par le biais de la supervision fine des cartes d'activation de classes ou de mécanismes attentionnels. De plus, nous proposons des critères de sélection pour l'apprentissage actif appliqué à la classification d'images, reposant sur les régions d'intérêt propres à chaque classe recherchée. Nos résultats montrent que l'échantillonnage des images générant le plus de contradictions entre les cartes d'activation et les scores d'attention du modèle surpassent les critères de sélection usuels. Finalement, nous mettons au point et exploitons un système d'apprentissage actif pour surmonter ses limitations, qui consiste en la parallélisation des tâches qui lui incombent dans des micro-services dédiés. Ces études ont été menées sur des données monomodales texte ou image, ou sur des paires multimodales texte/image, principalement des articles de presse en anglais et en français. À travers les chapitres de cette thèse, nous traitons plusieurs cas d'utilisation, parmi lesquels la reconnaissance du vague et des fausses nouvelles, la détection du manque d'avis contradictoires dans les articles et la classification d'articles comme abordant des sujets arbitrairement choisis, tels que les manifestations ou la violence.

Contents

1	Introduction	11
1.1	Context	11
1.1.1	Open Source Intelligence	11
1.1.2	Deep Learning	12
1.2	Contributions	15
1.3	Thesis Organization	16
I	Related Work	19
2	Deep Learning for the Processing of Image, Text and Multi-modal Data	21
2.1	Deep Learning	22
2.1.1	Neural Networks	22
2.1.2	Training	25
2.2	Image Processing	27
2.2.1	Convolutions	27
2.2.2	Pooling	27
2.2.3	Notorious CNN Architectures	29
2.2.4	CNN feature maps as descriptors	32
2.2.5	Object detection	33
2.3	Natural Language Processing	35
2.3.1	Word embeddings	35
2.3.2	Neural networks for text classification	37
2.4	Multi-modal Data Processing	43
2.4.1	Representation	43
2.4.2	Translation	44
2.4.3	Alignment	44
2.4.4	Fusion	46
2.4.5	Co-learning	48
2.5	Conclusion	49
3	Deep Active Learning	51
3.1	Active learning	52
3.2	Active learning scenarios	52
3.2.1	Pool-based active learning	53
3.2.2	Stream-based active learning	53
3.2.3	Query synthesis	54
3.2.4	Hybrid scenarios	55
3.3	Uncertainty-based strategies	56
3.3.1	Calibration of neural networks	57
3.3.2	Bayesian Neural Networks	58
3.3.3	Bayesian Active Learning by Disagreement	58
3.3.4	Ensembles	59
3.4	Information Density	60

3.5	Diversity-based criteria	62
3.5.1	A core-set approach for diversity	62
3.5.2	Contextual diversity	62
3.5.3	BatchBALD	65
3.6	Adversarial approaches	65
3.7	Semi-supervision	66
3.8	Conclusion	67
II	Contributions	69
4	An Interpretable Model to Measure Fakeness and Emotion in News	71
4.1	Introduction	72
4.2	Related Work	73
4.2.1	The fake news context	73
4.2.2	Machine learning on related text classification problems	74
4.2.3	Explainable AI and interpretable fake news	74
4.2.4	Discussion	75
4.3	TC-CNN: an interpretable model for biased news article classification	75
4.3.1	Spatially interpretable architecture	75
4.3.2	Predicting fake-like articles	76
4.3.3	Secondary task: predicting emotion	77
4.4	Experiments: TC-CNN at work	78
4.5	Measuring offences and biases in the press	79
4.5.1	Sources of 2020 data	79
4.5.2	Qualitative analysis on a sample	80
4.5.3	Statistical overview	80
4.6	Conclusion	83
5	Combining Vagueness Detection with Deep Learning to Identify Fake News	85
5.1	Introduction	86
5.2	Vagueness detection: VAGO	87
5.2.1	Vagueness and subjectivity	87
5.2.2	A typology of vague expressions	87
5.2.3	VAGO: detection and scoring	88
5.2.4	VAGO Implementation	89
5.2.5	Online tool	89
5.3	Comparison and combination	90
5.3.1	Validation of TC-CNN on unseen data	90
5.3.2	VAGO experimental settings	91
5.3.3	Comparison and correlation of VAGO and TC-CNN	92
5.3.4	Word-level analysis: exploiting the deep to find new keywords	92
5.4	Discussion	96
5.5	Conclusion	99
6	Deep Active Learning with Rationales for Text Classification	101
6.1	Introduction	102
6.2	Related Work	103
6.3	Learning with rationales	104
6.3.1	Background	104
6.3.2	Training a classifier with contextual and salient knowledge	105
6.4	Experimental results	109
6.4.1	Evaluation protocol	109
6.4.2	Contextualized Simulated Rationales	109
6.4.3	Experimental results	113
6.5	Conclusion	113

7	Active Learning with Rationales for Image Classification	117
7.1	Introduction	118
7.2	Active Image Classification with Rationales	119
7.2.1	Learner architecture and integration of rationale constraints	120
7.2.2	Automatic extraction of simulated rationales	126
7.2.3	Training an image classifier with rationales: experimental results	127
7.3	Sampling strategies for active learning with rationales	134
7.3.1	Notations	134
7.3.2	Sampling criteria	134
7.3.3	Active learning with rationales: experimental results	136
7.4	Conclusion and perspectives	140
III	Technical Solution	141
8	Real-Time Active Learning as Micro-Services	143
8.1	Introduction	144
8.2	Related Work: News Articles and Active Learning	145
8.2.1	Analyzing Media and Politics	145
8.2.2	The Need for Active Learning	146
8.2.3	Discussion	146
8.3	Real-Time Active Learning as Micro-Services	147
8.3.1	Motivations	147
8.3.2	Notations	149
8.3.3	Proposed Organization	149
8.4	Application on French Politics	155
8.4.1	Data Collection	156
8.4.2	Detection of Articles Conveying One-sided Opinions	157
8.4.3	Demonstration and Violence	158
8.5	Conclusion and Future Work	163
IV	Conclusion and Future Work	165
9	Conclusion	167
9.1	Contributions summary	167
9.1.1	Scientific contributions	167
9.1.2	Operational contributions	169
9.2	Limitations	169
9.3	Perspective and future work	170

List of Figures

1.1	A semi-log plot of transistor counts for microprocessors against dates of introduction, nearly doubling every two years, ourworldindata.org	13
1.2	Logos of deep learning frameworks TensorFlow, Keras and PyTorch. TensorFlow, the TensorFlow logo and any related marks are trademarks of Google Inc. PyTorch, the PyTorch logo and any related marks are trademarks of Facebook, Inc.	13
1.3	Images taken from the ImageNet dataset, mostly animals and objects. [153]	14
2.1	Representation of a neural network	22
2.2	The perceptron model	23
2.3	Activation functions sigmoid and tanh	24
2.4	ReLU activation	25
2.5	First convolutional layer with 2×2 kernel	28
2.6	A second convolutional layer with 2×2 kernel	28
2.7	Pooling examples	29
2.8	LeNet-5 architecture [101]	29
2.9	AlexNet architecture [96]	30
2.10	the first Inception module [172]	31
2.11	Inception v4 [173]	31
2.12	residual learning in ResNet	32
2.13	Faster R-CNN architecture [145]	33
2.14	Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors (Word2vec) of countries and their capital cities. [124]	36
2.15	1-dimensional convolutional neural network for text classification	38
2.16	2-layer recurrent neural network for text classification	39
2.17	Alignments discovered by the attention mechanism of a sequence-to-sequence translation network [14]. Attention scores acknowledge correspondences between words of both languages and highlight the reverse order of adjectives in noun phrases.	40
2.18	Self-attention scores on texts classified as a <i>good</i> or <i>bad</i> reviews. Red values accounts for high scores. The attention mechanism focuses on words responsible for class membership according to the trained model.	40
2.19	The Transformer architecture [183]	42
2.20	Joint representation versus coordinated representation [15]. A joint representation aggregates information from several modalities into a single representation, whereas a coordinated representation projects uni-modal information to a common space where representations can be compared regardless of their modality of origin.	44
2.21	Multi-modal autoencoder architectures [189]	45
2.22	Architecture of the visual-semantic attention network	46
2.23	Early versus late multi-modal fusion with neural networks	47
2.24	CentralNet [186]	48
3.1	Pool-based active learning scenario	52
3.2	Stream-based active learning scenario	53
3.3	Query synthesis active learning scenario	54
3.4	Alert-raising, hybrid pool/stream active learning scenario	55

3.5	Reliability diagrams for a 5-layer LeNet (left) and a 110-layer ResNet (right) on CIFAR-100 [57].	57
3.6	A limitation of uncertainty sampling: possibly uninformative outliers	59
3.7	Density estimations examples: a) histogram vs. Gaussian kernel density; b) narrow bandwidth vs. large bandwidth. Visual generated on astroml.org	61
3.8	100 data points (red) sampled from a distribution of 1,000 2-d points (blue) with the greedy core-set algorithm, favouring the diversity of sampled points. Unless it is explicitly taken care of, this method tends to sample peripheral outliers.	63
3.9	Variational Adversarial Active Learning [168]	66
3.10	State-Relabelling Adversarial Active Learning [201]	67
3.11	Active learning with pseudo labelling of unlabelled examples of high certainty	67
4.1	Extraction of Class Activation Maps from a tokenized sentence.	77
4.2	Examples of CAMs explaining a bias about Clinton: original (top), with replacements (bottom).	79
4.3	Words with highest CAMs, in their text	81
4.4	Comparing the datasets, by emotion (a), by presence of hate (b)	81
4.5	Article quality by source for <i>gab_trends</i> (a), for <i>news_outlets</i> and <i>the_onion</i> (b)	82
4.6	Emotion (a) and hatred (b) compared to fakeness, on the union of the three datasets	82
5.1	Results summary for text with three sentences	90
5.2	Classification of source corpora in the <i>test</i> dataset as <i>informative</i> (0) or <i>biased</i> (1) by TC-CNN.	91
5.3	Comparison between bias as predicted by TC-CNN and ratio of vague sentences (a), ratio of vague-subjective sentences (b) in texts, as predicted by VAGO.	93
5.4	Distribution of vague sentences ratio (a), vague-subjective sentences ratio (b) across sources.	94
5.5	Average “fakeness” scores of VAGO entries and other adjectives and adverbs according to TC-CNN class attention maps, defined in Equation 4.2.	95
6.1	An iteration of active learning with rationales in a pool-based scenario	102
6.2	LCS knowledge: class specificity of words taken in their context.	107
6.3	Knowledge transfer of spatial domain knowledge	107
6.4	Sparse spatial domain knowledge: label specificity of words taken in their context.	109
6.5	Extracted spatial domain knowledge, IMDB dataset (best viewed with colors)	110
6.6	Extracted spatial domain knowledge, WvsH dataset (best viewed with colors)	111
6.7	Simulated rationales ($\alpha = 1\%$)	112
6.8	Comparison of Lw/oR, LwDR and LwSR — IMDB dataset	114
6.9	Comparison of Lw/oR, LwDR and LwSR — WvsH dataset	114
7.1	Rationales for class membership can be evenly distributed across the image or strongly localized in a salient region. Membership to a hypothetical painting class is due to the overall texture information (a) whereas membership to the smoking class is strongly localized around the cigarette, the hand and the mouth (b).	120
7.2	RoI features extraction	121
7.3	Baseline method architecture	123
7.4	Neural network architecture for DMFM and DWFm approaches.	124
7.5	Neural network architecture for SAS approach.	126
7.6	Rationales $R(x)$ (in green) are simulated using oracle model O . Simulated rationales are then exploited by the learner model L when it is trained with a narrow training set \mathcal{T}_L	126
7.7	Simulated rationales from oracle model, Stanford40Actions dataset.	130
7.8	Simulated rationales from oracle model, Twitter Military dataset. No rationales were simulated for non-military images.	131
7.9	Stanford40Actions: validation accuracy when using rationales as a function of the training set size.	132
7.10	Stanford40Actions: validation accuracy using rationales during training (w/ rationales) and not using them (w/o rationales) for every studied method.	132
7.11	Twitter Military: validation accuracy when using rationales as a function of the training set size.	133
7.12	Twitter Military: validation accuracy using rationales during training (w/ rationales) and not using them (w/o rationales) for every studied method.	133
7.13	Learning curves obtained with different sampling criteria	137

7.14	Validation accuracy using rationales during training (w/ rationales) and not using them (w/o rationales), for every compared sampling criterion.	138
7.15	Difference in validation accuracy for each sampling criterion, between using rationales during training versus not using them.	139
8.1	ReALMS organization. Several docker services cooperate to carry out all time-consuming tasks required by active learning.	149
8.2	Template of models used in ReALMS	155
8.3	Domains and number of top-level comments of submissions crawled on reddit.com/r/france.	156
8.4	Domains and text lengths distributions of news articles crawled under the “Politique” (Politics) flair and used to train one-sided opinion detection.	157
8.5	Learning curves for one-sided opinion detection (left) and demonstration/violence detection (right). “New archit.” signifies that the model was obtained with a new, random architecture, as opposed to improving the already existing model.	158
8.6	Domains, title lengths and image size distributions of news articles composing the demonstration/violence unlabelled corpus.	159
8.7	Opinion score distribution across news outlets	161
8.8	Mean number of comments between factual and opinion-relaying articles, per news outlet	162
8.9	Opinion score distributions between articles mentioning – or not – “Macron” in their title (left); Difference in median opinion score between articles mentioning – or not – “Macron” in their titles, per outlet (right).	162
8.10	Demonstration score distribution of article previews (left) and violence score distribution of 8 articles previews detected as belonging to the <i>demonstration</i> class (right), across news outlets	164

List of Tables

2.1	Nearest neighbour to the word “play” using ELMo contextualized embeddings [140].	36
4.1	Classification scores	78
4.2	Most frequent words, excluding stopwords, on <i>news_outlets</i>	80
5.1	Thirty most bias-inducing terms according to TC-CNN, filtered to adjectives and adverbs with at least 10 occurrences. Entries are sorted by descending average CAM scores (avg column). VAGO terms are in bold. Miscategorized adjectives and adverbs are in gray.	97
5.2	Thirty least bias-inducing terms according to TC-CNN, filtered to adjectives and adverbs with at least 10 occurrences. Entries are sorted by ascending average CAM scores (avg column). VAGO terms are in bold. Miscategorized adjectives and adverbs are in gray.	98
7.1	Active learning results on validation data. Results are averaged over active learning iterations (omitting first bootstrap iteration).	136
8.1	Main documents table	150
8.2	labelled documents table	150
8.3	Sampled documents table	150
8.4	Sampling criteria.	152
8.5	Most positive and negative results for label “one-sided opinion” (from unlabelled corpus)	158
8.6	Target phrases used to pre-filter news articles.	159
8.7	Influence of input modalities on prediction scores	160
8.8	Highest predictions scores for label “demonstration” (left) and “violence” (right), taken from unlabelled corpus	160
8.9	Opinion scores of nouvelobs.com article whose titles contain “Macron”.	163

Chapter 1

Introduction

1.1 Context

1.1.1 Open Source Intelligence

The profusion of content, actors and interactions openly accessible that flourish on the web are targeted by analysts for intelligence, marketing or political purposes. Violence, threats, foreign propaganda and promotion of terrorism are a reality, even on open social media. These contents are taken very seriously by the intelligence agencies and justice. Just like harmful online behaviour can cause real-life harm, it can also have real-life consequences for the perpetrator. Online violence is now taken very seriously by the French authorities: the online expression of an opinion favourable to terrorism or of an incitement to terrorism are punishable by imprisonment¹. Also, online bullying and death threats even posted anonymously can lead to sentences, as illustrated by the very mediatized *Mila case* where online bullies were sentenced to suspended prison and fines².

Open source data are also a prime concern for marketing departments. Companies take care of their image by posting their communication on Twitter, Facebook, Instagram, etc., interacting with users and paying to promote their content. However, beyond that, companies have a need to analyse how much and how good their brand and their products are being talked about in the news and social medial, to evaluate their communication strategy and improve their reputation and exposition to the public.

Watching the immensity of open source data can hardly be successfully done without automated assistance. Machine learning and deep learning in particular have become unavoidable tools, proposing state-of-the-art techniques and solutions to help answer an analyst's needs.

¹<https://www.service-public.fr/particuliers/vosdroits/F32512>

²https://fr.wikipedia.org/wiki/Affaire_Mila

1.1.2 Deep Learning

With the explosion of uploaded data, notably on social media, open source intelligence has known an unprecedented growth during these last years. From now on, several operational services are in charge of following and detecting information of interest published on openly accessible platforms. The recent boom in machine/deep learning techniques has been favoured by several factors : the emergence of efficient and innovatory model architectures on some flagship tasks first, then on various tasks that benefit from them; the exponential performances of computing hardware, at an affordable price; the development of open-source deep learning libraries and the re-usability of major contributions; finally, the ever-growing amount of openly accessible data.

Efficient, innovative deep learning architectures. The last decade in machine learning has seen the actualization of well-established mechanisms together with the emergence of brand-new architectures. Feed-forward layers in neural networks are based on 1958's Perceptron [149]. Convolutional neural networks (CNN), who have taken over computer imagery by storm, are an adaptation of 1980's Neocognitron [44], whose authors were inspired by neurobiology. The recurrent neural networks (RNN) mechanism have been proposed in 1986 [152] and are still highly popular to process time series. Innovative architectures like Transformers [183] compete with older models in this teeming ecosystem. As time goes by, hybridization may blur the lines between neural network architectures: the Conformer, a convolution-augmented Transformer, achieves state-of-the-art performances in automatic speech recognition [55].

Moore's law. Deep learning applications were mostly theoretical during the XXth century, when memory and computing power were immensely scarcer. In 1965, electronics engineer Edgar G. Moore postulated that the complexity of semiconductors is supposed to double every year at a constant cost. A latter adjustment of this law in 1975 stated that the number of transistors on silicon microchips would double every two years. This prediction turned out to be surprisingly accurate (Figure 1.1). The exponential growth of computing capacities have permitted to exploit the potential of promising theoretical contributions and to implement, train and evaluate neural networks with millions of trainable parameters. Moreover, the capacities of Graphics Processing Units (GPU) have grown even faster in recent years. Originally designed for graphical rendering, GPUs have been found particularly well-suited to optimize highly parallelizable tasks like convolutions. Finally, storage and data exchange rates have known a significant improvement, allowing to store and quickly exchange voluminous datasets and model parameters.

Frameworks. In addition to the exponential growth of hardware capabilities, the booming/democratization of deep learning goes hand in hand with the emergence of dedicated frameworks, most of which are free to use and open source. Popular deep learning libraries TensorFlow, Keras and PyTorch (1.2) are propelled by active development teams and by the enthusiastic involvement of their user communities. Such deep learning frameworks propose interfaces for python and other programming languages like C++ (TensorFlow, PyTorch), provide CUDA support



Figure 1.3: Images taken from the ImageNet dataset, mostly animals and objects. [153]

the great diversity of classes featured in such datasets allow the models to learn and recognize various low and high level visual characteristics. Following the transfer learning paradigm, the models can then be used as fixed deep feature extractors or partially retrained to recognize unrelated classes.

In addition to such large publicly available datasets, the World Wide Web is a quasi-infinite source of open source unstructured data. For example, free access press articles and images, public databases, public contents posted by users on social media, and interactions between users can be used to detect or forecast events. It happens that publicly available data is pseudo-structured with human or algorithmic annotations: on many “subreddits” (thematic forums) of the Reddit website³, users can manually assign a “flair” to describe submissions. The GDELT project⁴ monitors events happening all over the world as they are mentioned in international press and events are sorted in a homemade ontology made of four main classes (verbal cooperation, material cooperation, verbal conflict, material conflict) and 20 CAMEO [156] codes (*make public statement, engage in diplomatic cooperation, exhibit force posture, etc.*).

The profusion of openly available data is counterbalanced by its unstructured and unlabelled aspects. Another limiting factor to its wide, free and open exploitation is embodied by laws and regulations that aim to protect the personal data of citizens. Worth mentioning in this domain, the General Data Protection Regulation (GDPR), which is a regulation in European Union law on data protection and privacy that became enforceable in 2018. It contains provisions and requirements related to the processing of personal data of individuals who are located in the European Economic Area and is vital to the citizens’ sovereignty over their personal data.

A final issue that is raised when collecting data to train a model is the presence of noise and/or bias. Depending on the critical character of the application, collected datasets must be thoroughly investigated. Thus, biased data

³<https://www.reddit.com/>

⁴<https://www.gdeltproject.org/>

leads up to biased models and brings about malfunctions or discriminatory behaviours.

1.2 Contributions

Efficient classification and detection methods are by definition supervised processes and therefore require large amounts of labelled examples to be trained. By putting the human in the loop and asking them to provide ground-truth labels for the most informative examples, active learning is a mechanism that is highly compatible with any scenario with a profusion of content but lacking labels, which is the case of open source intelligence (OSINT) scenarios like security watch, analysis of the political landscape, or automatic segmentation of aerial imagery. Moreover, open source data exists in a variety of modalities (text, image, audio, video) that may carry complementary, insightful information.

The implementation of classification models for intelligence or marketing purposes brings out a need for explainability, to earn trust of the operator or the regulator, to explain errors, to improve systems or to gain valuable insights on the studied data.

The work presented in this thesis is an effort towards bringing together the user and the machine, since neural networks are often considered today as indecipherable black boxes. A first contribution concerns the textual analysis and proposes to adapt an interpretation method to a Natural Language Processing (NLP) neural classifiers in order to demonstrate its usefulness to identify salient words and check a model's relevance. Neural networks compete with various methods, sometimes solving identical or neighbouring problems. Beyond performance metrics, we lack the ability to discriminate a neural network's behaviour compared to other methods. That is why, in this work, we investigate a in-depth comparison between methods that rely on completely different paradigms, which represents a crucial step towards enriching these methods with each other.

Users can benefit from the previous insights neural networks can provide us with. The opposite principle would consist in going further than only learning from examples, by harnessing human high level, expert knowledge during the training itself. This issue concerns our second family of contributions, which is related to the introduction of rationales into the learning process. Such additional information is particularly useful when the learning data is scarce, and thus can be beneficial in active learning scenarios. This issue is still poorly addressed by state-of-the-art methods. In our work, we have investigated different methods that allow injecting user's knowledge into a neural network's training, for both image and text classification.

Finally, from a technical point of view, we noticed the gap between abstract active learning setups in related state-of-the-art articles and the real-life conditions in which it operates: the main concern is related to the lack of availability of the human user responsible for providing ground-truth labels or the inefficiency of the sequential aspect of active learning loops. In a final contribution, we propose to implement a technical framework capable of real-life active learning for text, image and multi-modal classification.

1.3 Thesis Organization

This thesis consists of three distinct parts corresponding to the conducted research. Part I is a bibliographic part, divided into two chapters. Chapter 2 first presents general knowledge related to neural networks and their training process. Additionally, neural networks techniques and applications dedicated to the processing of images, texts, and multi-modal documents are discussed. This chapter ends with a conclusion expressing the interest of active learning as a method to overcome a shortage of ground-truth labels.

Then, chapter 3 presents and illustrates the active learning principle and its various scenarios. Reference is made to active learning sampling strategies, that follow criteria designed to select informative documents. They are notably characterized by their expression of uncertainty, information density, or diversity, and are sorted accordingly in this chapter. Furthermore, recent adversarial methods for active learning are mentioned, before concluding with the inadequate aspects of active learning investigated in our contributions.

Part II of this thesis presents our contributions. It explores the link between neural classifiers and the underlying knowledge hidden behind the classification task they are designed to solve. This knowledge incarnates in many ways: it can be an emanation of the data, features a trained model has learnt as rationales for class membership, or on the contrary, what the model's designer or user believes constitutes a valid rationale for class membership. To illustrate this duality, in this thesis, a particular importance is given to open source data. In chapter 4, we acknowledge that disinformation campaigns and the rise of online post-truth emphasize the need for automated support for the assessment of press trustworthiness. We propose an interpretable text classifier trained on an aggregation of datasets, and discuss the indications it provides on various article sources, including American alt-right media.

As a continuation of this work, in chapter 5 we combine the aforementioned convolutional approach with an algorithm that uses semantic rules combined with NLP techniques to measure vagueness and subjectivity in texts. After comparing the results of the two methods on four corpora, we present the mutual benefits yielded by this comparison.

In an early active learning setup, neural networks are penalized by the scarcity of the ground-truth labels. In chapter 6, we introduce a new learning strategy, which consists of inserting in the early stages of the active learning process some additional, local and salient knowledge, presented under the form of simulated, human like rationales. The experimental results obtained demonstrate that exploitation of such rationales permits to speed up the active learning process.

Similarly, in chapter 7, we first introduce different strategies that allow injecting expert spatial rationales into the learning process of an image classifier. The experimental results obtained demonstrate that the proposed self-attention supervision strategy permits to significantly improve the model's performances. A second contribution concerns the sampling strategies that consist of selecting, during the training phase, images whose rationales are

more likely to provide insightful, beneficial information.

Part III and chapter 8 describe our technical solution for active learning, named *ReALMS*, for *Real-Time Active Learning as Micro-Services*, designed to overcome a crucial limitation of active learning implementations, namely the iterative sequence of time-consuming tasks. The parallelization of said tasks in our framework enables the quick elaboration of models that learn to recognize arbitrary classes. Evidence of this capacity is given by a case study on French news outlets: the detection of subjectivity, demonstrations and violence.

Finally, concluding part IV and chapter 9 summarize our contributions, state limitations of our approaches and present some perspectives of future research.

Part I

Related Work

Chapter 2

Deep Learning for the Processing of Image, Text and Multi-modal Data

Abstract

During the last decade, neural networks and their associated techniques have permitted large steps forward in a variety of domains, among which the processing of images and natural language. Innovative neural architectures and training methods allowed to semantically classify images and multilingual texts, synthesize or transform documents, or describe their semantic content by a vector representation or a short textual description. The first section of this chapter introduces deep learning in general, while the second one describes various neural networks architectures dedicated to image processing. The neural networks dedicated to text classification, are then described as well as the multi-modal text and image processing approaches. Finally, the last section concludes this chapter with an analysis of constraints in terms of labelled data and their inadequacy with our objectives.

2.1 Deep Learning

2.1.1 Neural Networks

A neural network is a weighted, directed graph where vertices represent neurons and edges represent synapses. The neurons are usually arranged in layers: a first layer being the input layer, the last layer being the output layer and the middle ones being called hidden layers. When the output of each neuron in a given layer is connected to the input of every neuron in the following layer, we have a so-called *fully-connected neural network*. A network is said to be *deep* if it has a relatively important number of hidden layers. Thus, the *depth* of a network is commonly assimilated to its number of layers. Figure 2.1 represents a fully-connected deep neural network with five layers: an eight-neuron input layer, three nine-neuron hidden layers and a four-neuron output layer. Hidden layers may have different sizes (numbers of neurons).

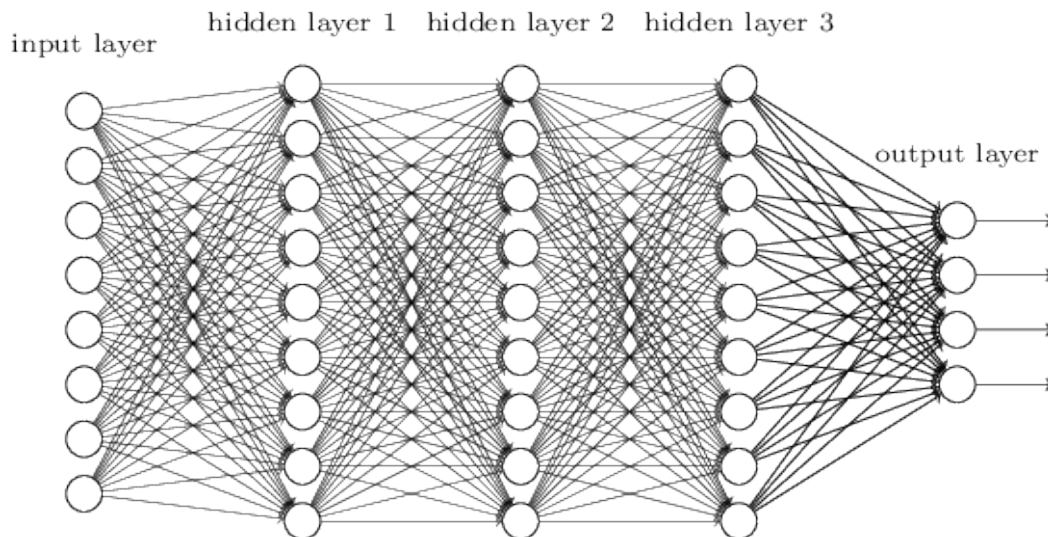


Figure 2.1: Representation of a neural network

Deep learning models are inspired from the human brain structure, hence the name of the fundamental unit: the *neuron*. Each neuron has several inputs x_1, x_2, \dots, x_n and a single output, referred to as its activation y . Each input is given a weight w_1, w_2, \dots, w_n that ranks its importance in the final result. An *activation function* f is applied to the linear combination of inputs; its result is the neuron's *activation*.

Decade 1950 marks the development of the first neural networks. The *perceptron* model [149] (Figure 2.2) performs a linear combination of its binary inputs x_i with the set of weights w_i and adds a *bias* b (equation 2.1). This defines the neuron's *function*.

$$z = \sum_{i=1}^n w_i x_i + b \quad (2.1)$$

In the original perceptron, the result z is thresholded by an activation function f whose output is either 0 or 1

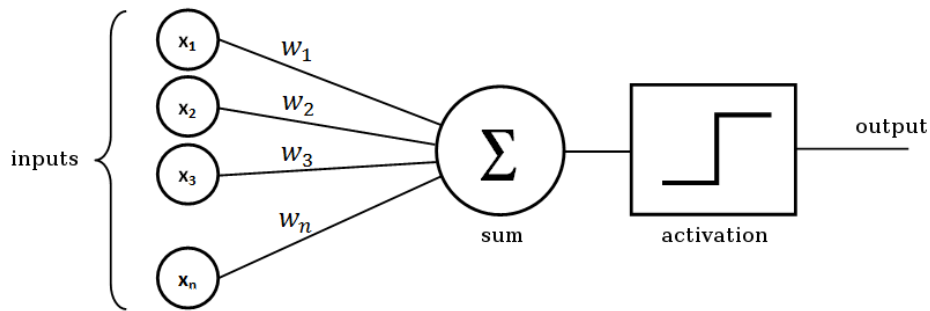


Figure 2.2: The perceptron model

(equation 2.2). The activation function is responsible of the over-all non-linearity of the system, which is mandatory when considering classification applications.

$$y = f(z) = \begin{cases} 1, & \text{if } z \geq \text{threshold} \\ 0, & \text{otherwise} \end{cases} \quad (2.2)$$

Modern neural networks typically involve thousands to millions of neurons in a single layer, and deep neural networks (DNNs) involve tens to hundreds of layers. In such large networks, billions of weights and biases need to be calculated. Adding larger layers of neurons increases the dimensionality of the data representation

In the purpose of classification, given an input X with a ground-truth class c_i among m possible classes, we aim at maximizing the value of the i^{th} output of the last layer.

To achieve supervised classification, a *learning process*, aiming at determining the sets of weights and biases for the neurons involved in the network needs to be achieved. To this purpose, a training set consisting of pairs of samples and corresponding labels $\{(x_i, c_i)\}$ is supposed to be available. The training process adjusts the parameters of the network thus that a global cost function, usually called *loss function* and expressing the error between predicted and ground truth labels is minimized. The learning process is further detailed in Section 2.1.2.

One of the key issues when designing neural networks concerns the choice of the activation function, which is responsible of the non-linearity of the system and can dramatically impact the efficiency of the learning process. In the original perceptron formulation, the step activation function takes only binary values. As a consequence, an activation change for a single neuron can have a determinant impact on the whole network.

Several alternative activation functions are available and deployed in modern neural architectures. The *sigmoid* function was the most used activation function among early neural networks:

$$y = f(z) = \frac{1}{1 + e^{-z}} \quad (2.3)$$

where $z = \sum_{i=1}^n w_i x_i + b$. This function is linear near to $z = 0$ but saturates for large values of $|z|$. Its derivative plays a major part during training. Note that:

- the maximum of the derivative is 0.25, for $z = 0$;
- the derivative goes to zero for large values of $|z|$.

Another choice is the hyperbolic tangent activation function, defined as :

$$\forall x \in \mathbb{R}, \tanh(x) = 2 \cdot \text{sigmoid}(2x) - 1. \quad (2.4)$$

The non-linearity of the hyperbolic tangent improves the quality of the training. However the small values of their derivatives makes the network challenging to train.

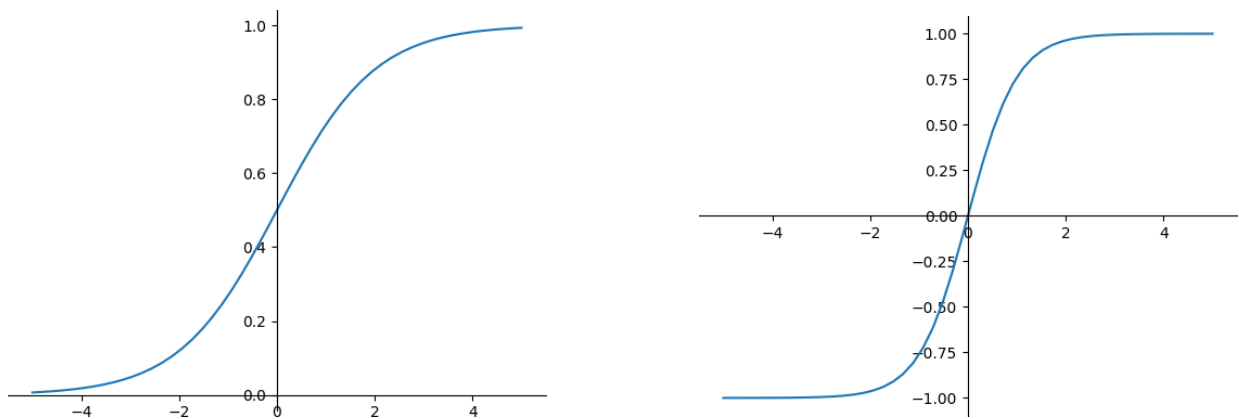


Figure 2.3: Activation functions sigmoid and tanh

To tackle this issue, modern DNNs have adopted the so-called Rectified Linear Unit (ReLU) (2.4), defined as:

$$y = \max(0, a) \quad (2.5)$$

The resulting networks are easier to train and remain responsive for large values of z . They outperform sigmoid-based networks on most tasks.

The Multi-Layer Perceptron (MLP), also called Feed-Forward Network, are called *networks* because they can be represented by the composition of several functions. For example, we might have three functions $f^{(1)}, f^{(2)}, f^{(3)}$ composed so that $f = f^{(3)} \circ f^{(2)} \circ f^{(1)}$. Here, each $f^{(i)}$ is the function of the i^{th} layer. The overall length of this composition chain gives the depth of the model.

An MLP aims at approximating some function f^* e.g. for a classifier, $y = f^*(x)$ maps an input x to a category y . A feed-forward network defines a mapping $y = f(x, \Theta)$ and learns the values of the parameters Θ that result in the best function approximation.

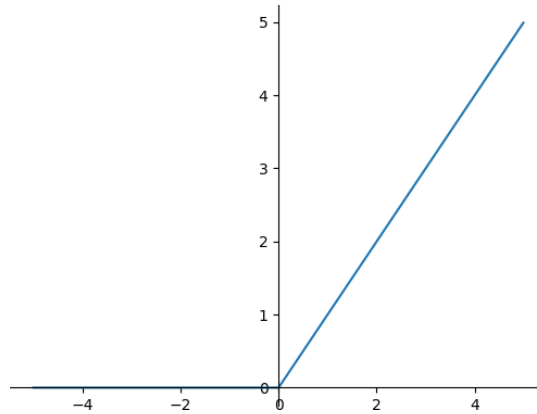


Figure 2.4: ReLU activation

2.1.2 Training

Backpropagation

Once a given model architecture retained (depth, size of each layer, activation functions...), the values of the network's parameters θ (weights and biases) need to be determined by *training* the network. A neural network is trained with the input values and the target patterns providing the network the ability to learn. In most multi-layer neural networks, the training algorithm used is *backpropagation* [148]. The input values x (a vector spanning the space of all input neurons) are propagated forward from the first layer to the last layer through the network. The difference between the obtained output \hat{y} and the expected output y (a vector spanning the space of all output neurons) is measured according to an objective function (or loss function) $L(\theta, x, y)$, providing an error value. The parameters of the neural network are then updated while propagating the error backwards from the last to the first layer to reduce the error value.

Gradient Descent

Typically, neural networks are optimized using *gradient descent* [151]. The objective of this algorithm is to minimize the objective function by updating the parameters in the opposite direction of the gradient of the objective function $\nabla_{\theta}L(\theta)$ averaged over all training samples, so that:

$$\theta \leftarrow \theta - \eta \cdot \nabla_{\theta}L(\theta) \quad (2.6)$$

where η is the learning rate weighting the parameter updates.

To avoid the inefficiency caused by the computation of gradients for redundant training samples, *Stochastic Gradient Descent* (SGD) updates parameters θ for every training sample (x_i, y_i) drawn in a random order:

$$\theta \leftarrow \theta - \eta \cdot \nabla_{\theta}L(\theta, x_i, y_i) \quad (2.7)$$

However, SGD may occasion high undesired fluctuations. A compromise resides in *Mini-Batch Gradient Descent*, which consists in averaging the gradient on random training batches, whose sizes may span from tens to hundreds. The consumption of a number of batches accounting for the entirety of the training dataset is called an *epoch*.

The high dimensionality of the model parameters can lead to very time-consuming highly oscillating gradient descents, or lead to undesirable local optima. Several variants of gradient descent techniques have been proposed to train deep and wide neural networks among which let us mention:

Gradient descent with momentum [141]: updating a gradient with a locally “gentle” slope can take a very long time. Gradient descent with momentum introduces an inertia term, pushing gradient descent in the previous step’s direction, weighted by a parameter ν typically between 0 and 1, in most cases set to 0.9:

$$\begin{aligned}m_t &\leftarrow -\eta \cdot \nabla_{\theta} L(\theta_t) + \nu \cdot m_{t-1} \\ \theta_{t+1} &\leftarrow \theta_t + m_t\end{aligned}\tag{2.8}$$

Nesterov accelerated gradient [128]: a variant where momentum is also employed to evaluate the gradient at the estimated position at the next iteration:

$$\begin{aligned}m_t &\leftarrow -\eta \cdot \nabla_{\theta} L(\theta_t + \nu \cdot m_{t-1}) + \nu \cdot m_{t-1} \\ \theta_{t+1} &\leftarrow \theta_t + m_t\end{aligned}\tag{2.9}$$

Adaptive Gradient algorithm [38] (*AdaGrad*): an SGD variant that maintains a per-parameter learning rate based on the accumulated squares of the gradient components, that improves performance on problems with sparse gradients (e.g. natural language and computer vision problems).

Root Mean Square Propagation [64] (*RMSPprop*): this SGD variant maintains a moving average of the square of gradients to divide the gradients by the root of this average. It is effective on noisy problems.

Adaptive moment estimation [90] (*Adam*): This algorithm calculates an exponential moving average of the gradient and the squared gradient, whose decay rates are controlled by two parameters β_1 and β_2 usually set close to 1 (respectively 0.9 and 0.999) that must be provided in addition to the base learning rate α . Adam is computationally efficient and requires few to no parameter adjustment: the default parameters perform well on most problems.

2.2 Image Processing

Deep learning and convolutional neural networks in particular have played a major role in recent progress in image classification, object detection in images, semantic segmentation, image generation, style transfer, indexation of images, scene understanding and visual question answering. These tasks greatly benefited from the supervised training of thousands to millions of parameters of stacked trainable convolution filters, outperforming hand-crafted methods to produce high-level semantic representations of visual content.

2.2.1 Convolutions

When they take images as inputs, Multi-Layer Perceptrons usually take the value of each pixel as input. Thus, in order to process three-channel images of size $W \times H$ pixels, the input layer should be of size $3 \times W \times H$, which is computationally prohibitive for common image resolutions. In addition, even though fully-connected networks can achieve good performance on simple classification tasks, they fail at taking the image spatiality into account and therefore show poor performance on high level tasks. Inspired from the animal visual cortex, neurons of Convolutional Neural Networks (CNN) are arranged in such a way that they cover small overlapping regions of the image. The visual cortex contains a complex arrangement of cells, where each cell is specialized and sensitive to a specific sub-region of the visual field [74]. By recreating this behaviour, we consider both global and local spatial behaviour.

Figure 2.5 represents the first layer of a CNN, whose input is a 3-channel image. Each neuron takes a 2×2 pixel region of the image as input, i.e. $2 \times 2 \times 3 = 12$ parameters, also referred to as a 2×2 *kernel*. The neuron output is given by $f(\sum_{i=1}^{12} w_i x_i + b)$ with f its activation function and x_i the pixel values. In a convolutional layer, the neurons whose outputs contribute to the same feature map (e.g. the yellow feature map in Figure 2.5) share the same learned weights w_i and bias b . Thus, each feature map is similar to the output of a 2×2 detector run over all the image. A convolutional layer can contain several neurons that produce a multichannel feature map, where every neuron is responsible for a channel. A subsequent convolutional layer can be applied to this feature map, where channels are considered like the colour channels of the input.

Usually, the *depth* of each layer output grows with the depth of the network. If a second convolutional layer were to follow, it would consider each previous feature map as an image colour channel, as illustrated in Figure 2.6. Usually, convolution is followed by some pooling operations.

2.2.2 Pooling

Pooling reduces the width and height of a feature map by applying some sub-sampling operations. Average-pooling applies a possibly overlapping sliding window to each feature map and recomposes a reduced map with the average values of each window. Max-pooling only keeps the highest value of each window. Figure 2.7 provides an example.

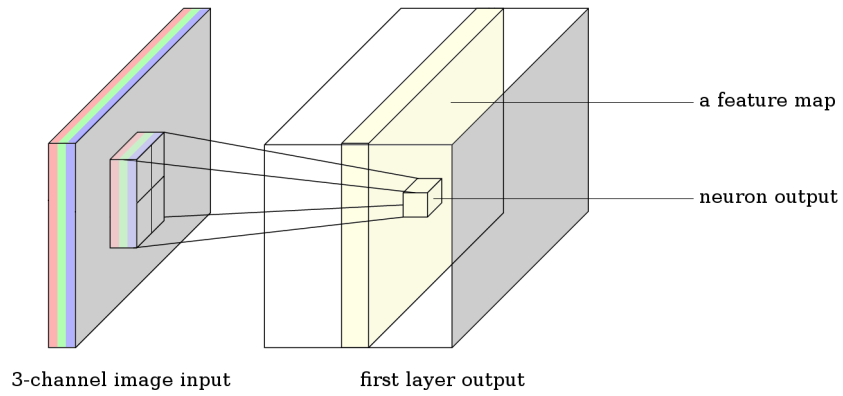


Figure 2.5: First convolutional layer with 2×2 kernel

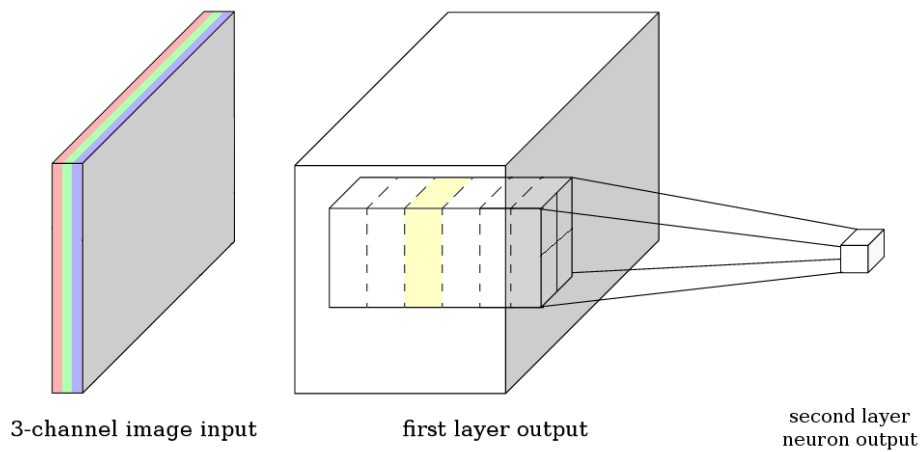


Figure 2.6: A second convolutional layer with 2×2 kernel

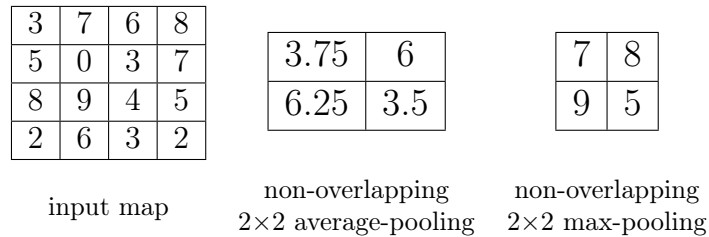


Figure 2.7: Pooling examples

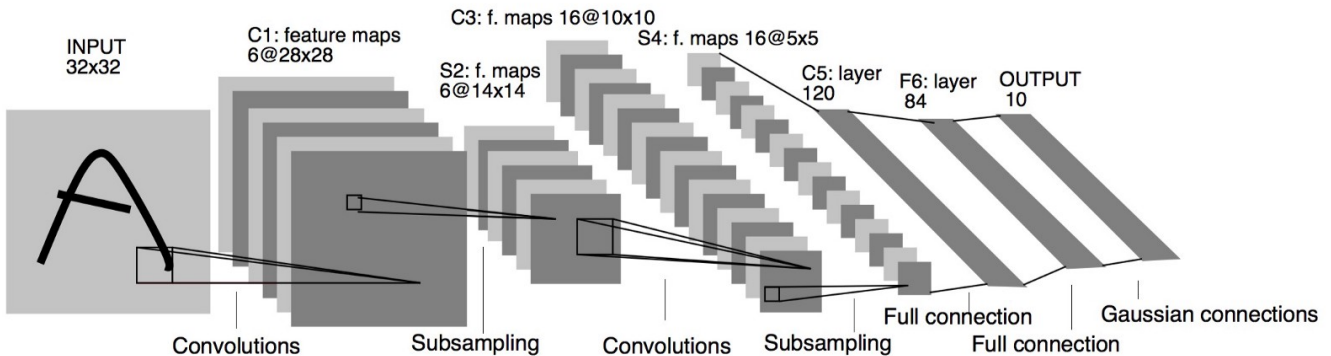


Figure 2.8: LeNet-5 architecture [101]

2.2.3 Notorious CNN Architectures

LeNet-5, AlexNet

In 1994, The convolutional neural network named LeNet-5 displays such an architecture and uses a combination of processing units still exploited today [101]. LeNet-5 is made of 2 convolution layers followed by 2 fully-connected layers. The first two layers are a sequence of convolution, pooling and non-linear activation. The activation used are the aforementioned *tanh* or *sigmoid* functions. Unfortunately, the computational power available in the 1990s was insufficient for CNNs to really propagate.

In a convolution layer, the obtained feature map has more channels than its input. The explosion of the number of parameters is avoided by sub-sampling the feature maps. In LeNet-5, subsampling is made using average pooling.

In 2012, an improvement of LeNet-5 run on two GPUs named AlexNet [96] won the ImageNet challenge [153] by a large margin. The improvements brought to LeNet-5 are the following:

- the inclusion of five convolutional layers,
- the utilization of ReLU activation functions,
- the use of a dropout technique [65] to avoid over-fitting,
- the use of a max pooling in place of average pooling.

The training of the network was rendered possible by the use of two Nvidia GTX 580 GPUs.

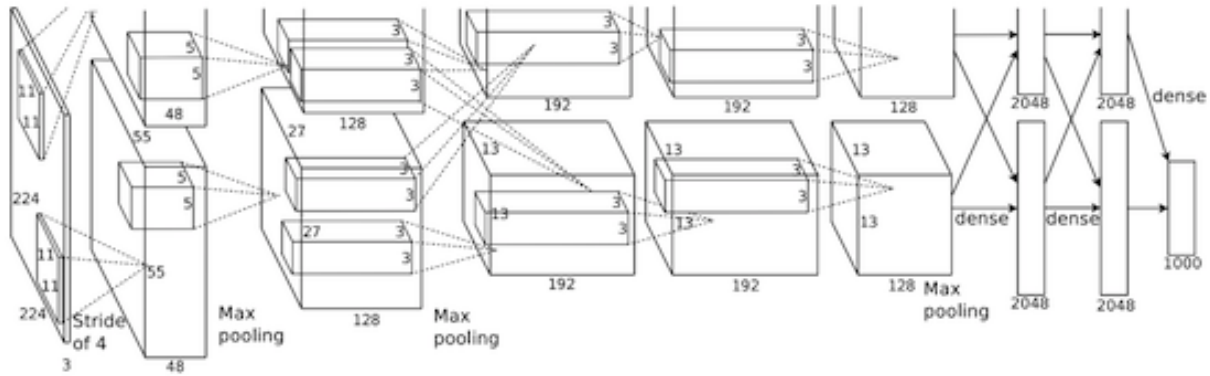


Figure 2.9: AlexNet architecture [96]

AlexNet scored a top-5¹ error rate of 16.4% on the classification task. The second score was 26.2% and was obtained using a mixture of SIFT [113] and SIFT-derived descriptors.

Since this breakthrough, convolutional neural networks have become predominant in image classification. Afterwards, models grew deeper and significant improvements have been proposed.

VGG, Network in Network

AlexNet proposed a first convolutional layer with a 11×11 kernel. In 2014, the VGG network from Oxford [167] successfully proposed several 3×3 filters instead, demonstrating that a succession of small filters can also detect large features. Network-in-network (NiN) went even further by inserting 1×1 convolutions between layers [106] to introduce non-linearities. Basically, 1×1 convolutions map input pixels with all their channels - without looking at their neighbouring pixels - to an output pixel. NiN's 1×1 are set to output feature maps with less channels than the input, *ie.* they perform *depth reduction*, whereas pooling reduces width and height. This mechanism tends to ignore less meaningful features: the output of NiN resembles a sparse version of its input. This design significantly lowers the number of required parameters.

GoogLeNet and the Inception Module

In [172], authors propose a network called *GoogLeNet*. The progressive deepening of neural networks has turned network training into a computational burden. To tackle this issue, GoogLeNet introduces sparsity in the feature maps. This role is played by the so-called *Inception* module (Figure 2.10), which consists in a parallel combination of 3×3 and 5×5 convolutional and 3×3 max-pooling filters. A crucial point is that every filter is followed or preceded by a 1×1 convolutional filter. A standalone 1×1 filter is also added. These NiN reduce the dimensionality of the produced feature maps and alleviate the computational complexity. Because of the dimensionality reduction, this module is also called *bottleneck*. GoogLeNet won the 2014 ImageNet competition.

¹For each test example, five proposals are allowed. Classification is reported wrong if all five tries are wrong.

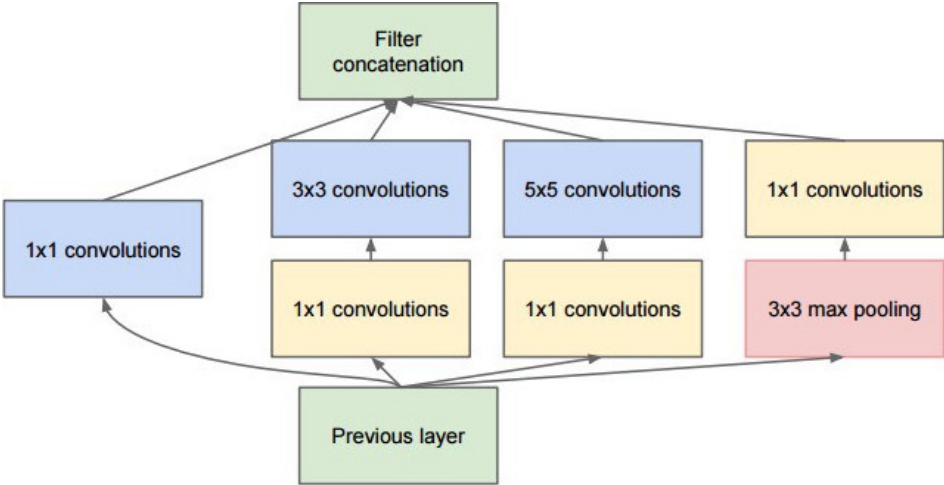


Figure 2.10: the first Inception module [172]

Afterwards, the *Inception* denomination tends to extrapolate to the overall network structure. Versions 2 and 3 of Inception brought batch normalization and a more complex bottleneck, with non-square convolutional filters (e.g. 1×3 kernels). The fourth Inception module [173] is illustrated in Figure 2.11.

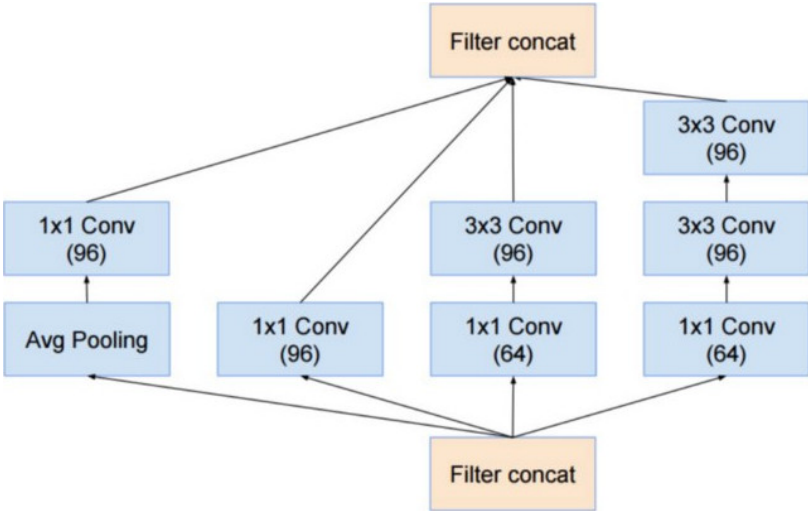


Figure 2.11: Inception v4 [173]

Deep Residual Learning

With great depth does not necessarily come great performance: deepest networks are subjects to over-fitting, meaning that they fail at generalizing classes they are meant to recognize. The responsibility for this undesirable behaviour is related to the vanishing gradient problem: because of backpropagation from the output to the input, the gradient of the loss function is close to zero for layers that are far from the output (ie. close to the input), which makes their training impossible. Deep residual learning and the so-called ResNet module [61] solves this problem by bypassing

two consecutive layers with their input before the second activation (Figure 2.12). The motivation behind this design is to virtually bring the first layer closer to the output with shortcuts, allowing the gradient to flow back deeper without vanishing. This strategy opened the way for networks of hundreds of layers, up to a thousand layers. The network submitted to ImageNet was composed of 152 layers.

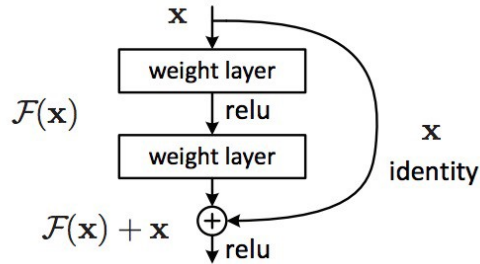


Figure 2.12: residual learning in ResNet

Eventually, the Inception and ResNet modules converged into Inception-ResNet. Authors claim that residual connections accelerate the training of their network. An ensemble of Inception-ResNet and Inception v4 networks [173] achieved 3.08 % percent top-5 error on the ImageNet Classification Challenge [153], which represents state of the art in this task.

Compact Networks

Although they pushed forward the boundaries of automatic image classification, the cost in depth and memory made very large CNNs unsuitable for small devices. For instance, network ResNet-152 dedicated to ImageNet classification contains 60.2 million parameters. The problem now resides in optimizing networks without losing performance. Xception [30] and ShaResNet [18] are attempts to slim very deep networks, respectively by using depth-wise separable convolutions and sharing the weights of convolutional layers between residual blocks. Depth-wise separable convolutions are a combination of depth-wise and point-wise convolutions that can be approximated to a factorisation of regular convolutions that drastically reduce the number of operations, thus the computational overhead, with almost no performance loss. They are notoriously used in MobileNet [71] and MobileNetv2 [155] architectures, intended for nomadic or in-vehicle use.

2.2.4 CNN feature maps as descriptors

The recent breakthroughs in image classification brought by convolutional neural networks opened the way for more solutions relying on CNNs. One question of particular interest in various computer visions/indexing and retrieval related tasks is the following : how can CNNs describe images? In a trained CNN, every neuron of every layer takes all sub-regions of the image (or of the previous layer) as inputs by applying a sliding window over it. Because the

neuron map is the same for every region of the image (or previous layer), the neuron acts like a space-invariant feature detector that preserves spatial information. Moreover, if the networks are trained for classification purposes, it is reasonable to assume that the parameters are such that the obtained feature maps are meaningful in regard to the classes the network has been trained on. For example, if a network classifies whether the image represents a cat or a dog, the deepest neurons are most likely to be detectors of cat-like or dog-like features. Hence, if it was trained on a large dataset encompassing a high number of categories, the network produces a great variety of feature maps, which can be used as image descriptors. For example, in [136], the fifth layer of network VGG19 is fed to an SVM for human action recognition. It is common practice to plug the backbone of a CNN trained on a large dataset like ImageNet to a “classification head” of one to several layers designed to recognize novel classes, under a transfer learning paradigm. Beyond classification, pre-trained CNN feature maps can be used as input for other tasks such as object detection, as described in the following section.

2.2.5 Object detection

The previous approaches are used for classification of overall images. Even though image classification is the figurehead of convolutional neural networks, they allowed to push computer vision forward in other domains, and notably they show their pertinence to object detection tasks. In many datasets and classification challenges, images are well-suited for direct classification: the object of interest is in the centre of the image, with few clutter in the background. However, in real-life applications, we have to deal with images that are not necessarily cropped around our object(s) of interest, with possibly clutter and obstruction. In this case, it becomes important to localize the object, by determining its regions of interest, prior to any classification task.

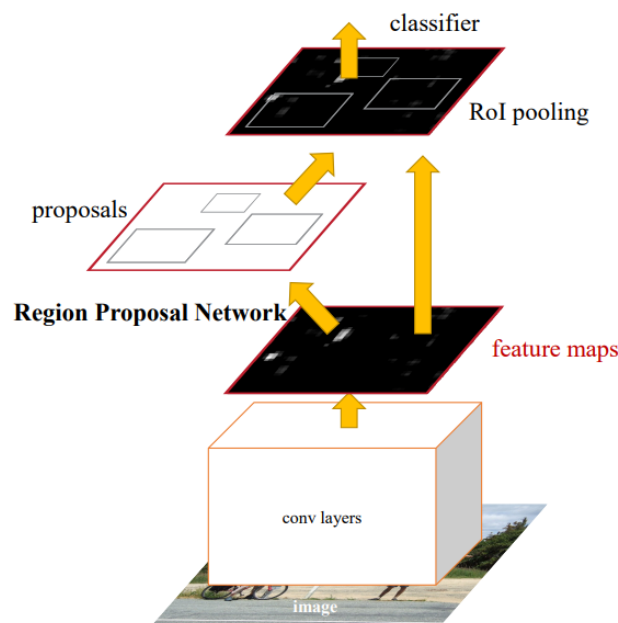


Figure 2.13: Faster R-CNN architecture [145]

Region-CNN (R-CNN) [50] is one of the first approaches designed with this intent. The network outputs Regions of Interest (RoI) supposed to contain objects, then proposes a classification for each of them. In this first version, the proposed regions are obtained with a method unrelated to CNNs, so-called Selective Search [180]. This method generates regions by associating pixels according to texture, colour and intensity at various scales. The regions most likely to be objects are fed into a customized AlexNet for classification, and finally into a Support Vector Machine (SVM) for deriving an *objectness score*, expressing the probability of the presence of an object in the determined region. If the region is confirmed as containing an object, its bounding box coordinates are refined and displayed, as well as its estimated class. Although it has shown good results, R-CNN is slow to train because it needs both AlexNet and SVM training, and slow to run because each region proposal needs a pass through AlexNet and SVM. In addition, the region proposal is still made using Selective Search, which is unrelated to the neural network and remains the bottleneck of the process.

The *Fast R-CNN* approach [49] introduces the Region of Interest Pooling (RoI-pooling) to speed up the process. The input image is fed only once to the AlexNet, and the features of each proposed RoI are obtained by cropping the resulting feature maps according to the RoI coordinates. Then, these cropped feature maps are fed to a softmax classifier and bounding box regressor, just as in R-CNN.

Faster R-CNN [145] carries out all steps of the process, including region proposal. The image is fed once to the network prior to any region proposal. Solely cropped feature maps are fed into the Region Proposal subnetwork and are given an *objectness* score.

Mask R-CNN [62] extends the Faster R-CNN workflow by targeting a pixel-level object segmentation. To achieve this goal, the last layer of the network is trained to adjust a binary map specifying where the detected object is supposed to be.

As suggested by its name, *YOLO* [144] (You Only Look Once) eludes the region proposal network. It requires only a single forward propagation to carry out object detection. In a single pass, from feature maps of various resolutions, YOLO models predicts a class label and bounding box coordinates (centre x and y, width, height). The single pass detection improves considerably the inference time. YOLO evolution YOLOv3 [143] is four times faster than its RetinaNet competitor [108] for equivalent performances.

Let us now analyze how the deep learning techniques have impacted the field of natural language processing.

2.3 Natural Language Processing

Natural Language Processing (NLP) is a discipline that focuses on the understanding, handling and generation of natural language by machines. NLP addresses various challenges such as automatic translation, sentiment analysis, named entity recognition, text classification (*e.g.* spam detection), conversational agents (chatbots), automatic text summarization, and many others. In this work, we mainly focus on text classification.

NLP undoubtedly is a bridge between computer science and linguistics, and because natural language is a high-level, complex representation of human thoughts and communications, NLP methods and algorithms must acknowledge such particularities. Unlike in images, there is no straightforward *pixel* language to extract meaning from characters, words, punctuation, grammatical groups and sentences are all semantic entities which must be taken into account. Somehow, language must be first turned into an exploitable data format. Characters, character n-grams, words, words n-grams sentences, *term frequency-inverse document frequency* (tf-idf) are all suitable input modalities when it comes to process natural language. In the vast majority, NLP neural networks use word embeddings as input.

2.3.1 Word embeddings

Word embeddings are an ensemble of learning methods aiming at representing words by fixed-length vectors. Such vectors are intended to project the input vocabulary onto a vector space that preserves the semantic similarity between words of the target vocabulary. To serve as inputs for neural networks, un-contextualized embeddings are concatenated in an embedding matrix W whose number of rows matches the vocabulary size. Like convolution layers in the trunks of vision-dedicated networks, embedding matrices can be randomly initialized and trained in the same time as the downstream layers. Embedding matrices can also be pre-trained on a given task and reused - frozen or trainable - to learn the down-stream task in a transfer learning fashion. Word2vec embeddings [123] are trained on a “skip-gram” pretext task that consists in predicting nearby words in source sentences. As illustrated in Figure 2.14, Word2vec is able to automatically organize concepts and learn implicitly the relationships between them, even though this information was never given at the training time.

Typically, one would allocate embedding rows for out-of-vocabulary words: unknown words are hashed and randomly assigned to one of the available out-of-vocabulary vectors, with the risk of assigning different words to the same embedding. FastText embeddings [77] overcome this limitation by computing a word representation from the character n-grams that words are made of. In this way, FastText ingeniously accounts for prefixes, roots and suffixes shared across words that link them together.

In the image domain, multi-layer CNN trunks pre-trained on huge labelled datasets (*e.g.* ImageNet) are vastly re-used to learn down-stream tasks, plugged into relatively shallow classification layers to try and recognize novel classes. Acknowledging that these pre-trained convolutional networks produce context-aware representations of image

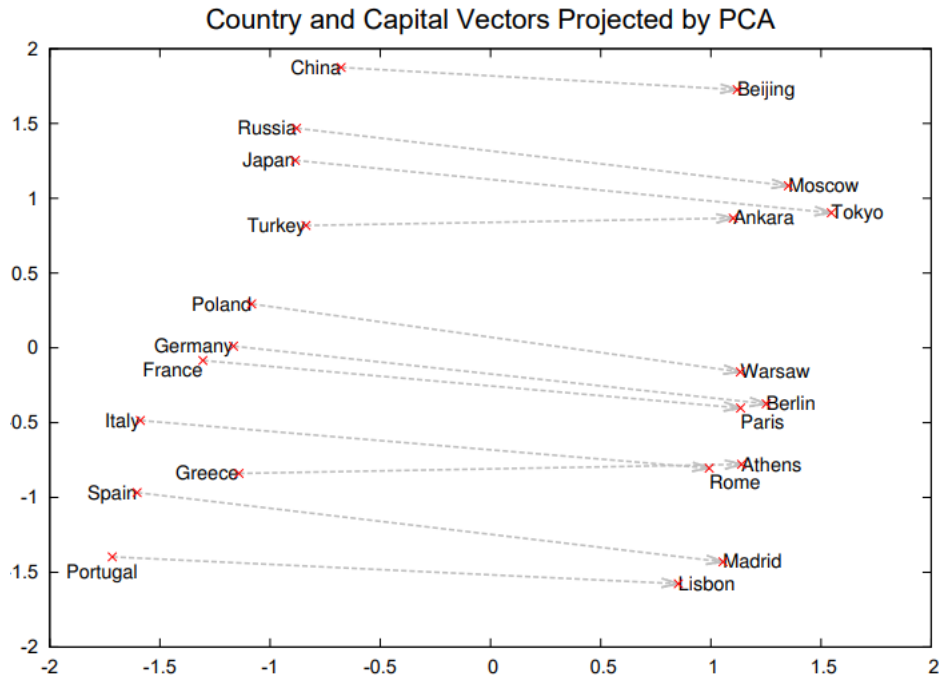


Figure 2.14: Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors (Word2vec) of countries and their capital cities. [124]

regions, the ELMo (Embeddings from Language Models) [140] approach proposes the pre-trained combination of a classical word embedding layer and stacked recurrent layers, whose assembly constitutes a “contextualized word embedding”, producing word representations accounting for the context surrounding words. Being context-aware, ELMo’s strength resides in its disambiguation capacity as illustrated in Table 2.1, where ELMo appears to overcome the polysemy around the noun “play”, given its context.

Source word in context	Nearest neighbour in context
Chico Ruiz made a spectacular play on Alusik’s grounder [...]	Kieffer, the only junior in the group, was commended for his ability to hit in the clutch, as well as his all-round excellent play .
Olivia De Havilland signed to do a Broadway play for Garson [...]	[...] they were actors who had been handed fat roles in a successful play , and had talent enough to fill the roles competently, with nice understatement.

Table 2.1: Nearest neighbour to the word “play” using ELMo contextualized embeddings [140].

BERT (Bidirectional Encoder Representations from Transformers) [37], *cf.* section 2.3.2) goes further by training a transformer-based architecture [183] to predict in a self-supervised manner, given an input sentence A and a sentence B that is likely to follow A, the masked words in A. BERT contextualized embeddings excel at solving various NLP problems including question answering, abstract summarization, sentence prediction, conversational response generation and sentiment classification.

2.3.2 Neural networks for text classification

The task of automatic text classification consists of automatically assigning a document to one or more membership classes, whether it is a question of sorting texts according to the topic they are addressing [192], detect toxic comments for moderation [82], sort out spam [117] or route support tickets to the correct person [133].

Convolutional neural networks (CNN) Figure 2.15 and recurrent neural networks (RNN) Figure 2.16 are two types of neural network architectures widely explored to handle NLP text classification tasks.

1-dimensional CNN

1-d convolutional layers of size k kernel (or filter width k) are used for representation learning from sliding k -grams. As illustrated in Figure 2.15, an input word sequence $x = \{x_0, \dots, x_n\}$ is converted into a word embedding sequence $e = \{e_0, \dots, e_n\} \in \mathbb{R}^{n \times d_e}$ of d_e -dimensional word embeddings. The sequence e is then fed to a convolutional layer parameterized by its output dimension d and kernel size k . The successive k -grams of e are concatenated and processed by a weight matrix $W_0 \in \mathbb{R}^{d \times kd}$ and bias vector $b_0 \in \mathbb{R}^d$ to produce a sequence of feature vectors $f_0 \in \mathbb{R}^{n \times d}$:

$$f_{0,i} = \sigma \left(W_0 \cdot \left(e_{i-\lfloor \frac{k-1}{2} \rfloor}, \dots, e_{i+\lceil \frac{k-1}{2} \rceil} \right) + b_0 \right), \quad (2.10)$$

where σ is an activation function (typically ReLU or tanh) and $\left(e_{i-\lfloor \frac{k-1}{2} \rfloor}, \dots, e_{i+\lceil \frac{k-1}{2} \rceil} \right)$ is the concatenated k -gram centred around e_i .

The sequence e is left and right zero-padded to ensure that there are as many feature vectors as word embeddings. Figure 2.15 displays a stack of two convolution layers with kernel sizes 3 and 5. The output of the last convolution layer can undergo global average or max pooling to obtain a global feature vector f_{global} that can be fed to a final classification layer.

Due to their principle of operation, 1-d CNN are well-designed to extract the most informative n-grams and close relationships between words. However, they are unable to detect long, cross-sentences relationships because of their narrow receptive field [188].

Recurrent neural networks

RNNs are neural networks designed to handle time series by recursively processing the concatenation of the current step and the last step output. An input word sequence $x = \{x_0, \dots, x_n\}$ is converted to a word embedding sequence $e = \{e_0, \dots, e_n\} \in \mathbb{R}^{n \times d_e}$ of d_e -dimensional word embeddings. In the most basic RNN architectures, every time step e_i of e is concatenated to the output of the previous step $f_{0,i-1}$ to be processed by a weight matrix $W_0 \in \mathbb{R}^{d \times dd_e}$ and bias vector $b_0 \in \mathbb{R}^d$ to produce a feature vector $f_{0,i} \in \mathbb{R}^d$:

$$f_{0,i} = \sigma \left(W_0 \cdot (f_{0,i-1}, e_i) + b_0 \right), \quad (2.11)$$

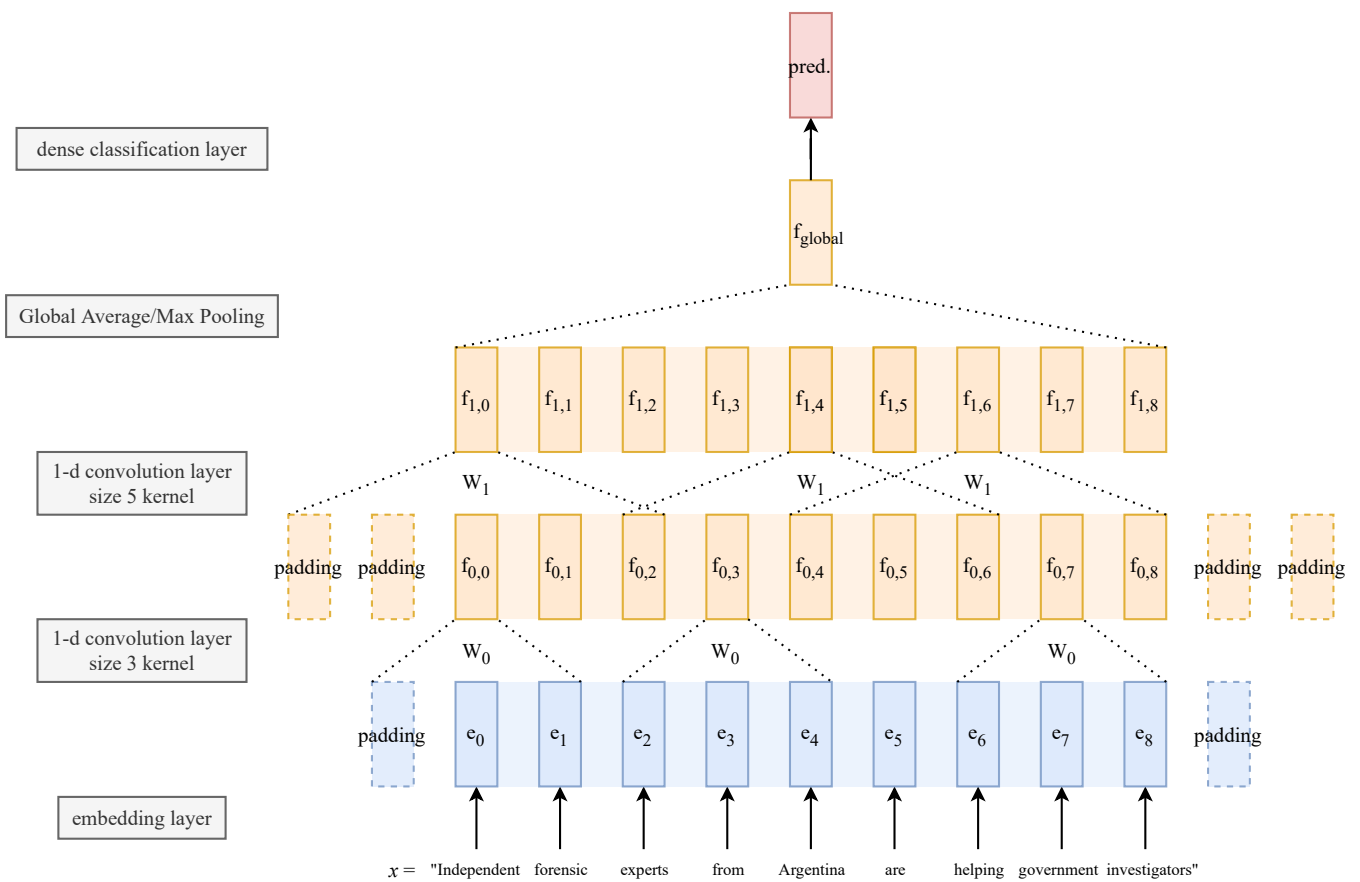


Figure 2.15: 1-dimensional convolutional neural network for text classification

where σ is an activation function. Recurrent layers can be stacked as depicted in Figure 2.16. For classification, a global feature vector summarizing the whole sequence is fed to a final classification layer. This global vector can be the last output of the last recurrent layer (as in Figure 2.16) or the global average/max pooling of every feature out of the last layer (as in CNNs, Figure 2.15).

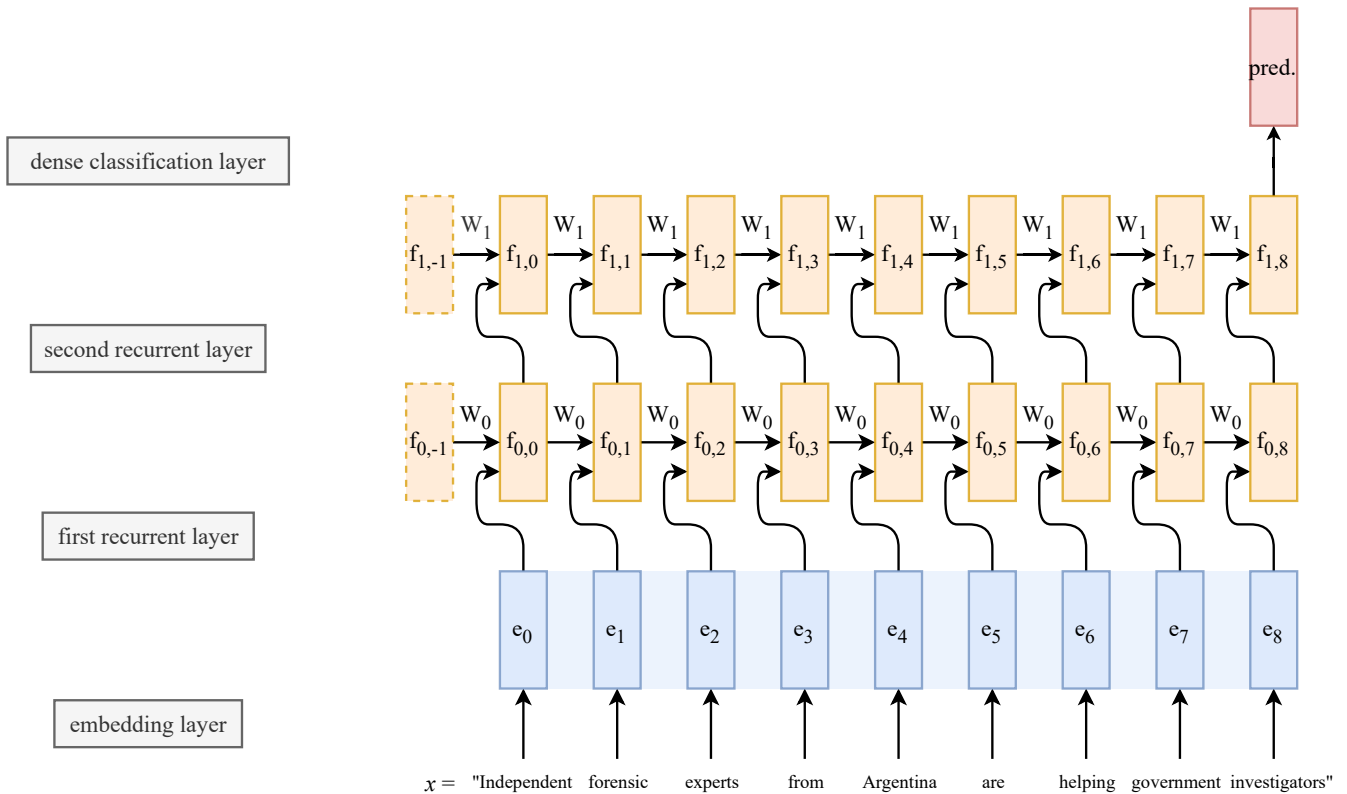


Figure 2.16: 2-layer recurrent neural network for text classification

Long Short-Term Memory (LSTM, [67]) and Gated Recurrent Unit (GRU, [27]) are upgrades in RNN architecture. LSTM introduces a cell, an input gate, an output gate and a forget gate. The cell memory state is transmitted and updated across time steps and allows detecting long term dependencies across words and sentences. Gates dictate whether memory states can be updated or forgotten. With their memory cell, LSMTs and GRUs are well suited to process long texts (e.g. document-level sentiment classification [175]). However, the causal dependency between steps makes parallelization harder than for CNNs. Also, because of the recurrent connections, an unfolded RNN ends up like a very deep network, harder to train than a CNN.

There is a lack of consensus regarding CNN/RNN selection for NLP. A CNN/LSTM/GRU comparison [197] supports the findings that CNNs and RNNs provide complementary information for text classification tasks: CNNs are better at detecting key phrases, while RNNs are more well-suited for semantically understanding of the whole sequence.

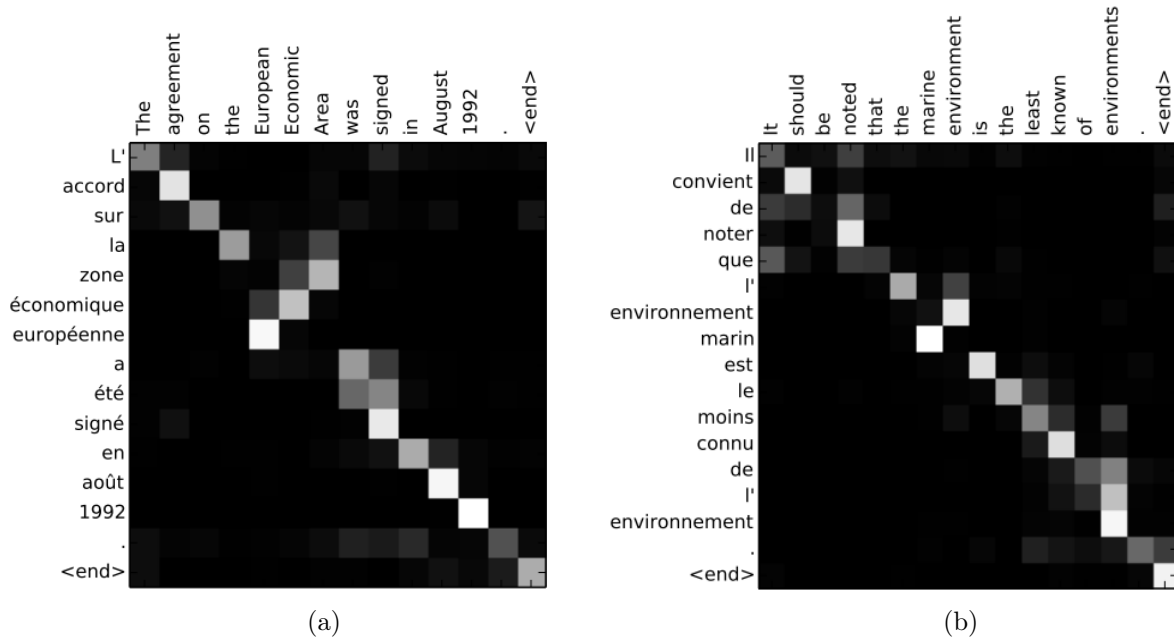


Figure 2.17: Alignments discovered by the attention mechanism of a sequence-to-sequence translation network [14]. Attention scores acknowledge correspondences between words of both languages and highlight the reverse order of adjectives in noun phrases.

Attention and self-attention

In Neural Machine Translation, sequence-to-sequence neural networks are trained to translate a text from a language to another. In the pioneering work entitled “Sequence to Sequence Learning with Neural Networks” [171], an encoder RNN produces a context vector (defined as its last hidden state) which is fed to a decoder RNN that iteratively outputs the translation’s words. If the encoder/decoder paradigm is still broadly used in a wide variety of applications, the use of a single context vector turned out to be a bottleneck for translation. Attention [14, 115] is a technique which highly improved the quality of machine translation by exploiting the encoder’s hidden states for all time steps and by allowing the decoder to focus on the relevant state at each decoding step. Typically, an attention mechanism is based on the outputs of one or several neural layers attributing a score to each encoder hidden state, corresponding to an input token. Hence, in addition to translation, an attention mechanism produces an alignment that can be exploited to explain, confirm or infirm the correspondences that the model discovered between the source and target languages, as illustrated in Figure 2.17.

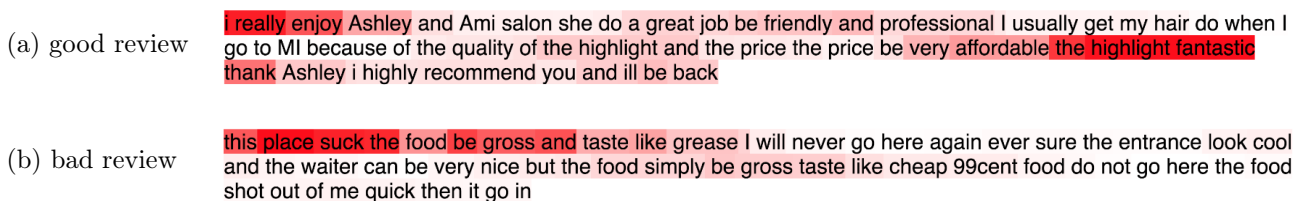


Figure 2.18: Self-attention scores on texts classified as a *good* or *bad* reviews. Red values accounts for high scores. The attention mechanism focuses on words responsible for class membership according to the trained model.

An attention mechanism can be applied directly by a decoder-less architecture by looking at its own hidden states. This mechanism is called *self-attention*. Self-attention has been used for text classification and produces helpful indications on the importance of words for the target task besides bringing interesting performance gains, as reported in [109] Figure 2.18. Here, self-attention is used as a replacement of Global Average Pooling (GAP) or Global Max Pooling (GMP) (as featured in Figure 2.15) to aggregate temporal hidden features into a single feature vector. In this case, the self-attention mechanism acts like a “global weighted average pooling” with attention scores as importance weights. It is worth to mention that several attention or self-attention mechanisms (also called attention *heads*) can be associated and trained in parallel to produce various mixtures of attention scores in what is called *multi-head* attention. As a result, several heads can learn to focus on different kind of words.

Self-attention can also be applied at every time step, so that every temporal hidden state – corresponding to an input word – produces its own self-attended output. This mechanism constitutes a kind of neural layer that is distinct from convolution and recurrence, two mechanisms that account for direct neighbours and previous words respectively. Instead, given a word’s embedding or hidden state, attention heads are trained to focus on what other words – neighbouring or not – it is meaningful to associate the input word with. This is the core mechanism on which the Transformer architecture [183] is based. Transformers include stacks of several multi-head self-attention layers that propagate token information in place of convolutional or recurrent layers. As a typical example let us cite BERT (Bidirectional Encoder Representations from Transformers [37]), which is a wide-spread Transformer-based contextual word embedding.

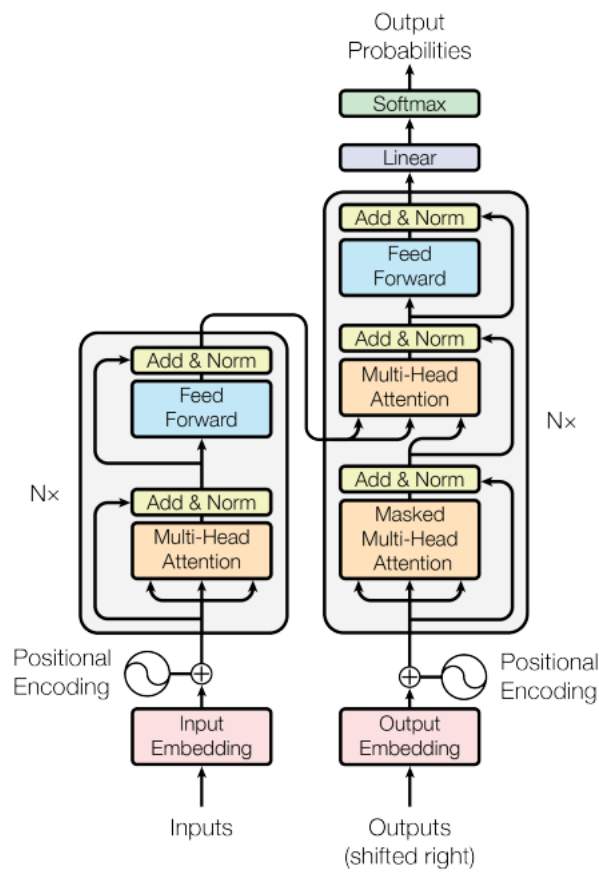


Figure 2.19: The Transformer architecture [183]

2.4 Multi-modal Data Processing

The multi-modal learning domain aggregates machine learning models or applications involving multiple modalities. A modality can be seen as a domain of data like an image, text or sound, or as a representation of such diverse data types, like local image descriptors, a Bag-of-Words, any sound transform, etc. In the frame of multi-modal learning, there are many configurations in which modalities can be found. It is possible to share a model's input or output between modalities, or to have a modality as input and another as output. Multiple modalities can be learned jointly, while one modality can assist in the learning of another. It finds applications in real life scenarios such as multimedia content retrieval, product recommendation systems, robotics... Multi-modal learning is challenging as models must handle different data representations (real against discrete values, dense vs. sparse values). Also, the combination of modalities can be an additional source of noise [3]. This section tends to focus on use cases where several modalities are used as input. Modalities of a document tend to bring complementary and correlated information, two characteristics we want to exploit in multi-modal learning. We also focus on methods based on deep learning networks, that have recently produced state-of-the-art results on both mono-modal and multi-modal tasks.

In their survey [15], Baltrušaitis et al. identify five core technical *challenges* surrounding multi-modal learning: representation, translation, alignment, fusion and co-learning.

2.4.1 Representation

Given a set of documents, it may be needed to craft a vector representation that manages to embed its specificities and preserve the similarities and dissimilarities that can be found within the documents' domain. Such features are used in content retrieval [9], as input of other models for various tasks [124] and can be used in active learning scenarios to compute an information density measure. In the field of multi-modal learning, joint representation is achieved by learning how to represent and summarize multi-modal documents as single entities while preserving some inter-documents similarity. At the opposite, coordinated representation is obtained by learning how to represent and summarize documents of heterogeneous modalities while preserving some inter-modality similarity (Figure 2.20)

Multi-modal autoencoders [130] are used to produce joint representations (or *multi-modal embeddings*) of multi-modal documents. This symmetrical neural network is trained to recreate its input by using one to several joint hidden layers. Its central hidden layer produces the desired embedding. Sporadically removing a modality from the input can be done so that the network can represent both modalities from only one (Figures 2.21a, 2.21b). In [189], authors propose a bidirectional cross-modal neural network with tied weights that learns simultaneously to reconstruct a first modality from the second and conversely, showing good results on cross-modal applications (Figure 2.21c).

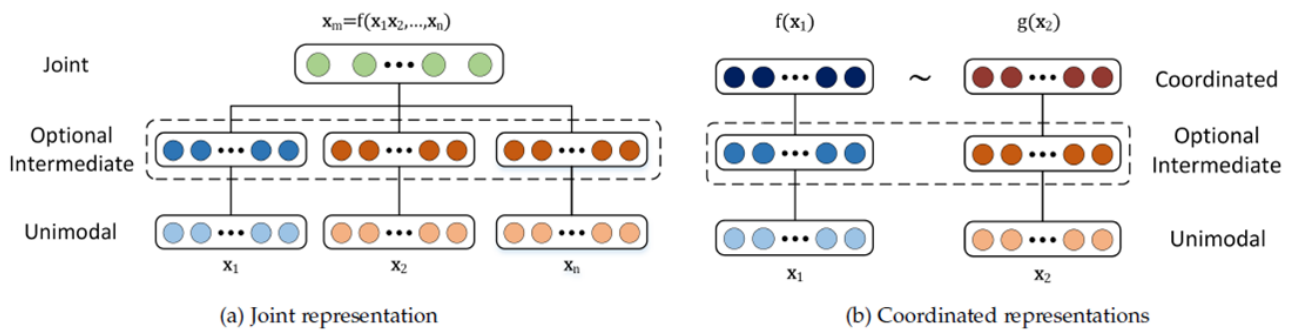


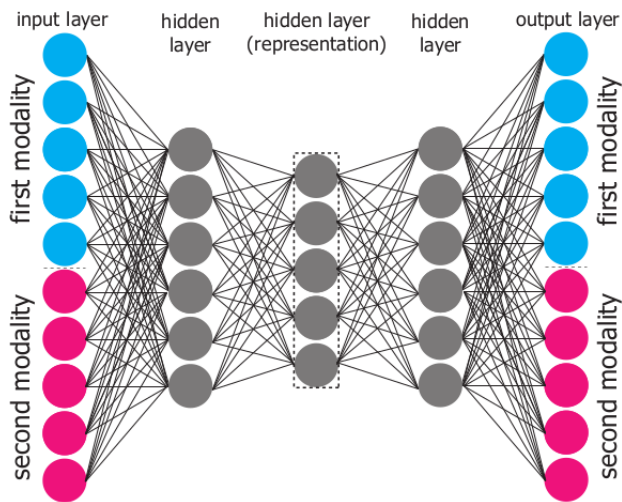
Figure 2.20: Joint representation versus coordinated representation [15]. A joint representation aggregates information from several modalities into a single representation, whereas a coordinated representation projects uni-modal information to a common space where representations can be compared regardless of their modality of origin.

2.4.2 Translation

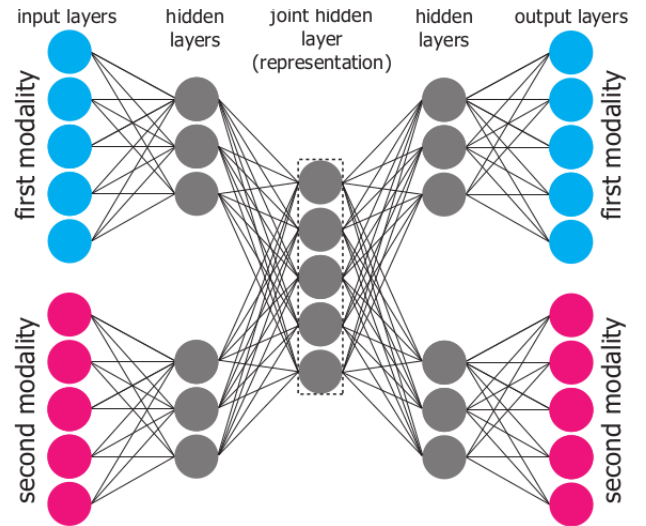
Translation is learning how to translate data from a modality to another by establishing a semantic correspondence between modalities. In the deep learning field, an intuitive way of implementing inter-modality translation is to establish a connection between the coordinated representations of modalities. In [120, 83], authors propose a deep image captioning model that carries out translation from the image modality to the text modality by preserving the semantics contained in each data domain. The approach reported in [127] proposes to carry out linguistic translation by first translating textual data to a hidden multimedia pivot obtained from the attached images, supposed to hold the same semantics as the text input. On top of this first step, a decoder module carries out a second translation towards the target language. Based on skip-thought vectors [92], Kiros et al. go further and propose to generate longer texts (short stories) with only images as input [91].

2.4.3 Alignment

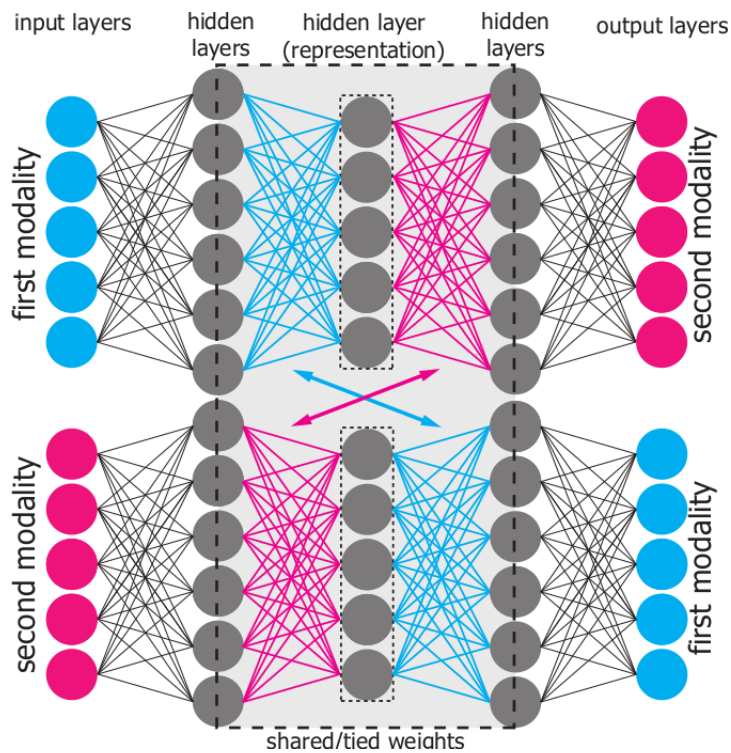
Alignment is learning how to establish a correspondence between sub-elements of two different modalities. For example, [83] proposes to match sentence elements from image captions to their corresponding image region with an alignment based on a combination of CNNs over image regions, bidirectional RNNs over sentences, and a structured objective that aligns the two modalities through a multi-modal embedding. In [176], authors aim at locating the speaker in a video stream by relying on both visual and audio features. In these two examples, alignment is the finality, but it can also be used as a way to improve performances in other tasks. For instance, [29] proposes to apply an attention mechanism in uni-modal or cross-modal encoder-decoders. Attention mechanisms are particularly suited for alignment in deep neural networks, as they can focus on meaningful segments of both text and image modalities when needed [193]. Indeed, when iterating over a time series, attention mechanisms can focus over time on various words or image regions. In [72, 73], a Visual-Textual Attention Model is used to automatically find the relation between words and image regions to produce a joint representation of multi-modal text-image pairs that captures



(a) concatenated representations at input and output, all hidden layers are joint



(b) separated inputs, outputs and hidden layers, a single hidden layer in common



(c) Bidirectional neural network with tied weights

Figure 2.21: Multi-modal autoencoder architectures [189]

the fine granularity correlation between image and text. It can be used for classification or tag suggestion. Given a pair of an image (divided into D regions represented by features of size M $R_i = \{r_{i,1}, \dots, r_{i,j}, \dots, r_{i,D}\} \in \mathbb{R}^{D \times M}$) and its textual description (made of L tokens represented by embeddings of size E $T_i = \{t_i^1, \dots, t_i^k, \dots, t_i^L\} \in \mathbb{R}^{L \times E}$), normalized attention between token t_i^k and image region $r_{i,j}$ is given by

$$\alpha_{i,j}^k = \text{softmax} [\varphi ((t_i^k)^T W r_{i,j} + b)] \quad (2.12)$$

where W and b are the weights and bias of the attention mechanism and $\varphi(\cdot)$ is \tanh activation.

For each token, the joint visual-textual feature c_i^k is obtained by concatenating the attended visual and textual features:

$$c_i^k = [u_i^k; t_i^k], C_i \in \mathbb{R}^{L \times (M \times E)} \quad (2.13)$$

Joint textual features C_i^k are meant to capture the alignment between the k^{th} token and the images regions. It is still a time series that can be fed to an RNN (Figure 2.22).

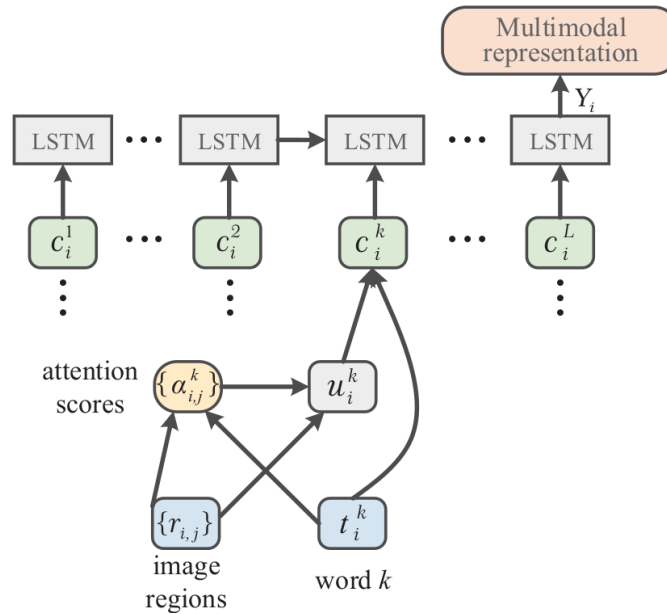


Figure 2.22: Architecture of the visual-semantic attention network

2.4.4 Fusion

Fusion is learning how to join information from multiple modalities in order to perform another task, such as classification or joint representation. Fusion methods are usually discriminated according to the level at which the fusion is done, typically *early* fusion against *late* fusion. Snoek et al. [169] come with the following definitions, illustrated in Figure 2.23:

- early fusion is a fusion scheme that integrates uni-modal features before learning concepts;

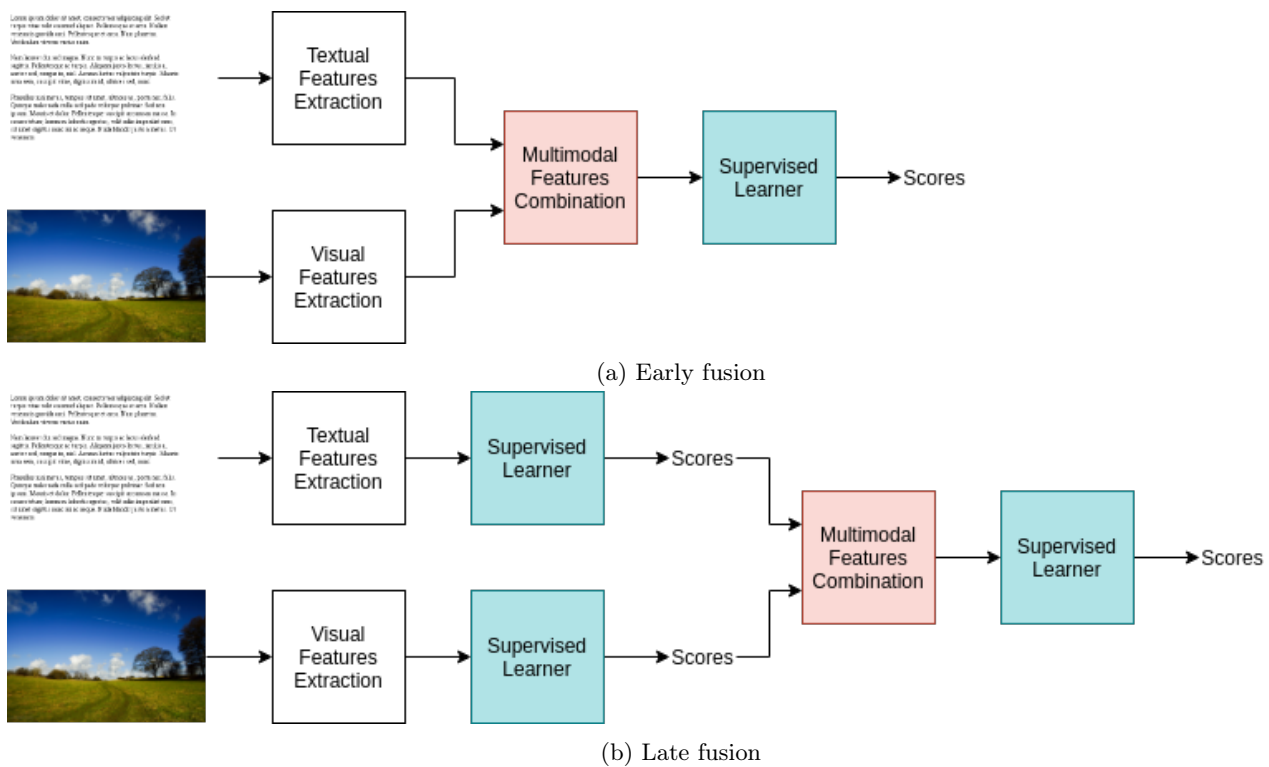


Figure 2.23: Early versus late multi-modal fusion with neural networks

- late fusion is a fusion scheme that first reduces uni-modal features to separately learned concept scores, whereupon these scores are integrated to learn concepts.

The preference of a method over another is task specific. Deep learning blurs the aforementioned definition of early and late fusion because it allows to merge modality features at different depths: the deeper the merging of features, the later the fusion. A multi-modal deep neural network can be trained in a single end-to-end fashion without separating uni-modal and multi-modal trainings. For neural nets, early and late fusion can be redefined as close to the data versus close to the decision. If modalities are somehow pre-aligned, it is possible to acknowledge this alignment when merging modalities: correlated video and audio streams are pre-aligned time series and can be merged according to their samples' timestamps; an image and its transform keep some spatial alignment that can be exploited during fusion: *via* compact bilinear pooling [47, 178] can be used to fuse an image and its associated noise stream [204].

Modalities are often not explicitly aligned, *i.e.* some text and its associated image. In such scenarios, deep neural networks merge modalities in a way that is agnostic to alignment:

- **Weighted average** of features or scores [120] with learnt weights;
- **Concatenation** of features or scores [72, 73];
- **ModDrop** [129]: a gradual fusion at several spatial and temporal scales involving random dropping of separate

channels;

- **Gated Multimodal Unit (GMU)** [10]: a module with gate neurons employed to decide whether each modality should participate or not to the internal encoding of the input sample based on the input content itself;
- **CentralNet** [186]: as its name suggests, implements a *central* neural net in between uni-modal neural networks, each hidden state of the central network being fed by a weighted sum of the previous activation of the central network and the outputs of the hidden states of the uni-modal neural nets. This architecture is displayed in Figure 2.24.

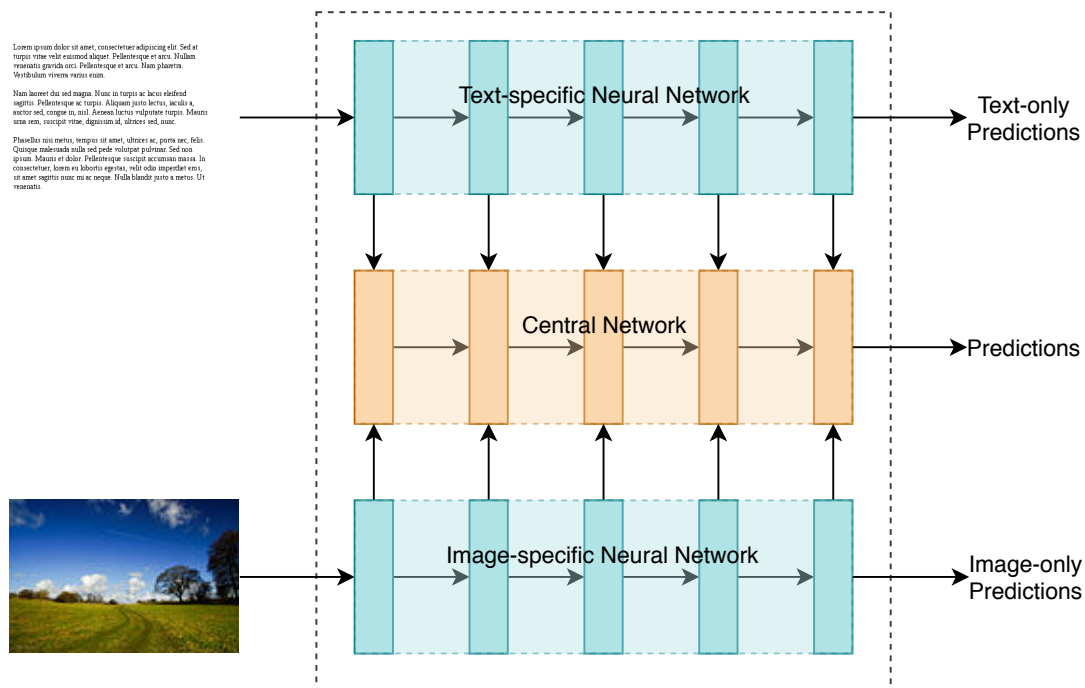


Figure 2.24: CentralNet [186]

2.4.5 Co-learning

Also known as knowledge transfer, co-learning exploits the correlation between two modalities by transferring an already trained predictive model and/or representation to the other modality. For instance, a multi-modal representation can be learnt by applying knowledge transfer from vision to sound [13]. With only the raw wave forms of the sound of unlabelled videos as input, a CNN is trained to reproduce the deep features of a teacher CNN processing the corresponding video channels. Then, the obtained sound representations are fed to an SVM for sound classification, with state-of-the-art results.

2.5 Conclusion

Deep Learning aggregates an abundant amount of techniques and practices whose effectiveness has been proven on a wide range of modalities and applications. Perceptron and gradient descent have come a long way to the high-level abstractions neural networks provide nowadays: language models and deep visual encoders produce powerful representations of their respective modalities, which can be fine-tuned and plugged into a wide variety of applications, granted the training data is provided in sufficient quantity and quality. The universal approximation theorem [69] states that feed-forward architectures, as functions approximators, can represent any continuous function between two Euclidean spaces as long as they are wide enough, deep enough, and the right set of parameters has been found. The network's size (width and depth) is subject to obvious memory limitations. But most importantly, finding the right set of parameters to approximate a high-level semantic function heavily relies on providing training data that actually holds the required meaningful knowledge, and that the chosen training approach converts that knowledge into the right set of parameters. The next chapters of this work will focus on overcoming the constraints in terms of labelled data and their inadequacy with the objective to train neural networks to acquire knowledge from open-source data. The following chapter presents active learning, a sub-field of machine learning in adequacy with this purpose.

Chapter 3

Deep Active Learning

Abstract

Active learning, sometimes called *online* learning or *query* learning, is a subfield of machine learning designed to address the scarcity of ground-truth values in supervised training applications. By carefully selecting or synthesizing unlabelled examples to receive ground-truth labels from, active learning methods aim at improving the training process. The rules that dictate what particular data points have to be chosen are designed as the *sampling strategy*. A common, straightforward strategy consists in sampling the most uncertain data point in between every training. However, related work has shown that certain applications, notably in the field of computer vision, can benefit from more intricate strategies. This chapter introduces the main active learning scenarios and presents the most effective sampling strategies exploited in the state of the art.

3.1 Active learning

Active learning [31], sometimes called *online* learning or *query* learning is a sub-field of machine learning designed to address the scarcity of ground-truth values in supervised training applications, whether it is caused by the lack of reliable labels in a pool of documents, or by the progressive exploitation of a stream of data. Many machine learning techniques have shown to be very effective in the field of automatic text or image classification. However, they often rely on large labelled datasets. Obtaining reliable ground-truth data in real-life applications may be considerably expensive and time-consuming. To tackle this issue, the key principle of active learning resides in estimating what examples are the most informative for training the considered machine learning model, before revealing the label of the sampled example. A human labeller – so-called *oracle* – participates in the learning process and enlightens the model by providing the correct ground-truth labels for the chosen examples. Let us mention that the human oracle may be replaced by an automated service. Since human labelling is generally expensive, the overall quality of an active learning process is evaluated based on the amount of human labeling required, which needs to be minimized.

3.2 Active learning scenarios

Active learning relies on cleverly selecting samples for annotation. Depending on the use cases, the profusion of available data, the material conditions of the studied use case and the real-life restrictions on the oracle, the active learning paradigm can present different incarnations, adapted to various scenarios. In [159], authors identify three main settings, which are *pool-based* active learning, *stream-based* selective sampling and *membership query synthesis*.

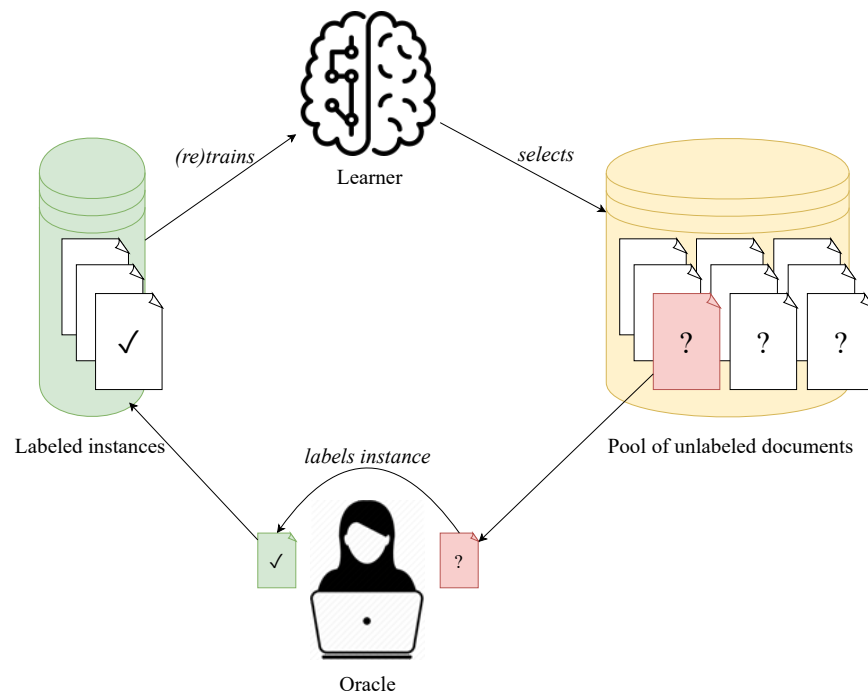


Figure 3.1: Pool-based active learning scenario

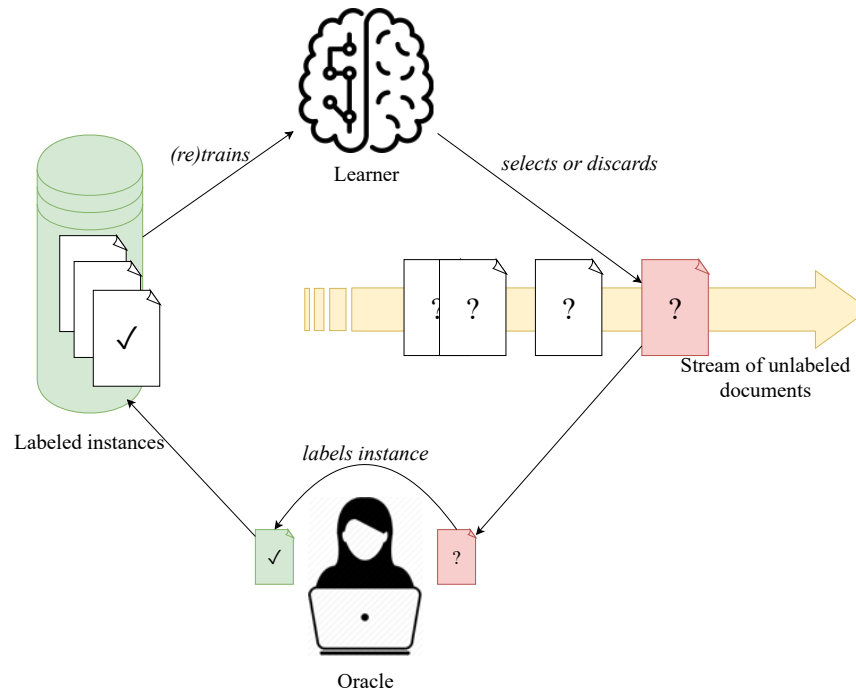


Figure 3.2: Stream-based active learning scenario

3.2.1 Pool-based active learning

The pool-based active learning scenario [104] assumes that a large *pool* of unlabelled data is already available (Figure 3.1, right) before any model training takes place. Also, it is assumed that a small subset of data have already been labelled (Figure 3.1, left) to kick-start the training of the learner model. Typically, pool-based active learning works by iterations, called *cycles*. During each cycle, the learner model is trained using the available labelled data. Then, one or several documents are *sampled* among the unlabelled pool according to some criterion (discussed in sections 3.3 to 3.6) The sampled documents are presented to the oracle for annotation. Once labelled, the documents are added to the set of labelled instances and the learner model re-trained. The quality of the sampling strategy is assessed by measuring the model’s improvement as a function of the number of manually labelled instances.

3.2.2 Stream-based active learning

In the case of the stream-based active learning scenario [31] (Figure 3.2), it is supposed that obtaining unlabelled documents from a continuous stream of unlabelled data is free or inexpensive, while the manual labelling of instances by the oracle still being considered costly. During the active learning cycle, it has to be decided if every encountered data point from the stream is informative enough to be worthy of oracle labelling. This scenario is suitable when the data stream is dense enough to never lack fresh unlabelled instances, such that storing unlabelled samples can be avoided.

3.2.3 Query synthesis

In this setting [8] (Figure 3.3), the unlabelled instances shown to the oracle for labelling are artificially synthesized from the input space, instead of being pulled from a set of examples met in the wild. This method is particularly helpful if the input space distribution is known, for example a set of spatial coordinates [32]. However, some complex input domains like real-life photographs, spoken or written natural language are hard to synthesize. Moreover, if the oracle is a human agent, the synthesized example must be semantically understandable and have a meaningful label in the output space.

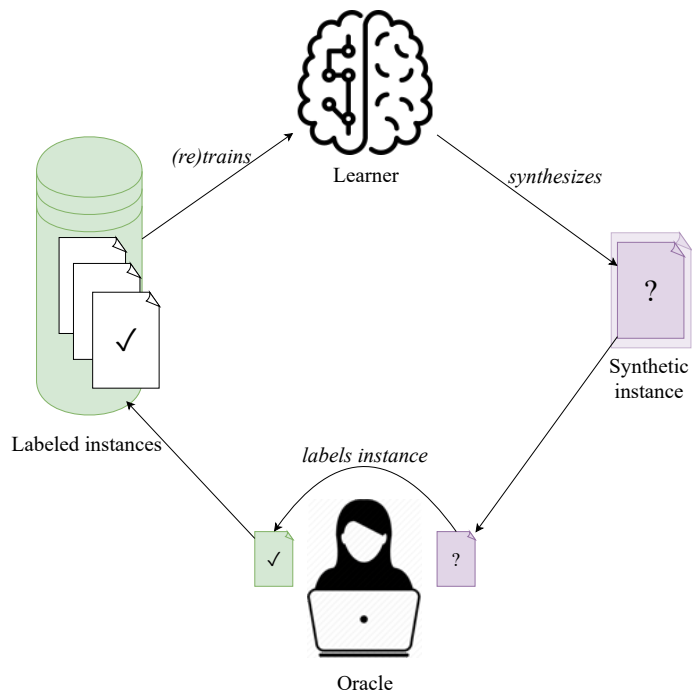


Figure 3.3: Query synthesis active learning scenario

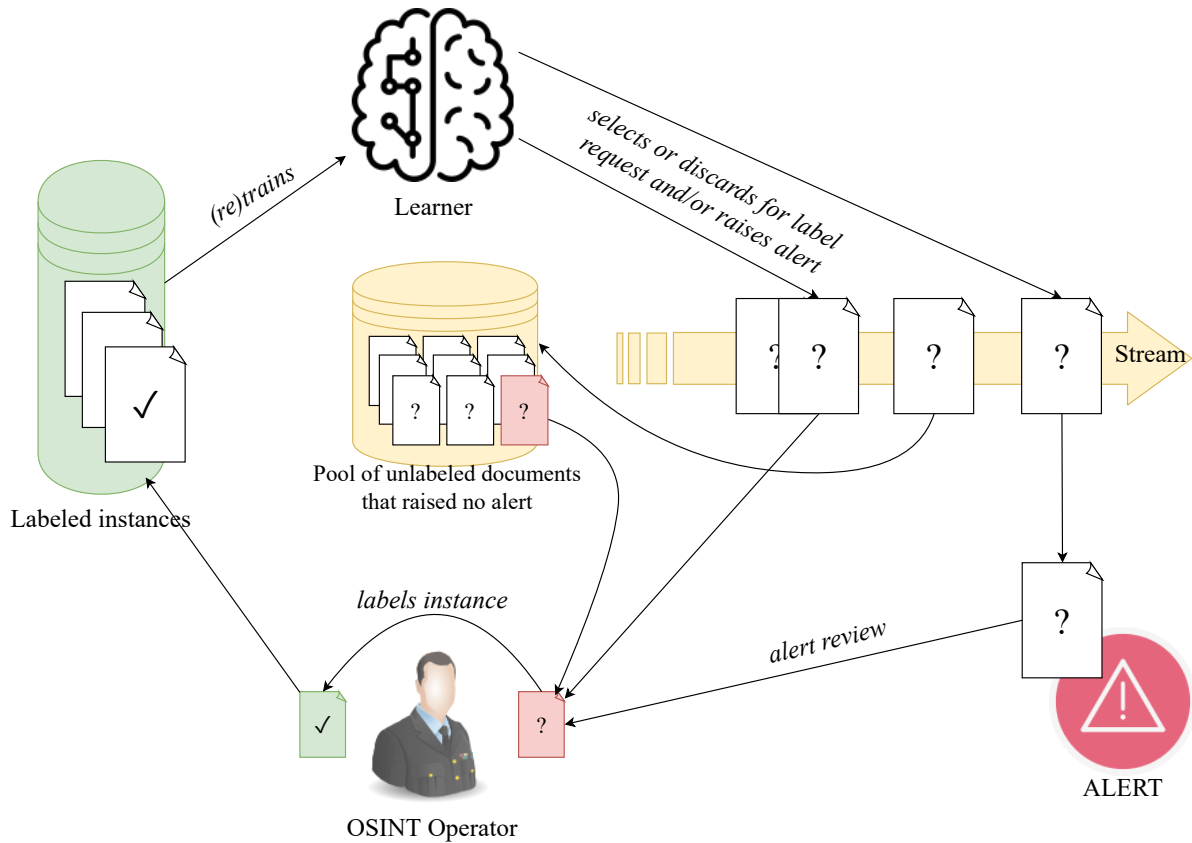


Figure 3.4: Alert-raising, hybrid pool/stream active learning scenario

3.2.4 Hybrid scenarios

In an OSINT context, users might be interested in training a (possibly) alert-raising model while keeping up to date with new, freshly crawled data, and without losing track of the less informative documents that may turn out to be informative later. Modern storage capacities allow for a hybrid pool/stream scenario where the pool is expanding with incoming crawled entities. As a benefit, the active learning framework is no longer dependent from the potential drying up or source shifts of the stream. Figure 3.4 illustrates this scenario.

The following sections study the various sampling strategies, processes and criteria according to which unlabelled examples are selected (*sampled*) to be labelled by the oracle. The global objective is to sample examples that are the most informative for the task the model is being trained to solve. Within this context, are considered as “informative” the data samples providing the greatest improvement to the model once added to the training set. To achieve this goal, several questions have to be answered. How much has a model improved after adding a new labelled document to the training set? Does it help the model to refine the frontiers between classes? Does it participate to better cover the input space? What parts of the input space are worthy of being covered? How does the new knowledge brought by a sampled document complement prior or ulterior knowledge?

3.3 Uncertainty-based strategies

In the subsequent sections, the following notations are employed.

- ω is a model's set of parameters;
- x is an input data point or a feature vector representing it;
- y is the ground-truth value of a data point;
- $\hat{y}_i = P(y | x_i, \omega)$ is the prediction for the data point x_i with model parameters ω ;
- $\hat{y}_{i,c} = P(y = c | x_i, \omega)$ is the output prediction for a class c .

Uncertainty sampling [104] is a quite straightforward active learning method for selecting informative examples, based on the following assumption: the more uncertain an example for the model being trained, the more likely it is to refine the class frontier. Uncertainty sampling have been broadly used with both shallow and probabilistic models like SVM [179] or KNN [76]. For binary classification, where the model's prediction \hat{y} belongs to the $[0, 1]$ interval, an elementary measure of confidence can be defined as the distance to middle point 0.5 [104, 103]. Thus, the most uncertain example x_{unc} is, by definition:

$$x_{\text{unc}} = \underset{i}{\operatorname{argmin}} |\hat{y}_i - 0.5| \quad (3.1)$$

In a C -class multi-class pool-based scenario, uncertainty can be measured by least confidence [104] with confidence being defined as the highest output class probability. The most uncertain example x_{unc} is:

$$x_{\text{unc}} = \underset{i}{\operatorname{argmin}} \max_{c \leq C} P(y = c | x_i, \omega). \quad (3.2)$$

In the same scenario, with *margin sampling*, the learner chooses the sample x_{unc} for which the difference between the two most probable classes is the smallest:

$$x_{\text{unc}} = \underset{i}{\operatorname{argmin}} \left[P(y = c_0 | x_i, \omega) - P(y = c_1 | x_i, \omega) \right] \quad (3.3)$$

The *Maximum entropy* approach [162] chooses data points that maximize the predictive entropy, as defined by:

$$\begin{aligned} \mathbb{H}(y | x) &= \mathbb{E}_{P(y)} [-\log \hat{y}] \\ &= - \sum_{c \leq C} \hat{y}_c \log \hat{y}_c \end{aligned} \quad (3.4)$$

The strength of maximum entropy sampling lies in the global consideration of the predictions for all classes.

3.3.1 Calibration of neural networks

Most active learning strategies heavily rely on the uncertainty of the unlabelled sampling candidates with regard to the trained model. Unfortunately, deep learning models are subject to overconfidence. For multi-class, mono-label classification, one typically uses the output of the softmax-activated final classification layer as a probability distribution among all possible classes, which has shown to produce overconfident results [190]. The resulting uncertainty sampling is susceptible to behave poorly, possibly worse than random sampling. Besides the improvement of uncertainty sampling, neural networks would benefit from being able to indicate how likely it is that they are incorrect, especially for critical applications. Ideally, this information should be embedded in the probability distribution output by the network, with a prediction of 1.0 corresponding to absolute confidence. On several labelled examples, this would translate to the prediction accuracy being somehow proportional to the model’s confidence. Such a neural network would be said to be well *calibrated*, hence *reliable* [131]. The class prediction confidence of a single example can range in $[0, 1]$ whereas the accuracy over a single class prediction is either 0 or 1, making reliability ill-defined for a single example. The reliability of a model over a pool of (labelled) instances can be visualized by gathering examples by confidence bins and plotting a reliability diagram, with the mean accuracy of confidence bins as a function of confidence (Figure 3.5). Reliable, well-calibrated models have a reliability diagram close to the identity function.

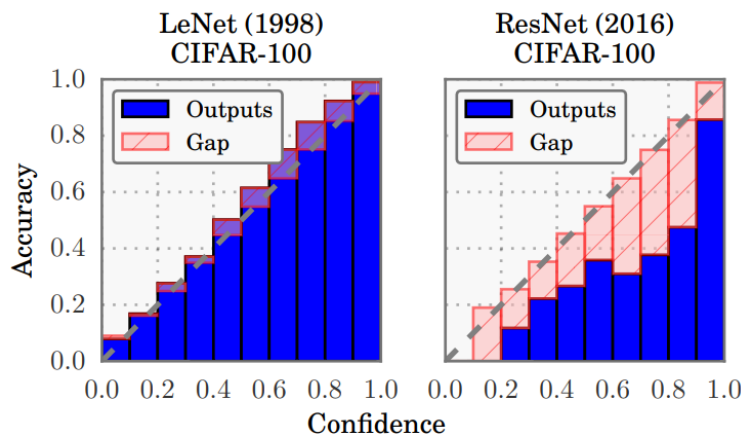


Figure 3.5: Reliability diagrams for a 5-layer LeNet (left) and a 110-layer ResNet (right) on CIFAR-100 [57].

Model calibration can be adjusted with temperature scaling [56]. Temperature scaling divides the logits $z = \{z_c\}_{c \in C}$ (inputs to the softmax function) by a learned scalar parameter T :

$$\hat{y} = \frac{\exp(z/T)}{\sum_c \exp(z_c/T)} \quad (3.5)$$

Calibration is done on the validation set, after the regular training, by optimizing T with respect to negative log-likelihood on the validation set. Because the parameter T does not change the maximum of the softmax function, the predicted class remains unchanged.

3.3.2 Bayesian Neural Networks

Bayesian Neural Networks (BNN) and their associated practices form a rigorous paradigm to train neural networks that are “aware” of their uncertainty by placing a probability distribution over their parameters. More specifically, Bayesian CNN are CNNs with prior probability distribution placed over the convolution kernels $\omega = \{W_1, \dots, W_L\}$: $\omega \sim P(\omega)$. A typical probability distribution is the Gaussian prior $P(\omega)$ [46]. In this case, the number of parameters is doubled, replaced by a mean and a standard deviation. *Variational approximation* is a technique providing a convenient proxy to BNN behaviour. In particular, stochastic regularization techniques like dropout [65] have shown to produce a good approximation of the BNN inference. In practice, kernel weights are randomly set to zero during model inference to sample from approximate prior. This technique is referred to as Monte Carlo dropout. It is easy to implement and can be set up on any network *a posteriori* by inserting (if needed) and activating dropout layers at the prediction time.

The expected prediction over k sampled parameters $q(\omega) = \{\hat{\omega}_j\}_{j \leq k}$ can be straightforwardly approximated as the mean over the k predictions:

$$\mathbb{E}_{P(\omega)} [P(y = c | x, \omega)] \approx \frac{1}{k} \sum_{j=1}^k P(y = c | x, \hat{\omega}_j) \quad (3.6)$$

BNNs allow exploiting an ad-hoc expression of uncertainty, measured as the mean standard deviation of predictions over the C available classes, used to produce uncertainty maps for scene understanding and semantic segmentation [85, 81]:

$$\sigma(x) = \frac{1}{C} \sum_{c=1}^C \sigma_c = \frac{1}{C} \sum_{c=1}^C \sqrt{\mathbb{E}_{q(\omega)} [P(y = c | x, \omega)^2] - \mathbb{E}_{q(\omega)} [P(y = c | x, \omega)]^2} \quad (3.7)$$

where $q(\omega)$ are the various parameters sets obtained with Monte Carlo dropout.

Variation ratio is another measure of uncertainty based on the disagreement of k Monte Carlo inferences, representing uncertainty as the proportion of predicted class labels that are not the modal class prediction:

$$\text{variation-ratio}[x] = 1 - \frac{1}{k} \max_{c \leq C} \sum_{j=1}^k \begin{cases} 1 & \text{if } \operatorname{argmax}_c (P(y = c | x, \hat{\omega}_j)) = c, \\ 0 & \text{otherwise.} \end{cases} \quad (3.8)$$

3.3.3 Bayesian Active Learning by Disagreement

Bayesian Active Learning by Disagreement [70] (BALD) is a method prioritizing samples that are expected to maximize the information gained about the model parameters. Given a training set $\mathcal{D}_{\text{train}}$ and the various dropped-out sets of model parameters ω who result from it, BALD exploits both the entropy of the true model posterior $\mathbb{H}(y | x, \mathcal{D}_{\text{train}})$ and the entropies obtained with individual variational Bayesian approximations $\mathbb{H}(y | x, \omega)$. The

BALD criterion \mathbb{I} of a pool point x defined as:

$$\begin{aligned} \mathbb{I}(y|x, \mathcal{D}_{\text{train}}) &= \mathbb{H}(y|x, \mathcal{D}_{\text{train}}) - \mathbb{E}_{P(\omega|\mathcal{D}_{\text{train}})} [\mathbb{H}(y|x, \omega)] \\ &\approx \sum_{c \leq C} \left(-\hat{y}_c \log \hat{y}_c + \frac{1}{k} \sum_{j=1}^k P(y=c|x, \hat{\omega}_j) \log P(y=c|x, \hat{\omega}_j) \right). \end{aligned} \quad (3.9)$$

BALD favours points with high uncertainty (high $\mathbb{H}(y|x, \mathcal{D}_{\text{train}})$), for which individual settings of the parameters ω produce confident outputs (low $\mathbb{E}_{P(\omega|\mathcal{D}_{\text{train}})} [\mathbb{H}(y|x, \omega)]$).

3.3.4 Ensembles

Several Monte Carlo Dropout inferences are used to produce disagreeing predictions, allowing the measurement of uncertainty. Instead of various dropout distributions over the same model parameters, an actual *ensemble* of models can be used to produce several predictions. Hence, any of the aforementioned uncertainty criteria can be harnessed with model ensembles. An ensemble of models with the same architecture and training data – but different parameter initialization – has shown to produce uncertainty criteria beneficial to active learning for image classification [16]. However, the need to carry out multiple trainings at every active learning iteration is a major drawback.

Another drawback of mere uncertainty sampling resides in the fact that, for each unlabelled instance, uncertainty results from the small distribution of labelled instances that have been used for prior training. They may not be representative of the underlying distribution of unlabelled data, as illustrated in Figure 3.6. Thus, the learner may choose highly uncertain, yet poorly informative outliers in the unlabelled distribution. If a sampled example has a barren neighbourhood, it only helps to refine class boundaries in regions that will scantily produce any improvement in a representative testing set. Worse, pure uncertainty is likely to yield outliers that are irrelevant to the use case and would simply be skipped by the labeller.

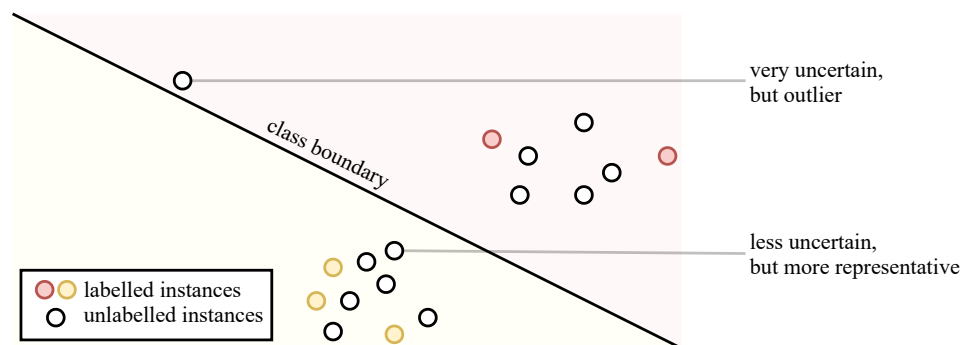


Figure 3.6: A limitation of uncertainty sampling: possibly uninformative outliers

3.4 Information Density

With uncertainty sampling as the only criterion for query selection, the learner is likely to pick non-representative outliers. However, the objective is to evaluate an active learning framework’s ability to perform well on a wide range of unknown examples it has never seen. For this reason, it is advantageous to sample representative examples which the learner can generalize from. Here, a desirable instance is said to be *representative*, and we use the concept of *information density* to indicate how representative it is. Methods exploiting information density need a way to measure the distance between samples, whether it is from input samples directly or from related feature vectors.

Usually, both uncertainty and information density criteria are used jointly in what is called density-weighted uncertainty. The participation weights of each criterion can be optimized with a method called *adaptive active learning* [105]: several instances are pre-sampled with a varying trade-off parameter β between uncertainty and information density, and the model is re-trained several times with the training set augmented by the pre-sampled instances and their predicted label. The finally sampled data point is the one yielding the lesser loss on the unlabelled dataset. To avoid the several re-training implied by this method, a fixed trade-off parameter can be used, still excluding outliers.

Several methods have been proposed to measure information density.

Cosine distance information density [160] straightforwardly measures the representativeness of a sample with the mean cosine similarity to the other N examples, as described in the following equation:

$$r(x) = \frac{1}{N} \sum_{i=1}^N \text{sim}(x, x_i), \quad (3.10)$$

where sim is the cosine similarity, which defines a measure of correlation between two vectors, expressed as their scalar product divided by the product of their corresponding norms:

$$\text{sim}(\vec{p}, \vec{q}) = \cos \angle(\vec{p}, \vec{q}) = \frac{\langle \vec{p}, \vec{q} \rangle}{\|\vec{p}\| \times \|\vec{q}\|}. \quad (3.11)$$

Probability density function estimation aims to draw a density map of the input space. The density at a sample’s coordinates is then used as its information density. The most straightforward method consists in building a histogram of the available input data. Thus, the information density of a sample corresponds to the population of the histogram bin(s) it falls into.

Kernel density estimation (or Parzen window density estimation) [134] is preferred to histograms since it overcomes the limitations related to the discretization of the bins parameters (centers and width) [166]. Given a learner’s input space populated by n instances x_1, \dots, x_n , the density at the position x of the input space is estimated as:

$$q(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right), \quad (3.12)$$

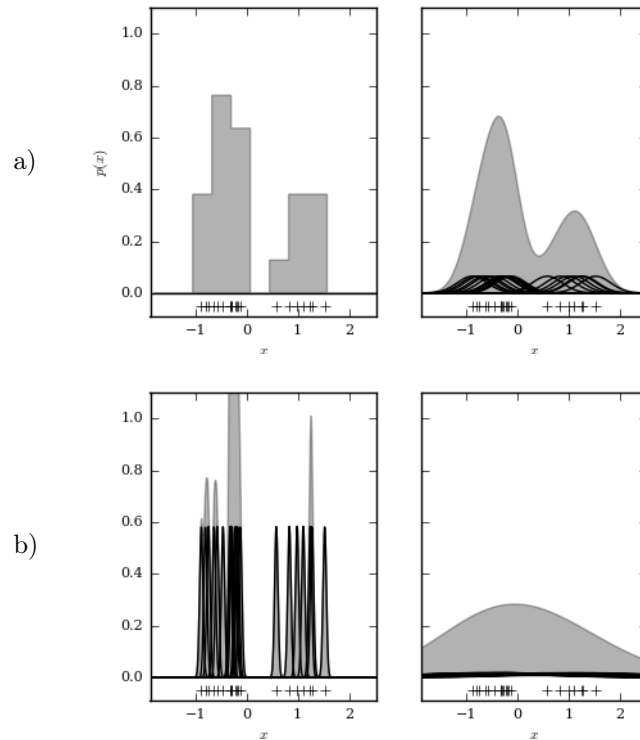


Figure 3.7: Density estimations examples: a) histogram vs. Gaussian kernel density; b) narrow bandwidth vs. large bandwidth. Visual generated on astroml.org.

where h is a bandwidth parameter and K is a kernel. More precisely K is a non-negative, real-valued, integrable function satisfying the following normalization and symmetry requirements:

$$\begin{cases} \int_{-\infty}^{+\infty} K(u) du = 1 \\ \forall u \in \mathbb{R}, K(u) = K(-u) \end{cases} \quad (3.13)$$

For normalized features, it is common to use an isotropic Gaussian kernel (with zero mean and unit variance) so that the density at the position x of the input space is given by:

$$q(x) = \frac{1}{nh\sqrt{2\pi}} \sum_{i=1}^n \exp\left(-\frac{\|x - x_i\|^2}{2h^2}\right) \quad (3.14)$$

where $\|\cdot\|$ is the Euclidean norm and h is the radius of the hyper-sphere centred on x_i .

Figure 3.7-a shows an overlay of 1-dimensional histograms and Gaussian kernels estimators. Concerning the bandwidth's choice, Figure 3.7-b illustrates a bandwidth *too large* resulting in a lack of resolution (left). In contrast, a bandwidth *too narrow* may result in local overfitting (right). In the active learning framework introduced in [202],

authors use a constant bandwidth based on the maximum distance between a sample and its closest neighbour:

$$h = \lambda \max_{k \leq n} \left(\min_{\substack{i \leq n \\ i \neq k}} \|x_k - x_i\| \right), \quad (3.15)$$

where λ is recommended to be between 1 and 10.

3.5 Diversity-based criteria

The aforementioned active learning criteria only select one unlabelled sample per iteration. This leads to a high computational burden, since the model is retrained once per active learning cycle. Gathering a whole batch of samples before retraining the model can alleviate this burden, with all samples selected according to the same sampling criteria (*i.e.* uncertainty). Unfortunately, similar (or identical) pool points may have similar (or identical) uncertainty and information density, although providing very little mutual information. To maximize how informative the sampled batch is, some works propose to put together *diverse* batches, gathered using a *diversity measure*. Batches can be collected by maximizing the angle between samples [19]. In addition to inter-batch dissimilarity, it has been observed that sampling examples dissimilar to the already labelled instances is beneficial to the active learning procedure [41].

3.5.1 A core-set approach for diversity

Core-set batch sampling [158] (Algorithm 1) completely puts aside uncertainty and diversity to gather large unlabelled batches for active learning. The authors demonstrate that deep learning better benefits from a wide variety of examples. Given a sampling budget b , the greedy core-set method chooses the b unlabelled data points that are the farthest from any already labelled data point *w.r.t.* their deep feature representation, one after the other. This results in a selection of wide-spread samples across the deep features' domain. An example is shown in Figure 3.8. To account for representativeness and avoid outliers, authors arbitrarily chose to exclude one out of 10^4 least representative data points.

3.5.2 Contextual diversity

Contextual Diversity for Active Learning (CDAL, [4]) is a sampling method based on a novel criterion that takes into account both uncertainty and its own conception of diversity. Designed for image classification, object detection and semantic segmentation, CDAL favours the sampling of batches of *regions* that are diverse among themselves based on their predicted classes instead of a latent space feature vector. Here, regions designate the entity being classified by the model, *i.e.* the whole image for image classification, the detected bounding boxes for object detection, and the individual pixels for semantic segmentation. Regions are considered similar if they are predicted to belong to the same class. This notion of diversity aims at covering all the possible confusions between classes in a sampled batch.

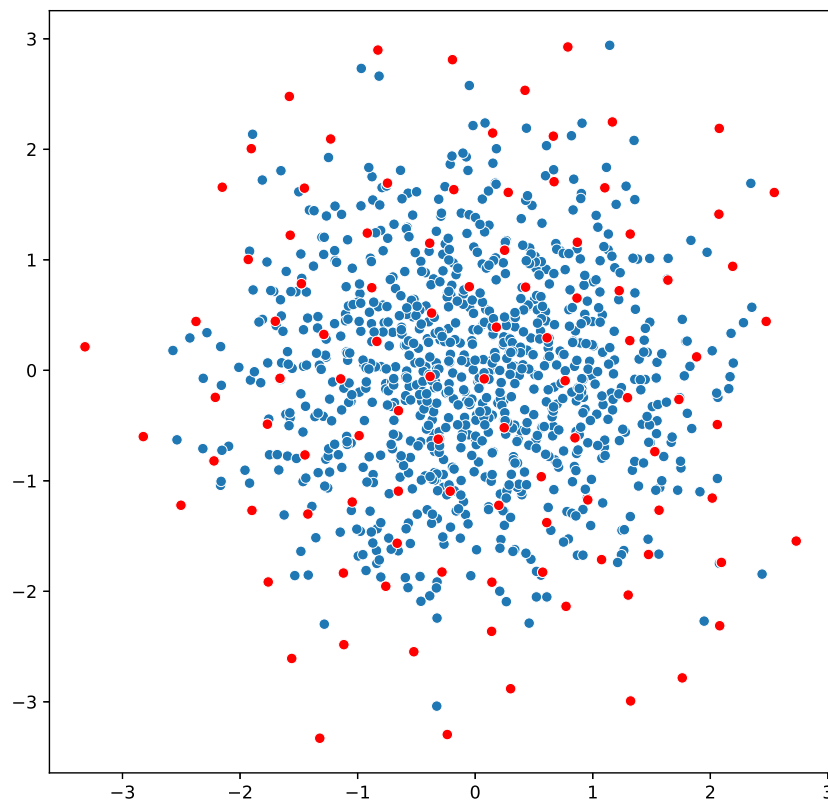
Algorithm 1: Core-set sampling for active learning**Data:** (i) An unlabeled pool \mathcal{U} ; (ii) an initial training set \mathcal{L} ; (iii) a labeling budget $B \in \mathbb{N}^*$ **Result:** $|\mathcal{L}| = B$ **while** $B > 0$ **do** **if** $\mathcal{L} = \emptyset$ **then** Randomly select x_s in \mathcal{U} ; **end** **else** Update distance matrix between \mathcal{U} and \mathcal{L} ; $x_s \leftarrow \operatorname{argmax}_{x_u \in \mathcal{U}} \min_{x_l \in \mathcal{L}} \|x_u - x_l\|_2$; **end** $\mathcal{L} \leftarrow \mathcal{L} \cup \{x_s\}$; $\mathcal{U} \leftarrow \mathcal{U} \setminus \{x_s\}$; $B \leftarrow B - 1$;**end**

Figure 3.8: 100 data points (red) sampled from a distribution of 1,000 2-d points (blue) with the greedy core-set algorithm, favouring the diversity of sampled points. Unless it is explicitly taken care of, this method tends to sample peripheral outliers.

Thus, unlabelled regions are represented by their matrix of *class confusions*, the arrays of the average scores \hat{y} of regions whose predictions fall in the same class, weighted by their entropies (not to be confused with the confusion matrix that maps ground-truth labels to predictions). Given C different classes, the class-specific confusion for class c of an image \mathbf{I} is a C -dimensional vector defined as:

$$\mathcal{P}_{\mathbf{I}}^c = \frac{\sum_{r \in \mathcal{R}_{\mathbf{I}}^c} \mathbb{H}(\hat{y}_r) \hat{y}_r}{\sum_{r \in \mathcal{R}_{\mathbf{I}}^c} \mathbb{H}(\hat{y}_r)}, \quad (3.16)$$

where $\mathcal{R}_{\mathbf{I}}^c$ is the set of regions of \mathbf{I} predicted as belonging to class c . The class confusions matrix of an image \mathbf{I} is the stacking of its individual class confusions:

$$\mathcal{P}_{\mathbf{I}} = \begin{bmatrix} \mathcal{P}_{\mathbf{I}}^1 \\ \vdots \\ \mathcal{P}_{\mathbf{I}}^C \end{bmatrix} \quad (3.17)$$

The dissimilarity between two images \mathbf{I}_1 and \mathbf{I}_2 is measured as their *pairwise contextual diversity*, the sum over c of the symmetric Kullback–Leibler divergences between class-specific confusions:

$$d(\mathbf{I}_1, \mathbf{I}_2) = \sum_{c=1}^C \begin{cases} \frac{1}{2} \mathcal{P}_{\mathbf{I}_1}^c \log \left(\frac{\mathcal{P}_{\mathbf{I}_1}^c}{\mathcal{P}_{\mathbf{I}_2}^c} \right) + \frac{1}{2} \mathcal{P}_{\mathbf{I}_2}^c \log \left(\frac{\mathcal{P}_{\mathbf{I}_2}^c}{\mathcal{P}_{\mathbf{I}_1}^c} \right) & \text{if } \mathcal{R}_{\mathbf{I}_1}^c \neq \emptyset \text{ and } \mathcal{R}_{\mathbf{I}_2}^c \neq \emptyset, \\ 0 & \text{otherwise.} \end{cases} \quad (3.18)$$

To sample a batch of B diverse unlabelled images w.r.t. contextual diversity, authors rank images with the greedy core-set algorithm (Algorithm 1) with class confusions matrices and pairwise contextual diversity instead of deep feature vectors and euclidean distance. This method appears to outperform the traditional core-set on several classification, detection and segmentation tasks.

Going further, at every active learning iteration, the authors train a bidirectional LSTM model to rank unlabelled images based on their class confusions matrices, in a reinforcement learning fashion. Here, a reward term is defined a weighted sum of the cumulative pairwise contextual diversities. Two supplementary rewards accounting for representativeness and class balancing, are also used. This variant outperforms the contextual diversity core-set on all tasks.

3.5.3 BatchBALD

BatchBALD [93] is an extension of BALD [70] that takes into account the information overlap between sampling candidates to greedily build a batch of *diverse* samples. The sampling batch is gathered by iteratively choosing the unlabelled data point that maximizes $\alpha_{\text{BatchBALD}}$, the mutual information of a batch $\{x_1, \dots, x_n\}$, defined as:

$$\alpha_{\text{BatchBALD}}(\{x_1, \dots, x_n\}) = \mathbb{H}(y_1, \dots, y_n) - \mathbb{E}_{\mathfrak{q}(\boldsymbol{\omega})} [\mathbb{H}(y_1, \dots, y_n | \boldsymbol{\omega})] \quad (3.19)$$

Following the authors notations, using a Monte-Carlo estimator with k sampled sets of parameters $\{\hat{\boldsymbol{\omega}}_j\}_{j \leq k}$, the first term is approximated by summing over all possible configurations $\hat{y}_{1:n}$ of $y_{1:n}$:

$$\begin{aligned} \mathbb{H}(y_1, \dots, y_n) &= \mathbb{E}_{\mathbb{P}(y_1, \dots, y_n)} [-\log \mathbb{P}(y_1, \dots, y_n)] \\ &= \mathbb{E}_{\mathbb{P}(\boldsymbol{\omega})} \mathbb{E}_{\mathbb{P}(y_1, \dots, y_n | \boldsymbol{\omega})} [-\log \mathbb{E}_{\mathbb{P}(\boldsymbol{\omega})} [\mathbb{P}(y_1, \dots, y_n | \boldsymbol{\omega})]] \\ &\approx - \sum_{\hat{y}_{1:n}} \left(\frac{1}{k} \sum_{j=1}^k \mathbb{P}(\hat{y}_{1:n} | \hat{\boldsymbol{\omega}}_j) \right) \log \left(\frac{1}{k} \sum_{j=1}^k \mathbb{P}(\hat{y}_{1:n} | \hat{\boldsymbol{\omega}}_j) \right). \end{aligned} \quad (3.20)$$

The second term is straightforwardly approximated by averaging entropies over the stochastic set of parameters $\mathfrak{q}(\boldsymbol{\omega})$:

$$\begin{aligned} \mathbb{E}_{\mathbb{P}(\boldsymbol{\omega})} [\mathbb{H}(y_1, \dots, y_n | \boldsymbol{\omega})] &= \sum_{i=1}^n \mathbb{E}_{\mathbb{P}(\boldsymbol{\omega})} [\mathbb{H}(y_i | \boldsymbol{\omega})] \\ &\approx \frac{1}{k} \sum_{i=1}^n \sum_{j=1}^k \sum_{c=1}^C \mathbb{P}(y_i = c | \hat{\boldsymbol{\omega}}_j) \log \mathbb{P}(y_i = c | \hat{\boldsymbol{\omega}}_j) \end{aligned} \quad (3.21)$$

On the MNIST dataset (a database of handwritten digits [36]), with a batch size of 10, BatchBALD has shown to produce learning curves equivalent to BALD with a batch size of 1, meaning that BatchBALD allows to divide by 10 the number of trainings for equivalent performances.

3.6 Adversarial approaches

A Generative Adversarial Network (GAN) is a framework in which two neural networks, so-called generator and discriminator, compete one against the other. The generator network is trained to generate meaningful images from a random feature vector. On the counterpart, the discriminator is trained to recognize real examples from the ones artificially synthesized by the generator [51]. Basically, the generator is trained by learning to fool the discriminator, helped by its discrimination scores that quantify how much an artificial image is close to realism.

Variational Adversarial Active Learning (VAAL) [168]: Derived from this mode of operation, VAAL employs a discriminator network in addition to its main classifier to separate unlabelled images from labelled images, and in

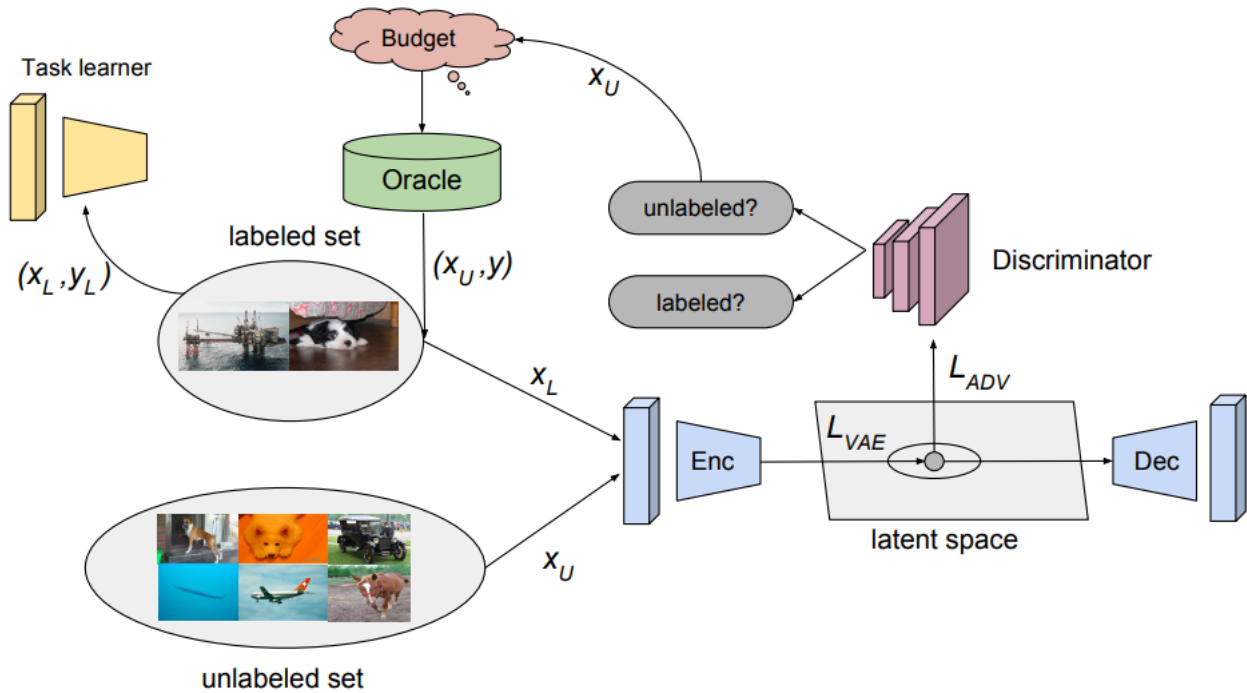


Figure 3.9: Variational Adversarial Active Learning [168]

particular what are the most obvious unlabelled images. In response to this, the classifier is re-trained with these images to try and fool the discriminator (Figure 3.9). This method has shown to outperform the core-set approach on various image classification tasks.

State-Relabelling Adversarial Active Learning (SRAAL) [201]: Similarly to VAAL, SRAAL consists in a unified representation generator and a labelled/unlabelled state discriminator. The generator embeds the annotation information into the final image features (unified representation) via a supervised target learner and an unsupervised image reconstructor (a variational autoencoder). The added value comes from the so-called *state-relabelling* process: to favour uncertain images, an *online uncertainty indicator* (OUI) is introduced to relabel the state of unlabelled samples and endues them with different importance factor. In this way, the target labels of unlabelled images for the discriminator range in the $[0, 1)$ interval. Finally, the state discriminator is updated through the labelled and unlabelled state losses and helps selecting the most informative samples. State-relabelling has shown to improve the active learning process over image classification on CIFAR and semantic segmentation on CityScapes.

3.7 Semi-supervision

Pseudo-labelling (PL) (Figure 3.11) proposes to incorporate to the training set the unlabelled examples that are the most certain with regard to the model being trained [53, 102]. This certainty is subject to evolve at each active learning iteration, and the ensemble of pseudo-labelled documents is to be kept up-to-date. PL have been notably

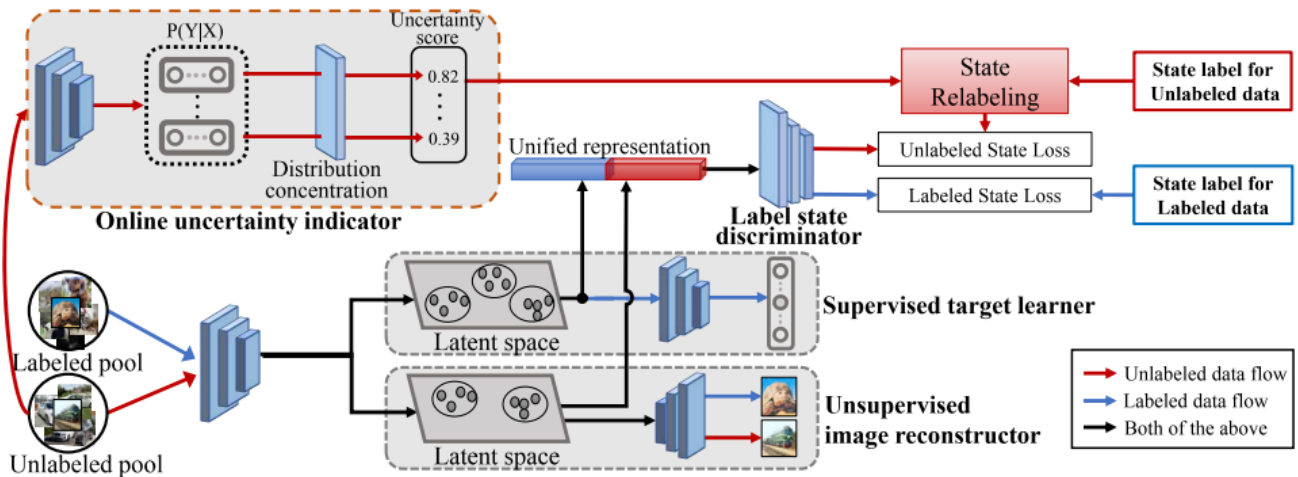


Figure 3.10: State-Relabelling Adversarial Active Learning [201]

used to improve image classification tasks [190] and automatic speech recognition [79]. Typically, documents are pseudo-labelled by following similar (except opposite) criteria to the uncertainty criteria described in the previous section, like entropy.

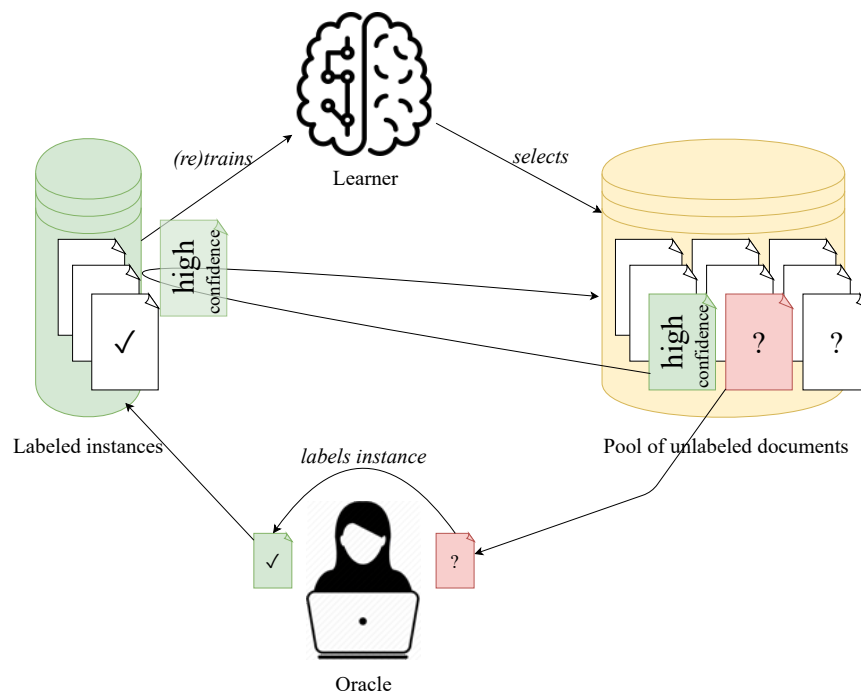


Figure 3.11: Active learning with pseudo labelling of unlabelled examples of high certainty

3.8 Conclusion

The abundance of data pipelines - open source or not - is propitious to the implementation of active learning strategies to learn an automated task. Sampling strategies are designed to choose what unlabelled data to unveil knowledgeable

labels from, in order to build a powerful model with as few labels as possible. Usual sampling strategies rely on various expressions of criteria such as uncertainty, information density and diversity. It is worth mentioning that in a setup of data profusion and label scarcity, semi-supervision can be complementary to active learning. Active learning still present a certain number of limitations. First, when working under the assumption that the human label provider is an expert of the task being learnt, the ground-truth label is only a fraction of the expert knowledge at stake. During Active Learning, the model being trained must deduce high level class membership rules by itself, based on a limited number of examples. We formulate the hypothesis that a greater level of knowledge can be taught from oracle to model. Such additional information may be expressed under the form of meaningful words among sentences or meaningful regions of interest in images. Second, active learning is usually carried out as a succession of iterations whose time-consuming steps are executed one after the other, including training, sampling and description of incoming unlabelled documents. Such drawbacks should be addressed to help getting closer to the goal of having the best model possible given the data being collected and the amount of labels already retrieved.

The following chapters gather our contributions to addressing such issues, by extracting interpretation cues from trained models to their re-injection during training in a data scarcity context.

Part II

Contributions

Chapter 4

An Interpretable Model to Measure Fakeness and Emotion in News

Abstract

Fake news and post-truth are present today everywhere. The huge number of online news outlets and the frequency of content creation underlines the demand for automatic information evaluation tools. Previous work usually focuses either on automatic fact-checking, or on fake-looking identification: the former tries to match a piece of content with trustable information, enough to confirm or infirm the claims. The latter gathers clues to help the reader's assessment of the piece of content. In this domain, there is no silver bullet: the reader desires verifiable information, thus the *fake news detector* should be interpretable or explainable.

In this chapter, we propose the TC-CNN interpretable text classifier. We apply it on two different tasks: fake news detection and emotion classification. A second contribution relies on these two classifiers, and on a third-party hate detector, to perform a case study on this year real and fake-news press articles, in a comparison between mainstream and alt-right media.

4.1 Introduction

Since 2015, all media observers have become aware of the “fake news” problem, even though there has always been information manipulation campaigns throughout history. The easiness to create convincing news outlets and to diffuse media content without any field reporters leads to a massive amount of dubious content [150]. These press- or news-looking content are then broadly diffused through the social media and trigger harmful consequences, notably observed under the form of a discredit towards journalists and politics. Computer sciences and AI already proposed a few approaches to deal with this challenge. Automatic fact-checking is already partially addressed, but presents intrinsic limitations [60] and often relies on third-parties fact-checkers. A second approach to detect manipulations consists in evaluating the fakeness of every piece of information. Most fake news are persuasive, however they often do not resist against a well-formed mind. Manipulation clues detection in texts has long been a research topic, helped notably by tools such as LIWC [137]. Such expert-based approaches produce an insightful analysis on discourses and official announcements, but require rare competences to be established and updated.

Today’s news articles already mutated since the beginning of the century, with clickbait techniques evolving relatively fast. We believe that machine learning techniques are more adapted to this task. Indeed, deep learning classification does not seem relevant to study whether a fact is false. However, it brings some valuable hints about the writing *style* of a piece of text. In this chapter, we propose the TC-CNN (Textual Class-activation-mappings Convolutional Neural Network) approach, a new model for deep learning based text classification, that jointly produces an interpretation of the input, highlighting the patterns supporting or opposing a label prediction. Indeed, the structure of a fully convolutional network defines receptive fields that are naturally adapted to automatically detecting groups of successive words that reveal local, contextual and salient knowledge [89, 203]. Such local patterns are then used as rationales for identifying the corresponding category in a classification task.

We apply this model on a fake news-related task, which concerns the detection of biased press articles, written in English. This topic is intrinsically complex, and this sole classifier cannot be considered as a silver bullet. We are aware of this and propose a broader framework, capable of evaluating emotion, bias and hatred levels in a text. The proposed technique is applied on real data: we perform a case study on recent pieces of news and “fake” news, comparing the writing style of major English language news outlets.

The rest of this Chapter is structured as follows. Section 4.2 gives some details about the challenging fake news problem and describes previous approaches to solve it. The proposed an interpretable model is described in Section 4.3, and its usage is discussed and validated in Section 4.4. Then, in Section 4.5, a comparative case study is performed, comparing articles stemming from three main sources or aggregators and enabling the reader to grasp the benefit to reveal the complexity of today’s media landscape. Finally, Section 4.6 concludes this chapter.

4.2 Related Work

4.2.1 The fake news context

Various definitions and categorizations have been proposed in the domain of fake news analysis; basically, a distinction is done between “serious fabrications”, where news articles are forged, mentioning events that never happened, “hoaxes” or rumours that only aim to be spread (and are sometimes referred to as *bullshit*), and “satire”: fabrications with an obvious humoristic goal (e.g., *The Onion*¹) [150]. Overall, the term “fake news” refers both to the globally speaking post-truth informational space, and to pieces of information that are intentionally diffused, while knowing they are false.

Beyond this term, the real problem deals with information manipulation, and the many ways to mix intent, information (e.g. messages) and knowledge (e.g. facts). An exhaustive formalization of this problem has been recently proposed, with a special focus on the act of *lying* [75]: everything is not only based on content falsehood, but also on content perception based on its source, and its propagation.

Social media are a primary environment for fake news propagation because of their intrinsic nature [164]. Social psychology already offers the ground for opinion acceptance by social contact, as in the classic models of opinion propagation [54]. This human behaviour is increased by the recommendation algorithms, creating at the same time echo chambers and viral diffusion. Nowadays, this space seem structured by clusters of like-minded people, exchanges of emotion-loaded content and a clear polarization of opinions.

Other recent initiatives are focuses on the automation (or at least, the up-scaling) of fact-checking, such as the CrossCheck² project launched in 2017 with Google News Lab in partnership with more than 20 media outlets. In parallel, Facebook associated with eight French media to reduce the amount of false information on its website. A similar project had already been launched in the United States with the support of ABC News, AP, FactCheck.org, Politifact and Snopes. CrossCheck is a collaborative journalism project that brings together editorial teams from all over the world to accurately deal with false, misleading or confusing statements circulating online, studying topics, comments, images and videos.

A very relevant survey on fake news detection proposed to split the problem into four challenging tasks, evaluating: the fact, the style, the propagation and the credibility of the emitters of a piece of news [205]. Within the *style* challenge, “clickbait” articles detection is specifically mentioned. The different kinds of *fakes* is still an open debate. Many papers propose their own taxonomies, such as *malicious / hoax / satire* [139], on social media texts; *fake / true* for posts combining text, image and propagation data [119] or even to evaluate whether the title of a press article is related, or opposed, to its own text [78]. Always more information can be combined, such as the system XFake, which independently analyses semantic, linguistic and contextual (e.g., the attributes of the source: author, media

¹<https://www.theonion.com/>

²http://www.lemonde.fr/les-decodeurs/article/2017/02/28/lutte-contre-les-fausses-informations-le-monde-partenaire-du-projet-crosscheck_5086731_4355770.html

outlet...) data [194]. A more classical approach, that we follow in our work is focused on classifying press articles along three classes: *biased* (actively promoting a point of view), *bullshit* (when the article only aims at attracting attention), and *legit* (the closer possible of a “true” label for a press article) [84].

4.2.2 Machine learning on related text classification problems

Text classification tasks have been widely addressed in numerous real applications over the last few years, such as emotion, sentiment and opinion classification [2, 22, 48], language identification [21] and even irony detection [23]. For the exploitation of raw textual data, the classification task is addressed by two main processes: feature extraction and classification itself. Common feature extraction methods (word embeddings) include GloVe [138], FastText [77], BERT [37]. On the classification side, Convolutional Neural Networks (CNN) [89] and Recurrent Neural Networks (RNN) [28] have recently become widespread.

Among text classification tasks, emotion detection has known a quick recent evolution, from dictionary-based [126] to machine learning based approaches: empowered by a 7-class, 40,000-tweet dataset, accuracies have increased, from 60% [17] to 70% [110]. Powered by bigger social media datasets, the project *Text-emotion*³ limited itself to 5 emotions (including neutral): various scientific opinions cohabit on the number of classes to keep, with regard to annotators agreement. This project reached a global accuracy of 62% on social media posts.

Entirely dedicated to the web content moderation, the detection of hatred has recently benefited from both theoretical and practical advances. Among the many contributions to this domain, *hatesonar*, a classifier embedded in a Python library (thus, very easy to integrate in a larger system), is grounded on a 3-classes hate detection: “hate” (directed towards a group of people), “offence” (directed towards an individual) or “neither” [35]. A widespread claim consists in combining different indicators to analyse Web2.0 content. As an example, the *Oasis* system aggregates topic detection, and two CNN-LSTM based classifiers for sentiment (3 classes) and emotion (4 emotions, and neutral) detection [110].

4.2.3 Explainable AI and interpretable fake news

The idea to detect fake news with a useful explanation is relatively new, with initiatives such as DEFEND [165] relying on user comments to automatically spot controversial claims in news articles. This approach heavily relies on a sane community of readers-commenters, which may not be the case for every media outlet.

More generally, in a classification setup, obtaining comprehensive class-membership cues is a highly tedious task, significantly more complex than a simple labelling one. To this purpose, two techniques are often considered to characterize sub-documents segments. A first one consists in observing the behaviour of an attention mechanism [14]. The second approach is based on the extraction of class activation maps [203].

³<https://github.com/tlkh/text-emotion-classification>

A self-attention mechanism was introduced in [109], where high attention scores assigned to crucial words are observed when performing text classification. Also, it has been shown that attention is transferable [198]. However, attention scores are flawed when considering the purposes of this chapter. They can be interpreted as the salience of a sub-part of the document relatively to a generic classification task, and not relatively to a precise label. They are ranging from zero (no interest) to one (high interest) without carrying information, whereas the considered sub-parts play in favour or against class membership.

A different approach concerns the so-called class activation maps (CAMs), introduced in [203]. CAMs are able to define the contribution of each sub-part of the document to each considered class. They can be seen as the penultimate neuron layer. They however require specific architecture conditions in order to conserve a logical matching with the inputs.

4.2.4 Discussion

Past work paved the way for better algorithm architectures, and for better taxonomies. The former enables us to propose an accurate prediction, combined with a mechanism for interpretation of the result: we decide to follow the class activation maps approach, and to apply it on textual input, following recent work [58]. The latter is focused on the application on real-world data: *fake news* cannot be reduced to a simple yes/no classification problem. Pseudo-fakes, ideological articles, hatred and emotion-loaded content are too often mixed-up. They all do participate in the *fake news* phenomenon and have to be measured separately, then combined, to build a complete analysis of the content of press-looking articles, avoiding black-box predictions.

Also, this combination cannot be seen uniquely from the domain of machine learning. Fake news corpora exist and are invaluable to train our methods. Yet, they are by nature out-dated, topic-specific and susceptible of bias. Such systems have to be illustrated and exploited with two goals: to show their relevance, and to update our awareness about today's media.

To tackle this intricate challenge, we introduce an interpretable text classifier, grounded on a CNN architecture while exploiting the CAMs. We propose to train two models, to predict emotions and style, and re-exploit a previous hate detection library. With these tools, we propose a case study on this year press articles, either real or fake.

4.3 TC-CNN: an interpretable model for biased news article classification

4.3.1 Spatially interpretable architecture

We intend to use a model architecture that would allow extracting spatially interpretable cues classification rationales R for any input text.

Initially used for image classification purposes [203], a recent work on text classification propose to adapt the CAM technique to textual data [58]. The method relies on the observation that spatiality is preserved across convolutional layers, whereas it is lost in the last fully-connected layers used by some CNNs. Hence, it only concerns fully-convolutional networks where global average pooling (GAP) or global max pooling (GMP) is applied to squash the T spatial features vectors associated to the deepest feature maps $F = \{F_1, \dots, F_T\} \in \mathbb{R}^{T \times K}$ into a single, global feature vector $F_g \in \mathbb{R}^K$ in which all spatiality is lost. Here, K denotes the size of the considered feature maps.

The model that we have adopted is a fully-convolutional neural network made of 3 layers of 128 kernels of size 5 followed by a global average pooling and a softmax classification layer. We used the pre-trained FastText word embedding [77]: a context-aware embedding such as BERT requires a higher-complexity architecture and would degrade the interpretability of the final result because it already takes into account the role of the neighbouring words without explaining how. In our model however, we directly trace the importance of each word's contribution.

Because we need to preserve the temporality across layers, we use the same padding for convolutions, so that there is exactly one output layer directly corresponding to each input token. Thus, the last convolutional layer presents a number of outputs that is equal to the number of input words. In this way, the t^{th} output of the last convolutional layer describes the t^{th} word of the input sentence, while taking into consideration its context within the convolutional receptive field.

The CAM extraction is explained hereafter. If there are C different labels, then the softmax input is defined as:

$$S = W^T F_g + b \quad (4.1)$$

where $W = \{w_c^k\} \in \mathbb{R}^{K \times C}$ and $b \in \mathbb{R}^C$ are the final weights and biases. For any input example x , the class activation map for label c at location t is obtained by summing the contribution $w_c^k F_t^k$ of each scalar feature F_t^k to the final score of label c , as described in the following equation:

$$\text{CAM}(x, c, t) = \sum_k w_c^k F_t^k \quad (4.2)$$

The CAM address the aforementioned limitations of the attention scores to fulfil our purpose: in a text classification context, it provides a signed contribution of each word to each class membership tackled by the model. Hence, we chose to use CAMs to extract dense, comprehensive knowledge from a network previously trained on a large set of documents, thus $R = \text{CAM}$. Our CAM-extracted model is illustrated in Figure 4.1.

4.3.2 Predicting fake-like articles

We used this architecture on two classification problems. To begin against fake news, we rely on two well-known press articles datasets to train the model:

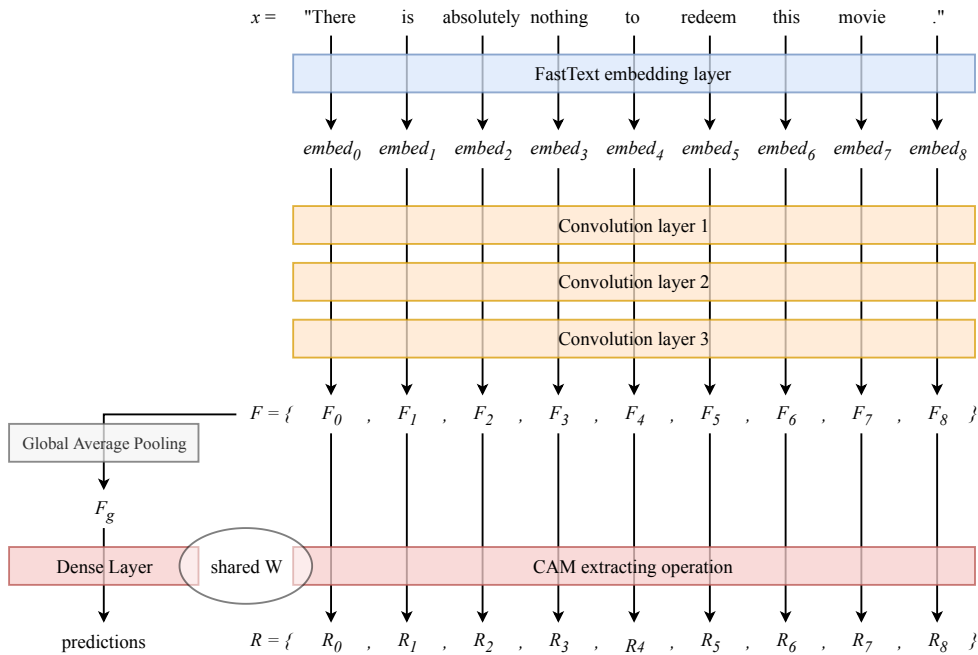


Figure 4.1: Extraction of Class Activation Maps from a tokenized sentence.

- Kaggle fake news⁴ is composed of 13,000 documents in English, tagged as “fake” and either *biased* or *bullshit*.
- Signal-Media⁵ gathers a huge corpus of press articles (in English), deemed to be legitimate. Following [84], we select a random sample so as to balance the classes.

The use of contextualized embeddings such as Bert or Elmo would be a return to the “black box” phenomenon, because they exploit the whole text to embed each token; instead we use the *fasttext* pre-trained embeddings, of dimension 300. The texts are only tokenized (including the punctuation), through the python library NLTK⁶. This pre-trained embeddings imply that the model has to grasp the context. A random train/test split of 20% is used to be able to compute accuracy scores. We adopted an early stop strategy, resulting in a total of 41 epochs.

4.3.3 Secondary task: predicting emotion

Fake news and biased content are often intricate with high emotional values: the intuition is that the emotive impact would replace rational facts to persuade the readers. We propose to train a second model, based on the same architecture, to enrich the analysis and combine “fakeness” predictions with emotive labels. The only modification to the model architecture is to output 5 class labels, to match the task labels. As a training corpus, we retain the dataset Text-emotion⁷, which consists of over 40,000 tweets in English. Even though the document size varies between press articles and tweets, we are confident that our model is sufficiently focused on the sentence patterns, to be able to

⁴<https://www.kaggle.com/mrisdal/fake-news>

⁵<http://research.signalmedia.co/newsir16/signal-dataset.html>

⁶<https://www.nltk.org/>

⁷<https://github.com/tlkh/text-emotion-classification>

tackle this second classification task.

4.4 Experiments: TC-CNN at work

A previous approach in the literature compared various machine learning approaches, highlighting that excellent accuracies could be reached using deep learning and more specifically, a recurrent neural network fed with pre-trained GloVe embeddings [84]. In Table 4.1, we include the score reported in their article. However, as we may have discrepancies (including in the train/test split strategy), we also include our 300d-GloVe-based recoded baseline, *CNN-prev*, which consists of three convolutional layers with kernel size 3, a dense layer, and a global max pooling. We extend the comparison to another broadly used classification framework, *fasttext* [77].

Table 4.1: Classification scores

Task	Classifier	Precision	Recall	Weighted F_1
Fake news	As in [84]	-	-	0.85
	<i>CNN-prev</i>	0.911	0.907	0.907
	<i>fasttext</i>	0.878	0.876	0.871
	<i>TC-CNN-fake</i>	0.918	0.919	0.918
Emotion	<i>Text-emotion</i> ⁸	-	-	0.62
	<i>fasttext</i>	0.597	0.600	0.590
	<i>TC-CNN-emo</i>	0.686	0.688	0.683

From Table 4.1, we can observe that our model performs equivalently or relatively better than the previous CNN-based architecture and than *fasttext*, while conserving a reasonable network size and more importantly, while bringing a huge gain: interpretable results.

A recurrent objection to fake news classification is the risk to overfit the classifier on a specific dataset, overfocusing on a few words. This problem is particularly hard to analyse, because of the scarce amount of similarly-labelled, thematically different datasets. The introduced model presents an advantage to investigate this point, through the class activation mappings. We propose to compare an example present amongst the biased articles of the Kaggle dataset in Figure 4.2, typical of the 2016 fake news: it deals with the Clinton affairs. “Clinton” being a recurrent target of disinformation campaigns, is likely to be determining in the output of the classifier. Figure 4.2 (upper) shows the activation values, in blue (positive) and orange (negative) for the class *bias*, which is the predicted class for this text. The classifier provides an interpretable output, which is the weight of each token in the prediction; we propose some hints about the presence of such weights, using red boxes. To begin with, the model has learnt a vocabulary: James Comey and Hillary Clinton were present in a number of biased articles, thus highlighted in blue. The model also learnt patterns: the abusive use of quotes every few words suggest bias. Precise information about places and dates are highlighted in orange, suggesting non-bias (thus looking more legitimate).

By replacing all family names by more neutral ones, we can see the differences in Figure 4.2 (lower): “Smith” is not as strongly linked to the *bias* class as “Clinton” (notably in the phrase “Hillary Clinton’s corruption”, in the last

Explaining class: bias

The Clinton email scandal has taken an unexpected twist Friday as Federal Bureau of Investigation Director James Comey notified key members of Congress that the agency will be reopening their investigation against former Secretary of State Hillary Clinton. In a letter to Congress, Comey wrote that the FBI has recently learned of the existence of emails that appear to be pertinent to the investigation regarding Clinton's use of a private server during her tenure at the State Department. While Comey did not elaborate on what those emails contain, the director that the emails were discovered in connection with an unrelated case. Via FoxNews He told lawmakers the investigative team briefed him on the information a day earlier, and I agreed that the FBI should take appropriate investigative steps designed to allow investigators to review these emails to determine whether they contain classified information, as well as to assess their importance to our investigation. He said the FBI could not yet assess whether the new material is significant and he could not predict how long it will take to complete this additional work. Trump, speaking to cheering supporters Friday afternoon in Manchester, N.H., praised the FBI for having the "courage" to right the horrible mistake that they made – saying he hopes that is corrected. Hillary Clinton's corruption is on a scale we have never seen before, Trump said. We must not let her take her criminal scheme into the Oval Office. In a nod to the significance of the FBI's announcement, Trump quipped: The rest of my speech is going to be so boring. We will continue to update as new details surface.

Explaining class: bias

The Smith email scandal has taken an unexpected twist Friday as Federal Bureau of Investigation Director James Fitz notified key members of Congress that the agency will be reopening their investigation against former Secretary of State Mary Smith. In a letter to Congress, Fitz wrote that the FBI has recently learned of the existence of emails that appear to be pertinent to the investigation regarding Smith's use of a private server during her tenure at the State Department. While Fitz did not elaborate on what those emails contain, the director that the emails were discovered in connection with an unrelated case. Via FoxNews He told lawmakers the investigative team briefed him on the information a day earlier, and I agreed that the FBI should take appropriate investigative steps designed to allow investigators to review these emails to determine whether they contain classified information, as well as to assess their importance to our investigation. He said the FBI could not yet assess whether the new material is significant and he could not predict how long it will take to complete this additional work. Wesson, speaking to cheering supporters Friday afternoon in Manchester, N.H., praised the FBI for having the "courage" to right the horrible mistake that they made – saying he hopes that is corrected. Mary Smith's corruption is on a scale we have never seen before, Wesson said. We must not let her take her criminal scheme into the Oval Office. In a nod to the significance of the FBI's announcement, Wesson quipped: The rest of my speech is going to be so boring. We will continue to update as new details surface.

Figure 4.2: Examples of CAMs explaining a bias about Clinton: original (top), with replacements (bottom).

red box). However, the sentence structure itself is still determining, enabling the classifier to maintain its prediction and recognize a bias: as an example, the model appreciates as factual (here in orange) to fully give the title and name of a person.

4.5 Measuring offences and biases in the press

4.5.1 Sources of 2020 data

We have constituted three news article datasets: *news_outlets*, contains a random sample of 30 articles from each of seven world-level English press references: CNN, MSNBC, FoxNews, The Guardian, BreitbartNews, The Daily Express and The BBC. In a separated *the_onion* dataset, an eighth media outlet is also considered: The Onion. We keep it segregated, because of its satirical nature: this outlet does not aim to propagate news, but humour. The included articles have been redacted in January and February 2020.

The dataset *gab_trends* has been collected by downloading the proposed links from a news aggregator, "Gab Trends". Gab is a microblogging platform (Twitter-like), US-based, that brands itself as championing free speech: some hosted content would be banned on mainstream platforms. This dataset contains 1135 articles from a total of 150 different web domains. We crawled it from the 15th of December 2019, to the 15th of February 2020.

While *gab_trends* indeed contains well-known news outlets such as the BBC, the Daily Mail and Fox News, it also refers to other websites, which do not enjoy the same journalistic quality. Breitbart is the most cited (over 200

Table 4.2: Most frequent words, excluding stopwords, on *news_outlets*

Prediction	Words	Nb Docs
bias	Democrats, House, time, –, com, Iran, like, president, U, 2020, President, also, one, would, people, —, said, Trump	861
bullshit	campaign, Court, v, appointee, Pelosi, American, also, Senate, Louisiana, one Republican, trial, two, News, CNN, people, Democrats, voted, would, witnesses court, House, president, case, Donald, vote, President, impeachment, abortion, said, Trump	30
factual	first, two, like, —, told, Trump, time, year, also, one, would, people, ”, “, said	484

times), followed by YouTube (which may refer to any type of content); ZeroHedge (150) and InfoWars⁹ (cited less than 40 times) are at least controversial.

4.5.2 Qualitative analysis on a sample

A first deep dive in both the *news_outlets* and the *gab_trends* corpora consists in genuinely looking at the most frequent words for each predicted label, as indicated in Table 4.2. As a classic frequency count retrieves generic words, we decided to hide the stop words. A first observation is the dominance of politics related terms: “Trump” is the most frequent person in all three classes. The bullshit class does not have the same importance as the two others, because of the scarce amount of articles that were tagged this way.

A second opportunity is offered, thanks to the CAMs, to display the words that trigger each class of our classifier. In Figure 4.3, we retain the words with class-relative-CAM highest values (coloured words in the middle) and their immediate text for three documents predicted in this class. The second biased example is typical, with the reference to “mainstream media”; the bullshit class seems attracted by Twitter *follow me* requests. On the legitimate articles side, the first example is very well documented (title, name and date); the second illustrate the focus on precise dates; the last suggest the presence of a given media outlet in the training dataset.

4.5.3 Statistical overview

For the three datasets, the proportion of each emotion and hate presence is presented in Figure 4.4. On the emotion side, the majority is sad. While the “neutral” state would have been expected, we believe that the semantic fields around politics, geopolitics and economics are globally closer to sadness. We conserved the original taxonomy for the emotions, thus the emotion “hate” refers to anger, while the hate label “Hate” refers to the presence of at least one hateful or offensive sentence in the article. A continuously high level of hate in a whole article is somehow improbable. Gab Trends proposes higher presence rates of either happy or hateful articles. The Onion somehow manages its goal with a sadness level comparable to serious newspapers: we confess that our emotion detector does not spot humour. On the hate side, only a handful of articles are marked as “hateful”, all in our *gab_trends* dataset. They do not go

⁹“InfoWars is a far-right American conspiracy theory and fake news website”, <https://en.wikipedia.org/wiki/InfoWars>

LABEL is: 0 bias

among these are Life , Liberty and the pursuit of Happiness , " Thomas Jefferson wrote in the Declaration of Independence : Again , all people are created equal . Black people should not be put in a special class

counts . The blocking feature works in a similar way to an ad blocker and allows users to choose specific mainstream media outlets they want to remove from their search results and recommendations or alternatively block all the mainstream media

Anyone who took High School economics remembers the names Milton Friedman and John Maynard Keynes . And anyone who majored in Economics would have had those names seared into

LABEL is: 1 bs

Rick Scott picks the wrong adjective to defend Trump on scandal On Ukraine scandal , Rick Scott keeps praising Trump ' s " transparency . " I don

as a juror and listen to the evidence and vote in accordance with the evidence : " Follow Pam Key on Twitter @ pamkeyNEN

working for them in trying to work with this president to get things done . " Follow Jeff Poor on Twitter @ jeff _ poor

LABEL is: 2 legit

but acts of vandalism , like graffiti or swastikas scrawled on places like synagogues , Police Commissioner Dermot Shea said in September , when he was the department ' s chief of detectives ; But anti - Semitic incidents are

18 , 000 refugees can be resettled in the U . S . between October 1 , 2019 ; and September 30 , 2020 . This is merely a numerical limit and not a goal federal officials are supposed to

jungle with tensions having built so high that it is going to explode at some point ; " Visit Business Insider ' s homepage for more stories . Former Vice President Joe Biden is facing increased scrutiny over his record

Figure 4.3: Words with highest CAMs, in their text

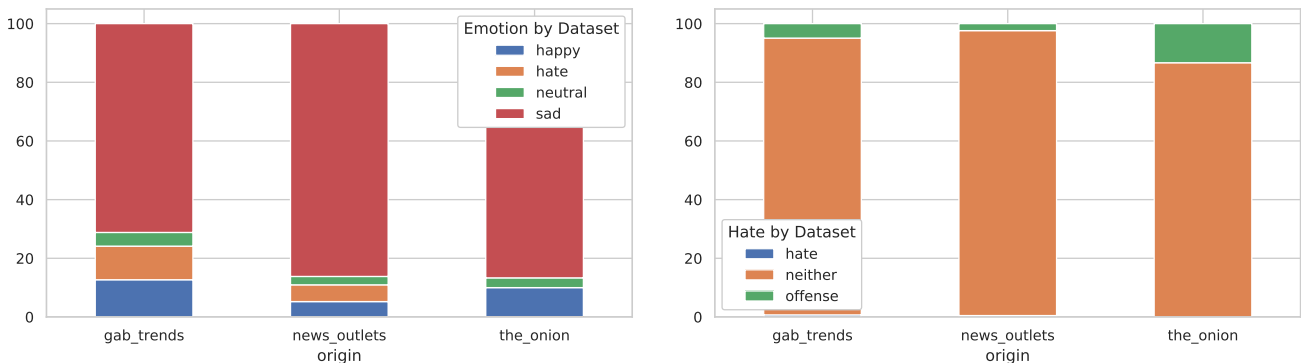


Figure 4.4: Comparing the datasets, by emotion (a), by presence of hate (b)

beyond clichés about demographics (white people share among the US population), or about feminism.

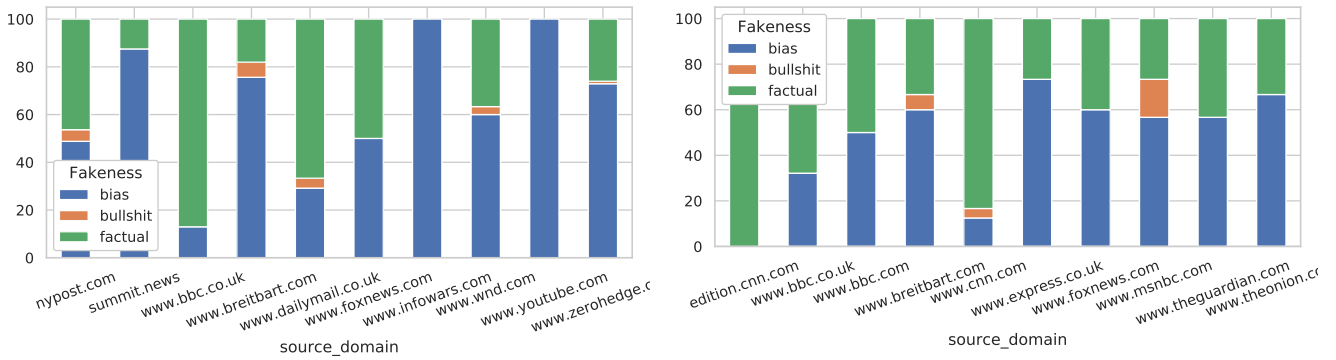


Figure 4.5: Article quality by source for *gab_trends* (a), for *news_outlets* and *the_onion* (b)

The proportion of biased articles is grouped by dataset in Figure 4.5: for the articles shared on *Gab_trends* (a), and for our selection of news outlets (including *The Onion*) on (b). As the referred article sets are different, the scores of fakeness can vary for a same source present in both (a) and (b). On the left side, our model identifies the classic newspapers (Nypost, BBC, Daily Mail and Fox) even though they cover a variety of styles: as an example, the N.Y. Post is a tabloid. The model is however quite critical towards far-right and/or conspiracy outlets *summit*, *wnd* (WorldNetDaily) and *infowars*, whose content is predicted as biased. On the right, CNN and the BBC are split in two domain names: their writing styles are globally tagged as factual journalism. Fox News is also present and reasonably factual. Our selection of Breitbart articles results more factual than Gab’s selection.

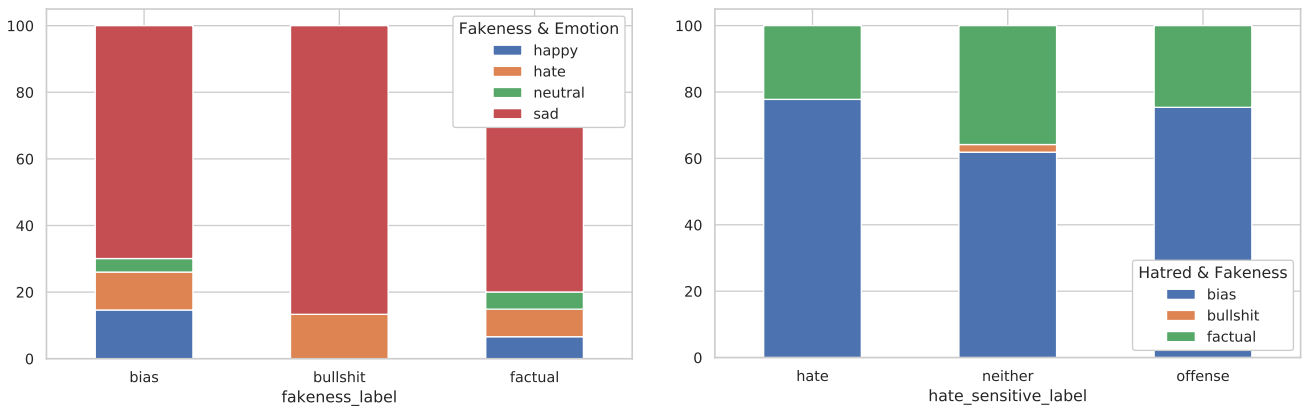


Figure 4.6: Emotion (a) and hatred (b) compared to fakeness, on the union of the three datasets

An interesting point is shown in Figure 4.6, comparing the repartition of dominant emotion along the fakeness (a), and comparing hatred and fakeness (b). Here, we have aggregated the three datasets. It appears that classic factual articles are expected to be sad. A second recurrent trend for biased articles is to bring either happiness or anger. On the right-hand side, the figure shows the distribution of hatred with regard to the fakeness. Let us note that most of the articles are not hateful, represented in the “neither” column. We observe that hate and offence are most likely found in biased articles.

4.6 Conclusion

Fake news and information manipulation detection has longly been a difficult, intricate problem, which needs to face many scientific challenges. While automatic fact-checking always requires a set of trusted sources of truth, content quality can be learnt offline. In this chapter, we followed our intuition to build generic content evaluation tools, to help the readers understand what is aimed by the writers and see the lack of references and facts. We think that fake news cannot be seen only as a classification problem: how could one trust an algorithm when one does not even trust newspapers? The role of the algorithm and the weight to give to its prediction has to be well thought, as the aim is to detect bias, not to propagate one. In this regard, our method displays a great advantage, thanks to its interpretability: the reader sees why and what triggered the prediction. We have also combined this spectrum of analysis with related tasks of hate and emotion detection.

The case study conducted on recent news articles, redacted at the beginning of 2020, gives hints about the current mixing of factual and persuasive articles on the media landscape, on “mainstream” and “non-mainstream” news outlets. Notably, the studied “alternative media” mixes high- and poor-quality articles and outlets, equally presenting valuable sources (the BBC) and scam websites (infowars). As part of future works, we aim to aggregate more content evaluation modules to enable a holistic analysis of fake content propagation on social media.

A combination of the TC-CNN approach with a vagueness analysis technique is further developed and presented in the following Chapter.

Chapter 5

Combining Vagueness Detection with Deep Learning to Identify Fake News

Abstract

In this chapter, we combine two independent detection methods for identifying fake news: the algorithm **VAGO** uses semantic rules combined with NLP techniques to measure vagueness and subjectivity in texts, while the **TC-CNN** classifier introduced in Chapter 4 relies on Convolutional Neural Network classification and supervised deep-learning to classify texts as biased or legitimate. We compare the results of the two methods on four corpora. We find a positive correlation between the vagueness and subjectivity measures obtained by **VAGO**, and the classification of text as biased by **TC-CNN**. The comparison yields mutual benefits: **VAGO** helps explain the results of **TC-CNN**, conversely **TC-CNN** helps us corroborate and expand **VAGO**'s database. The use of two complementary techniques (rule-based vs data-driven) proves a fruitful approach for the challenging problem of identifying fake news.

5.1 Introduction

The computational verification of textual claims is motivated by the proliferation of misinformation on the Web and online resources. Such methods are needed because human verification does not scale up to the diffusion speed of online misinformation, which spreads up to six times faster than real news [187]. It is also worth mentioning that up to 50% of the shares of viral claims happens within a few minutes after their appearance [200].

Existing fake news detection methods have focused on verifying textual claims given reference information, based on corpora of previously checked claims [177, 161]. Some approaches exploit the thousands of reports written by journalists in fact-checking organizations to automatically verify if a claim is true or not. Other works use textual documents as trusted information to be analysed in order to validate claims. Recently, there have been new methods to verify claims using reference-structured data, for example by adapting transformers to let them model datasets [24, 63].

In this chapter, we propose to exploit lexical vagueness as a cue for the automatic identification of fake news conveying biased information. The motivation is twofold: vague claims are less sensitive to factual refutation than precise claims [40]. Moreover, vague terms are more prone to subjective interpretation than precise terms, particularly in the adjectival domain [80]. On the technical side, we use and compare two independent detection methods for identifying fake news. The first, deployed in the so-called VAGO algorithm, detects and measures vagueness and subjectivity in texts using a semantic-based approach combined with NLP techniques. The second, deployed in the TC-CNN classifier, relies on a Convolutional Neural Network to classify texts as biased or legitimate. The motivation behind this comparison is to investigate whether VAGO's and TC-CNN's respective classification results actually converge, at text level and word level. Our aim is thus to confront and to relate a semantic method with a deep-learning method and to pool their results.

The rest of the chapter is organized as follows. In section 5.2, we present the expert-based approach of VAGO. We first motivate our emphasis on vagueness by explaining the way in which some vague expressions convey subjectivity. We present a typology of vague terms used by VAGO to classify corpora as being either *vague* or *precise*, *opinion* or *factual*, as well as simple scoring rules used to implement an online text analyser.

In section 5.3, we compare the categorization by VAGO between *vague* or *precise*, *opinion* or *factual*, with the classification of news articles by TC-CNN between *legitimate* or *biased*. In terms of explainable AI, we expect that doing so will help us see more clearly into the determinants of the TC-CNN classification.

Finally, in Section 5.4 we discuss the opportunities and gains in terms of explainability of mixing expert-based systems with deep-learning, data-driven classifiers.

5.2 Vagueness detection: VAGO

5.2.1 Vagueness and subjectivity

Vague words are expressions whose meaning is indeterminate and compatible with an open-ended range of possible interpretations [154]. Typical examples of vague words include gradable adjectives like “tall”, “rich”, “intelligent”, whose extension is left open by the speaker [86]. When hearing “John is tall”, the listener generally cannot infer a precise representation of John’s height, but only a set of more or less probable values [99]. This is different if we hear the precise sentence “John is exactly 187 cm tall”, which eliminates more possibilities.

There is no direct relation between vagueness and either truth or falsity [40]. An utterance with precise truth conditions can perfectly be false (“the Eiffel Tower is 96 meters high”). Conversely, a vague sentence can be judged uncontroversially true (“the Eiffel Tower is a tall building”). As noted in [154], however, a vague sentence has higher chances of being judged true than a precise one, because it is compatible with more possible states of affair. Relatedly, vague expressions are generally seen as more subjective than precise expressions, because both speaker and hearer can interpret them in different ways [184]. Thus, a large class of gradable adjectives is described as subjective, or even as evaluative [87, 121, 170, 80].

In principle, a seasoned liar could use only precise language to create fake news. But it may be harder to ensure coherence, and it may thereby add up to the cognitive cost of having to make up a story (see [185]). In contrast, using vague language can be a cheap way of making one’s utterances easier to accept, because less sensitive to factual refutation [40]. Importantly, vague language is also used cooperatively to minimize error and to communicate uncertainty [182, 40, 39]. But an overwhelming reliance on vague language may signal that the discourse is possibly less factual, and more prone to bullshitting or to the spreading of biased information.

5.2.2 A typology of vague expressions

To detect vagueness and subjectivity in texts, IJN and Mondeca have developed a lexical database and associated algorithm called VAGO. The inventory relies on the typology proposed in [40], which distinguishes four types of lexical vagueness: approximation (V_A), generality (V_G), degree-vagueness (V_D), and combinatorial vagueness (V_C).

Expressions of approximation here include mostly modifiers such as “around”, “about”, “almost”, “nearly”, “roughly”, which loosen up the meaning of the expression they modify (compare “around 10 o’clock” and “10 o’clock”). Expressions of generality include determiners such as “some” and modifiers like “at most”, “at least”. Unlike the former, these expressions have precise truth-conditions, but fail to give a maximally informative answer to the question “how many” (compare “some students/at least three students left” to “eleven students left”).

The class of degree-vague and combinatorially vague expressions includes mostly one-dimensional adjectives on the one hand (like “tall”, “old”, “large”) and multidimensional adjectives on the other (“beautiful”, “intelligent”, “good”, “qualified”), which combine several dimensions or aspects of comparison. The opposition between degree-vagueness

and combinatorial vagueness is adapted from [6]. In Alston’s approach, combinatorial vagueness concerns not just adjectives but also common nouns and verbs. The class of degree-vague and combinatorially vague expressions consists mostly of adjectives in VAGO (see Section 5.3), knowing that in the case of adjectives, the degree vs combinatorial distinction is congruent with the distinction proposed by Kaiser and Wang between simple-subjective vs complex-subjective adjectives [80].

Thus, we assume that subjectivity is introduced foremost by expressions of type V_D and V_C . Two competent speakers can have a non-factual disagreement whether someone is “tall” or “old”, regardless of the existence of common scales of physical measurement, and even as they share the same knowledge of height or age [87]. And the more dimensions attached to an expression, the more disagreement is expected to arise between competent speakers, as evidenced in the case of evaluative adjectives like “beautiful” or “smart” for which no standard scale of measurement is available [121]. By contrast, we assume that expressions of type V_A and V_G give rise to no subjectivity, or at least to limited subjectivity, since the interpretation of these expressions is number-relative and less context-sensitive (“roughly 20” is relative to the precise value 20, unlike “tall”, and similarly “some students” literally means “more than 0 students”).

As a result, expressions of type V_A and V_G are treated as factual expressions, and expressions of type V_D and V_C as subjective expressions. This means that , the class of vague expressions is not co-extensional with the class of subjective expressions, but forms a superset.

5.2.3 VAGO: detection and scoring

In VAGO, vagueness and subjectivity are detected and scored in a bottom-up manner, from words to sentences and then to pieces of larger texts. For a given sentence, its vagueness score is defined simply as the ratio of vague words to the total number of words in the sentence:

$$R_{vague}(\phi) = \frac{\overbrace{(|V_G|_\phi + |V_A|_\phi)}^{factual} + \overbrace{(|V_D|_\phi + |V_C|_\phi)}^{subjective}}{N_\phi} \quad (5.1)$$

where N_ϕ designates the total number of words in the sentence ϕ and $|V_G|_\phi$, $|V_A|_\phi$, $|V_D|_\phi$ and $|V_C|_\phi$ respectively denote the number of terms of each of the four types of vagueness (generality, approximation, degree-vagueness and combinatorial vagueness).

Similarly, the subjectivity score of a sentence is calculated as the ratio of subjective expressions to the total number of words in the sentence.

$$R_{subjective}(\phi) = \frac{|V_D|_\phi + |V_C|_\phi}{N_\phi} \quad (5.2)$$

Both the vagueness score and the subjectivity score of a sentence vary between 0 and 1. When a sentence contains

at least one vague term, the degree of vagueness of the sentence is non-zero, $R_{vague}(\phi) > 0$. When $R_{vague}(\phi) = 0$ this implies that the sentence does not contain vague vocabulary, and similarly for subjectivity.

For sets of sentences, the vagueness and subjectivity scores can be defined as the proportion of sentences with non-zero vagueness and non-zero subjectivity scores respectively. More fine-grained measures can be proposed, but these scores suffice to characterize the prevalence of each feature. Thus, if T is a text and N_T denotes its total number of sentences, then:

$$R_{vague}(T) = \frac{|\{\phi \in T | R_{vague}(\phi) > 0\}|}{N_T} \quad (5.3)$$

$$R_{subjective}(T) = \frac{|\{\phi \in T | R_{subjective}(\phi) > 0\}|}{N_T} \quad (5.4)$$

5.2.4 VAGO Implementation

VAGO’s back-end is built on top of the popular GATE [33] framework for NLP content processing. It also leverages the semantic content annotator CA-Manager [26], a semantic-based tool for knowledge extraction from unstructured data. The current pipeline automatically detects the language of the corpus (English or French) using TexCat.¹

The back-end uses a workflow composed of different analysis engines, combining NLP processing from GATE and semantic technologies with a UIMA-based infrastructure.² UIMA architecture allows enriching and customizing engines to address specific needs in the application. VAGO is also available through HTTP REST API.

5.2.5 Online tool

The online tool VAGO, available from Mondeca’s website,³ provides a graphical interface for the representation of these scores using two barometers. A first barometer represents the degree to which a text is vague or precise. The second one indicates the degree to which the text reports an opinion or is factual (the proportion of subjective vs objective vocabulary). A section of detailed results explains for each sentence the vague triggers detected and the corresponding category (V_X).

The online tool uses an anonymous profile, with some restrictions to call the service such as a limit number of characters as input (e.g., 1200), and some predefined short sample texts. The online application automatically detects the language of the text entered by the user (French or English).

As a toy example, consider the text $T = \{\text{“Most sensational news articles are sometimes hard to believe”, “Two plus two equals four”, “Mary left Paris around 2pm”}\}$, comprised of three sentences. T contains five vague terms (“most”, “sensational”, “sometimes”, “hard”, “around”), with “sometimes” and “most” instantiating generality, “sensational” and

¹<https://www.let.rug.nl/vannoord/TextCat/index.html>

²Unstructured Information Management Architecture (<http://uima.apache.org>)

³<https://research.mondeca.com/demo/vago/>

“hard” pertaining to combinatorial vagueness, and “around” to approximation. While the sentence “Two plus two equals four” is precise, the other two are vague. Moreover, “Mary left Paris around 2pm” is only factual but “Most sensational news articles are sometimes hard to believe” contains both factual and subjective terms. The barometers reporting the proportion of vague and subjective sentences in the text T are illustrated in Figure 5.1.

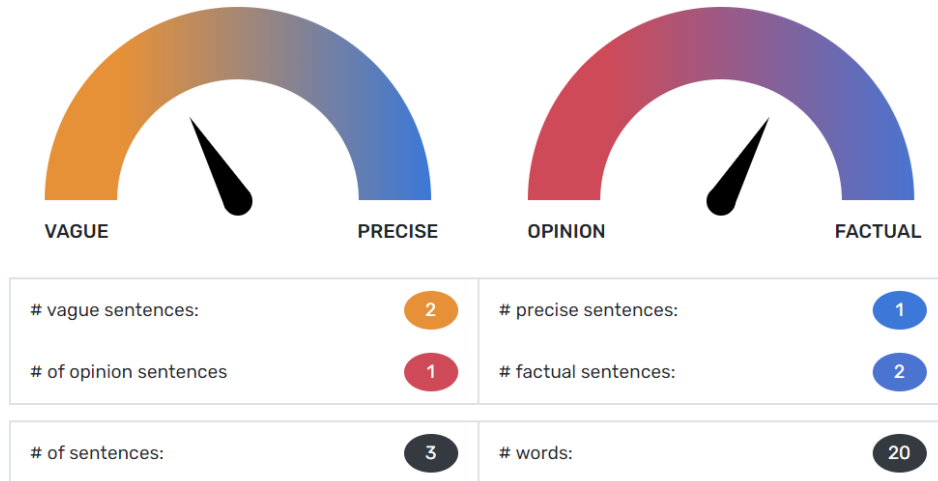


Figure 5.1: Results summary for text with three sentences

5.3 Comparison and combination

Let us now jointly analyze the classification results obtained by TC-CNN and the measures produced by VAGO.

5.3.1 Validation of TC-CNN on unseen data

The architecture of TC-CNN is described in section 4.3. To train this architecture to detect “fake news” like articles, we rely on well-known press articles datasets. **Kaggle fake news**⁴ is composed of 13,000 documents in English, tagged as “fake”: either *biased* or *bullshit*. **ISOT**⁵ contains 21,417 “true” and 23,481 “fake” articles, subsequently referred to as ISOT-True and ISOT-Fake. **SignalMedia1M**⁶ gathers a huge corpus of press articles (in English), deemed to be legitimate [84]. From this resource, we select two random samples:

- 15,064 articles to train the classifier in a balanced manner,
- 20,071 other articles for the evaluation part.

On Kaggle and ISOT, a random train/test split policy is applied (80%/20%). The resulting datasets are then aggregated with their SignalMedia1M complement.

⁴<https://www.kaggle.com/mrisdal/fake-news>

⁵<https://www.uvic.ca/engineering/ece/isot/datasets/fake-news/index.php>

⁶<http://research.signalmedia.co/newsir16/signal-dataset.html>

Figure 5.2 presents the measure of the model performance with respect to the four corpora. The y-axis is the output score; 0 means “*informative*” while 1 stands for a “*biased*” content. The x-axis shows the source corpora. The letter-value plots⁷ illustrate the distribution of articles that were part of the test dataset (20% of total data, excluded from the training set).

ISOT true and ISOT fake are two facets of the ISOT dataset. Informative or legitimate articles, labelled as “true news”, and biased articles, labelled as “fake”, are very well recognized and separated. The same trend is also remarkable in separating SignalMedia1M and Kaggle’s datasets, even though the latter shows a fuzzier distribution. Overall, the classification performance on these test datasets results in an F1-score = 0.955, indicative of high accuracy.

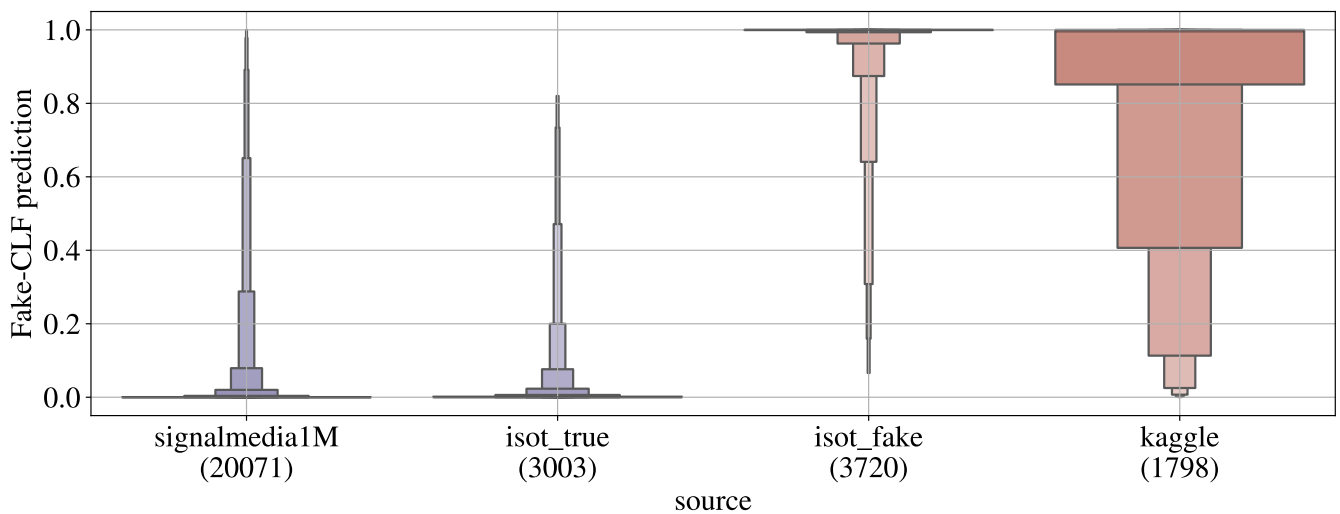


Figure 5.2: Classification of source corpora in the *test* dataset as *informative* (0) or *biased* (1) by TC-CNN.

5.3.2 VAGO experimental settings

The lexicon used by VAGO at the time of this experiment consists of 1,527 English terms, and 1,1150 French terms (version of the database of 12/22/2020) [12]. The English lexicon includes 95% of adjectives, all in the classes V_D and V_C , the whole vocabulary being distributed as follows: $|V_A| = 8$, $|V_G| = 14$, $|V_D| = 35$ and $|V_C| = 1,470$. Hence, 96% of the English lexicon consists of V_C items. The French lexicon includes 97% of adjectives, again all in the categories V_D and V_C . In French, we have $|V_A| = 7$, $|V_G| = 15$, $|V_D| = 28$ and $|V_C| = 1,100$. Accordingly, 96% of the French lexicon consists of V_C vocabulary. Although the corpora used contained mostly English sentences, they included some French excerpts, which were included for analysis. Hence, part of the French lexicon was used.

⁷The widest box stretches between the first and last quartiles, the second-widest boxes stretch between the first/last quartile and the first/last octile, the third-widest boxes stretch between the first/last octile and the first/last hexadeciles, and so forth [68].

5.3.3 Comparison and correlation of VAGO and TC-CNN

We measure the relation between vagueness and how many articles are predicted to be manipulative by TC-CNN by looking at two classes of vague items detected by VAGO, including: all vague items ($V_A + V_G + V_D + V_C$), and subjective vague items ($V_D + V_C$). With regard to these two classes, Figure 5.3 displays the score distributions of texts detected by TC-CNN as legitimate (in blue) and biased (in red).

The relation between articles predicted to be manipulative and the presence of vagueness is measured in Figure 5.3 a), using VAGO as our detector of vague sentences, and the vagueness measure defined in Eq. (5.3). On the left-hand side, the x-axis shows the TC-CNN predicted class (legitimate in blue, biased in red). The y-axis displays the percentage of sentences containing markers of vagueness (the higher, the more vague). The Pearson correlation coefficient between these two variables (biased, and vague) is 0.208: biased articles tend to contain more vague sentences, but this is not the sole determining factor.

A similar analysis is performed using VAGO as a detector of *subjective* sentences, also described as opinion sentences, using the subjectivity measure defined in Eq. (5.4). The results obtained are presented in Figure 5.3 b). Here, the y-axis displays the ratio of sentences that contain V_C and V_D type keywords. In this case, the correlation between these two variables is 0.271: articles detected as manipulative tend to contain more markers of subjectivity.

A split according to the source corpora provides further insights. Regarding subjectivity, Figure 5.4 b) shows that corpora follow the tendency of their legitimate/biased affiliation: ISOT-true and SignalMedia1M texts are detected as less subjective than ISOT-fake and Kaggle texts. While ISOT-true documents and ISOT-fake documents contain the same proportion of vague terms, all types included (Figure 5.4 a), the difference between them concerns the occurrence of subjective vocabulary (Figure 5.4 b).

5.3.4 Word-level analysis: exploiting the deep to find new keywords

TC-CNN not only provides a prediction at the article-level, but also a word-level contextual score to enable an understanding of its predictions. Figure 5.5 displays the distribution of the average bias (or fakeness score) of the occurring VAGO keywords, as they are perceived by the classifier. Each dot represents the average fakeness score of a vocabulary entry across all its occurrences. For each category, a box plot sums up the score distribution. V_A and V_G words used to identify *vague-factual* sentences are in blue; V_C and V_D words used to identify *vague-subjective* sentences are in red. Gray bars give the scores for other adjectives and adverbs, as a reference, as VAGO contains mostly adjectives. Non-VAGO entries have been tagged as adjectives or adverbs by jointly using TextBlob⁸ and NLTK part-of-speech tagging.

Figure 5.5 gathers the average CAM scores of VAGO and some non-VAGO vocabulary entries. As described in section 4.3, for every token in a sentence the CAM provides a signed score accounting for the contribution of this token and

⁸<https://github.com/sloria/TextBlob>

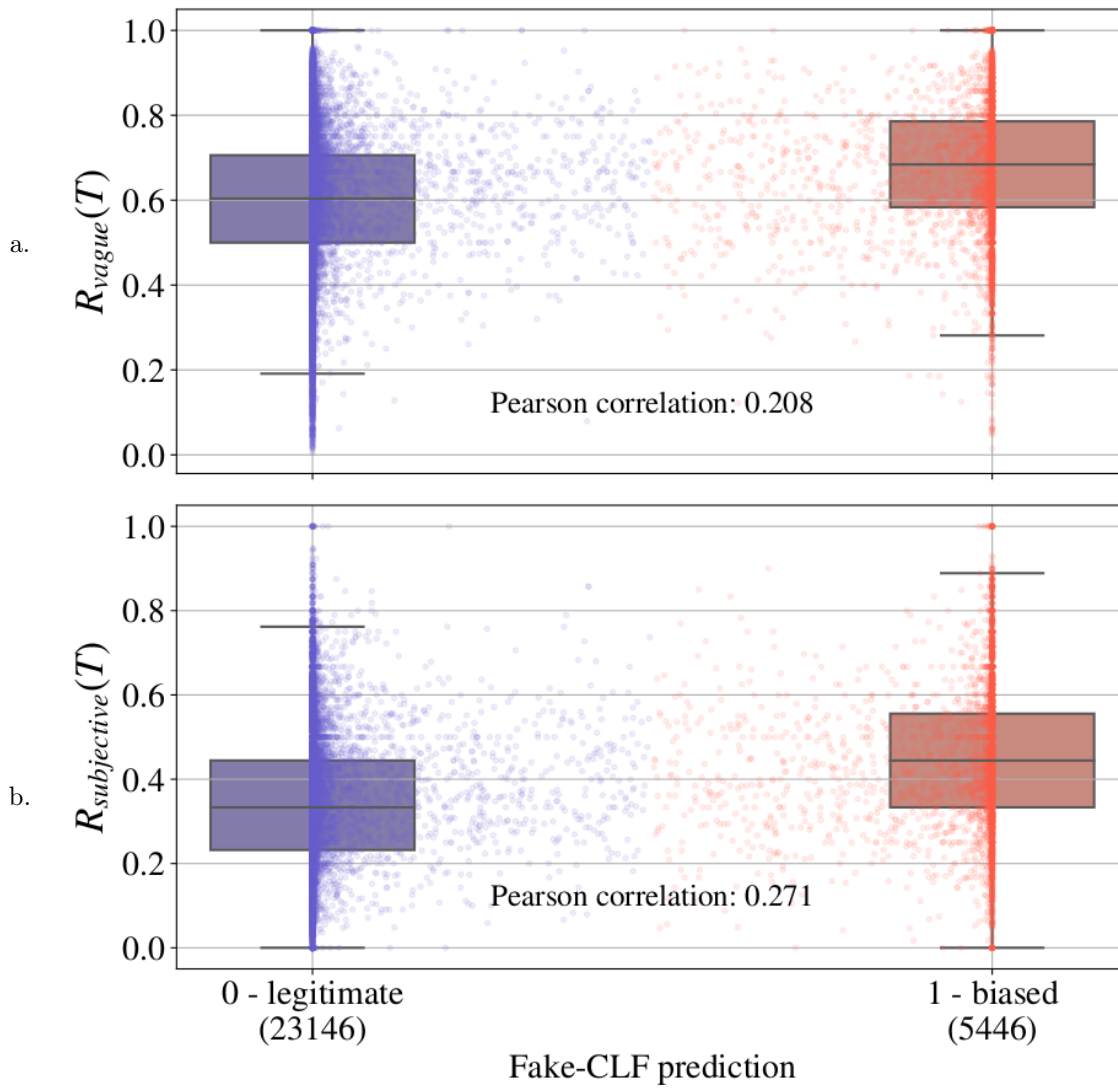


Figure 5.3: Comparison between bias as predicted by TC-CNN and ratio of vague sentences (a), ratio of vague-subjective sentences (b) in texts, as predicted by VAGO.

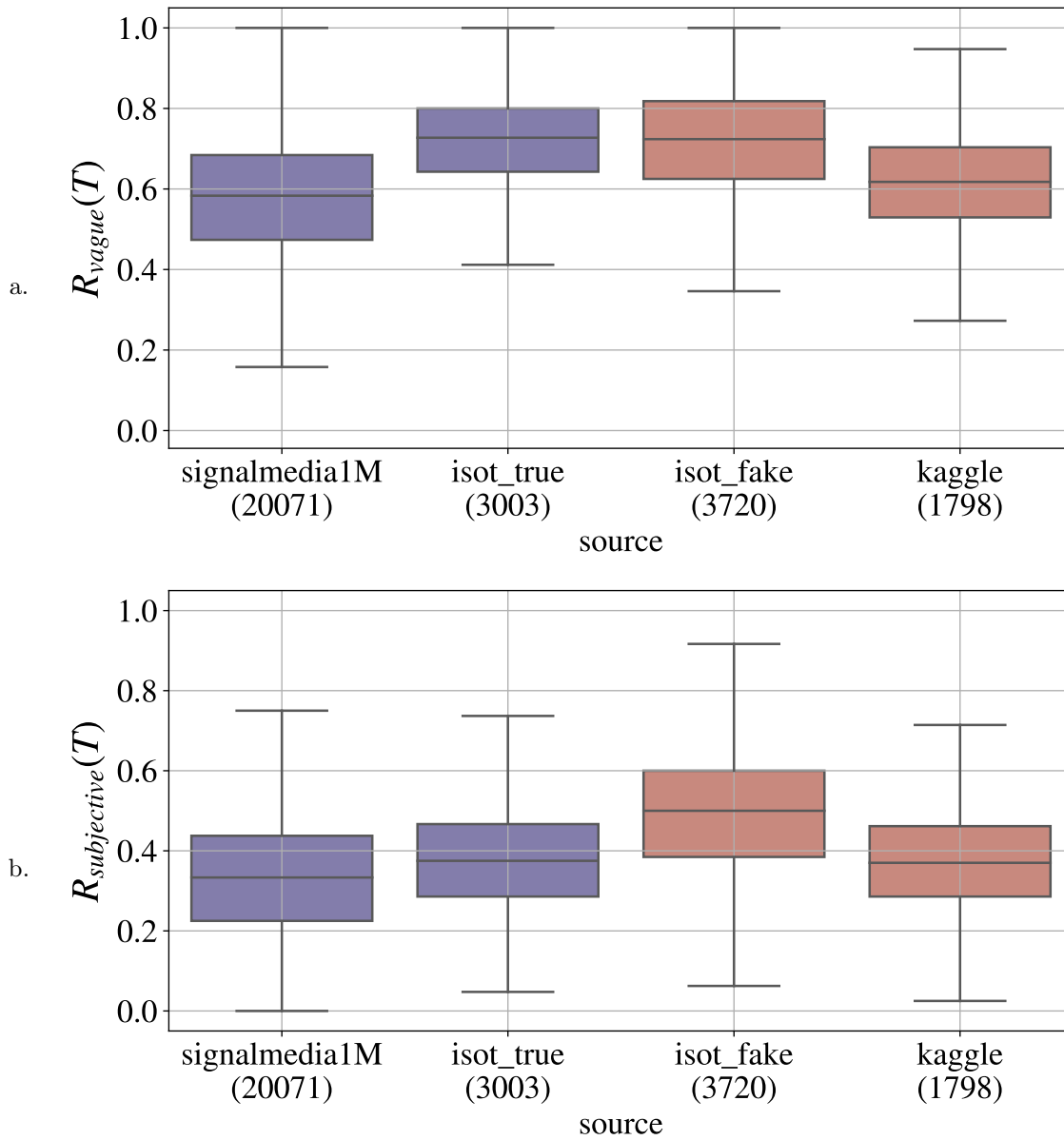


Figure 5.4: Distribution of vague sentences ratio (a), vague-subjective sentences ratio (b) across sources.

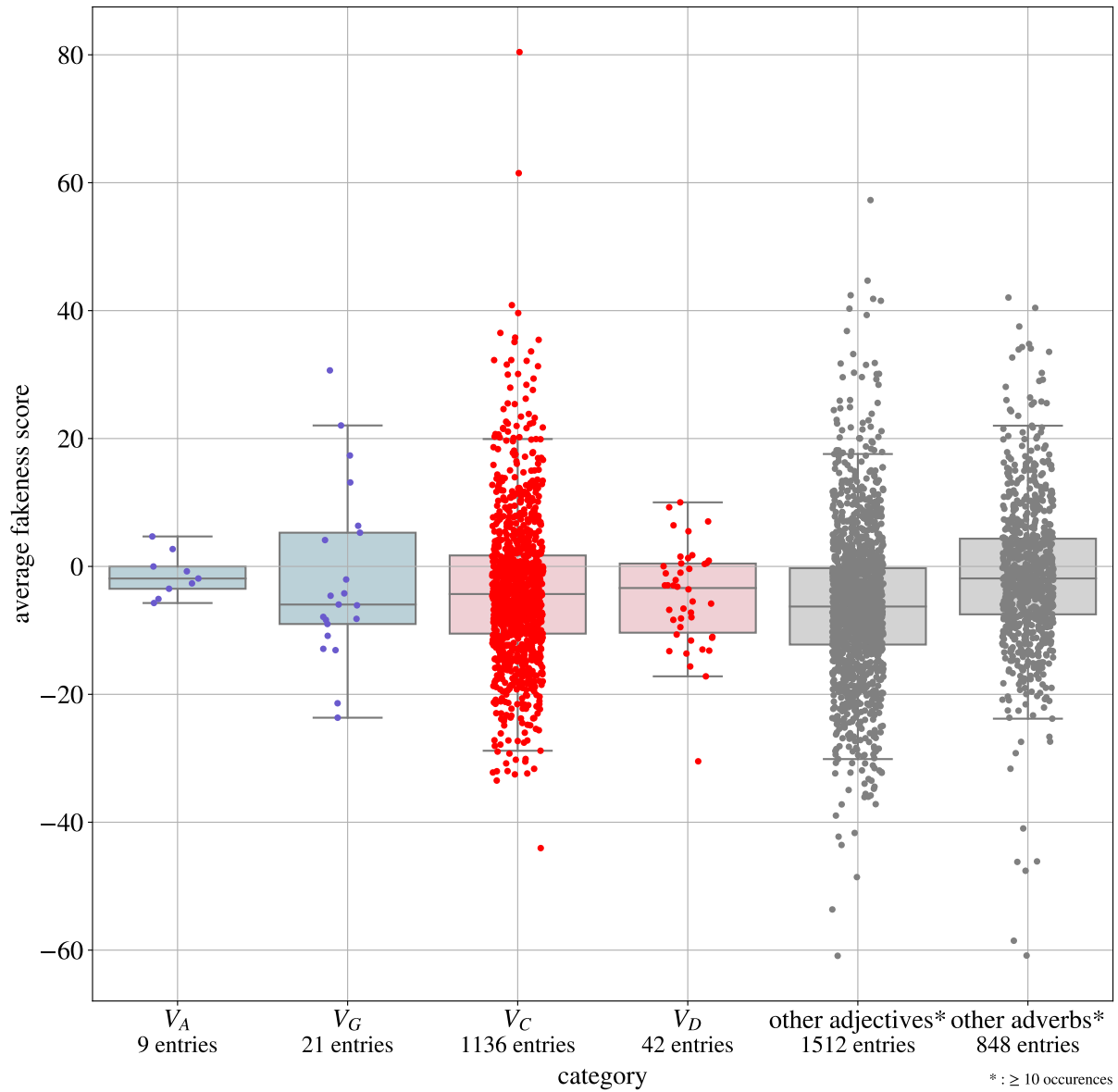


Figure 5.5: Average “fakeness” scores of VAGO entries and other adjectives and adverbs according to TC-CNN class attention maps, defined in Equation 4.2.

its near context to the prediction of the whole sentence. Thus, we consider that a high CAM score stands for a highly bias-inducing word according to TC-CNN. We can see from this figure that belonging to a vague category is not sufficient to trigger a biased or legitimate label; however, some words remain good cues in that respect. In Table 5.1, we propose a list of the thirty most bias-inducing adjectives and adverbs (considering the prevalence of adjectives in VAGO and the proximity of adverbs to adjectives). The first two columns present the number of occurrences of each term and its average “fakeness” score. The third gives the semantic type (when listed in VAGO) and the fourth the syntactic category.

Even though two part-of-speech tagging tools were used to identify “other adjectives” and “other adverbs”, some artifacts (displayed in gray) are wrongly tagged as adjectives or adverbs. Several evaluative adjectives like “disgusting” and “racist” are identified by VAGO as markers of subjectivity; others like “Orwellian”, “sociopathic” and “arrogantly” are not in the VAGO database but are also clear vehicles of subjectivity and evaluativity. In fact, disregarding the two artifacts marked in gray, all terms in Table 5.1 are adjectives or adverbs that fall under the category of combinatorial vagueness and that could be added as such to the VAGO lexicon.

A comparison with least bias-inducing terms (scores ranging $[-60; -30]$, Table 5.2) shows a more heterogeneous set. Fewer terms are already part of the VAGO lexicon. Furthermore, among adjectives and adverbs, several are non-gradable and precise (“40th”, “unbeaten”, “upwardly”, “fortnightly”, “Tanzanian”, “unauthorized”). Fewer correspond to evaluative adjectives or to adjectives that would be eligible for the type V_C (“interactive”, “unrivalled” may be exceptions).

5.4 Discussion

Our hypothesis in this chapter was that because vague terms are often subjective, the prevalence of vagueness in discourse can be used as a cue that the discourse conveys fake news (whether biased, or bullshit). In order to test this hypothesis, we pooled two distinct methodologies. The first methodology involves the semantic-based algorithm VAGO, which provides a measure of vagueness vs precision in text, and a measure of subjectivity vs objectivity in text. The second methodology involves the deep-learning classifier TC-CNN, relying on CNN and CAM techniques.

By comparing the results of TC-CNN with the measures of vagueness and subjectivity obtained by VAGO on four distinct corpora pre-identified as containing “fake” vs “true” articles, we found a positive correlation between vagueness and fakeness, as well as between subjectivity and fakeness. The association with fakeness is stronger for subjective vague terms (of type V_D and V_C in VAGO’s taxonomy). While an overwhelming majority of the terms in the VAGO database are marker of subjectivity, this stronger association also confirms that not all types of vagueness introduce subjectivity.

From a methodological point of view, the two approaches discussed in this chapter can be viewed as complementary. In one direction, TC-CNN identifies as markers of fakeness some adjectives not originally in the VAGO database,

word	occ	avg	VAGO	part-of-speech
sociable	32	80.42	V _C	adj.
disgusting	231	61.50	V _C	adj.
stumble	28	57.28		adj.
Orwellian	13	44.69		adj.
sociopathic	12	42.40		adj.
arrogantly	13	42.04		adv.
misogynistic	37	41.84		adj.
entire	2246	41.54		adj.
idiotic	48	40.86	V _C	adj.
courageously	15	40.43		adv.
neoconservative	25	40.30		adj.
neoliberal	45	39.31		adj.
coincidentally	24	37.52		adv.
subliminal	14	36.81		adj.
delusional	61	35.78	V _C	adj.
pitiful	18	35.43	V _C	adj.
frighteningly	10	34.78		adv.
disturbingly	16	34.30		adv.
shamelessly	26	34.08		adv.
blatantly	75	33.92		adv.
astonishing	100	33.62	V _C	adj.
laughably	11	33.56		adv.
sic	101	33.21		adj.
outrageously	18	32.67		adv.
racist	971	32.27	V _C	adj.
massive	1314	32.14	V _C	adj.
devious	13	31.81		adj.
transnational	52	31.73		adj.
deplorable	85	31.56	V _C	adj.
Siberian	11	31.51		adj.

Table 5.1: Thirty most bias-inducing terms according to TC-CNN, filtered to adjectives and adverbs with at least 10 occurrences. Entries are sorted by descending average CAM scores (avg column). VAGO terms are in bold. Miscategorized adjectives and adverbs are in gray.

word	occ	avg	VAGO	part-of-speech
Carly	16	-60.89		adj.
Emily	72	-60.84		adv.
provisionally	28	-58.53		adv.
40th	32	-53.64		adj.
Experian	40	-48.58		adj.
vSphere	14	-47.59		adv.
upwardly	11	-46.20		adv.
fortnightly	11	-46.13		adv.
cloudy	99	-44.04	V _C	adj.
sectoral	14	-43.56		adj.
unbeaten	113	-42.26		adj.
trimble	19	-41.68		adj.
premiere	57	-40.97		adv.
Sebastian	80	-38.96		adj.
directorial	12	-37.21		adj.
playable	18	-37.18		adj.
Tanzanian	11	-36.07		adj.
treble	19	-36.01		adj.
undertaken	25	-35.82		adj.
interactive	378	-35.57		adj.
semifinal	19	-34.96		adj.
topographic	16	-34.77		adj.
shareable	10	-34.44		adj.
Dorian	10	-33.97		adj.
procedural	92	-33.50		adj.
muddy	46	-33.48	V _C	adj.
unrivalled	13	-33.24		adj.
comprehensive	1001	-32.52	V _C	adj.
moody	146	-32.37	V _C	adj.
unauthorised	14	-32.36		adj.

Table 5.2: Thirty least bias-inducing terms according to TC-CNN, filtered to adjectives and adverbs with at least 10 occurrences. Entries are sorted by ascending average CAM scores (avg column). VAGO terms are in bold. Miscategorized adjectives and adverbs are in gray.

but which clearly belong to the class V_C of adjectives exemplifying multidimensional vagueness, and conveying subjectivity. The correlation found is therefore expected to increase by the inclusion of a larger vocabulary. In the converse direction, the typology deployed in **VAGO** helps to make the results of **TC-CNN** more easily interpretable and explainable. Indeed, while the inductive generalizations operated by **TC-CNN** remain opaque, **VAGO** rests on a transparent semantic architecture. More work is needed to narrow the gap between the two approaches, but the consistency in the correlation between the two classifications is a step toward increased explainability.

5.5 Conclusion

Focusing on lexical vagueness as a marker of subjectivity, we have combined a semantic-based NLP engine **VAGO** with a deep-learning classifier **TC-CNN** to improve the detection of fake news contents. The two approaches yield convergent results, and they each provide useful input to improve the other.

Several points remain for further elaboration. First, the **VAGO** lexicon was still limited at the time of this comparison. We have seen how it can be expanded using **TC-CNN**, but the base will keep evolving. In particular, other expressions beside adjectives (and adverbs) can introduce vagueness and subjectivity. One non-adjective that was included in **VAGO** at the time of this comparison is the modal “should”. “Should” was tentatively classified as V_C considering its expressive dimension in deontic utterances stating moral prescriptions. As it turns out, **TC-CNN** predicts that “should” is not among the most bias-inducing terms when more entries than adjectives and adverbs are considered for analysis. Further work is needed to adjudicate the potential vagueness and subjectivity of “should” and related modals.

Secondly, the correlation found between fakeness scores and subjectivity scores is low, and thus subjectivity explains only part of the variance in the classification of **TC-CNN**. This is not surprising, since subjectivity is only one among several aspects that can contribute to misinformation. Fake news consist not only in opinionated texts, but also in clever fabrications, hoaxes, or simply mistaken factual reports [150]. The type of integration proposed between **TC-CNN** and **VAGO** should therefore be extended to explore other sources of fakeness beside bias.

Chapter 6

Deep Active Learning with Rationales for Text Classification

Abstract

Neural networks have become a preferred tool for text classification tasks, demonstrating state-of-the-art performances when trained on a large set of labelled data. However, in an early active learning setup, the scarcity of the ground-truth labels available severely penalizes the generalization capability of the neural network. In order to overcome such limitations, in this chapter, we introduce a new learning strategy, which consists of inserting in the early stages of the learning process some additional, local and salient knowledge, presented under the form of simulated, human like rationales. We show how such knowledge can be automatically extracted from documents by analysing the class activation maps of a convolutional neural network. The experimental results obtained demonstrate that the exploitation of such rationales permits to significantly speed up the learning process, with a spectacular increase of the accuracy rates, starting from a very reduced number of documents (10-20).

6.1 Introduction

Numerous classification techniques and neural networks in particular have shown to be very effective when large labelled, ground truth data sets are available for training. However, obtaining reliable ground-truth in real-life, user-specific applications may be an extremely expensive, tedious and time-consuming task. Also called *query learning*, *active learning* [181, 8] is a sub-field of machine learning designed to address the scarcity of ground-truth samples in supervised training applications, whether it is caused by the lack of reliable labels or by the progressive exploitation of a stream of unlabelled data. Such learning strategies are mandatory for various applications, including the classification of open source content (online press, social networks, sentiment analysis, etc.) or medical imagery [195, 52]. The underlying principle is the following. Over a sequence of active learning cycles, unlabelled documents are sampled or synthesized, according to criteria dictated by a sampling strategy, and presented to a human, expert oracle for labelling. The labelled documents are then successively added to the training set and the neural network is re-trained with these new examples. However, in a usual active classification setup, the oracle only provides a label information for a given document. Yet, as an expert of the classification task, the oracle holds a more detailed knowledge than solely the class membership.

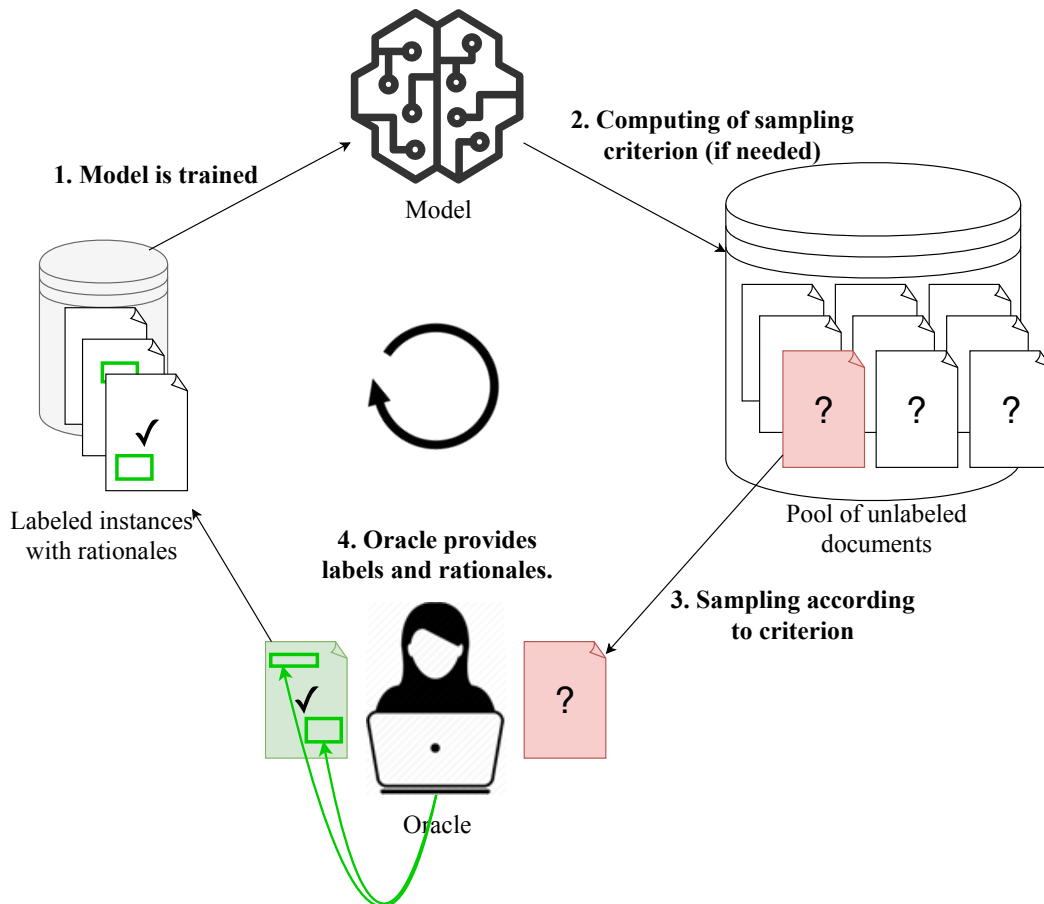


Figure 6.1: An iteration of active learning with rationales in a pool-based scenario

This observation is notably exploited in [199] and [163]. In order to further benefit from the oracle’s expert knowledge, the authors propose to ask the oracle, in addition to a simple label, for rationales that are guiding the labelling process, for text categorization applications. In this scenario, called *active learning with rationales* (Figure 6.1), the oracle has the possibility to specify a set of words in a given document that guided the classification choice. This information is further exploited in order to speed up and enhance, in terms of classification performances, the learning process. In this chapter, we present a method to transfer local rationales to a fully-convolutional neural network dedicated to text classification, within an active learning setting. The rest of the chapter is organized as follows. Related work is presented in section 6.2. In section 6.3, we describe our approach for training a neural text classifier with additional rich spatial knowledge and with sparse, human like rationales. Experimental results are presented and discussed in Section 6.4. Finally, section 6.5 concludes the chapter and opens perspectives of future work.

6.2 Related Work

Text classification tasks have been widely addressed in numerous real applications over the last few years, such as emotions, sentiments and opinion classification [2, 22, 48], language identification [21] and even irony detection [23]. For the exploitation of raw textual data, the classification task is addressed by two main processes: feature extraction and classification technique. Common feature extraction methods are Term Frequency-Inverse Document Frequency (tf-idf)[147], Word2Vec[123], GlobalVectors for Word Representation (GloVe) [138], FastText[77], ELMo [140], BERT[37]. Naive Bayes, Logistic Regression and Support Vector Machines are traditional classification techniques, whereas Convolutional Neural Networks (CNN) [89] and Recurrent Neural Networks (RNN) [28] have been recently increasing in popularity.

The key principle of active learning methods consists in selecting the examples which are the most appropriate for boosting the learning process. Most often a human labeller, called oracle, is involved in the learning process. His mission is to provide the correct labels for a set of sampled examples. Depending on the active learning scenario, samples may come from a static pool, from a continuous stream of unlabelled documents, or they can be even synthesized by generative models [42].

Active learning frameworks are characterized by their sampling strategy, which dictates what examples are presented to the oracle. Some common pool-based scenario sampling criteria that have shown to be efficient are uncertainty [103, 31], information density [105, 159], diversity [19, 41], or query by committee [34]. For evaluation purposes, active learning iterations are simulated by splitting the full-size labelled training set into an initial restrained training set and a larger pool of candidates for sampling. The restrained training set is then progressively enriched during the learning process with samples that satisfy the most the criterion under consideration. The performance of an active learning framework is generally evaluated by: (1) the amount of human labelling that is necessary to reach a

given accuracy score, or (2) examining the model’s accuracy as a function of the size of the training set.

Within the framework of active learning with rationales, to further benefit from the underlying knowledge and understanding of the oracle, in [199, 163] authors propose to ask the labeller, in addition to a simple label, for rationales that are guiding the labelling process, for text categorization applications. In this scenario, the oracle specifies which words are determinant for the classification choice. The influence of the words picked as rationales is then artificially inflated during the classification process. To this purpose, various methods can be employed, including multinomial naïve Bayes, logistic regression or support vector machines. By considering solely sets of words, such approaches do not take into account the local context of the rationales. However, the significance of a word and its relevance with respect to a given category strongly depends on the context of its appearance.

The issue of active learning with rationales has been until now poorly addressed within the context of deep convolutional neural networks (CNNs). In our work, we notably tackle this issue and propose the following contributions: (1) We first show that the structure of a fully convolutional network defines receptive fields that are naturally adapted to automatically detecting groups of successive words that reveal local, contextual and salient knowledge that can be used as rationale for identifying the corresponding category. (2) We show how these rationales can be efficiently transferred from a teacher to a student network within the framework of an active learning process. (3) We finally derive a binarized rationale representation that is resembling with human like rationales and demonstrate its pertinence for active learning objectives. The proposed methodology is described in details in the following section.

6.3 Learning with rationales

6.3.1 Background

This section introduces the notation used throughout the chapter and defines the concepts of dense, comprehensive spatial domain knowledge and sparse spatial domain knowledge.

Active learning without rationales

Let $\mathcal{L} = \{(x, y)\}$ be a set of document-label pairs and $\mathcal{U} = \{x\}$ a set of unlabelled documents. Documents are 1-dimensional arrays of words and punctuation characters. In a C -label multi-label setup, a label y is represented as a vector in $\{0, 1\}^C$. At each active learning iteration, the underlying model \mathcal{M} is trained on \mathcal{L} , then documents from \mathcal{U} are sampled and shown to the oracle for labelling. These newly-labelled documents are removed from \mathcal{U} and added to \mathcal{L} with their respective labels.

Active learning with rationales

Let $\mathcal{L} = \{(x, y, R)\}$ be a learning set of document-label-rationale triplet. The new component R is a matrix containing expert spatial domain knowledge. As an example, let $x = \{x_t\}$ be a text document of length T (with T denoting

the number of words). The rationales matrix $R = (r_{c,t})_{1 \leq c \leq C, 1 \leq t \leq T}$ stores the influence of each in-context word x_t on the label-memberships expressed by y . A high, positive $r_{c,t}$ value indicates a high positive influence of word x_t , in favour of label c . On the contrary, a low negative $r_{c,t}$ value represents a high negative influence of word x_t in the detriment of label c . A close to zero value tends to express neutrality towards the concerned label. The interaction with the user is carried out as in the previous case. A set $\mathcal{U} = \{u\}$ of unlabelled documents is supposed to be available and presented to the oracle. At each iteration, for a set of sample documents from \mathcal{U} , the oracle is asked to provide the corresponding label vectors as well as the rationale matrices R . Finally, the learning set \mathcal{L} is updated with the new documents (which are in the same time withdrawn from \mathcal{U}) and the learning process is iterated on the updated set \mathcal{L} .

6.3.2 Training a classifier with contextual and salient knowledge

This section presents our main contribution, which defines how to extract and inject expert knowledge within a context of restrained ground-truth, as encountered in an active learning setup. We propose original solutions for the following issues: (1) the automatic generation of *local, contextual and salient* (LCS) knowledge that provides rationales for the learning process; (2) the transfer of such knowledge to another network in an active learning setup; (3) the pruning of the contextual knowledge in order to make it similar to human-generated rationales. Under this framework, the central concept is the one of LCS knowledge. We claim that useful rationales should be: (1) *local*: solely some sub-parts of the document are relevant for determining the corresponding category; (2) *contextual*: words become meaningful only when considered within their context; (3) *salient*: the selected words should be discriminative for establishing a given category. Let us now detail how such LCS knowledge can be extracted automatically from textual documents.

Automatic extraction of local, contextual and salient knowledge.

Obtaining comprehensive class-membership cues from human labellers is a highly tedious task, significantly more complex than a simple labelling one. For this reason, we propose an automatic approach for generating rationales, based on a preliminary teacher model that is further exploited for knowledge transfer towards a student model. The teacher neural network is trained on a large set of labelled data (different from the one considered in the user's application) in order to extract some form of dense, comprehensive domain knowledge. To this purpose, two techniques are often considered to characterize sub-documents segments. A first one consists in observing the behaviour of an attention mechanism [14]. The second approach is based on the extraction of class activation maps [203]. Attention mechanisms represent a first manner to obtain spatial or temporal saliency maps with neural networks. In [14], the authors address the issue of text translation and point out that attention scores produce a meaningful unsupervised alignment between input and output tokens. In [109], a self-attention mechanism is introduced. Here, high attention scores assigned to crucial words are observed when performing text classification.

Also, in [198] authors show that attention is transferable. However, attention scores are flawed when considering the purposes of this chapter. They can be interpreted as the saliency of a sub-part of the document relatively to a generic classification task, and not relatively to a precise label. They are ranging from zero (no interest) to one (high interest) without carrying information, whereas the considered sub-parts play in favour or against class membership. For example, in [109], a self-attention mechanism is used to predict opinion polarity: the observed attention scores are high for both positively and negatively polarized words, without distinction. We would benefit from a finer characterization technique, acknowledging for negative and positive contributions of words to class membership.

A different approach concerns the so-called class activation maps (CAMs), introduced in [203]. The CAMs are able to define the contribution of each sub-part of the document to each considered class. Initially used for image classification purposes, we propose to adapt this technique to textual data. The method relies on the observation that spatiality is preserved across convolutional layers, whereas it is lost in the last fully-connected layers used by some CNNs. Hence, it only concerns fully-convolutional networks where global average pooling (GAP) or global max pooling (GMP) is applied to squash the T spatial features vectors associated to the deepest feature maps $F = \{F_1, \dots, F_T\} \in \mathbb{R}^{T \times K}$ into a single, global feature vector $F_g \in \mathbb{R}^K$ in which all spatiality is lost. Here, K denotes the size of the considered feature maps.

The model that we have adopted is a fully-convolutional neural network made of 3 layers of 128 kernels of size 5 followed by a global average pooling and a sigmoid classification layer. We used the pre-trained FastText word embedding [77].

Because we need to preserve the temporality across layers, we use the same padding for convolutions, so that there is exactly one output layer directly corresponding to each input token. Thus, the last convolutional layer presents a number of outputs that is equal to the number of input words. In this way, the t^{th} output of the last convolutional layer describes the t^{th} word of the input sentence, while taking into consideration its context within the convolutional receptive field.

The CAM extraction process is explained hereafter. If there are C different labels, then the softmax input is defined as:

$$S = W^T F_g + b \quad (6.1)$$

where $W = \{w_c^k\} \in \mathbb{R}^{K \times C}$ and $b \in \mathbb{R}^C$ respectively denote the final weights and biases. For any input example x , the class activation map for label c at location t is obtained by summing the contribution $w_c^k F_t^k$ of each scalar feature F_t^k to the final score of label c , as described in the following equation:

$$\text{CAM}(x, c, t) = \sum_k w_c^k F_t^k \quad (6.2)$$

The CAM address the aforementioned limitations of the attention scores to fulfill our purpose: in a text classification context, it provides a signed contribution of each word to a class membership, for every class tackled by the model.

Hence, we chose to use CAMs to extract dense, comprehensive LCS knowledge from a network previously trained on a large set of documents, thus $R = \text{CAM}$. Our CAM-extracted model is illustrated in Figure 4.1. Figure 6.2 is an example of LCS knowledge.

label	“supplies	for	deployed	troops”
sports	0.22	0.12	-0.89	-0.91
military	0.41	0.15	0.92	1.00

Figure 6.2: LCS knowledge: class specificity of words taken in their context.

Knowledge transfer of LCS knowledge.

We now formulate the assumption that we are considering under the following active learning circumstance. At the active learning bootstrap stage and at each iteration, some unlabelled documents are sampled and shown to a simulated oracle who provides (1) the actual labels L of the sampled documents and (2) the dense, comprehensive domain knowledge matrix $R \in \mathbb{R}^{C \times T}$ as described in section 6.3.1, extracted from a learned teacher model.

At each active learning iteration, a fully-convolutional neural network is trained on the (L, R) ground-truth pairs. From a knowledge transfer perspective, we consider this model as a student model whereas the model used for the extraction of transferred knowledge R is a teacher model, as illustrated in Figure 6.3. Both teacher and student network use the architecture described above, and illustrated in Figure 4.1. First, the teacher neural network is trained on a large set of labelled documents. Then, the rationales matrix R of each document are extracted using the trained teacher model after what active learning iterations are carried out to train the student model. When a document x is sampled, the student model gains access to its label y and rationales matrix R .

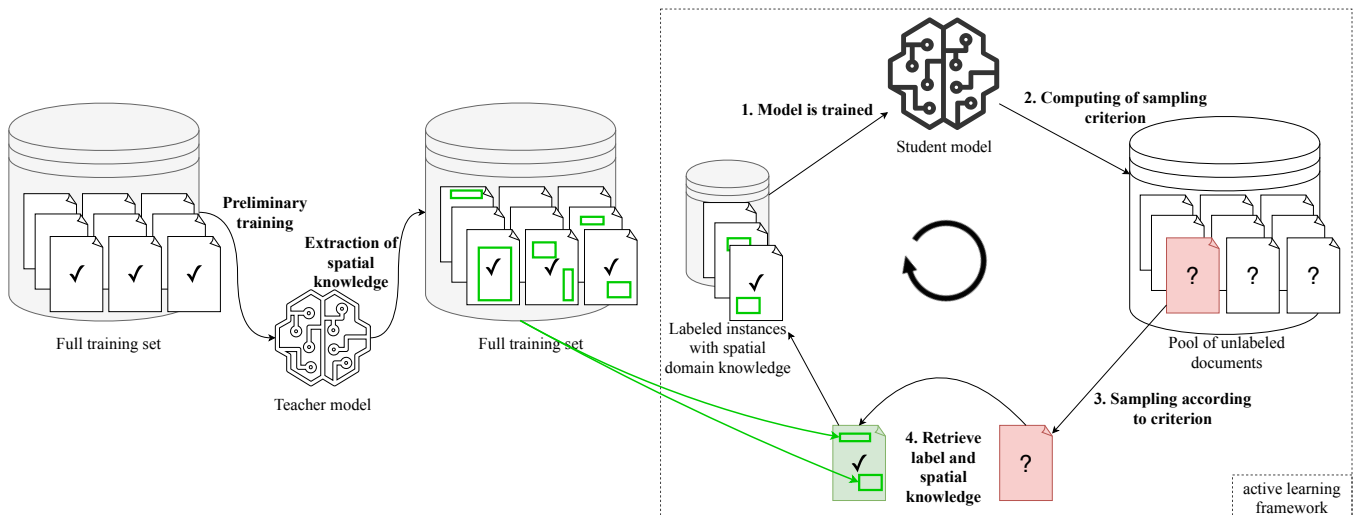


Figure 6.3: Knowledge transfer of spatial domain knowledge

The objective is twofold. On one hand, we want to train the student model to correctly classify the scarce available documents. On the other hand, we aim at correctly learning the class specificity for each sub-part of these

documents. To achieve this purpose, we introduce a rationales loss L_R that is the Mean Squared Error (MSE) between the spatial knowledge R and the CAMs directly extracted from the student model during training. This new loss function term is added to the standard cross-entropy classification loss function. The hypothesis backing this process is that the accurate spatial knowledge embedded within an even scarce set of documents is likely to be beneficial to the generalization power of the network.

Human like simulated rationales.

An active learning strategy can be evaluated without a human oracle, provided that we have access to a large labelled dataset. The label y of a document $x \in \mathcal{U}$ is obfuscated at first, and revealed when x is selected to join \mathcal{L} . To avoid the tedious task of gathering human rationales, we propose to do the same with class rationales by simulating them beforehand. To this purpose, [163] decided of a vocabulary specific to each label. Every time a word from this vocabulary is encountered, it is considered as a simulated rationale, no matter the context of this word. Our method brings some contextualization to simulated rationales, since the same word in a different context may or may not be retained as a rationale depending on the words surrounding it.

We propose to differentiate the *dense, comprehensive* LCS knowledge (not likely to be informed by a human) from the *sparse* LCS knowledge, more likely to constitute realistic simulated rationales. We used Class Activation Mappings to extract the former. We propose to prune CAM values according to the following hypotheses: (1) a human oracle may input zero, one or more rationales per text for his labelling choice; (2) it would be tedious for a human oracle to input a score for each rationale.

Hence, we used a threshold to prune-out and discretize CAM values, so texts may keep from zero to many rationales. First, the teacher model is used to extract all CAM values for all (token, label) couples. For texts actually belonging to a certain label, only a percentage α of the highest values are kept as rationales and set to 1. Similarly, for negative CAM values, the same percentage α of the lowest (negative) values are kept as rationales, and set to -1. All other values are set to 0. Here, the percentage threshold α is a user-defined parameter. In our work, we have considered a variation range from 1% to 10%. Also, rationales loss L_R is a hinge loss function modified to ignore null values:

$$L_R(R, \hat{R}) = \frac{1}{C} \sum_{c=1}^C \frac{1}{|R'_c|} \sum_{t=1}^T \begin{cases} \max(0, 1 - \tanh(\hat{r}_{c,t})) & \text{if } r_{c,t} = 1 \\ \max(0, 1 + \tanh(\hat{r}_{c,t})) & \text{if } r_{c,t} = -1 \\ 0 & \text{if } r_{c,t} = 0 \end{cases} \quad (6.3)$$

where $R = (r_{c,t})_{1 \leq c \leq C, 1 \leq t \leq T}$ are the ground-truth rationales and $\hat{R} = (\hat{r}_{c,t})_{1 \leq c \leq C, 1 \leq t \leq T}$ are the CAM values extracted from the student network at training time, corresponding to labels c and words offsets t ; and $R'_c = \{r_{c,t} \in R_c | r_{c,t} \neq 0\}$.

Figure 6.4 presents pruned, binarized spatial knowledge.

label	“supplies	for	deployed	troops”
sports	0	0	-1	-1
military	0	0	1	1

Figure 6.4: Sparse spatial domain knowledge: label specificity of words taken in their context.

6.4 Experimental results

In this section, we describe the datasets and the active learning settings used for our experiments. Then, we present the simulated, sparse rationales obtained when applying the CAM-pruning method presented in section 6.3.2. Finally, we present and comment the learning curves obtained with learning without rationales (Lw/oR), learning with dense, comprehensive rationales (LwDR) and learning with sparse, human like rationales (LwSR).

6.4.1 Evaluation protocol

We used the following text classifications datasets:

- The IMDB dataset consists of 25K film reviews [116] labelled as positive or negative.
- The WvsH dataset is a 20 Newsgroups dataset [98] reduced to Windows versus Hardware topics.

In order to compare our results with those of [163], we have used the AUC (Area Under ROC Curve) measure as evaluation metric. For LwSR, the values considered for the α parameter are of 10, 3 and 1. We have used a budget of 200 documents to feed the active learning process.

Concerning the active learning process, we have used a bootstrap of 10 random documents, and we labelled 5 documents at each active learning iteration, which have been sampled according to the uncertainty criterion [104]. In this binary classification context, our uncertainty function is defined by

$$u(x) = \frac{1}{2} - \left| \hat{y} - \frac{1}{2} \right| \quad (6.4)$$

where $\hat{y} \in [0, 1]$ is the student model’s prediction. At the beginning of each active learning iteration, the 5 most uncertain unlabelled documents are selecting using this criterion.

The Lw/oR, LwDR and LwSR scenarios have the same initial documents boot-straps and all models are initialized with the same set of random weights and biases.

6.4.2 Contextualized Simulated Rationales

Figures 6.5 and 6.6 illustrate Class Activation Maps extracted from the teacher model, and the results of their pruning to get sparse class rationales. First, the extracted dense, comprehensive spatial knowledge shows that the trained teacher model is able to detect what words are discriminative for establishing a given category: in these examples,

	Spatial domain knowledge (green: positive, red: negative)
Dense, comprehensive	<p>This is one of the worst movies I have ever seen . However , the little slave girl , Alice and Jared Harris imitating Christopher Walken is what makes this movie entertaining . [...]</p>
Sparse ($\alpha = 10\%$)	<p>This is one of the worst movies I have ever seen. However, the little slave girl, Alice and Jared Harris imitating Christopher Walken is what makes this movie entertaining. [...]</p>
Sparse ($\alpha = 3\%$)	<p>This is one of the worst movies I have ever seen. However, the little slave girl, Alice and Jared Harris imitating Christopher Walken is what makes this movie entertaining. [...]</p>
Sparse ($\alpha = 1\%$)	<p>This is one of the worst movies I have ever seen. However, the little slave girl, Alice and Jared Harris imitating Christopher Walken is what makes this movie entertaining. [...]</p>
Dense, comprehensive	<p>This is the definite Lars von Trier Movie , my favorite , I rank it higher than " Breaking the waves " or the latest " Dancer in the Dark "... I simply love the beauty of the pictures ... The framing is so original ; acting is wonderful , A MUST SEE .</p>
Sparse ($\alpha = 10\%$)	<p>This is the definite Lars von Trier Movie, my favorite, I rank it higher than "Breaking the waves" or the latest "Dancer in the Dark"... I simply love the beauty of the pictures... The framing is so original; acting is wonderful, A MUST SEE.</p>
Sparse ($\alpha = 3\%$)	<p>This is the definite Lars von Trier Movie, my favorite, I rank it higher than "Breaking the waves" or the latest "Dancer in the Dark"... I simply love the beauty of the pictures... The framing is so original; acting is wonderful, A MUST SEE.</p>
Sparse ($\alpha = 1\%$)	<p>This is the definite Lars von Trier Movie, my favorite, I rank it higher than "Breaking the waves" or the latest "Dancer in the Dark"... I simply love the beauty of the pictures... The framing is so original; acting is wonderful, A MUST SEE.</p>

Figure 6.5: Extracted spatial domain knowledge, IMDB dataset (best viewed with colors)

	Spatial domain knowledge (red: Windows, blue: hardware)
Dense, comprehensive	I have a DFI Handy Scanner Model HS - 3000Plus and a little bit of software running under dos to use it . I ' d like to make more extensive use of this device (in particular , write a driver for it on unix) . So , can anyone give me a description of how to talk to this device . It connects to the system via it ' s own interface card . Any info would help , it can ' t be too difficult to talk to : -)
Sparse ($\alpha = 10\%$)	I have a DFI Handy Scanner Model HS - 3000Plus and a little bit of software running under dos to use it . I ' d like to make more extensive use of this device (in particular , write a driver for it on unix). So , can anyone give me a description of how to talk to this device. It connects to the system via it ' s own interface card . Any info would help, it can ' t be too difficult to talk to :-)
Sparse ($\alpha = 3\%$)	I have a DFI Handy Scanner Model HS - 3000Plus and a little bit of software running under dos to use it . I ' d like to make more extensive use of this device (in particular, write a driver for it on unix). So , can anyone give me a description of how to talk to this device. It connects to the system via it ' s own interface card . Any info would help, it can ' t be too difficult to talk to :-)
Sparse ($\alpha = 1\%$)	I have a DFI Handy Scanner Model HS - 3000Plus and a little bit of software running under dos to use it . I ' d like to make more extensive use of this device (in particular, write a driver for it on unix). So , can anyone give me a description of how to talk to this device. It connects to the system via it ' s own interface card . Any info would help, it can ' t be too difficult to talk to :-)
Dense, comprehensive	I remember reading about a program that made windows icons run away from the mouse as it moved near them . Does anyone know the name of this program and the ftp location
Sparse ($\alpha = 10\%$)	I remember reading about a program that made windows icons run away from the mouse as it moved near them. Does anyone know the name of this program and the ftp location
Sparse ($\alpha = 3\%$)	I remember reading about a program that made windows icons run away from the mouse as it moved near them. Does anyone know the name of this program and the ftp location
Sparse ($\alpha = 1\%$)	I remember reading about a program that made windows icons run away from the mouse as it moved near them. Does anyone know the name of this program and the ftp location

Figure 6.6: Extracted spatial domain knowledge, WvsH dataset (best viewed with colors)

the highest values are “worst” (negative review), “wonderful” (positive review), “interface” (hardware) and “windows” (Windows). It appears that the teacher model is sensitive to context, as shown by the apparent continuity of values across words. Two examples notably contains contradictory series of words. The teacher model detects positiveness when a film reviewer admits being entertained by one of the worst movies he/she has ever seen; the teacher model detects software references in an overall hardware related message. The transfer of this very comprehensive knowledge to a student network corresponds to our LwDR scenario. The continuity of the CAM values makes it difficult to prune them to realistic, human like rationales. Pruning with $\alpha = 10\%$ tends to contain more information than a human oracle could efficiently provide with a highlighting tool, whereas pruning with $\alpha = 1\%$ is more realistic but may omit every salient word of some sentences (Figure 6.5). The transfer of these sparse, simulated rationales to a student network corresponds to our LwSR scenarios.

IMDB positive reviews	IMDB negative reviews
Kusturica brilliantly examines this theme compellingly explores the emotional chasm well-written and well are remarkably futuristic today. perfect supporting cast and great definitely recommend this. 9 I really liked Dana Plato Everyone was great! 10 the unmatchedably billiant and ingenious Brashear deeply touches the heart	lack of much evidence of of confusion and disappointment preposterously ugly and annoying girl disrespectful boring shame that will was a crass attempt at horrible acting, lame porn was pretty crap-freaking boring with its clichés and create a cheap idiotic show was the worst ending I
WvsH Windows messages	WvsH hardware messages
in the apps icon (for Windows? Will it accounts. Windows Recorder does print drives for Windows for for MS-Windows v3 Ms-Windows logo, a program in windows such of Window Menu, Help into the Windows startup Brad made windows icons run away	with 2MB of DRAM. . Are SCSI drives faster floppies. The controller and a 50MHz motherboard would seem radiation emission monitors besides NEC with other modems that are clock hardware interrupt and BIOS the connector on the back that floppy.. BURN 8507 IBM monitor (19

Figure 6.7: Simulated rationales ($\alpha = 1\%$)

Figure 6.7 presents simulated rationales obtained with our most restricting pruning setting $\alpha = 1\%$. Words are displayed within their nearby context. With three layers of kernel size 5, the size of the full theoretical receptive field of the network is 13. However, according to [114], the effective receptive field is a Gaussian distribution centred around the “central” token (the one aligned with the observed CAM value). So, only 5-wordgrams with the highest contribution to the receptive field are displayed. Words near the beginning or the end of a text are displayed in shorter n-wordgrams.

We observe that our simulated rationales grasp a broader context than the one-vocabulary-per-label method used in [163]. The central word “written” is not positive by itself, however, when combined with other words in its vicinity,

as in the case of this example (“well-written”) it obviously bears a positive valence. Similarly, the adjective “futuristic” takes a positive signification as soon as it is précised to be “remarkably futuristic”. Some 5-grams like “boring with its clichés and” encapsulate several close negative words and depict a rather negative region of the text. Regarding rationales for WvsH Windows class, they Rationales for the WvsH Windows class are basically collection of the “windows” word, whereas rationales for the WvsH hardware class gather a broader specific vocabulary. We claim that such simulated rationales resemble those that can be reasonably provided by a human oracle with a highlighting tool. Thus, we consider them suitable for evaluating the contribution of rationales to learning curves. The following section present the results obtained in Lw/oR, LwDR and LwSR settings.

6.4.3 Experimental results

Figures 6.8 and 6.9 present the learning curves for the Lw/oR, LwDR and LwSR scenarios. First, let us observe that the LwDR approach drastically outperforms the others. It must be emphasized that the teacher model used for the generation of the transferred comprehensive rationales have been trained on a large set of documents, that would not be available in real active learning settings. Yet, given the CAM values of only 10 to 25 documents, the student model is able to reach a very high AUC. This spectacular result shows that refined information about salient regions of a few documents can lead to a considerable improvement of a CNN’s generalization capacity.

The LwSR performances are between those of LwDR and Lw/oR. Such results were predictable since LwSR transfers more than the sole label information of Lw/oR, but it is a pruned, binarized version of the comprehensive knowledge of LwDR. Moreover, the higher the number of provided rationales (symbolized by parameter α), the higher the results. Relying on the same CNN architecture, LwSR constitutes a significant improvement over Lw/oR, for both datasets.

Our method outperforms the framework introduced in [163] on the IMDB task. However, it is outperformed on the WvsH task, especially at the earliest stages of active learning (10-50 documents). Although we might have chosen another CNN architecture, CNNs are not necessarily the best classification tool for every type of text documents. Our CNN benefited from contextualized rationales to grasp the sometimes tortuous and verbose opinions of film reviewers. Yet, a simple word count method is a sufficient, straightforward method to detect membership to a category such as “windows”.

6.5 Conclusion

Our results have shown that, for text classification, even a few training examples carry a much more valuable information than their sole label: indeed, some contextualized words or groups of words are responsible, more than others, for class membership. When given the precise map of class specificity of only a few training documents (called local, contextual and salient knowledge in this chapter), a student convolutional neural network has access to

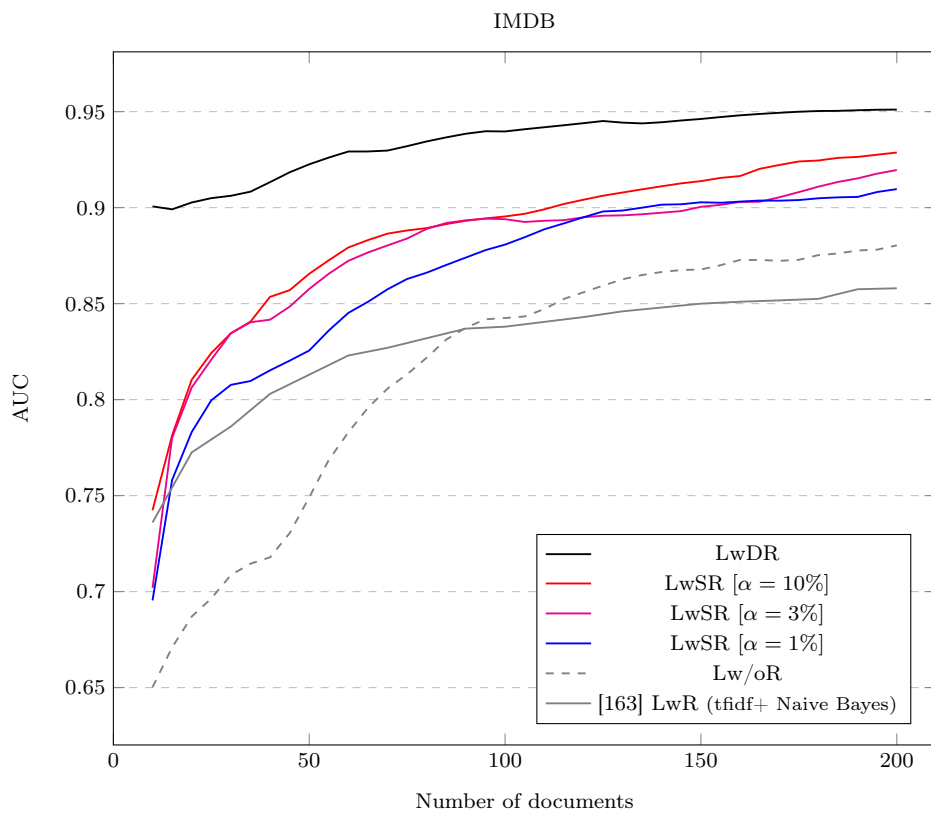


Figure 6.8: Comparison of Lw/oR, LwDR and LwSR — IMDB dataset

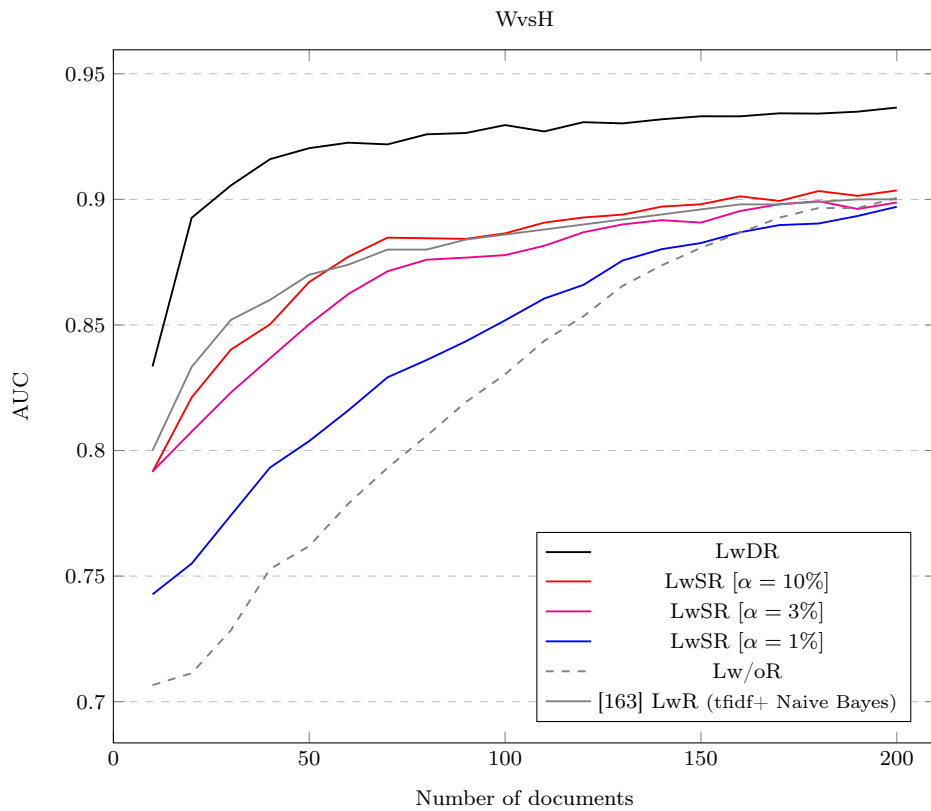


Figure 6.9: Comparison of Lw/oR, LwDR and LwSR — WvsH dataset

an extremely rich knowledge allowing for strong generalization on unseen data. To get such knowledge, we used Class Activation Mappings extracted from a teacher convolutional neural network, in a knowledge transfer fashion. When pruning this additional ground-truth, we get simulated rationales that reasonably resemble the rationales a human with a highlighting tool could have provided. The more we prune and simplify the transferred spatial knowledge, the more we deteriorate the learning curves, although the process keeps beneficial at the earliest stages of active learning, where ground-truth is at its scarcest. Providing class rationales as rich as possible seems to be a good practice when initiating an active learning framework. Also, our study introduced rationales into deep active learning.

This chapter opens up a perspective for deep active learning applied to image classification, and for the use of real, non-simulated human rationales. Also, rationales could be exploited to favour the sampling of documents containing the most conflicting groups of words.

Chapter 7

Active Learning with Rationales for Image Classification

Abstract

Deep neural networks have demonstrated state-of-the-art performances in image classification tasks, when very large sets of labelled documents are available for training. However, in an early active learning setup, because of the extreme scarcity of the available labelled images, the model fails to achieve a satisfying generalization capacity. In the same time, techniques such as class activation mapping and attention visualization make it possible to identify the regions that a trained neural network considers as rationales for its predictions. Following a reverse approach, in this chapter, we first introduce different strategies that allow injecting expert spatial rationales into the learning process. Such additional information is particularly useful in the case where the training set is too scarce, and we show that it can significantly enhance the network's generalization capabilities. The experimental results obtained demonstrate that the proposed self-attention supervision strategy permits to significantly improve the model's performances. A second contribution concerns the sampling strategies that consist of selecting, during the training phase, images whose rationales are most likely to provide useful, beneficial information. Experimental results obtained demonstrate that the proposed self-attention/Grad-CAM misalignment criterion outperforms traditional sampling approaches.

7.1 Introduction

Deep neural networks have demonstrated state-of-the-art performances in automatic classification tasks when large, labelled ground-truth datasets are available for training.

In particular, for image classification tasks, convolutional neural network harness the abundant training examples to discriminate the high-level visual features responsible for class membership. Assembling such training datasets requires collecting thousands, sometimes millions of documents and manually annotating them with reliable ground-truth. Such an expensive, tedious and time-consuming task is incompatible with real-life, urgent, user-specific applications for which no pre-existing dataset is available. Thus, when the training data is scarce, neural image classifiers suffer from overfitting and lack of the generalization capacity to recognize prominent, high level features in unseen images.

A first solution to this problem consists in considering a transfer learning paradigm [132]. Convolutional neural network architectures such as AlexNet [95], VGG [167], GoogLeNet [172], ResNet [61] and Inception-v4 [173] have obtained state-of-the-art results on ImageNet Large Scale Visual Recognition Challenge [153]. When pre-trained on very large image datasets with a great number of object classes like ImageNet, OpenImages [94] or Common Objects in Context [107], such architectures become highly performant and offer versatile feature extractor trunks, on which one or several task-specific additional layers can be plugged, under a transfer learning paradigm. Following transfer learning variants, the pre-trained trunks can be either frozen or fine-tuned while the added custom layers are necessarily trained.

A second solution, that can boost the transfer learning approaches, consists of considering an active learning process [181, 8]. During an active learning iteration, new training images are either synthesized or drawn from a pool of unlabelled images according to criteria dictated by a sampling strategy. They are then presented to a human, expert oracle for labelling. Finally, the classification model is re-trained with the new labelled samples. The key ingredient of active learning methods consists in sampling the images which are the most likely to boost the learning process. Multiple sampling criteria can be considered, such as uncertainty [103, 31], information density (or representativeness) [105], diversity [19, 41], or query-by-committee [34]. In a traditional active learning setup, the oracle solely provides label information for sampled images. Yet, in the scarce ground-truth context of early-stage active learning, the oracle can provide additional information under the form of rationales that explain why he has assigned a certain label to a given sample. Such rationales can be exploited during the learning phase, by forcing the loss function to take them into account. This principle has been considered in the field of text classification [163, 58], where its application is straightforward: the annotator has simply to specify sets of words/sentences in a document that justified his labelling. The results reported in [163, 58] demonstrate that rationales can significantly improve the classification performances. In the case of images, the objective is to specify, in a given image, the salient regions (e.g., with the help of some bounding boxes) that are relevant for the label provided by the annotator. However, the

application of such a technique is not straightforward. The difficulty comes from the intrinsic nature of the image data. Thus, in the case of text, the class activation maps (CAMs) [203] can be directly used for injecting rationales, since the spatial features are naturally aligned with the salient sentence words. When dealing with images, the successive convolutional layers of a CNN yield rectangular spatial feature maps whose receptive fields in the input image are uniform grids of equally-sized cells. Generally, such grids are not aligned with the salient objects that are present in the image. In [125], the authors propose to project fine-grained human rationales on a grid, in order to ultimately fine-tune a model by supervising an attention mechanism. Since the feature map grid does not match the semantic entities that are present in the images, the human labeller is asked to provide a precise segmentation of the discriminative regions by drawing as many circular “bubbles” as needed to match their shapes. This tedious interactive process represents the main drawback of the method. This issue is also addressed in [24], for image captioning and visual question answering [5]. Here, authors propose to exploit the object proposal features produced by a Faster R-CNN object detector [145]. In this way, they replace the 2D uniform feature grid with a 1D list of object features that can directly describe the objects detected in the images.

In this chapter, we notably tackle the issue of image-based active learning with rationales. Our contributions are twofold: (1) We first introduce a method that transfers spatial rationales to an image classifier built on top of a Faster R-CNN object detector. We notably show that objects responsible for class membership can be hinted to the learning model to boost its generalization capacity, with a minimal amount of images (i.e., starting from a single image per class). (2) We then propose a novel sampling criterion based on a form of intra-image uncertainty and show that it outperforms usual active learning sampling criteria. The rest of the chapter is organized as follows. Section 7.2 describes the proposed approaches for training an image classifier with human-like simulated rationales as additional spatial information. In section 7.3, we propose new sampling criteria that take into account fine-grained spatial information within the sampling candidates. Finally, section 7.4 concludes the chapter and opens some perspectives of future work.

7.2 Active Image Classification with Rationales

During the early stages of active learning for classification, the trained model only has access to a very narrow training set of images. Independently of choosing an adequate sampling strategy to enrich this set, training the model with a potentially very narrow training set is a key challenge of early-stage active learning. In an active learning scenario encompassing a classification task, the oracle possesses the expert knowledge needed to provide reliable labels for sampled documents. More specifically, the oracle knows the intrinsic aspects of a document that are responsible for its class membership. For some image classification tasks, such high level features may be distributed all over the image, without a strongly localized saliency.

This is typically the case of place recognition tasks [9] with evenly distributed background elements, of genre

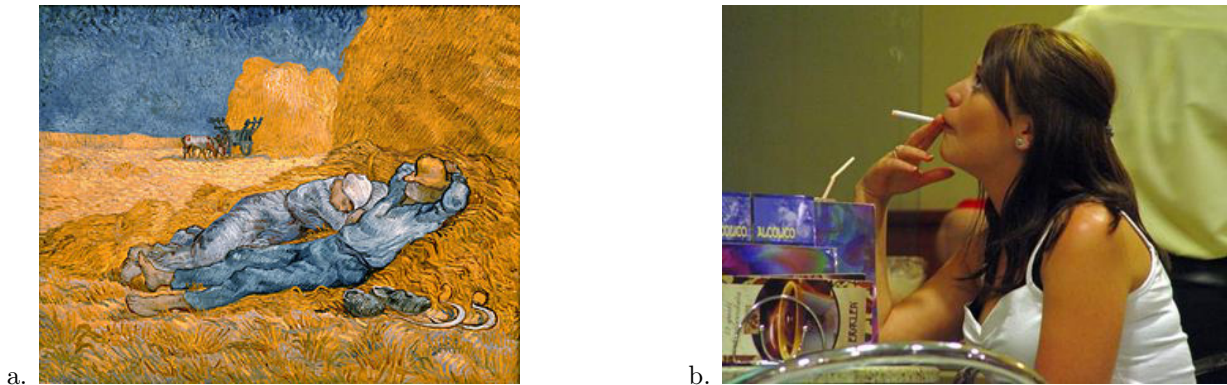


Figure 7.1: Rationales for class membership can be evenly distributed across the image or strongly localized in a salient region. Membership to a hypothetical painting class is due to the overall texture information (a) whereas membership to the smoking class is strongly localized around the cigarette, the hand and the mouth (b).

prediction from film posters [97], or of painting style recognition [100] where the key content is the overall style of the image (Figure 7.1a). On the contrary, certain image classification tasks do rely on strongly localized objects, without being reducible to object detection. For example, the recognition of an action may be specifically centred on the person, body part or object involved in the action, but is not reducible to the simple presence of this person, body part, or object (Figure 7.1b). The approaches proposed in this chapter are specifically dedicated to this second category of situations. If some localized spatial information is relevant for the classification task, we claim that taking into account spatial rationales when training the classifier with a narrow training set makes it possible to improve the model performances. This hypothesis is based on the assumption that with few training examples (starting from only one image per label), the provided rationales can force the repetition of class-specific patterns across a broader training set.

The rationales are defined as a set of rectangular image regions. The straightforward manner to obtain such rationales consists of asking a human expert to manually select the corresponding image regions that are considered as responsible/relevant for the given label. Then, the challenge consists in defining appropriate methods for exploiting this additional, spatial information. The proposed approaches are detailed in the following section.

7.2.1 Learner architecture and integration of rationale constraints

All the proposed models are built on top of the object feature extractor of a pretrained Faster R-CNN object detector [145], which has been successfully used for image captioning and visual question answering tasks [7]. This frozen neural network backbone extracts object-related rectangular regions of interest from images and yields a fixed-size, 2048-dimensional feature vector for each determined bounding box (Figure 7.2). Let us note that solely the regions where the corresponding objectness score (which can be interpreted as the probability of the presence of an object in the given region, independently of its class) provided by Faster R-CNN exceeds a confidence threshold are retained.

The Faster R-CNN is applied as a post-processing, refinement phase to each image provided by the oracle. The

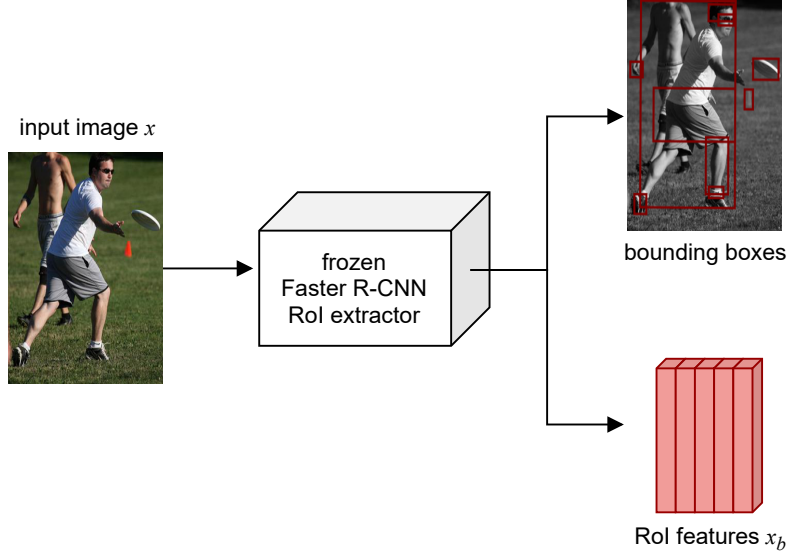


Figure 7.2: ROI features extraction

Faster R-CNN regions that overlap with the rationale regions provided by the oracle are considered as the final rationales. In this way, we ensure consistency between regions provided by the oracle and those determined by the learner model.

The set of rationales derived from the oracle (O) region specification is finally represented, for each image-label pair (x, y) , as a binary support vector $R^O(x, y) = (R_1^O(x, y), \dots, R_b^O(x, y), \dots, R_{B(x)}^O(x, y))$ of size $B(x)$, where $B(x)$ is the *total* number of regions detected by the Faster R-CNN model for the image x . In this representation, the value $R_b^O(x, y)$ equals 1 if the b^{th} region is a rationale region and 0 otherwise.

Under this framework, let us now assume that a set $\mathcal{D}_L = \{(x, y, R^O(x, y))\}$ of image-label-rationale triplets is available for the learner model L . The set \mathcal{D}_L is split into a training set \mathcal{T}_L and a validation set \mathcal{V}_L .

For convenience, we assimilate every image x as its set of $B(x)$ features associated to the retained rationale objects $x = \{x_b\} \in \mathbb{R}^{B(x) \times 2048}$. The extracted box features are fed to N_L fully-connected layers with $K = 512$ units, ReLU activation and dropout (with 0.5 rate), yielding a transformed feature matrix $x' = \{x'_b\} \in \mathbb{R}^{B(x) \times K}$. This feature matrix can be interpreted as a set of K feature maps z_k , each of dimension $B(x)$:

$$\forall k \in \{1, 2, \dots, K\}, \quad z_k(x) = (x'_{1,k}, x'_{2,k}, \dots, x'_{B(x),k}) \quad (7.1)$$

Each feature vector z_k is then globalized into a single scalar value $f_k(x)$, defined as:

$$\forall k \in \{1, 2, \dots, K\}, \quad f_k = \sum_{b=1}^{B(x)} \alpha_b x'_{b,k} \quad (7.2)$$

This process can be assimilated as a weighted, *extended* form of *global average pooling* [106] applied to each feature vector z_k , that we will denote hereafter by e-GAP.

The values $f_k(x)$ are finally stored into a global feature vector $f(x) = (f_0(x), \dots, f_k(x), \dots, f_K(x))$ that is further used for classification purposes, which can be written as:

$$\forall k \in \{1, 2, \dots, K\}, f_k(x) = \sum_{b=1}^{B(x)} \alpha_b x'_{b,k} = \text{e-GAP}(z_k(x)) \quad (7.3)$$

The question now is how to take advantage of this weighting mechanism in order to incorporate into the α_b coefficients some additional information, derived from the available rationales.

In the last years, within the context of an increasing research effort dedicated to the explanation of networks behaviour, various saliency analysis approaches have emerged. Among the most popular techniques, let us cite the class activation mappings (CAM) [203], their Grad-CAM extension [157] or the self-attention (SA) mechanisms [183]. The generic principle consists of associating to each image-label pair (x, y) within the learning set, a saliency vector $S(x, y)$ of size $B(x)$, globally expressing the contribution of each detected region of interest to the classification decision. In other words, $S(x, y)$ is a vector indicating the importance of each region of interest for the final prediction score of label y . Most of the time, $S(x, y)$ is defined as a linear combination of the features maps z_k , as described in the following equation:

$$S(x, y) = \sum_{k=1}^K w_k^y z_k(x) \quad (7.4)$$

where the coefficients w_k^y are weighting the contribution of the k^{th} feature map z_k to the classification prediction y . In the following, we will denote by $S^L(x, y)$ the saliency vector determined by the learner model L .

This saliency information can be compared to the rationales $R^O(x, y)$ provided by the oracle. In particular, a binarized version $R^L(x, y)$ of $S^L(x, y)$ will correspond to a prediction of rationales, determined by the learning model. In our work, we have adopted the following binarization scheme:

$$\forall b \in \{1, \dots, B(x)\}, \quad R_b^L(x, y) = \begin{cases} 1 & \text{if } S_b^L(x, y) \geq 0.9 \times \max_{\beta \in \{1, \dots, B(x)\}} S_\beta^L(x, y) \\ 0 & \text{otherwise} \end{cases} \quad (7.5)$$

The threshold value 0.9 has been empirically chosen so that 1 to 3 rationales are retained for most images. The principle of the various methods proposed consists in forcing, during the learning stage, the learner model L to adapt its saliency vectors $S^L(x, y)$ and the corresponding predicted rationales $R^L(x, y)$ to those provided by the oracle $R^O(x, y)$. This principle is put into practice with the help of two different layers. A first one consists in adapting the e-GAP weights, and a second one concerns the specification of the loss function.

A first method considered is the direct extension to the case of images of the approach initially introduced in [58] for text classification purposes.

Baseline method: direct supervision of class activation mappings Direct-CAM

Here, box features replace the word embeddings previously considered in [58]. However, since image box features are listed in an arbitrary order that has no relation with their corresponding spatial positions, 1-D convolutions do not make sense in this case. The considered architecture is illustrated in Figure 7.3.

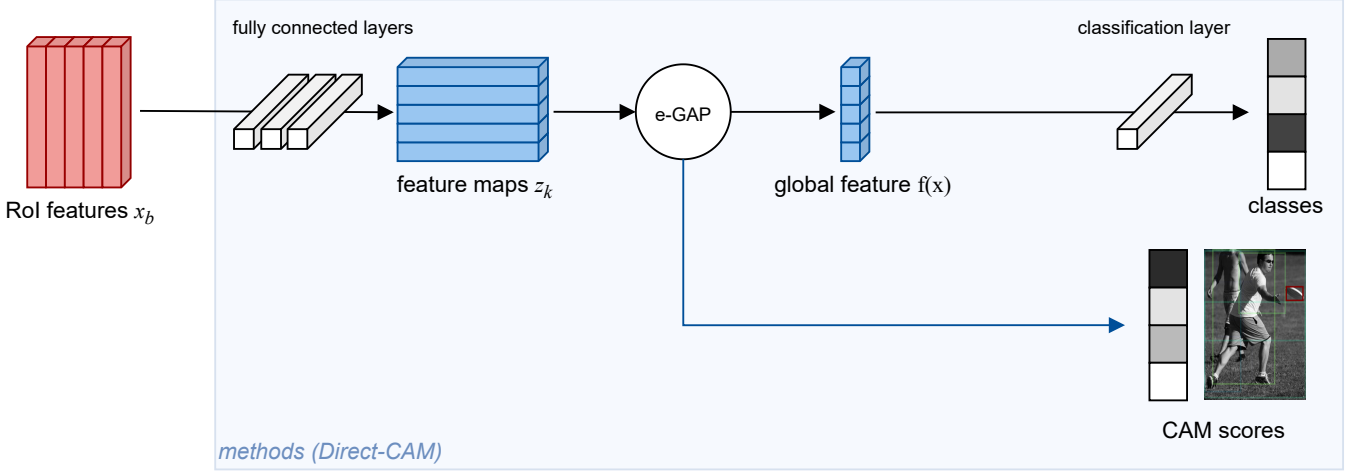


Figure 7.3: Baseline method architecture

Processed feature maps are in this case merged by a standard GAP. Thus, the coefficients α_b in equation 7.2 are all equal to 1. Because this method is based on CAM [203], the final classification layer is necessarily applied directly just after the GAP stage. It is a dense layer with as many outputs as labels, and softmax activation (respectively sigmoid activation) for mutually exclusive classes (respectively for single-output binary classification).

For all $(x, y) \in \mathcal{D}_L$, $S^L(x, y)$ is computed as the learner's model CAM with regard to the ground-truth label y , using the pre-GAP features z_k . In order to take rationales into account and guide the learner model into focusing on the correct regions of the training images, a CAM loss function is defined as:

$$\text{loss}_{\text{CAM}}(x, y) = \frac{1}{\|R(x)\|_1} \sum_{b \leq B(x)} \begin{cases} 1 - \tanh S^L(x_b, y), & \text{if } R_b^O(x, y) = 1 \\ 0, & \text{if } R_b^O(x, y) = 0 \end{cases} \quad (7.6)$$

During training, the CAM loss function is added to the classification loss that takes ground-truth label y and model prediction \hat{y} as inputs. With C mutually exclusive classes, we use categorical cross entropy:

$$\text{loss}(x, y, \hat{y}) = - \sum_{i=1}^C y_i \log \hat{y}_i + \text{loss}_{\text{CAM}}(x, y) \quad (7.7)$$

For single-output binary classification, we use binary cross entropy (7.8):

$$\text{loss}(x, y, \hat{y}) = -y_i \log \hat{y}_i - (1 - y_i) \log (1 - \hat{y}_i) + \text{loss}_{\text{CAM}}(x, y) \quad (7.8)$$

The following two methods, so-called Dynamic masking of feature maps (DMFM) and Dynamic weighting of feature maps (DWFM), share a common network architecture illustrated in Figure 7.4.

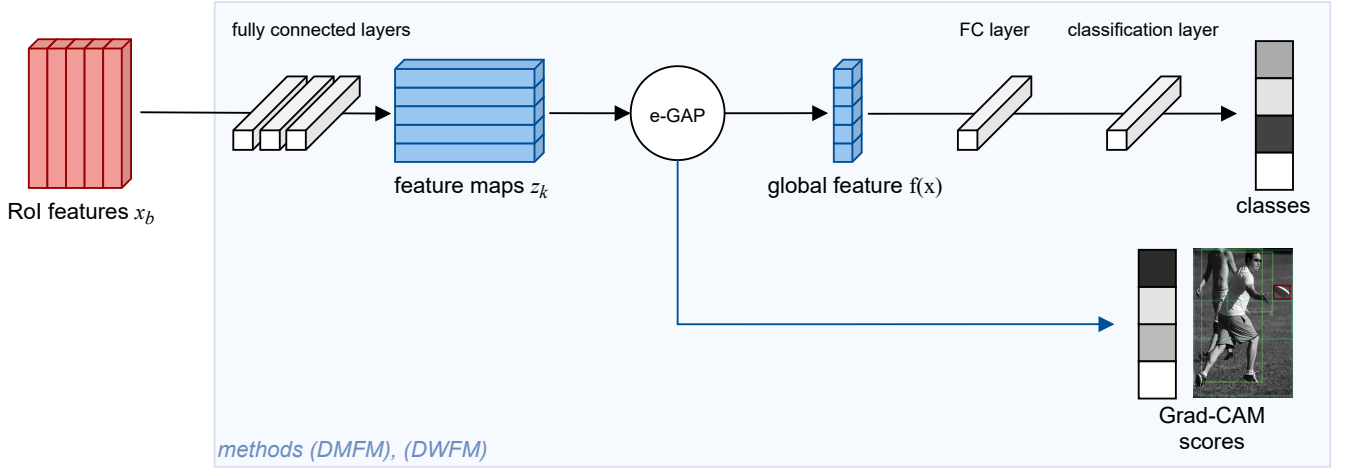


Figure 7.4: Neural network architecture for DMFM and DWFM approaches.

The main difference with the baseline approach concern the e-GAP phase, which replaces the traditional GAP, the introduction of an additional fully connected layer with 512 units, ReLU activation and dropout (with 0.5 rate), prior to the classification one, which is made possible by the replacement of the CAM with a Grad-CAM technique. The final classification layer uses softmax activation (respectively sigmoid activation) for mutually exclusive classes (respectively single-output for binary classification).

Dynamic masking of feature maps DMFM

During the training of the model, the only supervised loss considered is the cross-entropy classification loss. For all image-label pairs (x, y) in the training set, we introduce a masking rate $\rho(x, y)$ that measures how much the learner model is considering the relevant regions of x as responsible for membership to class y . The masking rate is defined as the global intersection rate between oracle and learner rationales, as described in the following equation:

$$\rho(x, y) = 1 - \frac{\sum_{b \leq B(x)} R_b^O(x, y) \times R_b^L(x, y)}{\sum_{b \leq B(x)} R_b^O(x, y)} \in [0, 1] \quad (7.9)$$

The masking rate is 0 when $R^O(x, y)$ and $R^L(x, y)$ perfectly overlap, whereas it is 1 when the learner model L fails to acknowledge salient regions. The value of $\rho(x, y)$ is updated at every epoch during the training process.

In order to take rationales into account and guide the learner model into focusing on the correct regions of the training images, out-of-rationales boxes, *i.e.*, regions for which $R_b^O(x, y) = 0$, are randomly masked with a probability

of $\rho(x, y)$ during the e-GAP phase. Thus, the coefficients α_b characterizing the e-GAP process are defined as:

$$\alpha_b = \frac{\alpha'_b}{\sum_b \alpha'_b}, \quad \text{with } \alpha'_b = \begin{cases} 1, & \text{if } R_b^O(x, y) = 1 \\ 1, & \text{if } R_b^O(x, y) = 0 \text{ and } \text{rand}(\xi) \geq \rho(x, y) \\ 0, & \text{if } R_b^O(x, y) = 0 \text{ and } \text{rand}(\xi) < \rho(x, y) \end{cases} \quad (7.10)$$

where $\text{rand}(\xi)$ denotes a random trial of a random variable ξ uniformly distributed in the $[0, 1]$ interval. The random trial is performed at each epoch for each set of out-of-rationales boxes.

If at a given epoch the model is “wrong” in its Grad-CAM for image x and label y , out-of-rationales boxes are likely to be masked out. Otherwise, if the model is “right” in its Grad-CAM at another epoch, out-of-rationales boxes are more unlikely to be masked-out. In this way, at the beginning of the learning process, the training is forced to focus on the regions provided as rationales. On the contrary, towards the end of the training process, when the learner and oracle rationales are supposed to become more and more similar, some out-of-rationales boxes (that can be interpreted as additional, contextual elements of information) are allowed to be taken into account.

Dynamic weighting of feature maps DWF

This method is highly similar with the dynamic masking method. The differences come solely from the weighting mechanism of the e-GAP process. We introduce a weighting factor $\omega(x, y) = 1 - \rho(x, y) \in [0, 1]$, updated at every epoch. In order to take rationales into account and help the learner model into focusing on the correct regions of the training image, features corresponding to out-of-rationales regions are down-weighted by a factor $\omega(x, y)$. More precisely, the coefficients α_b characterizing the e-GAP process become in this case:

$$\alpha_b = \frac{\alpha'_b}{\sum_b \alpha'_b}, \quad \text{with } \alpha'_b = \begin{cases} 1, & \text{if } R_b^O(x, y) = 1 \\ \omega(x, y), & \text{if } R_b^O(x, y) = 0 \end{cases} \quad (7.11)$$

In this way, the more the learner model is “wrong” in its Grad-CAM, the more the out-of-rationales regions are ignored.

The final approach is based on a self-attention supervision SAS technique.

Self-attention supervision SAS

Here, the processed box features are merged by a self-attention mechanism [25, 183]: a dense layer with a softmax activation outputs the weighting coefficients α_b of each deep region feature x'_b .

In order to take the rationales into account and guide the learner model into focusing on the correct regions of the training images, the supervision of the post-softmax attention scores is treated like a multi-label classification task. As a loss function, we have considered the categorical cross-entropy weighted by the inverse number of ra-

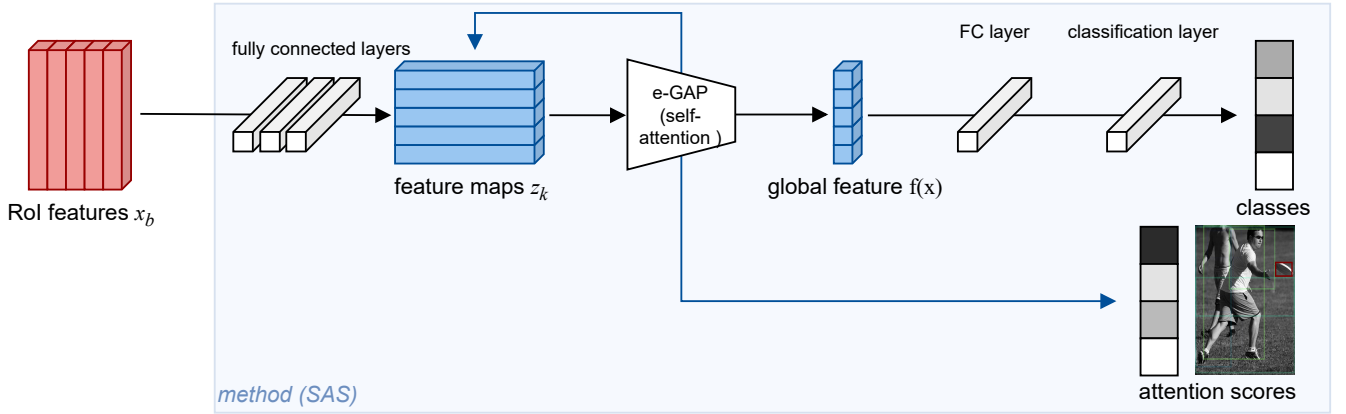


Figure 7.5: Neural network architecture for SAS approach.

tionales $\frac{1}{\|R(x,y)\|_1}$, which yields better results than the binary cross-entropy function frequently used in multi-label classification problems, as indicated in [118]. More precisely, the loss function supervising attention scores is defined as:

$$\text{loss}_{\text{SAS}} = -\frac{1}{\|R^O(x,y)\|_1} \sum_{b=1}^{B(x)} R_b^O(x,y) \log(\alpha_b) \quad (7.12)$$

Whatever the considered technique, the availability of rationales is a fundamental issue that needs to be solved. The proposed solution is described in the following section.

7.2.2 Automatic extraction of simulated rationales

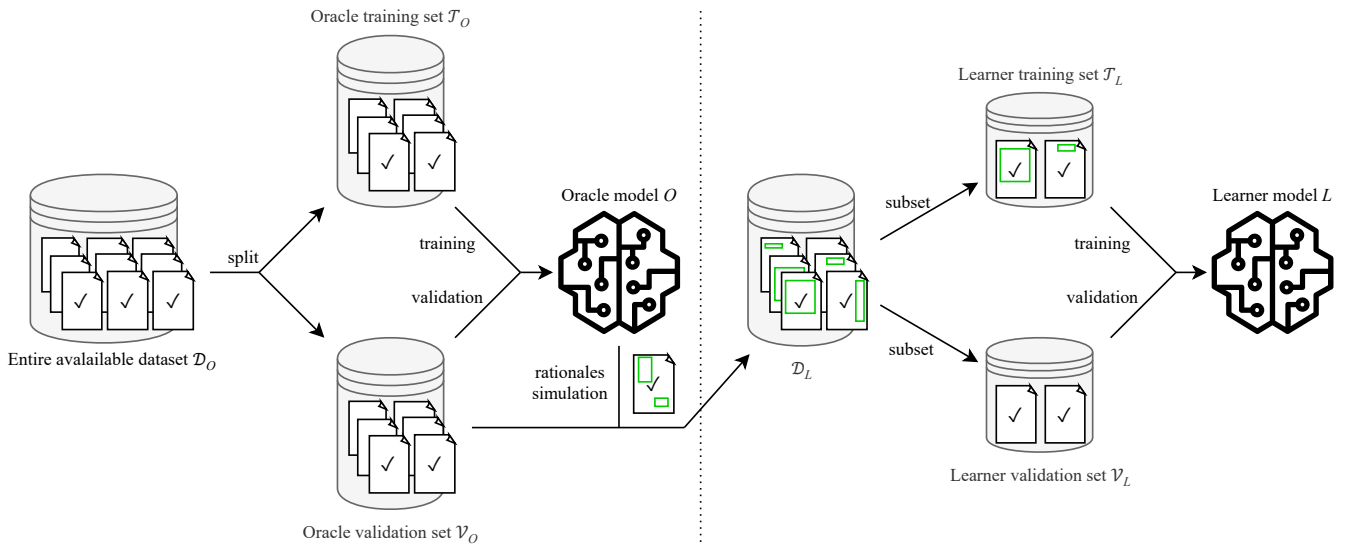


Figure 7.6: Rationales $R(x)$ (in green) are simulated using oracle model O . Simulated rationales are then exploited by the learner model L when it is trained with a narrow training set \mathcal{T}_L .

In order to overcome such difficulties, we propose to generate such rationales in a fully automatic manner. The underlying principle follows the method initially proposed in [58] for text classification purposes. Thus, a reliable

oracle model is used to generate automatically human-like simulated rationales. The process is illustrated in Figure 7.6

The oracle model is trained on a *large* training set several orders of magnitude larger than the *narrow* training set used to train the learner model.

Let $\mathcal{D}_O = \{(x, y)\}$ be the entire available set of image-label pairs used by the oracle network O . \mathcal{D}_O is split into subsets T_O and V_O , respectively representing the training and validation sets.

By extension, let $\mathcal{D}_L = \{(x, y, R(x))\}_{(x,y) \in \mathcal{V}_O}$ be the set of image-label-rationale triplets derived from the oracle's validation set \mathcal{V}_O . From \mathcal{D}_L we randomly extract subsets \mathcal{T}_L and \mathcal{V}_L , respectively denoting the training and validation sets for the learner model (Figure 7.6). Let us underline that the two learning sets are completely independent. The oracle model typically obtains good results on a generic classification task and is able to provide salient activation maps for any image input. In order to convert this information into rationales that are similar to those provided by a human labeller, the activation maps are binarized ((cf). equation 7.5) into a set of bounding rectangles that correspond to the strongest regions of interest. From now on, such discretized maps will be called *simulated rationales*.

The only constraint that needs to be satisfied is that the oracle's training dataset include all the categories targeted by the learner model.

7.2.3 Training an image classifier with rationales: experimental results

In this section, we describe the datasets and the active learning settings used for our experiments. Then, we present the simulated rationales obtained from oracles models as described in section 7.2.1. Finally, we present and discuss the learning curves obtained for all methods, with varying \mathcal{T}_L sizes.

Evaluation protocol

We used the following image classification datasets:

Stanford40Actions. The Stanford 40 Actions Dataset contains 9,532 images of humans performing 40 actions [196].

Twitter Military. A homemade dataset of around 20,000 images. Images from Twitter accounts considered to be exclusively focused on military topics have been crawled, as well as images gathered from accounts centred on topics like cooking, music or fashion to constitute a class of non-military images.

For both datasets, the oracle model O is trained on the training set \mathcal{T}_O . For Stanford40Actions, we use the pre-existing train/valid split. Simulated rationales are extracted for all image-label pairs of validation set \mathcal{V}_O to constitute $\mathcal{D}_L = \{(x, y, R(x))\}$. For Military Twitter, only images belonging to the *military* category have class rationales. Images belonging to label *other* have their rationales set to $\{0\}_b \leq B$.

For both datasets, we compared the performance of various learner models presented in section 7.2.1: $L_{\text{Direct-CAM}}$, L_{DMFM} , L_{DWFM} , L_{SAS} .

To measure the influence of rationales on training, we introduce their following counterparts:

- $L_{\text{Direct-CAM}}^0$: the learner model for method **Direct-CAM**, without any CAM supervision;
- L_{DxFM}^0 : the learner model for methods **DMFM** and **DWFM**, with no dynamic masking or weighting;
- L_{SAS}^0 : the learner model for method **SAS** without self-attention supervision.

A number of k_t documents *per label* are randomly taken from \mathcal{D}_L to constitute a narrow learning set \mathcal{T}_L . Across experiments,

- k_t is taken in $\llbracket 0, 10 \rrbracket$ for Stanford40Actions, so $|\mathcal{T}_L| \in \llbracket 40, 400 \rrbracket$;
- k_t is taken in $\{1, 2, 3, 4, 5, 6, 7, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$ for Twitter Military, so $|\mathcal{T}_L| \in \llbracket 2, 100 \rrbracket$.

Similarly, k_v documents per label are taken from $\mathcal{D}_L \setminus \mathcal{T}_L$ to constitute a validation set \mathcal{V}_L .

- $k_v = 25$ for Stanford40Actions, so $|\mathcal{V}_L| = 1,000$;
- $k_v = 500$ for Twitter Military, so $|\mathcal{V}_L| = 1,000$.

With narrow training and validation sets, erroneous ground-truth labels can cause severe degradation of performances. For Twitter Military, all military documents in $\mathcal{T}_L \cup \mathcal{V}_L$ have been manually labelled by authors as actually belonging to the military class. Were actually considered “military” images containing military uniforms, military gear, combat vessels, fighter aircraft and combat helicopters.

For every value of k_t , models $L_{\text{Direct-CAM}}^0$, $L_{\text{Direct-CAM}}$, L_{DxFM}^0 , L_{DMFM} , L_{DWFM} , L_{SAS}^0 and L_{SAS} are trained using Adam optimizer, a learning rate of 10^{-4} , early stopping on \mathcal{V}_L (100 epochs), a categorical cross-entropy loss for Stanford40Actions and a binary cross-entropy loss for Twitter Military (in addition to rationales loss of methods **Direct-CAM** and **SAS**). For every value of k_t , results are averaged over 8 experiments.

Oracle model

For both datasets, the oracle model O used to extract simulated rationales is trained on the *large* training set \mathcal{T}_O . This model has the same architecture as $L_{\text{Direct-CAM}}$. On validation set \mathcal{V}_O , this model achieves a 0.813 categorical accuracy on Stanford40Action and 0.905 accuracy on Twitter Military.

Simulated rationales

Samples of simulated rationales are presented in Figures 7.7 and 7.8. For all methods, regions retained as simulated rationales are distinctive of the image’s label. For Stanford40Actions, the produced rationales are centred on the tool, object, body part or animal specific to the action. For Twitter Military, the produced rationales focus on camouflage textures, uniforms and military vehicles. We observe that near and/or overlapping bounding boxes tend to be selected indifferently. We claim that this is not an issue because (1) overlapping bounding boxes hold similar

content, thus are described by similar feature vectors; (2) if a human was asked to draw an arbitrary bounding box as rationale, several overlapping object proposal regions could match and would be kept in $R(x)$. The incorporation of un-simulated human bounding boxes into the proposed methods is out of the scope of this paper and could be tackled in future work. Finally, we observe that the oracle model tends to arbitrarily give more importance to some objects to the detriment of some very similar others. For example, only two soldiers out of three are retained as rationales in Figure 7.8-b. Aware of this shortcoming, proposed methods **Direct-CAM**, **DMFM**, **DWFM** and **SAS** for classification with rationales do not penalize a learner model that would pay attention to extra, unprovided regions.

Quantitative results

Figures 7.9 and 7.11 present the learning curves obtained on both datasets. On Stanford40Actions, method **SAS** is the most effective by a broad margin. It is 5 to 10 % above other methods and up to 7 % above its unsupervised-attention counterpart. On the other hand, weighting method **DWFM** shows interesting results when training data is very scarce, but method **SAS** appears to be superior, with a significant increase of performances starting from around 30 images. The baseline **Direct-CAM** model is under-performing on both tasks.

We proposed and evaluated strategies to take rationales into account when training an image classification model with a narrow training set. Such narrow training set setting is met especially during the earliest stages of an active learning scenario. At each active learning iteration, the underlying classification model is trained, then unlabelled documents are sampled according to some sampling criterion. In the next section, we propose and discuss sampling criteria tailored to enrich the training set with images knowing they will be given a label and a set of rationales.

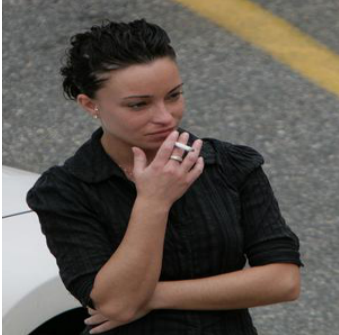








	label	image	$R(x)$	$\cup R(x)$
a.	smoking			
b.	texting			
c.	rowing a boat			
d.	riding a horse			

Figure 7.7: Simulated rationales from oracle model, Stanford40Actions dataset.

label	image	$R(x)$	$\cup R(x)$
a. military			
b. military			
c. military			
d. military			

Figure 7.8: Simulated rationales from oracle model, Twitter Military dataset. No rationales were simulated for non-military images.

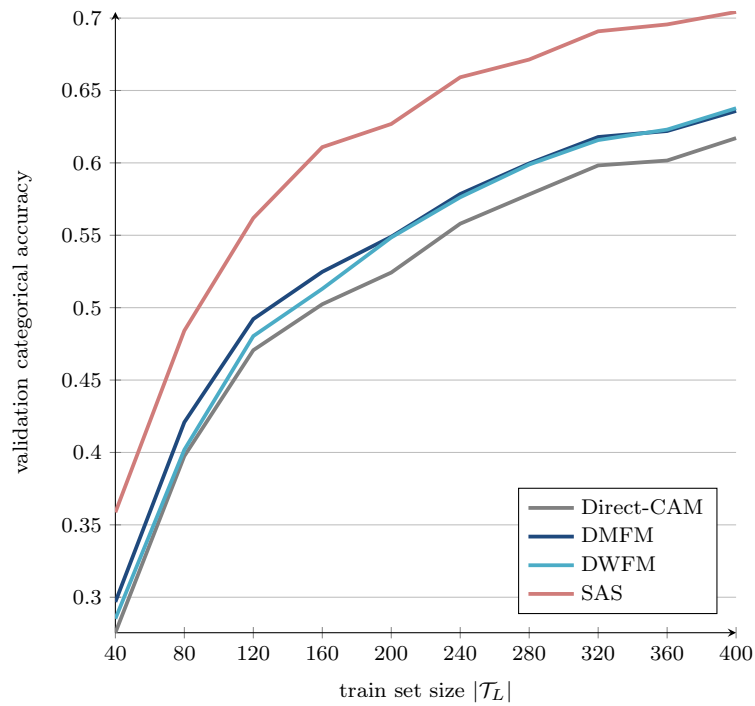


Figure 7.9: Stanford40Actions: validation accuracy when using rationales as a function of the training set size.

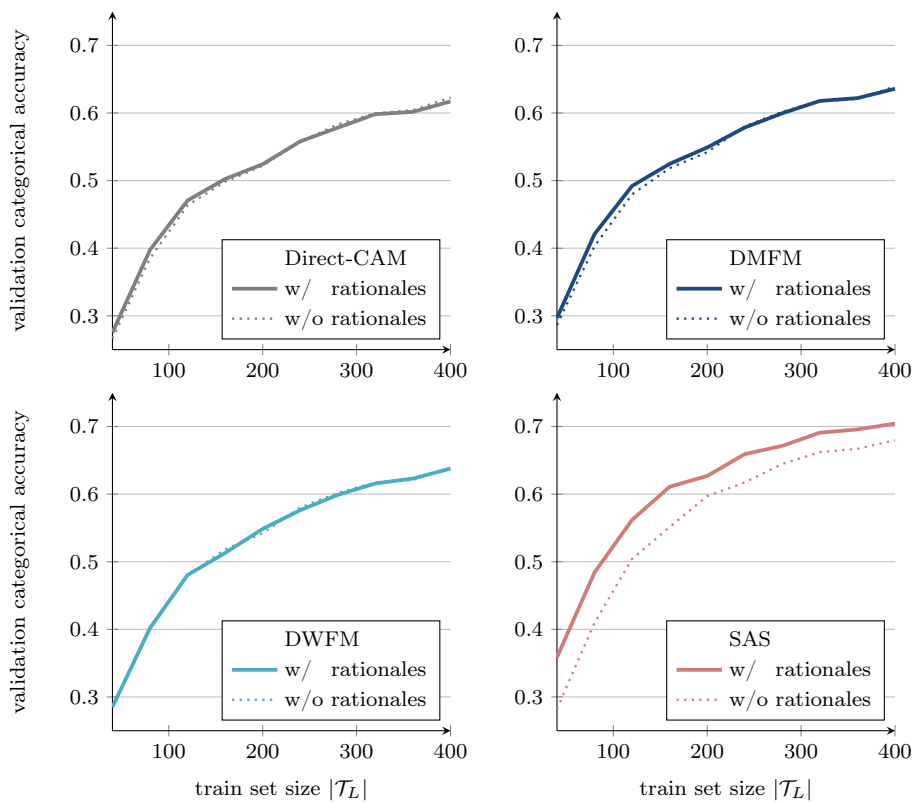


Figure 7.10: Stanford40Actions: validation accuracy using rationales during training (w/ rationales) and not using them (w/o rationales) for every studied method.

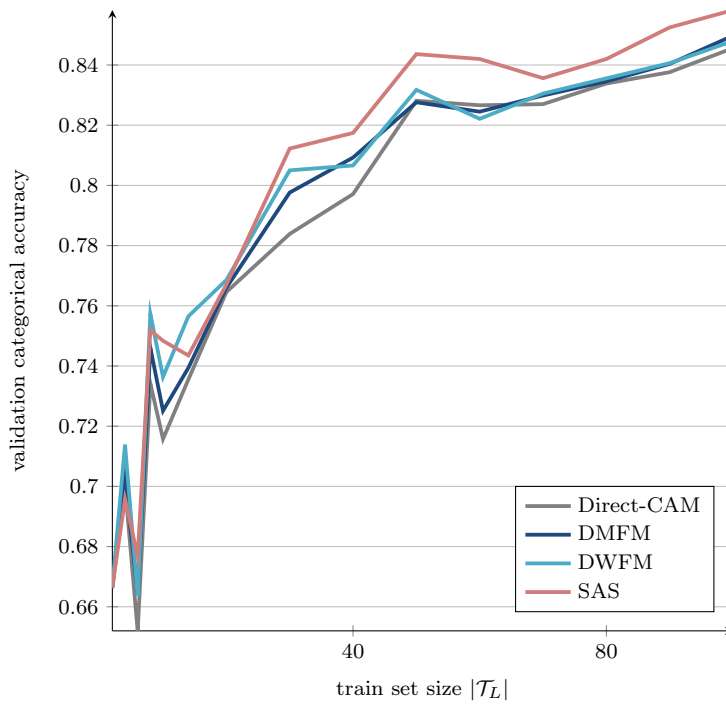


Figure 7.11: Twitter Military: validation accuracy when using rationales as a function of the training set size.

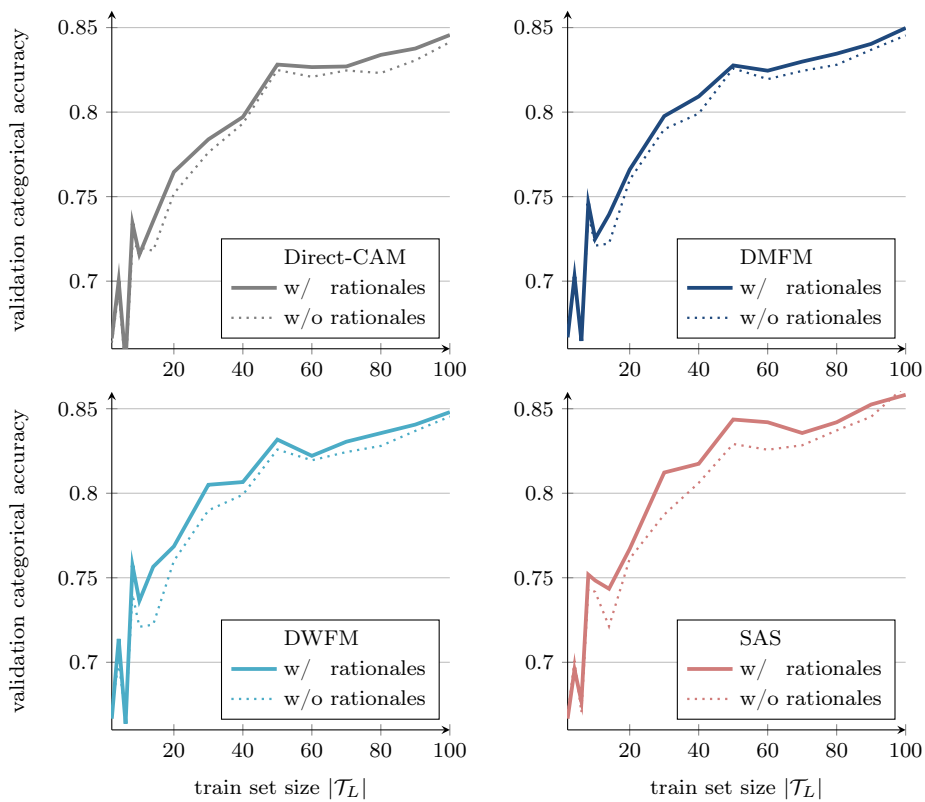


Figure 7.12: Twitter Military: validation accuracy using rationales during training (w/ rationales) and not using them (w/o rationales) for every studied method.

7.3 Sampling strategies for active learning with rationales

In this section, we propose sampling criteria meant to be exploited alongside the training of a learner model L with rationales. Experimental results are presented and discussed.

Experimental results presented in section 7.2.3 showed that among all proposed methods, attention supervision of a self-attention model (method **SAS**) is the best method for training an image classification model with rationales on a *narrow* set of images. Thus, in this section, we consider solely the $L = L_{\text{SAS}}$ learner model. Here, the objective is to sample documents for which the label-rationale pair will yield the largest improvement of the model, starting from early learning stages.

7.3.1 Notations

Let $\mathcal{D}_L = \{(x, y, R(x))\}_{(x,y) \in \mathcal{V}_O}$ represent the image-label-rationales dataset available for both the learning and the validation phases. \mathcal{D}_L is split into two disjoint sets: a training set \mathcal{T}_L and a validation set \mathcal{V}_L .

Before the sampling stage of the n^{th} iteration, \mathcal{T}_L is split into the pool of untreated sampling candidates \mathcal{U}_n and the current training set \mathcal{T}_n . The set of documents already sampled at the n^{th} iteration is denoted by \mathcal{S}_n . At the beginning of the training process, we set $\mathcal{T}_0 = \emptyset$ and $\mathcal{U}_0 = \mathcal{D}_L \setminus \mathcal{V}_L$. The set of rationale regions, described in terms of feature vectors, at iteration n , is denoted by R_n .

7.3.2 Sampling criteria

The proposed three new sampling criteria are detailed in the following.

a. self-attention uncertainty

Classification uncertainty is a well-known sampling criterion for active learning. It is based on the assumption that uncertain documents are more likely to help the learner model refine its inter-class frontiers. Self-attention uncertainty derives from the following assumption: rationales of high self-attention uncertainty may help training the self-attention mechanism. For each image $x = \{x_b\}_{b \leq B(x)} \in \mathcal{U}_n$, the self-attention uncertainty is defined as the entropy of the self-attention scores associated to its regions of interest:

$$\forall x = \{x_b\}_{b \leq B} \in \mathcal{U}_n, \quad \text{crit_a}(x) = -\frac{1}{B} \sum_{b \leq B} \text{self-att}(x_b) \log \text{self-att}(x_b) \quad (7.13)$$

Let us note that the self-attention uncertainty criterion is computed for each image in the training set, independently of any other sample previously provided to the learner model.

b. diversity w.r.t. previously provided rationales

Before the n^{th} sampling iteration, we already have rationales in \mathcal{T}_n under the form of features vectors of regions marked by the oracle as distinctive for some ground-truth label y . Inspired by the diversity criterion, we propose to sample images having no regions similar to a previously provided rationale. The criterion is the smallest distance between a feature vector x_b of x and the features of all previously provided rationales:

$$\forall x = \{x_b\}_{b \leq B} \in \mathcal{U}_n, \quad \text{crit_b}(x) = 1 - \max_{b \leq B, r \in R_n} \text{sim}(x_b, r) \quad (7.14)$$

where $\text{sim}(x_b, r)$ denotes the cosine similarity between the 2048-dimensional features describing regions x_b and r , provided by the Faster-RCNN model.

c. self-attention and Grad-CAM misalignment

Let x be an image belonging to class y . Because x belongs to y , the most salient, attention-worthy part of x is likely to be distinctive of label y . We want the self-attention scores $\hat{S}(x)$ extracted from the learner model L to be focused on the salient parts of x responsible for membership to class y . Also, the saliency map for any class c according to learner model L can be explicitly obtained as the Grad-CAM of x w.r.t. label c . Since the self-attention mechanism of L is trained with the rationales of sampled documents, we formulate the assumption that L is most likely to benefit from the rationales of images with the lowest alignment between self-attention and Grad-CAM. We define our alignment criterion as follows:

$$\forall x \in \mathcal{U}_n, \quad \text{crit_c}(x) = 1 - \min_{c \leq C} \text{sim}(\text{self-att}(x), \text{Grad-CAM}_c(x)) \quad (7.15)$$

where sim is the cosine similarity.

We compare our proposed sampling criteria to state-of-the-art sampling criteria such as uncertainty, representativeness and diversity [104, 160, 41]. We also include in our comparison a simple random sampling strategy.

For recall, the **uncertainty criterion** crit_d is computed as the entropy of predictions \hat{y} :

$$\forall x = \{x_b\}_{b \leq B} \in \mathcal{U}_n, \quad \text{crit_d}(x) = - \sum_c \hat{y}_c \log \hat{y}_c \quad (7.16)$$

The **diversity criterion** crit_e is defined as:

$$\forall x = \{x_b\}_{b \leq B} \in \mathcal{U}_n, \quad \text{crit_e}(x) = 1 - \max_{x' \in \mathcal{T}_n} \text{sim}(x, x') \quad (7.17)$$

where $x = \frac{1}{B} \sum_{b \leq B} x_b$ and sim is the cosine similarity.

Finally, the **representativeness criterion** `crit_f` is computed as follows:

$$\forall x = \{x_b\}_{b \leq B} \in \mathcal{U}_n, \quad \text{crit}_f(x) = \frac{1}{100} \sum_{x' \in 100\text{nn}(x)} \text{sim}(x, x') \quad (7.18)$$

where $100\text{nn}(x)$ are the 100 nearest neighbours from x according to the euclidean distance between their feature maps, and sim is the Euclidean similarity.

7.3.3 Active learning with rationales: experimental results

Evaluation protocol

First, the self-attention learner model is pre-trained with a balanced set of 40 images for Stanford40Actions, with simulated rationales. Then, for all criteria, at every iteration, the training set \mathcal{T}_n is enriched with the top 5 images from \mathcal{U}_n according to their updated sampling criterion and the learner model is re-trained.

Quantitative results

sampling strategy	1. average validation accuracy	2. improvement with rationales
random sampling	0.380	0.024
a. self-att. entropy	0.403	0.063
b. diversity w.r.t. rationales	0.403	0.063
c. self-att./Grad-CAM misalignment	0.446	0.066
d. entropy	0.378	0.047
e. diversity	0.423	0.052
f. representativeness	0.376	0.077

Table 7.1: Active learning results on validation data. Results are averaged over active learning iterations (omitting first bootstrap iteration).

Figure 7.13 shows the evolution of validation accuracy over active learning iterations for all presented sampling criteria. Column 1 of Table 7.1 shows these values averaged over all iterations. We observe that our proposed attention/Grad-CAM misalignment sampling criterion produces the best performance among all compared criteria, especially when the train set contains over 100 images. In our experiment, the self-attention entropy criterion produced better results than well-known predictions entropy criterion. However, diversity w.r.t. rationales is beaten by the classical diversity criterion. We note that well-known criteria like uncertainty and representativeness perform worse with rationales than random sampling, whereas proposed criteria `crit_a`, `crit_b`, `crit_c` are in all cases superior to random sampling.

Parallel to trainings with rationales, we conducted iterations of active learning with the same sampling criteria but without self-attention supervision. Figure 7.14 shows the comparative improvement brought by self-attention supervision at various iterations, for all sampling criteria. These results are averaged in Table 7.1, column 2. We

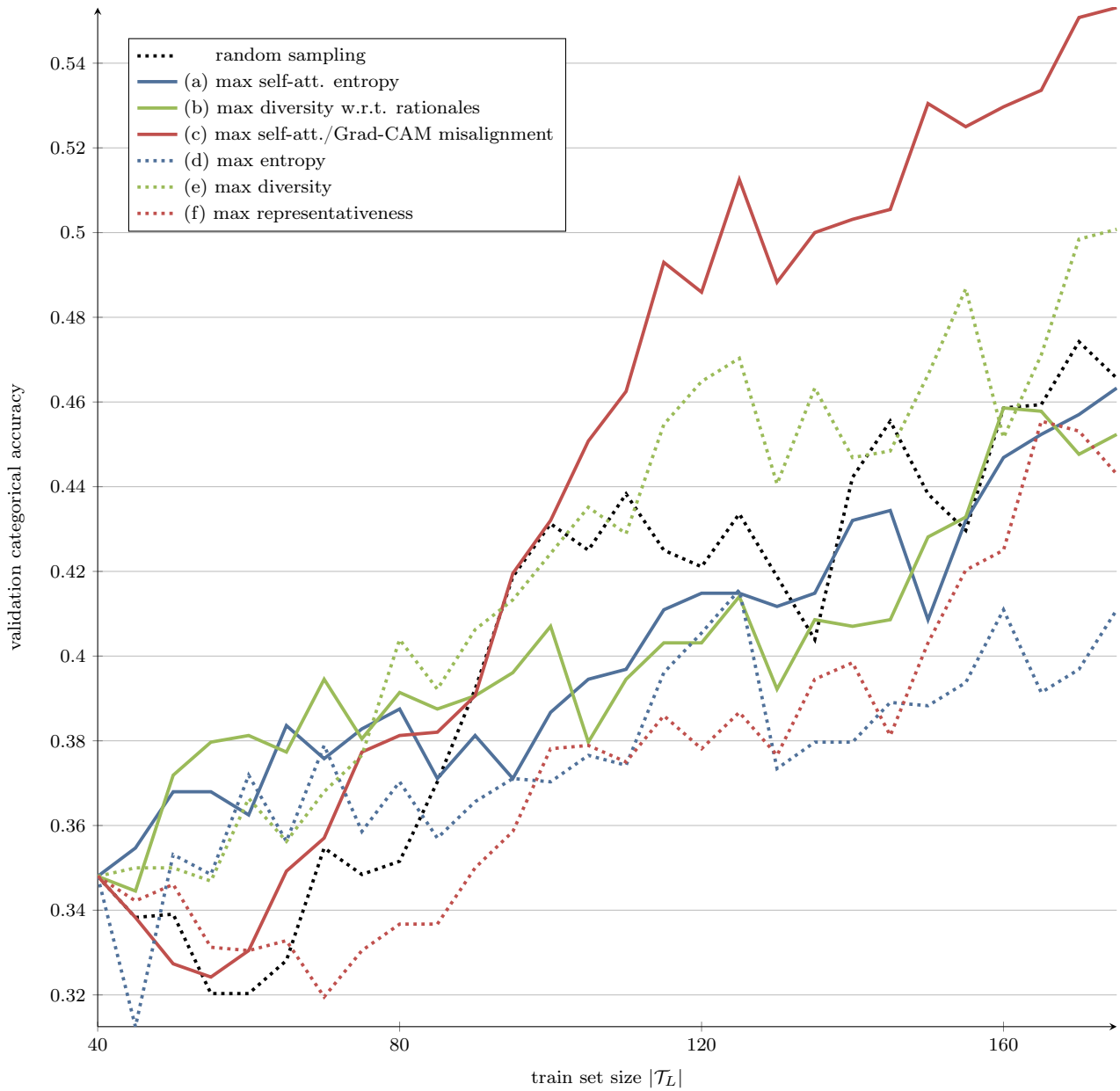


Figure 7.13: Learning curves obtained with different sampling criteria

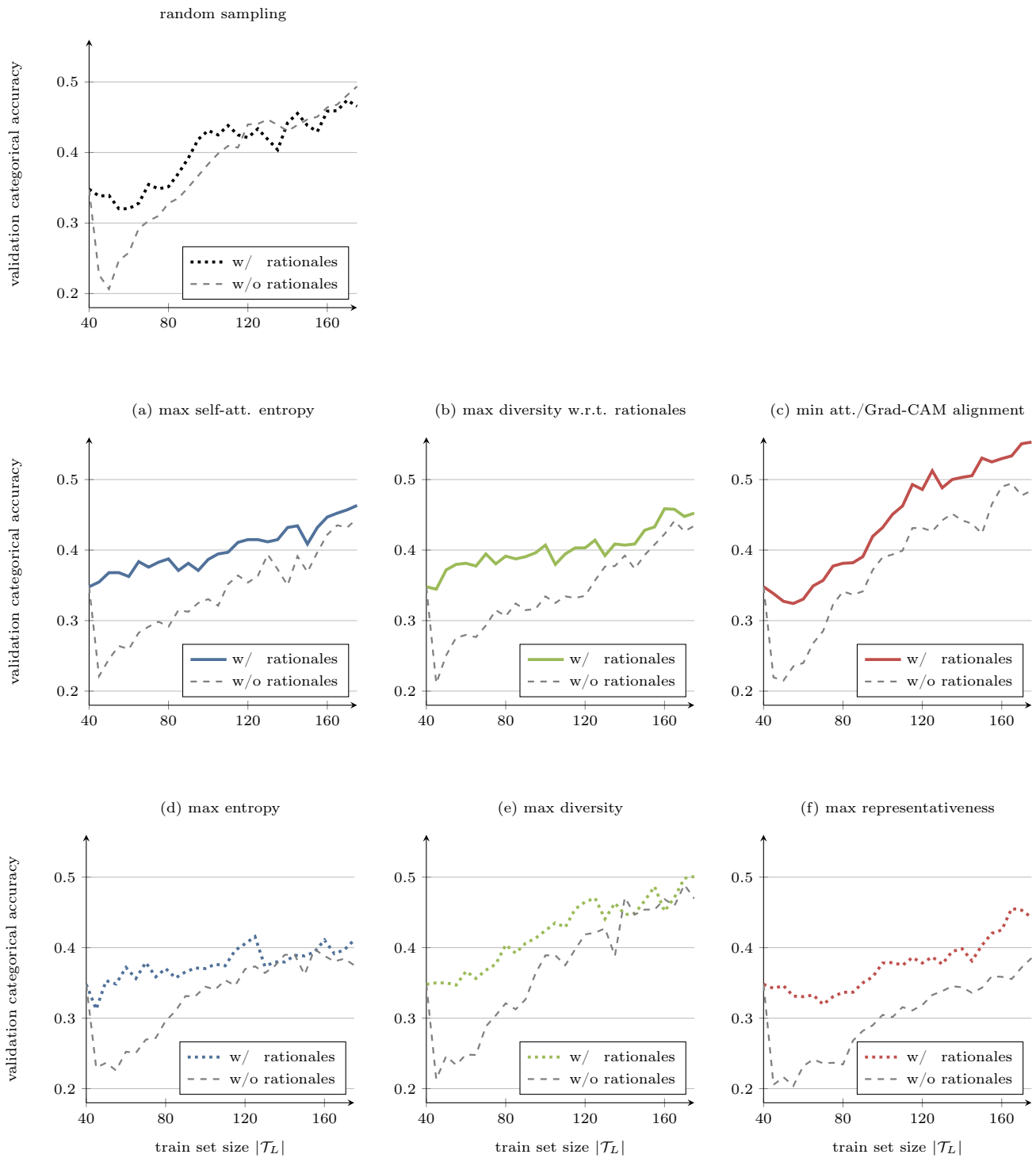


Figure 7.14: Validation accuracy using rationales during training (w/ rationales) and not using them (w/o rationales), for every compared sampling criterion.

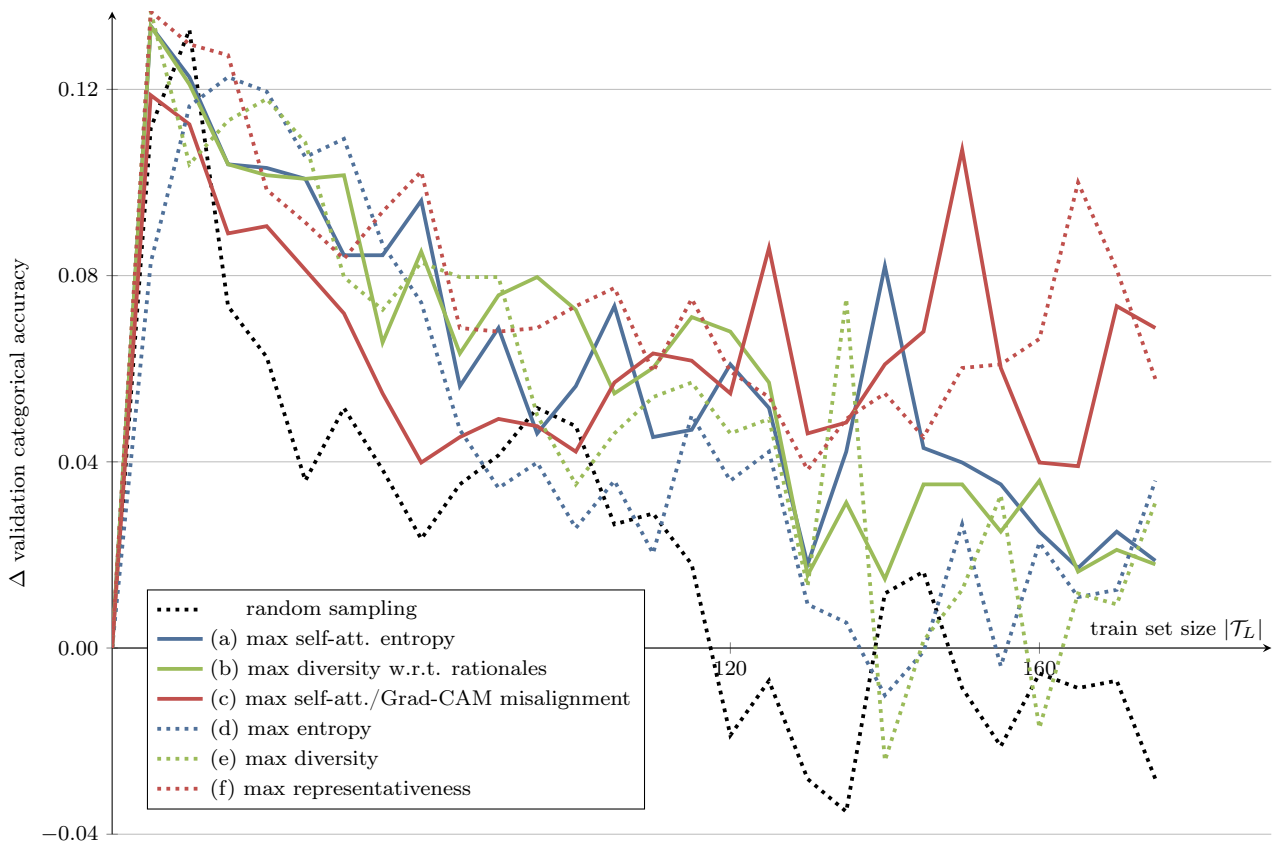


Figure 7.15: Difference in validation accuracy for each sampling criterion, between using rationales during training versus not using them.

observe that self-attention supervision is significantly beneficial and makes representativeness criterion `crit_g` and proposed criteria `crit_a`, `crit_b` and `crit_c` behave more efficiently.

Usually, a combination of several sampling criteria are used simultaneously to choose unlabelled images [105]. For instance, representativeness can prevent other criteria from sampling outliers and diversity can prevent other criteria from adding near duplicates in the training set. Here, criteria were benchmarked on their own and left free to assemble an unbalanced training set. However, even in these conditions the robustness of our proposed criterion based on the lack of alignment between Grad-CAM and self-attention showed interesting results. Moreover, it is also usable without rationales.

7.4 Conclusion and perspectives

Our results have shown that the incorporation of simulated rationales in deep neural networks for image classification with a scarce dataset is effective. The direct supervision of a self-attention mechanism built on top of a high level feature extractor can bring significant improvement to an action classification task where informative content is strongly localized. For a different task like topic classification of images, where informative content is more diverse, a straightforward method like under-weighting uninformative spatial features can help “kickstart” a neural network with less than 10 images. To automatize our evaluation, spatial rationales were simulated by an *oracle model* and obtained *via* class activation mapping. Finally, we proposed three sampling criteria for active learning explicitly based on the spatial intrinsic information of images. Our results have shown that our proposed criteria favouring the lack of alignment between self-attention and Grad-CAM surpass state-of-the-art criteria such as uncertainty, representativeness, diversity and random sampling.

This chapter opens up a perspective for the use of real, non-simulated human rationales and for the use of rationales in multi-modal active learning.

Part III

Technical Solution

Chapter 8

Real-Time Active Learning as Micro-Services

Abstract

News articles analysis may be oversimplified when restricted to detecting classes of interest already benefiting from trustworthy labelled datasets, like political affiliation or fakeness. Behind an apparent neutrality, an editorial slant may be embodied by favouring one-sided interviews, avoiding topics or choosing oriented illustrations. These challenges, seen as machine learning problems, would require a tedious annotation task. We introduce ReALMS, an active learning framework capable of quickly elaborating models which detect arbitrary classes in multi-modal text and image documents. Evidence of this capability is given by a case study on French news outlets: the detection of subjectivity, demonstrations and violence.

8.1 Introduction

The last five years have seen the worldwide dissemination of the *fake news* buzzword, associated with various phenomena in politics and in the media. The role of journalism as information providers also includes a part of *attention retainer*, with some news outlets apparently renouncing to information (replacing it with *contents* or *stories*).

The machine learning domain already proposes great algorithms to tag news articles as “normal” or as “fake-looking”. Limiting the analysis to single documents, classic approaches are automatic fact verification [11] and classification based on stylistic features [84, 45]. While successfully detecting rubbish news, these works seem to dismiss the fact that news outlets are entitled to promote opinions, tribunes, open letters and editorials.

Political science has a long history of studying the political leaning of news outlets [142], counting the number of articles per topic and scrutinizing the coverage of events with regard to their impact in an election. These analyses are particularly focused on the United States bipartisan political system, where supporters of a party must be opponents of the other. However, this does not reflect the realities of many other countries, cultures and languages.

We take some distance from classic US-centred machine learning aided analysis, motivated by the following reasons: as non-US people, we have no working knowledge of US politics. Other countries often expose a political system that is not based on bipartisan dialogue, which results in a more subtle political stance detection challenge. Semantics in politics evolve quite fast, underlining the importance to have continuously learning classifiers. Finally, we wanted to work on our own language, which is French.

In this chapter, we propose to consider two new tasks of press article classification: one-sided opinion classification, and on a limited topic, violence level classification. We have the intuition that these challenges will help to better understand the media landscape. We instantiate these classifiers through a novel active learning framework, named ReALMS, agnostic to the labelling user’s availability, thus bringing substantial advantages in terms of the effective time spent to obtain a satisfying classification model.

This chapter is structured as follows: Section 8.2 gives some details about the news article classification problems and describes previous approaches. Then we introduce ReALMS, our active learning framework, in Section 8.3. Section 8.4 presents the collection of Reddit data used to carry out our real life case study of French press articles and the training of ReALMS on this data, and the obtained models are put to contribution in to characterize news outlets in a novel manner, resulting in an insightful analysis of the French media landscape. Finally, Section 8.5 concludes this chapter.

8.2 Related Work: News Articles and Active Learning

8.2.1 Analyzing Media and Politics

An intuitive scale for media analysis consists in measuring the political leaning of all major US news outlets along a “liberal-conservative” axis, which can be based on newspaper headlines, using sentiment analysis, and taking into account economic events [112]. This kind of investigation has also been performed while limited to *editos*, that is excluding objective, purely informative articles [66].

Media bias can be measured without reading any article: a way to do so is to rely on demographics data supplied by advertisement companies such as Facebook [146], resulting in an efficient qualification of news outlets along ideological (liberal-conservative), identity (age, race, gender) and income. Another way is to follow the *retweet* links on a social network, tagging each news outlet account with a red/blue colour [191]. The political non-binary stance attribution has only been the focus of a few works; only recently have some datasets been constituted, opening this as a frame of social network analysis [43].

Previous machine learning based analysis provided invaluable insight. First, that most news outlets are frequently “informative only” (i.e., non-partisan). Second, that the agenda setting behaviour is probably more subtle than initially thought, as “*little evidence exists of systematic differences in story selection*” [20]. The growing success of AI techniques results in attempts to rationalize the analysis of newspapers in a current context of “fake news”: either to detect fake pieces of information (fact-checking) [11], or stylistic features correlated with low-quality journalism [45].

Various definitions and categorizations have been proposed in the domain of fake news analysis; basically, a distinction is done between “serious fabrications”, when news articles are forged, mentioning events that never happened, “hoaxes” or rumours that only aim to be spread (and are sometimes referred to as *bullshit*), and “satire”: fabrications with an obvious humorous goal (e.g., *The Onion*¹) [150].

Overall, the term “fake news” refers both to the globally speaking post-truth informational space, and to pieces of information that are intentionally diffused, while knowing they are false. The importance of stylistic features and emotion intensity has been underlined in a recent survey work [205], suggesting that “persuasive” news outlets prefer to insist on impacting content, up to fear-inducing, such as raw violence.

Violence detection is often seen as a video analysis task, with a direct exploitation concerning movies [88]. For textual news articles, some resources exist, mostly focusing on the topic of gun violence in the United States; they enable topic classification and entity identification [135]. Another stream of work investigates the “framing”, i.e. the main aspect of gun violence that is relayed by a given article, instantiated as a classification task (Politics, Economic consequences, 2nd Amendment, Gun control, Ethnicity, ...) [111].

¹<https://www.theonion.com/>

8.2.2 The Need for Active Learning

Automating this analysis is possible as current technologies propose effective methods to process text and image modalities. News articles are text-centred documents: text classification tasks have been widely addressed in numerous applications, among which emotions, sentiments and opinions classifications [2, 22, 48], language identification [21] or irony detection [23]. The processing of textual data is usually addressed by two main processes: feature extraction and then classification. Widespread text feature extraction methods are pre-computed word embeddings according to which words are represented by fixed-length vectors such as Word2Vec [123], GloVe [138], FastText[77], or contextual word embeddings that produce vectors by considering the neighbouring words in sentences (ELMo [140], BERT [37]). Text classification can be done by feeding word embeddings to 1-dimensional convolutional neural networks (CNN) [89] or recurrent neural networks (RNN) [28]. Images are also present in news articles and are a significant part of their embedded previews on social media. A very popular solution to image classification is transfer learning [132]. CNN architectures such as VGG [167], GoogLeNet[172], ResNet [61] or EfficientNet [174] offer performing and versatile feature extractor trunks when pre-trained with very broad datasets like ImageNet [153] beforehand or Common Objects in Context [107] beforehand.

The aforementioned text and image classification methods often rely on a significant number of labelled instances, ranging from thousands to millions. Evolving trends and the diversity of taxonomies make so that many problems lack or will lack the reliable datasets that could ease their solving. A solution consists in considering the active learning process [181, 8]. During an active learning iteration, new documents are either synthesized or drawn from a pool – or stream – of unlabelled documents according to criteria dictated by a sampling strategy. They are then presented to a human expert – or oracle – for labelling. The key principle of active learning consists in selecting the examples which are the most likely to benefit training. Multiple sampling criteria can be considered, such as uncertainty [103, 31] , information density (or representativeness) [105? , 159], diversity [19, 41] or query-by-committee [34].

8.2.3 Discussion

First, we estimate that too few works explore the non-US, non-bipartisan setups. Most systems are specifically designed for English language news analysis, while most of the world does rely on non-English “local” media. Thus, we propose to explore the French media landscape through one online social entry point, the Reddit r/france page.

Second, most of the related work requires comprehensive expert-annotated datasets, which limits thoroughly the possible questions. To empower the analyst to feel free to ask and research whatever question they need, we introduce ReALMS (Real-time Active Learning as Micro-Services), an active learning framework, enabling to sketch and refine analytics from non-annotated corpora and datasets. The use cases displayed in this work showcase the opportunities offered by active learning in general and by our proposed framework in particular.

8.3 Real-Time Active Learning as Micro-Services

To solve a classification problem, active learning encompasses methods and practices exploiting a large pool (or stream) of unlabelled documents and the expert labelling of as few documents as possible. It is highly compatible with the processing of open-source content, that is widely available by definition but may lack the descriptors for which an analyst would be interested in finding. Our motivation is that active learning could provide analysts with a powerful tool to assist them in their work.

8.3.1 Motivations

Usually, the evaluation of an active learning method is based on a separate testing pool: by measuring a performance metric (e.g. accuracy) after consuming a labelling budget, or by counting the required labelling steps before a desired metric threshold is reached. More generally, active learning performance can be assessed by the concavity of the learning curve, which plots the performance metric as a function of the (growing) training set. Granted that the sole number of labelling is a determining cost, it alone conceals crucial drawbacks of active learning when conducted in real life conditions:

1. The active learning process is usually carried out as a succession of *iterations* or *cycles*, during which several time-consuming steps are executed one after the other, including training, sampling and description of incoming unlabelled documents. This raises two issues: although precious, this elapsed time is usually not accounted for. Moreover, it may interfere with the availability of a busy human oracle.
2. In real-life experimental conditions, there exists no independent validation set that can be used to measure the improvement of the trained model.
3. The setting of the model's hyperparameters is often determined by the time and data consuming cross-validation of various configurations. Optimal hyperparameters would become obsolete with the growing population of the training set, thus new configurations must be continuously investigated.

To address these drawbacks, we propose ReALMS (Real-time Active Learning as Micro-Services), a technical solution in the form of a set of cooperating docker [122] microservices, carrying out the tasks required by active learning: receive documents and labels, train neural networks, update document descriptions, and sample documents that are candidates for labelling. Our proposed solution includes a variety of supervised models applied to a variety of modalities: as part of this work, on text and on multi-modal (text and image) classification. Furthermore, we claim that our proposed technical solution can be quickly integrated in a broader media analysis / business intelligence framework.

The active learning process is usually carried out as a succession of *iterations* or *cycles*, during which several steps are executed one after the other. Algorithm 2 aggregates the steps constituting an active learning iteration with a

fixed labelling budget: update descriptions, sampling, labelling, and training. These four steps are time-consuming and are blocking the others while they are unfinished.

Algorithm 2: Sequential active learning

Data: (i) An unlabelled pool/stream \mathcal{U} ; (ii) an initial training set \mathcal{L} ; (iii) a labelling budget $B \in \mathbb{N}^*$

Parameters: Number of sampled documents per iteration s

Result: A classifier \mathcal{M} trained on labelled data

while $B > 0$ **do**

 update description of all documents in \mathcal{U} ;
 $s \leftarrow \min(B, s)$;
 sample s documents $\{x_1, \dots, x_s\}$ from \mathcal{U} according to sampling criterion;
 $\mathcal{U} \leftarrow \mathcal{U} \setminus \{x_1, \dots, x_s\}$;
 wait for their gold labels $\{y_1, \dots, y_s\}$;
 $\mathcal{L} \leftarrow \mathcal{L} \cup \{(x_1, y_1), \dots, (x_s, y_s)\}$;
 train classifier \mathcal{M} on labelled data \mathcal{L} ;
 $B \leftarrow B - s$;

end

Update of descriptions. The sampling criterion usually relies on sub-criteria designed to sample the most informative documents such as uncertainty, representativeness and diversity. The global sampling criterion – or *sampling score* – is a combination of all sub-criteria. The uncertainty criterion requires obtaining prediction scores for each document in \mathcal{U} according to the last model \mathcal{M} . The representativeness and diversity criteria require extracting a feature vector for each document in \mathcal{U} . \mathcal{M} being a neural network, the output of deep, trainable layers is susceptible to embed high-level semantic information. In this case, the extracted deep features must be updated at each step. Both prediction and feature extraction require the time-consuming processing of each unlabelled document by \mathcal{M} .

Sampling. The representativeness sub-criterion requires the time-consuming task of comparing the candidate document with other documents to penalize outliers. The diversity sub-criterion requires the time-consuming task of comparing the candidate document with already labelled documents. The overall sampling score must be computed for all candidate documents.

Labelling. The labelling of sampled documents is a strong bottleneck. The human oracle may take any amount of time to label a sampled document, may discard documents, and may be unavailable for long periods of time.

Training. Depending on its number of trainable parameters and the size of its training set, fitting a neural network may take from seconds to days. Moreover, a common mechanism like early stopping may result in inconstant training times. Having a busy oracle waiting for a long and unpredictable delay before labelling is a major drawback.

Within our proposed technical solution, these tasks are carried out by interconnected microservices having access to each other's results in real time. Figure 8.1 illustrates the organization of our proposed solution. These microservices and their overall organization are described in the following subsections.

8.3.2 Notations

Throughout this chapter, the following notations are used. To solve multi-class or multi-label classification problem with c classes, in an active learning setting, $\mathcal{U} = \{(x_i, \hat{x}_i, \hat{y}_i)\}_{i \leq u}$ is the set of all unlabelled documents stored in the framework, represented by their frozen embeddings x , their deep vector representation \hat{x} and prediction scores $\hat{y} \in \mathbb{R}^c$ (cf. section 8.3.3.) $\mathcal{L} = \{(x_i, \hat{x}_i, y_i)\}$ is the set of all labelled documents stored in the framework, where $y \in \{0, 1\}^c$ is the ground-truth label provided by an expert oracle. \mathcal{L} is split into training set \mathcal{T} and validation set \mathcal{V} . The best classification model at a given moment is noted \mathcal{M} , while a candidate model being trained is noted M . For the sake of simplicity, we liken a document to its frozen embedding x .

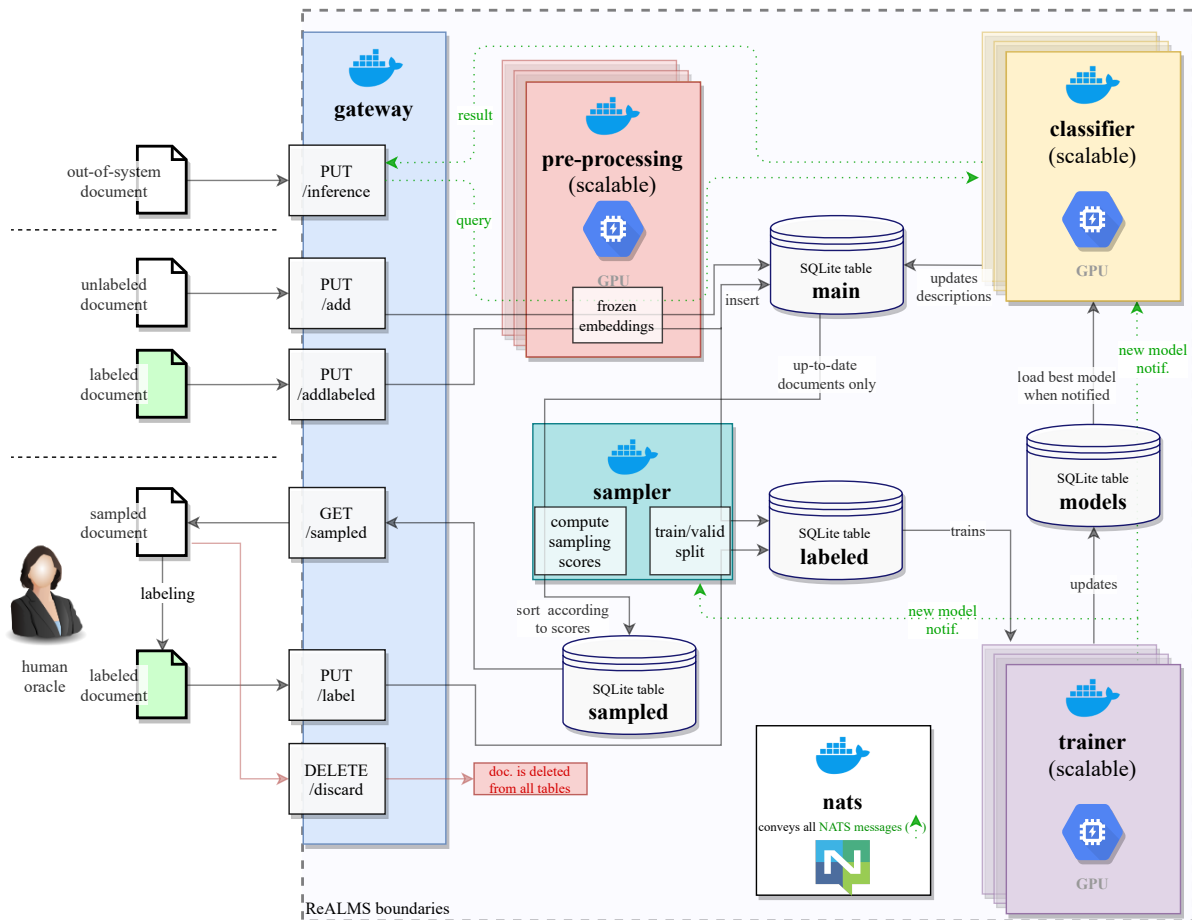


Figure 8.1: ReALMS organization. Several docker services cooperate to carry out all time-consuming tasks required by active learning.

8.3.3 Proposed Organization

Our proposed framework aims to parallelize time-consuming tasks that are usually conducted sequentially: training, sampling and description of unlabelled documents. Moreover, we blur the boundary between pool-based (sampling from a static pool of unlabelled documents) and stream-based (sample or discard incoming documents) active learning scenario by allowing the continuous reception of new documents during the lifespan of the system. The

overall organization of our framework is illustrated on Figure 8.1. The following sections describe the networked microservices.

id	text	lock	version	embeddings	features	prediction	uncertainty
1	lorem ipsum...	-	4	[...]	[...]	[...]	0.23
2	dolor sit amet...	-	4	[...]	[...]	[...]	0.31
3	consectetur...	labelled	4	[...]	[...]	[...]	0.08
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
5135	est laborum...	-	-	[...]	-	-	-

Table 8.1: Main documents table

id	labels	split
3	a	valid
103	b	valid
⋮	⋮	⋮
92	a	train

Table 8.2: labelled documents table

id	score	uncertainty	representativeness	diversity
18	0.2565	0.3662	146.27	0.00479
115	0.2560	0.3661	263.17	0.00266
⋮	⋮	⋮	⋮	⋮
3	0.0720	0.3012	191.19	0.00125

Table 8.3: Sampled documents table

Pre-processing Service.

The pre-processing service is in charge of computing frozen vector embeddings x for all incoming raw documents. Unless deleted, a document embedding is stored during the whole lifespan of the system in the *main* table of the database (displayed on Figure 8.1) to avoid multiple re-computing. The trainer service uses the frozen embeddings $\{x_i\}$ as inputs once per training epoch to train a neural network (cf. trainer service); the classifier service uses the frozen embeddings $\{x_i\}$ to update the representation of documents (cf. classifier service). Text inputs are transformed into a list of FastText 300-dimensional embeddings of their tokens. Input images are resized to 380×380 pixels and fed to a pre-trained EfficientNet-B4 trunk producing 1792-dimensional embeddings. This service is scalable: in case of a heavy stream of incoming documents, several instances can subscribe to the same NATS queue. Finally, it also produces the embeddings for on-demand inference queries from the user.

This framework is designed to receive raw unprocessed documents, as texts, images or a combination of both modalities. The pre-processing service is in charge of computing frozen vector embeddings x for all incoming documents. Unless a document is deleted, its produced embedding is stored during the whole lifespan of the system in the dedicated *main* table of the database (displayed on Figure 8.1). This avoids to re-compute the embeddings several times when they are used as inputs by the trainer and classifier services: the trainer service uses the frozen embeddings $\{x_i\}$ as inputs once per training epoch to train a neural network (cf. section 8.3.3); the classifier service

uses the frozen embeddings $\{x_i\}$ to update the representation of documents (cf. section 8.3.3). The pre-processing service tokenizes text inputs and produces a list of the FastText 300-dimensional embeddings of its tokens. Input images are resized to 380×380 pixels and fed to an EfficientNet-B4 trunk producing 1792-dimensional embeddings. The pre-processing service is scalable: in case of a heavy stream of incoming documents, several pre-processing instances can work concurrently by subscribing to the same NATS[1] queue. The pre-processing service also produces frozen embeddings for on-demand inference queries from the user.

Classifier Service.

In active learning, the classification model \mathcal{M} is intended to be upgraded several times. The classifier service always has to keep up-to-date the description of all documents: prediction results $\{\hat{y}_i\}$ and deep feature vectors $\{\hat{x}_i\}$. It is implemented as a TensorFlow container with GPU support. The service accesses the *main* table representing the pool of registered documents, represented by their frozen embeddings $\{x_i\}$. It is notified whenever a new model version is discovered by the trainer service. When this occurs, the parameters of the new model \mathcal{M} are loaded and the service updates the documents whose descriptions are obsolete, by random batches. Also, the classifier service answers on-demand inference queries from the user. Several instances can run simultaneously in separate containers to speed up the description update. From now on, documents whose description is up-to-date are called *up-to-date documents*; others are called *obsolete documents*.

During active learning, the classification model \mathcal{M} is intended to be upgraded several times. The classifier service is designed to keep the description of all documents up-to-date at any time. The description encompasses prediction results $\{\hat{y}_i\}$ and the extraction of deep feature vectors $\{\hat{x}_i\}$. It is implemented as a TensorFlow container with GPU support. The classifier service has access to a shared table (*main* table) representing the pool of every registered document represented by their frozen embeddings $\{x_i\}$. The classifier service is notified whenever a new model version is discovered by the trainer service. When this occurs, the classifier service loads the parameters of the new model \mathcal{M} and proceeds to update the documents whose descriptions are obsolete, by random batches. Also, the classifier service answers on-demand inference queries from the user. Several classifier services can run simultaneously in separate containers to speed up the description update. From now on, documents whose description is up-to-date are called *up-to-date documents*; others are called *obsolete documents*.

Sampler Service.

Sampling task. The sampler service relies on the up-to-date document descriptions $\{(\hat{x}_i, \hat{y}_i)\}$ produced by the classifier service to compute sampling scores. The highest sampling score document will be the next to be labelled by the oracle. Only up-to-date documents are candidates for sampling. Unless the list of up-to-date documents has ceased to expand, the sampler continuously carries out the sampling task. This mechanism ensures that the incompleteness of the description task is not an obstacle to sampling. In that respect, when the oracle requests a document for

Table 8.4: Sampling criteria.

Criteria	Definition	Formula
Uncertainty	the prediction vector's entropy	$c_u(x_i) = - \sum_{i=1}^c \hat{y}_i \log \hat{y}_i$
Representativeness	opposite of the mean distance of a document to its nearest neighbors	$c_r(x_i) = 1 - \frac{1}{n} \sum_{j=1}^n \text{dist}(\hat{x}_i, \hat{x}_j)$
Diversity	smallest distance between an unlabelled document and the already labelled documents	$c_d(x_i) = \min_{x_j \in \mathcal{L}} \text{dist}(\hat{x}_i, \hat{x}_j)$
Global	product of sub-criteria c_u , c_r and c_d	$C(x_i) = c_u(x_i) \times c_r(x_i) \times c_d(x_i)$

labelling at any given time, they receive best document possible at this moment.

The sampling criterion according to which unlabelled documents are sorted is the combination of three sub-criteria: uncertainty c_u , representativeness c_r and diversity c_d , which are defined in Table 8.4. The distance used is the cosine distance. Up-to-date unlabelled documents are sorted according to their global sampling scores, decreasingly, to populate a sampling queue. In practice, the sampling queue is truncated to 1000 items to reduce the database writing time.

Uncertainty criterion: defined as the prediction vector's entropy:

$$c_u(x_i) = - \sum_{i=1}^c \hat{y}_i \log \hat{y}_i \quad (8.1)$$

Representativeness criterion: based on the mean distance of a document to its nearest neighbours. The greatest this score, the more a document is similar to its neighbours.

$$c_r(x_i) = 1 - \frac{1}{n} \sum_{j=1}^n \text{dist}(\hat{x}_i, \hat{x}_j) \quad (8.2)$$

Diversity criterion: aimed at avoiding to sample documents which are similar to already labelled documents. It is the smallest distance between an unlabelled document and the already labelled documents.

$$c_d(x_i) = \min_{x_j \in \mathcal{L}} \text{dist}(\hat{x}_i, \hat{x}_j) \quad (8.3)$$

Global sampling criterion: the score according to which documents are sorted is the product of sub-criteria c_u , c_r and c_d :

$$C(x_i) = c_u(x_i) \times c_r(x_i) \times c_d(x_i) \quad (8.4)$$

Fine-tuning of diversity criterion: two documents A and B with similar (or identical) representations $f_A \approx f_B$ are likely to have similar (or identical) sampling scores and to be neighbours in the sorted sampling queue. If active learning was to be carried out sequentially, the labelling of A would lead up to $c_d(f_B)$ approximating 0, and thus B

would most likely not be sampled. However, in our case, the oracle may quickly sample and label A and B before the sampling service updates B 's diversity criterion. Here, B would be sampled despite being similar (or identical) to already labelled document A , defeating the purpose of the diversity criterion. Hence, we propose to fine-tune the diversity criterion so that it includes all above-placed documents in the sampling queue, as described in algorithm 3. Moreover, documents whose diversity score equals zero are immediately removed from the framework.

Algorithm 3: Diversity criterion fine-tuning

Data: A pre-sorted sampling queue $\mathcal{S} = \{x_1, x_2, \dots, x_s\}$ where x_1 is the first document that will be sampled for labelling.

Result: \mathcal{S} is re-sorted according to a fine-tuned diversity criterion.

```

for  $i \leftarrow 1$  to  $s - 1$  do
  for  $j \leftarrow i + 1$  to  $s$  do
     $d \leftarrow \min_{k \in \llbracket 1, i \rrbracket} \text{dist}(\hat{x}_j, \hat{x}_k);$  // find smallest distance to  $x_j$ 
     $c_d(x_j) \leftarrow \min(c_d(x_j), d);$  // update diversity criterion
     $C(x_j) \leftarrow c_u(x_j) \times c_r(x_j) \times c_d(x_j);$  // update global criterion
  end
   $j \leftarrow \operatorname{argmax}_{j \in \llbracket i+1, L \rrbracket} C(x_j);$  // rank of document promoted to rank  $i+1$ 
   $x_{i+1}, x_j \leftarrow x_j, x_{i+1};$  // swap  $j^{\text{th}}$  and  $i+1^{\text{th}}$  document in  $\mathcal{S}$ 
end

```

Training/Validation Split Strategy. The split decision is performed when a document is labelled, and it is final: a document marked as *validation* will be used for validation by all future models. ReALMS is initialized with a target validation ratio r_V dictating the proportion of labelled documents to use as a validation set \mathcal{V} , isolated from the training set \mathcal{T} . When a labelled document is received, its training/validation split is determined by its label and the current state of the validation set. The incoming document is assigned to \mathcal{V} if \mathcal{V} is lacking members of the received document's classes $y = \{y_1, \dots, y_c\} \in \{0, 1\}^c$.

$$c_V(y) = \min_{i \in \llbracket 1, c \rrbracket} \frac{|\{y' \in \mathcal{V}; y'_i = y_i\}|}{|\{y' \in \mathcal{V} \cup \mathcal{T}; y'_i = y_i\}|} \quad (8.5)$$

Thus defined, c_V accounts for the validation ratio of the most depleted label (positive or negative) in \mathcal{V} . If $c_v(y) < r_V$, the received document is assigned to validation. Otherwise, it is assigned to training.

For the particular case of multi-class classification, c_v can be simply redefined as follows:

$$c_V(y) = \frac{|\{y' \in \mathcal{V}; y' = y\}|}{|\{y' \in \mathcal{V} \cup \mathcal{T}; y' = y\}|} \quad (8.6)$$

In a multi-class setting, we also incorporate a criterion for the class balancing of \mathcal{V}

Trainer Service.

Training routine. The trainer service exploits \mathcal{T} to produce a model that performs best on \mathcal{V} . The service continuously follows a training routine described by algorithm 4. Each trained model is compared to the *best* model \mathcal{M} . Because the validation set \mathcal{V} grows over time, the f1-score obtained by the best model \mathcal{M} at the time of its creation is obsolete. For this reason, we re-evaluate \mathcal{M} on the current state of \mathcal{V} . Several trainer containers can run simultaneously to speed up the discovering of new best models. ReALMS is initialized with a random architecture ratio $r_R \in [0, 1]$ dictating the probability of starting a new training from scratch with a new model M or to re-load and resume the training of \mathcal{M} . Also, confident documents may be added to \mathcal{T} for training [190].

Algorithm 4: Training routine of trainer service

Data: Training set \mathcal{T} , validation set \mathcal{V} , unlabelled set \mathcal{U} , best model \mathcal{M}
Parameters: Random architecture ratio r_R , uncertainty upper-bound threshold t_p
Result: Successive new versions \mathcal{M} trained on \mathcal{T} and validated on \mathcal{V}

```

while true do
   $r \leftarrow \text{rand}(0., 1.)$ ;
  if  $r < r_R$  then
    |  $M \leftarrow$  new model with a randomized architecture;
  else
    |  $M \leftarrow \mathcal{M}$ ;
  pull  $\mathcal{T}$  and  $\mathcal{V}$  from the database;
   $t \leftarrow \text{rand}(0., t_p)$ ; //  $t$  is a random threshold between 0 and  $t_p$ 
  /* unlabel documents whose uncertainty is below  $t$  are pseudo-labelled */
   $\mathcal{T} \leftarrow \mathcal{T} \cup \{x_i \in \mathcal{U}; c_u(x_i) < t\}$ ;
  train  $M$  on  $\mathcal{T}$  with early stopping on  $\mathcal{V}$ ;
   $f_M \leftarrow \text{f1-score}(M, \mathcal{V})$ ;
  load best model  $\mathcal{M}$ ;
   $f_{\mathcal{M}} \leftarrow \text{f1-score}(\mathcal{M}, \mathcal{V})$ ;
  if  $f_M > f_{\mathcal{M}}$  then
    |  $\mathcal{M} \leftarrow M$ ;
    | send NATS notification to other services;
end

```

Model Architectures. Figure 8.2 represents the overall architecture of the models used in our framework. For mono-modal classification, the fusion step is bypassed. Layers outputs are regularized by dropout and weight decay and are ReLU-activated, except for the final classification layer using a softmax (respectively sigmoid) activation for multi-class (respectively multi-label) classification. We use a random number (between 0 and 4) of 1-d convolutional layers to process the text modality, each with a random number of filters (between 2^6 and 2^{10}) and a random kernel size (3 or 5). The global pooling method is chosen randomly between global max pooling, global average pooling and self-attention [109]. Text and image feature vectors are processed by a random number of dense layers (between 0 and 4) with a random number of units (between 2^6 and 2^{10}), whose output are merged following one randomly chosen fusion method out of late fusion, most certain and a multi-layer CentralNet[186] subnetwork.

- late fusion: multi-modal prediction is the average of the mono-modal predictions.

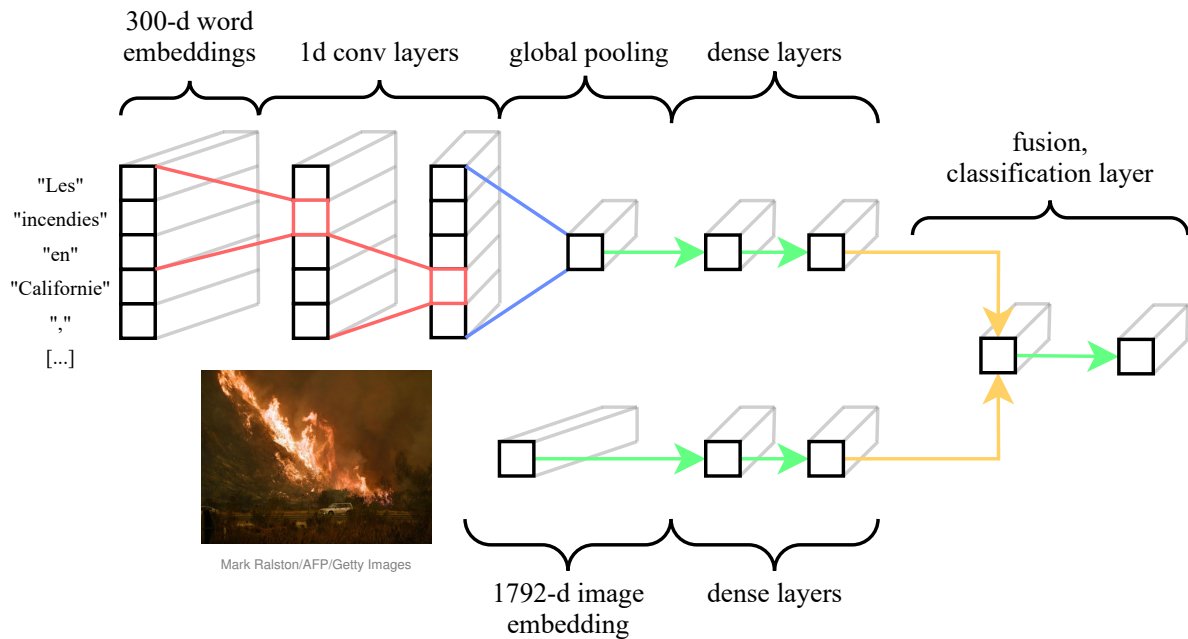


Figure 8.2: Template of models used in ReALMS

- most certain: keep the most certain output label-wise before applying softmax/sigmoid activation.
- CentralNet (sum): mono-modal representations are merged through a CentralNet [186] with a random number of layers (between 1 and 4) and random number of units (between 2^6 and 2^{10}).
- CentralNet (concatenation): a CentralNet fusion network where modalities are concatenated rather than summed.

Helper Services.

As shown on Figure 8.1, ReALMS relies on additional services. The gateway service exposes a web interface to manage all incoming documents and labels, enabling a continuous reception. It also exposes a labelling GUI as a web page. Nats service is a container running message broker NATS², conveniently conveying asynchronous queries and notifications (as dotted green arrows on Figure 8.1) without the need to expose a web interface on every microservice.

8.4 Application on French Politics

This section approaches the peculiarities of French politics, it presents the nature of the data collected fed to our framework to build our analytics tools and displays detection results.

French politics present a few differences with its US counterparts, in terms of language (French) and political landscape structure. Stance is not a binary classification task, because the centre-right government is criticized

²NATS, <https://docs.nats.io/>

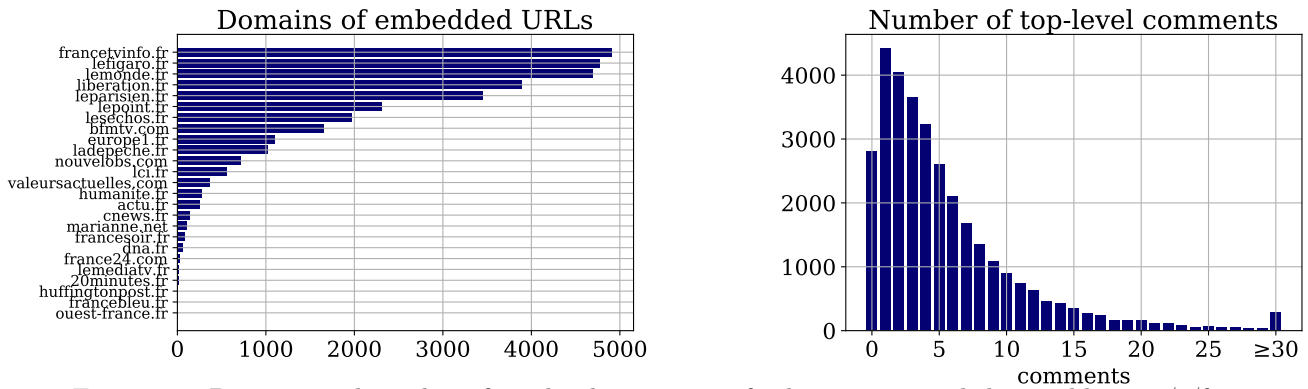


Figure 8.3: Domains and number of top-level comments of submissions crawled on reddit.com/r/france.

from both sides by left and right wings oppositions. However, as in the USA, most news articles are informative, although their political leaning is perceivable from the addressed topics. Some citizens feel like news outlets should be informative only, not promoting a stance. In this chapter, we propose to compare the emission and engagement received by objective versus opinionated articles. We believe this may help to qualify the behaviour of news outlets.

Inspired by [111] and the detection of frames (i.e., the difference in topical coverage of an event), we propose to count the references to *demonstrations* and to *violence* on articles that thematically cover the introduction of a new security law. We believe this is a first step to measure the narratives used by the different news outlets to encourage or discourage people to participate in the protests.

8.4.1 Data Collection

We aggregated data from the r/france *subreddit*³ by targeting submissions that embed URLs from a list of 25 French news outlets, using the Pushshift API⁴. The titles, bodies and main images of the embedded articles are retrieved using NewsPlease [59]. The number of top-level comments per submission is retrieved with the Python Reddit Application Programming Interface (API) Wrapper⁵. Once duplicates have been removed, we have a total of 39,611 news articles. The collected news outlets are presented in Figure 8.3, on the left. Among the most present, *francetvinfo*⁶ is the web portal of a radio and a TV, displaying mostly information reports; *lefigaro* is the liberal / right-wing reference; *lemonde* is considered centrist and *liberation* the left-wing reference.

On the upper-right side, the *flairs* are purely a Reddit thing: on r/france they are commonly used to tag or label a submission. Most submissions in the dataset do not bear any tag. Topics are not exclusive (Politics, Society, News are the most common).

On the lower-left side, the date of submission are aggregated by year. More than 9,000 news articles have been published on r/france during the year 2019, the peak of our dataset.

On the right side, the count of top level comments is displayed. On Reddit, people can comment either the

³reddit.com/r/france

⁴github.com/pushshift/api

⁵PRAW: The Python Reddit Api Wrapper, <https://praw.readthedocs.io/en/v3.0.0/>

⁶*francetvinfo*.fr and *franceinfo*.fr redirect to the same site and have been merged.

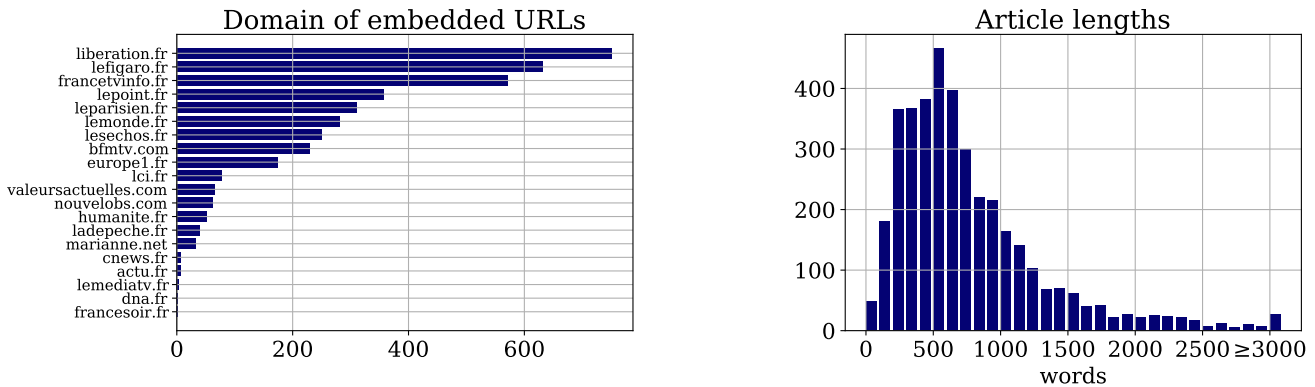


Figure 8.4: Domains and text lengths distributions of news articles crawled under the “Politique” (Politics) flair and used to train one-sided opinion detection.

submission directly (top-level comment) as well as other comments, resulting in a comment tree. Some branches do evolve into fully-fleshed discussions, often forgetting the topic of the original post: we limit the measure of popularity of an article to the number of top-level comments. Around 2,800 articles did not receive any comment; almost 1,000 have received exactly 10 comments. Mean article length is 821 words. The following subsections present the use cases and the models obtained after labelling articles with ReALMS. The resulting corpora are published⁷ in order to facilitate future research on these topics.

Data for the detection of one-sided opinion in news articles.

For our first use case, we chose to restrict our study to news article marked with the “Politics” flair, exploiting their titles and full text bodies. Figure 8.4 displays, on the left, the distribution of the number of articles explicitly tagged as politics and, on the right, their length. The most likely article length is around 500 words, which is hiding a long tail (more detailed articles being not uncommon).x

8.4.2 Detection of Articles Conveying One-sided Opinions

This text classification problem consists in detecting articles that relay one-sided opinions, such as interviews, one-sided paraphrased opinions and straight tribunes. For this first use case, we chose to restrict our study to news articles marked with the “Politics” flair. Input documents consist in the junction of their titles and bodies. This problem is tackled as a binary classification task. Target validation ratio r_V is set to $1/5$. During labelling, articles dealing with foreign politics are discarded. Figure 8.5.a shows the learning curves obtained while sampled articles were manually labelled by the authors. Vertical dotted lines stand for hours-long interruptions of the labelling task. After labelling 100 articles, our model reached 0.941 f1-score (0.889 precision, 1.0 recall) on 20 validation articles. The final neural network’s architecture is $\{conv1d(137 \text{ filters of size } 3), conv1d(124 \text{ filters of size } 3), global \text{ average pooling}, dense(66 \text{ units})\}$. 0 pseudo-labelled, confident documents were exploited during its training. Table 8.5 contains titles

⁷URL will be given in camera ready as web hosting is not solved yet

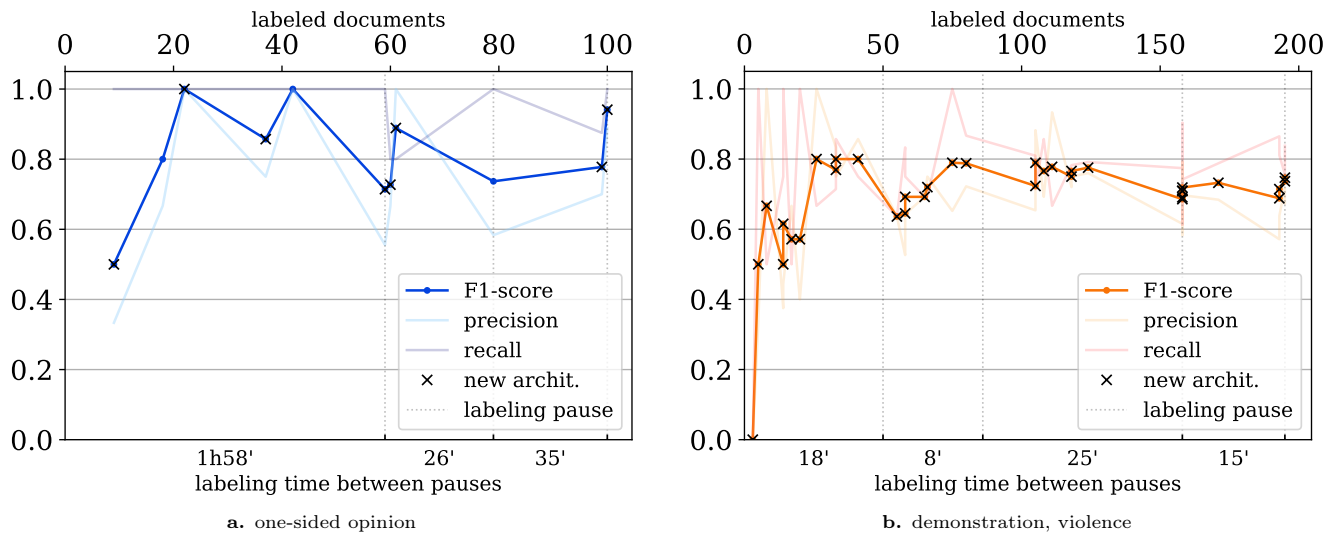


Figure 8.5: Learning curves for one-sided opinion detection (left) and demonstration/violence detection (right). “New archit.” signifies that the model was obtained with a new, random architecture, as opposed to improving the already existing model.

of unlabelled articles whose prediction scores are the strongest. Testing on 20 independent documents yields a 0.82 f1-score.

Table 8.5: Most positive and negative results for label “one-sided opinion” (from unlabelled corpus)

pred.	translated title
0.990	François Hollande warns against Donald Trump’s victory that many think is impossible
0.986	Worried about Mélenchon’s breakthrough, Hollande calls for “renewal”
0.985	VIDEO. Alain Juppé “screws” those who find him “very conventional”
0.983	Presidential election: Montebourg would call Mélenchon if he won the primary
0.982	Hollande: there has been a “serious” identity crises and “for a long time” in France
0.005	Brexit: the (simple) graphic to summarize five months of procrastination between London and Brussels
0.008	Presidential election: Macron at 24.01%, Le Pen at 21.30%, according to definitive first round results
0.008	Presidential election: 9.8 million French people followed the debate on TF1
0.009	The vigorous unblocking of the Fos-sur-Mer refinery, in 42 seconds
0.011	Nearly 224,000 migrants reached Europe via the Mediterranean in 2015

8.4.3 Demonstration and Violence

Our second use case consists in classifying text/image pairs as belonging to zero, one or two of arbitrary “demonstration” and “violence” categories. Article lacking a valid image are discarded. We used keywords to quickly filter on a controversial topic about police violence. Articles are retained if their body contain one or more of the following keywords: *manifestation*, *manifs*, *sécurité globale* (a controversial French bill), *violence(s) policière(s)* (police violence), *CRS* (French riot police), *émeute* (riot), *dégradations* (degradation). These words are not specifically stance-based, even though the different political factions either amplify or diminish the phenomenon. To emphasize the importance of the accompanying image, we chose to only use the article title as the text modality. Conveniently, title/image

target phrase	meaning
<i>manifestation, manif</i>	demonstration
<i>sécurité globale</i>	controversial French bill
<i>violence(s) policière(s)</i>	police violence
<i>crs</i>	French riot police
<i>émeute</i>	riot
<i>dégradations</i>	degradation

Table 8.6: Target phrases used to pre-filter news articles.

pairs are the constituents of the previews displayed on social media.

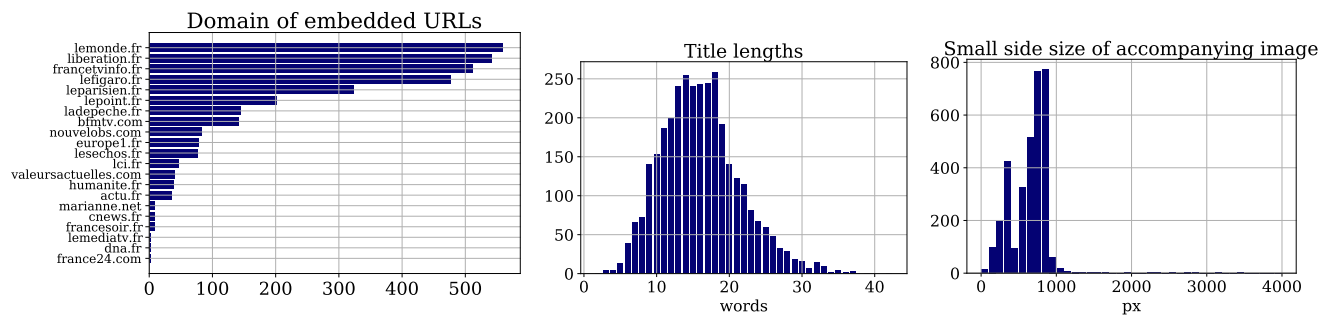


Figure 8.6: Domains, title lengths and image size distributions of news articles composing the demonstration/violence unlabelled corpus.

More descriptive elements are given about the multi-modal learning task in Figure 8.6. Among the main domains, the same reference outlets are still the most present. About the textual modality, titles are much shorter than the body of the article; ten to twenty words are the most usual. About the image modality, we identify two main clusters: low quality pictures (~ 300 pixels for the small side) and higher quality, usually between 700 and 1000px; some outliers propose very high quality pictures).

This text and image classification problem consists in detecting articles previews that mention or depict demonstration and/or violence. Input documents consist in article previews as they appear when they are shared in social networks: a title and an accompanying image. This problem is tackled as a multi-label classification task. Target validation ratio r_V is set to $1/4$. During labelling, articles dealing with foreign politics are discarded. Figure 8.5.b shows the learning curves obtained while sampled articles were manually labelled by the authors. Vertical dotted lines stand for hours-long interruptions of the labelling task.

After labelling 195 articles, our model reached 0.747 micro f1-score (0.674 micro precision, 0.838 micro recall) on 50 validation articles. The final architecture performs a late fusion between the text branch $\{conv1d(89 \text{ filters of size } 5), conv1d(635 \text{ filters of size } 3), global \text{ average pooling}, dense(140), dense(72), dense(122)\}$ and the image branch $\{dense(65), dense(367), dense(75), dense(81)\}$. Results in Table 8.7 confirm that both input modalities have an impact on prediction results. Table 8.8 contains titles and images of unlabelled articles previews whose prediction scores are the strongest. Testing on 20 independent documents yields a 0.70 f1-score.

In this section, we perform a case study in two steps: first as a text classification task, detecting one-sided opinion

Table 8.7: Influence of input modalities on prediction scores









translated input text	input image	 <p>Jean-Christophe MARMARA / Le Figaro</p>	 <p>Xavier DE FENOYL / La Dépêche</p>
	Unemployment: the number of job seekers fell sharply in 2019	demonstration: 0.091 violence: 0.086	demonstration: 0.473 violence: 0.504
Evacuation of the Sully bridge: the CRS commander himself lost consciousness “by suffocation of tear gas”	demonstration: 0.561 violence: 0.577	demonstration: 0.943 violence: 0.994	

Table 8.8: Highest predictions scores for label “demonstration” (left) and “violence” (right), taken from unlabelled corpus

image	translated title	image	translated title
	EXCLUSIVE. <i>Black bloc, féminicide, dégagisme...</i> The new words of the Larousse	 <p>LP/Jean-Baptiste Quentin</p>	A same-sex couple assaulted by about ten armed young people in Seine-Saint-Denis
 <p>© FRANCOIS LO PRESTI / AFP</p>	Yellow jackets, act 9: towards an action in “the centre of France” or in La Défense?	 <p>©SIPA/AP/LM Otero</p>	Dallas: five police officers killed, suspects arrested, another is dead
 <p>MAXPPP</p>	LIVE. Pension reform: unions call for local actions on Thursday 12 December and for a national demonstration on Tuesday 17 December	 <p>REUTERS/Charles Platiau</p>	“Purge” against police officers: the caller placed in custody

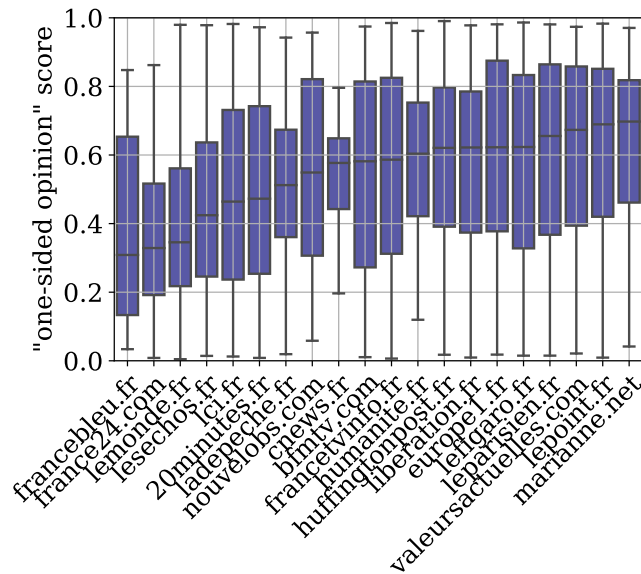


Figure 8.7: Opinion score distribution across news outlets

articles, quantifying their impact and qualifying their sources. Second, as a multi-modal identification task limited to a political subtopic about public demonstrations: we propose to qualify the news outlets by their display of violence to qualify these events.

One-sided Opinion Articles.

One of the working questions bears on the fact that newspapers produce both *informative* or *opinion-sided* articles. Each article receives a score of opinion, ranging from 0 (informative) to 1 (expressing one person’s opinion). On Figure 8.7, these scores are aggregated by news outlet, and sorted by their median score. *FranceBleu*, the local public information channel, can be qualified as informative while *LePoint* and *Marianne*, at the other side of the chart, are better diffused on Reddit when they relay opinions.

Opinionated articles are often supposed to be quicker to spread on social networks. Figure 8.8 answers this question: we arbitrarily split the documents as “factual” (opinion score below 0.4, in grey) and “opinion-relaying” (score over 0.6, in blue). The global trend is indeed for opinion-relaying articles to receive some more top-level comments; this trend is however not very strong. Another phenomenon to underline is the great diversity between news outlets: in terms of engagement, the continuous information TV channels *LCI* and *BFM* seem to consistently trigger more engagement. At the extreme opposite, *france24* does not receive as many comments; this channel is also more watched abroad than inside France. Among the news outlets for which there is no gain in engagement when their articles are opinionated, *ValeursActuelles*, *LePoint* and *Humanité* have a reputation of being relays of their political factions. It seems that they do not gain more attention when explicitly relaying an opinion.

French President Macron holds a special place in the political landscape, being at the centre and receiving opposition from both the left and right wings. In Figure 8.9, we compare the distribution of opinion-sided articles

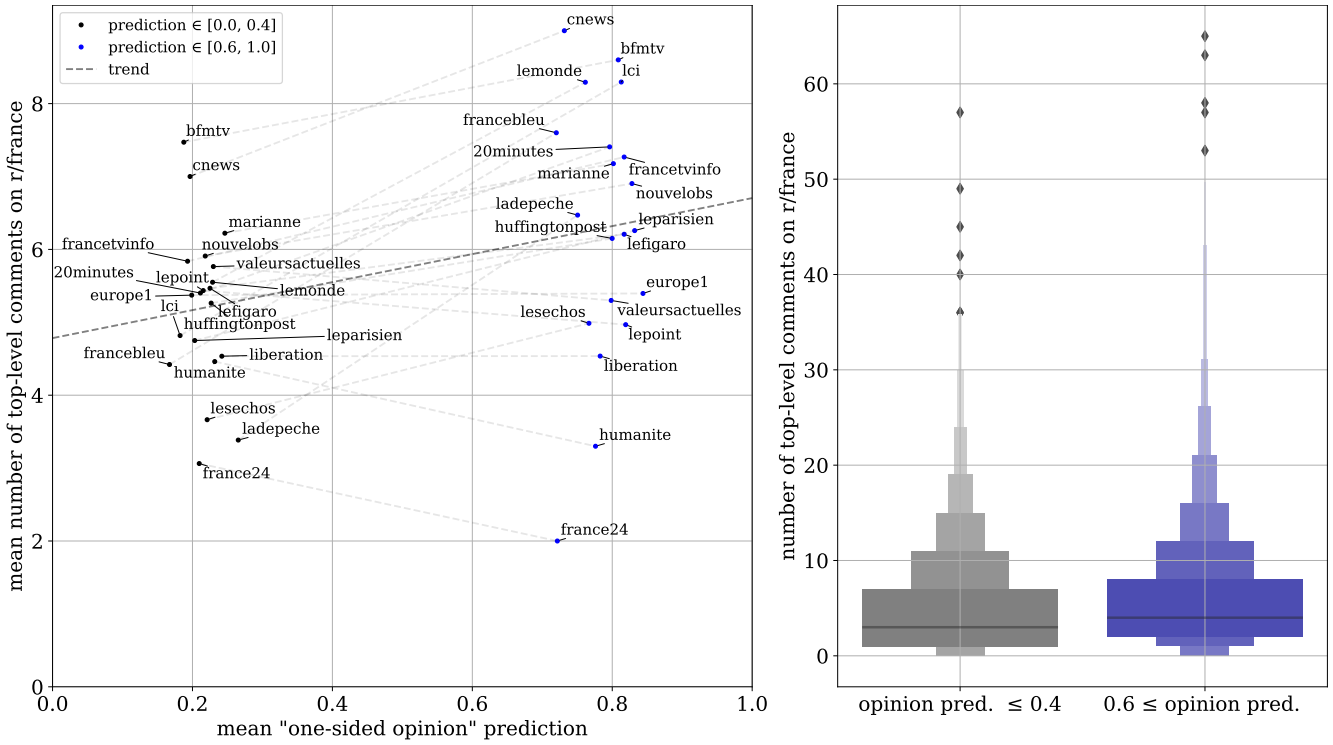


Figure 8.8: Mean number of comments between factual and opinion-relaying articles, per news outlet

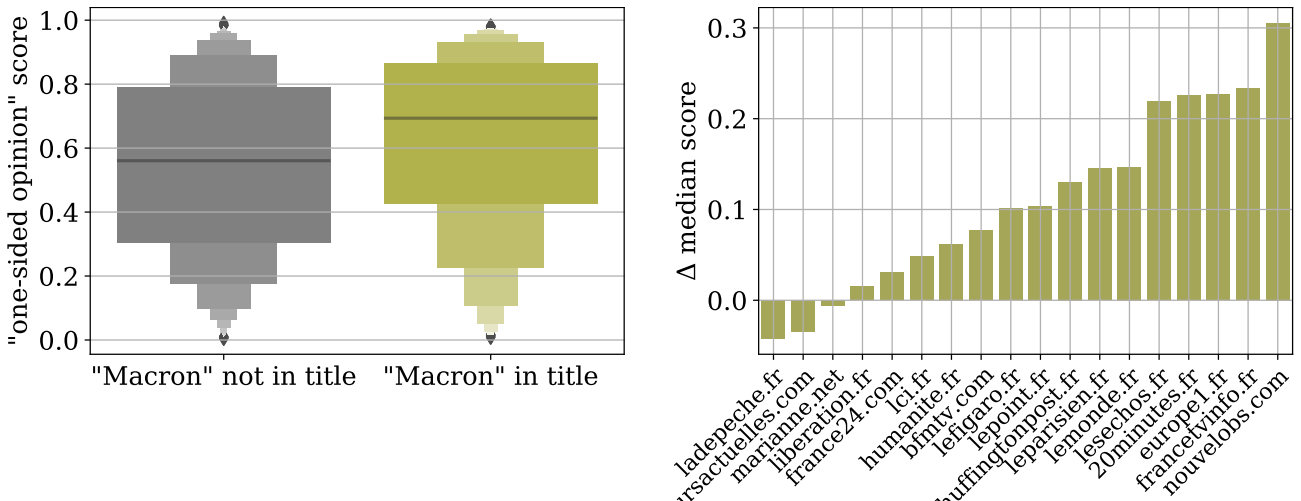


Figure 8.9: Opinion score distributions between articles mentioning – or not – “Macron” in their title (left); Difference in median opinion score between articles mentioning – or not – “Macron” in their titles, per outlet (right).

containing (in yellow) or not (in grey) “Macron” in the title; and the difference in opinion score for each news outlet. The most extreme is *NouvelObs*, for which “Macron” articles are often tagged as opinionated. However, only a few directly relay Macron’s opinion. The others consist in interviews of other political figures, who explain their ties and past history with the president, often coming along some critics.

pred.	title	translation
0.14	Kim Jong-un a remplacé Macron dans les cadres photos de la Maison-Blanche	Kim Jong-un replaced Macron in White House photo frames
0.61	Immigration : Macron fait-il pire que Sarkozy ?	Immigration: Is Macron doing worse than Sarkozy?
0.80	Pour Chevènement, tout est bon dans Macron	For Chevènement, everything is good in Macron
0.81	Convention climat : voici les trois propositions écartées par Emmanuel Macron	Climate convention: here are the three proposals rejected by Emmanuel Macron
0.87	Emmanuel Macron : “La communauté homosexuelle trouvera toujours en moi un défenseur”	Emmanuel Macron: “The homosexual community will always find a defender in me”
0.88	Macron persiste et signe : “Moi où j’habite, on en trouve du travail”	Macron persists and signs: “Where I live, we find work.”
0.93	François Bayrou : “Macron va devoir changer de logique”	François Bayrou: “Macron will have to change his logic”
0.96	“Il m’avait fait bonne impression” : quand Hollande raconte le poker menteur de Macron	“He made a good impression on me”: when Hollande recounts Macron’s lying poker

Table 8.9: Opinion scores of nouvelobs.com article whose titles contain “Macron”.

Demonstration and Violence.

There are many ways to cover events; this case study is focused around the global security law proposal in France at the end of 2020, which has triggered demonstrations, including violent scenes. News outlets did choose different editorial paths, arguably to encourage or dissuade people to gather in the streets. Figure 8.10 displays the news outlets, sorted according to their median scores (whether the articles cover demonstrations).

We propose a similar view about violence, for the articles that are predicted as covering demonstrations, in Figure 8.10. Indeed, a well spread narrative suggests that all demonstrations always result in violence and riots: this chart brings some quantifiable insights on this topic.

8.5 Conclusion and Future Work

In this chapter, we introduced an efficient implementation for Active Learning, and demonstrate its relevance on two classification tasks in a setup of political media analysis. The framework is highly scalable and solves many pain points, notably the parallelization of time-consuming tasks.

The case study on French language articles posted on Reddit illustrates the possibilities given by this technology: it facilitates the development of AI models not only for English, and not limited to one modality. We believe that active learning will help practitioners to adopt AI in their analysis, enabling them to explore more research questions and to quantify more subtle phenomena.

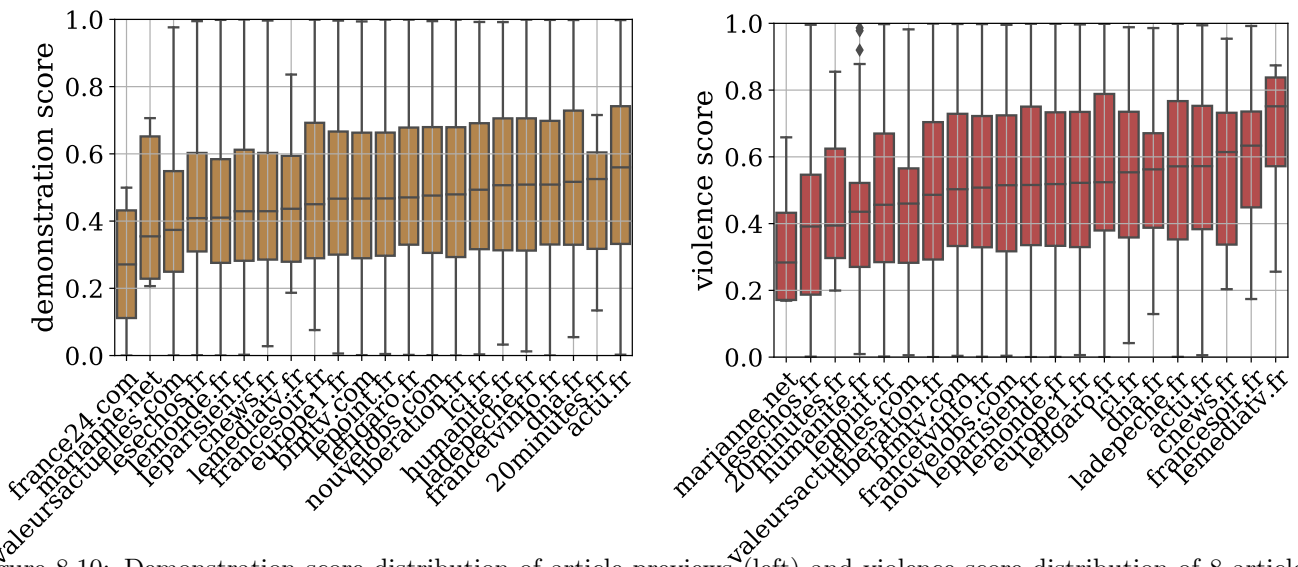


Figure 8.10: Demonstration score distribution of article previews (left) and violence score distribution of 8 articles previews detected as belonging to the *demonstration* class (right), across news outlets

Improvement leads are twofold: first in the AI domain, to take into account the recent progress in explainability, ensuring that the models learns with the correct rationales. Second, in the media analysis domain, we would like to explore and to deepen our analysis, in order to continue qualifying and characterizing the main information sources of the citizen.

Part IV

Conclusion and Future Work

Chapter 9

Conclusion

In this chapter, we summarize the contributions and limitations of our propositions regarding active learning for the detection of objects of operational interest in open-source multimedia content. Then, we identify the limitations of our approaches. Perspectives and future work are discussed in the final section.

9.1 Contributions summary

The second and third chapters of this thesis present existing works in the domain of deep neural networks and active learning. Although recent propositions in neural networks architectures have demonstrated strong encoding and downstream tasks capacities for image and text modalities, the required knowledge is obtained through the use of massive training datasets. If dealing with a great amount of open-source unlabelled data, active learning – the careful selection of informative samples to be labelled – becomes a beneficial option to train neural networks. Active learning brings together an oracle, who is the human holder of the complex knowledge of the downstream task, and the trained model, whose decision-making abilities evolve during the active learning process. By default, the model’s decision-making rationales are opaque and unrelated to those of the expert task tackler, the oracle. In our contributions, our goal is to observe the rationales behind the model’s decisions, put them into perspective with human rationales, and try and make the neural network align its decision-making rationales with those of a teacher to improve its performances.

9.1.1 Scientific contributions

In the era of both massive disinformation campaigns and growing distrust of the established press outlets, the automatic evaluation of the trustworthiness of a press article would be a powerful asset, but its trustworthiness could be disputed in the image of the content it has been trained from. With an aggregation of publicly available datasets, we trained a convolutional neural network to classify press articles as trustworthy or biased. In order to observe

and interpret the rationales behind the studied model's decisions, we propose to exploit word-wise class activation maps. These maps are extracted from the model once it is trained and allow us to draw conclusions on the input word n-grams that are considered meaningful by the model for its decision-making. Thereby, we can observe that our model is triggered by some phrases expressing vagueness or approximation, which are unwanted in reliable reporting. However, we also notice that the model is triggered by specific names, a behaviour we attribute to a biased training set. We hope our approach is a step towards the detection of unreliable press releases and an essential practice to consolidate the trust in the model in charge of this task.

Given that the aforementioned convolutional neural network appears to be triggered by phrases expressing vagueness, we carried out a comparison with VAGO, a model specifically designed to measure vagueness in texts, based on counting in sentences the number of words that belong to an expertly put together vocabulary expressing different categories of vagueness. On the one hand, we compared the overall classification decisions of both models. On the other hand, we observed the neural network's class activation scores for VAGO words expressing vagueness, in the context of the sentences where they appear. The results show that there exist a slight correlation between VAGO scores and the neural net's scores, highlighting the partial role played by vagueness in the characterization of biased articles. To confirm this, our study has also shown that many VAGO adjectives are given a high bias-inducement score by the neural net, among many others, which opens up prospects for future work.

The rationales behind's a model's decisions can be examined and compared with those a human would provide for their own decision. Going further, we intend to approach the induced alignment of the model's rationales with expert rationales, in order to improve the model's performances in a scarce ground-truth setup, as is that of an early active learning scenario. This study was conducted on two binary text classification tasks (positive vs. negative film reviews, software vs. hardware topic messages on an online board). The transfer of fine-grained word-wise knowledge from a teacher model to a student model *via* the direct supervision of class activation maps has shown that active learning *with rationales* is indeed beneficial to the training of the student model. Like an active learning curve tends to grow asymptotically, the benefits of this transferred additional knowledge fades as the training set grows.

In a subsequent study, we adapted this principle to image classification. Teacher neural classifiers were trained on the entirety of image datasets to learn how to recognize actions performed by people, and whether an image displays military-themed content. For each task, the rationales for the teacher model's decision were extracted as the regions inducing the highest activation or subjected to the highest attention scores. These rationales can be considered as the most informative regions with regard to the task being solved by the teacher model. Our results have shown that supervising the attention mechanism of the student model to focus on these most meaningful regions brings about steeper learning curves, when training on a very narrowed training set. Going further, we introduced and evaluated alternative sampling criteria for active learning based on attention uncertainty, region diversity and disagreement between attention scores and class activation maps, the last one of which yielded the best results.

9.1.2 Operational contributions

Finally, we introduced an efficient implementation for active learning and demonstrated its relevance on two classification tasks in a setup of political media analysis. Named ReALMS (Real-time Active Learning as Micro-Services), our active learning framework has been designed to be highly scalable and solve active learning pain points, notably the parallelization and real-time interaction of time-consuming tasks that are usually conducted in an iterative, time-wasting way: data reception, model predictions, sampling, reception of ground-truth labels and model training. The case study on French language articles posted on Reddit illustrates the possibilities given by this framework: it facilitates the development of AI models from openly accessible (open-source) yet unlabelled documents, not only for English and not limited to one modality since we exploited article titles, text bodies and featured images. Our framework allows instantiating a classification model, inject unlabelled documents, receive the most informative one according to the sampling criteria, and send it back with its label. Simultaneously, the sampling queue and the deep features describing documents are being updated, and the inner model is being re-trained with varying hyperparameters. The model can be requested for predictions at any time. In particular, using our framework, we instantiated two classification models using arbitrary classes. The first one is a text classifier trained to recognize whether an input press article expresses opposing views or a one-sided opinion; the second one is a multi-modal text and image classifier trained to recognize press articles addressing demonstrations and/or violence. The trained model is then used to compare how French news outlets cover events. We believe that our framework and active learning in general will help practitioners to adopt AI in their analysis, enabling them to better identify arbitrary topics and contents and to quantify more subtle phenomena.

9.2 Limitations

Our work has led to scientific and operational contributions in the field of neural networks and active learning. However, they include limitations relating to our approaches.

The efforts to put in parallel a handcrafted vocabulary expressing vagueness and strong class activation from a convolutional fake news classifier is biased by the using of different classes targeted by the two models: vagueness versus fake news. On the one hand, the VAGO model and its vocabulary were put together with vagueness in mind, as its name suggests, a poor indicator of press article honesty, but not the only one. On the other hand, our neural classifier learnt to recognize words and word patterns typical of low-quality, biased news outlets. However, fake news consist not only in imprecise texts, but also in clever fabrications, hoaxes, or simply mistaken factual reports. It cannot be excluded that the divergences between the two models' behaviours find causes in this difference of the targeted classes. Also, the comparison between the two models is only descriptive whereas the results call for a mutual improvement of the considered approaches: the enrichment of the VAGO vocabulary and the re-training of the CNN with VAGO words as rationales to lay stress on vagueness.

The attempt to incorporate rationales in text classification is limited by the fact that we used synthetic rationales produced by a teacher model trained on a full dataset. Words retained as rationales – those that trigger strong predictions from the teacher model – may be different words than those likely to be retained by a human label provider. It could be argued that synthetic rationales could be particularly favourable to the improvement of a student model with a structure resembling that of the teacher, more so than those issued from a person.

As stated in dedicated chapter 7, the hypothetical acquisition of human rationales for image classification can only be considered if target classes have strong spatially bounded recognizable features that can be pointed out on an image. In this study, our target classes are actions being performed (smoking, throwing a Frisbee...) and military-themed content. In both cases, class membership resides in strong regions of interest (objects being held, outfits, vehicles...). We can imagine incompatible scenarios, like trying to predict painting technique used to paint an entire picture. Also in this study, we introduced and compared sampling criteria for active learning that account for spatial peculiarities. Due to the computational burden of the experimental process, standalone criteria were compared in a one-shot experiment, whereas criteria can benefit from being used simultaneously and the results may vary with the outcome of the preliminary random sampling.

Regarding the exploitation of the results obtained with our active learning framework, we are fully aware of the necessity to reach a satisfying accuracy score on a significantly large enough testing set before qualifying news articles and outlets in light of the model's predictions. Putting this framework in the hand of the analyst must be accompanied by clear indicators of how good the model behaves and on the size of the samples this score is measured from.

9.3 Perspective and future work

Subsequent works have been initiated to pursue the mutual enrichment of VAGO and neural approaches for fake news classification. We hope these efforts will make a change in the understanding of what constitutes honest and unbiased journalism.

The direct supervision of attention mechanisms appears to be an efficient way to train a neural network *with rationales* in a scarce ground-truth setting. Transformers architectures relying on numerous attention heads, their preeminence in language modelling tasks opens up perspectives for clever methods and rules to train them with rationales.

Efforts have been made to integrate our active learning framework to Airbus products. We wish to put this tool in the hands of analysts in real-life scenarios and that it yields insightful feedback and results. Also, adding the capacity to learn *with rationales* to our framework would make a logical continuation as an interweaving of our contributions.

Publications

- Deep active learning with simulated rationales for text classification, *P. Guélorget, B. Grilheres and T. Zaharia*, International Conference on Pattern Recognition and Artificial Intelligence. Springer, Cham, 2020, p363-379.
- An interpretable model to measure fakeness and emotion in news, *G. Gadek and P. Guélorget*, KES, 2020.
- Active learning to measure opinion and violence in French newspapers, *P. Guélorget, G. Gadek, T. Zaharia and B. Grilheres*, KES, 2021,
- Combining vagueness detection with deep-learning to identify fake news, *P. Guélorget, B. Icard, G. Gadek, S. Gahbiche, S. Gatepaille, G. Atezing and P.Égré*, Fusion, 2021

Bibliography

- [1] NATS. URL <https://docs.nats.io/>.
- [2] A. Abbasi, H. Chen, and A. Salem. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Transactions on Information Systems (TOIS)*, 26(3):12, 2008.
- [3] A. Abdelkader. *Multimodal Deep Learning*.
- [4] S. Agarwal, H. Arora, S. Anand, and C. Arora. Contextual diversity for active learning. In *European Conference on Computer Vision*, pages 137–153. Springer, 2020.
- [5] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Batra, and D. Parikh. VQA: Visual Question Answering. [arXiv:1505.00468 \[cs\]](https://arxiv.org/abs/1505.00468), Oct. 2016. URL <http://arxiv.org/abs/1505.00468>. arXiv: 1505.00468.
- [6] W. P. Alston. *Philosophy of Language*. Prentice Hall, 1964.
- [7] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018.
- [8] D. Angluin. Queries and Concept Learning. *Machine Learning*, 2(4):319–342, Apr. 1988. ISSN 0885-6125, 1573-0565. doi: 10.1023/A:1022821128753. URL <https://link.springer.com/article/10.1023/A:1022821128753>.
- [9] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. [arXiv:1511.07247 \[cs\]](https://arxiv.org/abs/1511.07247), Nov. 2015. URL <http://arxiv.org/abs/1511.07247>. arXiv: 1511.07247.
- [10] J. Arevalo, T. Solorio, M. Montes-y Gómez, and F. A. González. Gated Multimodal Units for Information Fusion. [arXiv:1702.01992 \[cs, stat\]](https://arxiv.org/abs/1702.01992), Feb. 2017. URL <http://arxiv.org/abs/1702.01992>. arXiv: 1702.01992.
- [11] P. Atanasova, P. Nakov, L. Màrquez, A. Barrón-Cedeño, G. Karadzhov, T. Mihaylova, M. Mohtarami, and J. Glass. Automatic fact-checking using context and discourse information. *Journal of Data and Information Quality (JDIQ)*, 11(3):1–27, 2019.

- [12] G. Atemezing, B. Icard, and P. Égré. Multilingual gazetteers to detect vagueness in textual documents, Apr. 2021. URL <https://doi.org/10.5281/zenodo.4718530>.
- [13] Y. Aytar, C. Vondrick, and A. Torralba. SoundNet: Learning Sound Representations from Unlabeled Video. [arXiv:1610.09001 \[cs\]](https://arxiv.org/abs/1610.09001), Oct. 2016. URL <http://arxiv.org/abs/1610.09001>. arXiv: 1610.09001.
- [14] D. Bahdanau, K. Cho, and Y. Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. [arXiv:1409.0473 \[cs, stat\]](https://arxiv.org/abs/1409.0473), Sept. 2014. URL <http://arxiv.org/abs/1409.0473>. arXiv: 1409.0473.
- [15] T. Baltrušaitis, C. Ahuja, and L.-P. Morency. Multimodal Machine Learning: A Survey and Taxonomy. [arXiv:1705.09406 \[cs\]](https://arxiv.org/abs/1705.09406), May 2017. URL <http://arxiv.org/abs/1705.09406>. arXiv: 1705.09406.
- [16] W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9368–9377, 2018.
- [17] M. Bouazizi and T. Ohtsuki. A pattern-based approach for multi-class sentiment analysis in twitter. *IEEE Access*, 5:20617–20639, 2017.
- [18] A. Boulch. Sharesnet: reducing residual network parameter number by sharing weights. [arXiv preprint arXiv:1702.08782](https://arxiv.org/abs/1702.08782), 2017.
- [19] K. Brinker. Incorporating Diversity in Active Learning with Support Vector Machines. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning, ICML'03*, pages 59–66, Washington, DC, USA, 2003. AAAI Press. ISBN 978-1-57735-189-4. URL <http://dl.acm.org/citation.cfm?id=3041838.3041846>.
- [20] C. Budak, S. Goel, and J. M. Rao. Fair and balanced? quantifying media bias through crowdsourced content analysis. *Public Opinion Quarterly*, 80(S1):250–271, 2016.
- [21] D. W. Castro, E. Souza, D. Vitória, D. Santos, and A. L. Oliveira. Smoothed n-gram based models for tweet language identification: A case study of the Brazilian and European Portuguese national varieties. *Applied Soft Computing*, 61:1160–1172, 2017.
- [22] C. Catal and M. Nangir. A sentiment classification model based on multiple classifiers. *Applied Soft Computing*, 50:135–141, 2017.
- [23] B. Charalampakis, D. Spathis, E. Kouslis, and K. Kermanidis. A comparison between semi-supervised and supervised text mining techniques on detecting irony in greek political tweets. *Engineering Applications of Artificial Intelligence*, 51:50–57, 2016.

- [24] W. Chen, H. Wang, J. Chen, Y. Zhang, H. Wang, S. Li, X. Zhou, and W. Y. Wang. Tabfact: A large-scale dataset for table-based fact verification. In ICLR, 2020.
- [25] J. Cheng, L. Dong, and M. Lapata. Long Short-Term Memory-Networks for Machine Reading. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 551–561, 2016.
- [26] H. Cherfi, M. Coste, and F. Amardeilh. CA-manager: a middleware for mutual enrichment between information extraction systems and knowledge repositories. In 4th workshop SOS-DLWD “Des Sources Ouvertes au Web de Données, pages 15–28, 2013.
- [27] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259, 2014.
- [28] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. June 2014. URL <https://arxiv.org/abs/1406.1078>.
- [29] K. Cho, A. Courville, and Y. Bengio. Describing Multimedia Content using Attention-based Encoder-Decoder Networks. IEEE Transactions on Multimedia, 17(11):1875–1886, Nov. 2015. ISSN 1520-9210, 1941-0077. doi: 10.1109/TMM.2015.2477044. URL <http://arxiv.org/abs/1507.01053>. arXiv: 1507.01053.
- [30] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.
- [31] D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. Machine Learning, 15(2): 201–221, May 1994. ISSN 0885-6125, 1573-0565. doi: 10.1007/BF00993277. URL <https://link.springer.com/article/10.1007/BF00993277>.
- [32] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. Journal of artificial intelligence research, 4:129–145, 1996.
- [33] H. Cunningham. Gate, a general architecture for text engineering. Computers and the Humanities, 36(2): 223–254, 2002. ISSN 1572-8412. doi: 10.1023/A:1014348124664. URL <http://dx.doi.org/10.1023/A:1014348124664>.
- [34] I. Dagan and S. P. Engelson. Committee-Based Sampling For Training Probabilistic Classifiers. In In Proceedings of the Twelfth International Conference on Machine Learning, pages 150–157. Morgan Kaufmann, 1995.
- [35] T. Davidson, D. Warmley, M. Macy, and I. Weber. Automated hate speech detection and the problem of offensive language. In Eleventh international aaai conference on web and social media, 2017.

- [36] L. Deng. The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine, 29(6):141–142, 2012.
- [37] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs], Oct. 2018. URL <http://arxiv.org/abs/1810.04805>. arXiv: 1810.04805.
- [38] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. Journal of machine learning research, 12(7), 2011.
- [39] P. Égré, B. Spector, A. Mortier, and S. Verheyen. On the optimality of vagueness: “around”, “between”, and the Gricean maxims. arXiv preprint arXiv:2008.11841, 2020.
- [40] P. Égré and B. Icard. Lying and vagueness. In J. Meibauer, editor, Oxford Handbook of Lying. Oxford University Press, 2018.
- [41] E. Elhamifar, G. Sapiro, A. Yang, and S. S. Sarsry. A Convex Optimization Framework for Active Learning. In 2013 IEEE International Conference on Computer Vision, pages 209–216, Dec. 2013. doi: 10.1109/ICCV.2013.33.
- [42] M. Enzweiler and D. M. Gavrila. A mixed generative-discriminative framework for pedestrian classification. In 2008 IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, June 2008. doi: 10.1109/CVPR.2008.4587592. ISSN: 1063-6919.
- [43] O. Fraïssier, G. Cabanac, Y. Pitarch, R. Besançon, and M. Boughanem. # élysée2017fr: The 2017 french presidential campaign on twitter. In Proceedings of the International AAAI Conference on Web and Social Media, volume 12, 2018.
- [44] K. Fukushima. Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. Biological Cybernetics, 36:193–202, 1980.
- [45] G. Gadek and P. Guélorget. An interpretable model to measure fakeness and emotion in news. Procedia Computer Science, 176:78–87, 2020.
- [46] Y. Gal and Z. Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. arXiv preprint arXiv:1506.02158, 2015.
- [47] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell. Compact Bilinear Pooling. arXiv:1511.06062 [cs], Nov. 2015. URL <http://arxiv.org/abs/1511.06062>. arXiv: 1511.06062.
- [48] M. Giatsoglou, M. G. Vozalis, K. Diamantaras, A. Vakali, G. Sarigiannidis, and K. C. Chatzissavvas. Sentiment analysis leveraging emotions and word embeddings. Expert Systems with Applications, 69:214–224, 2017.

- [49] R. Girshick. Fast r-cnn. In Proceedings of the IEEE international conference on computer vision, pages 1440–1448, 2015.
- [50] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 580–587, 2014.
- [51] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. arXiv:1406.2661 [cs, stat], June 2014. URL <http://arxiv.org/abs/1406.2661>. arXiv: 1406.2661.
- [52] M. Gorriz, A. Carlier, E. Faure, and X. G. i. Nieto. Cost-Effective Active Learning for Melanoma Segmentation. CoRR, abs/1711.09168, 2017. URL <http://arxiv.org/abs/1711.09168>.
- [53] Y. Grandvalet, Y. Bengio, et al. Semi-supervised learning by entropy minimization. CAP, 367:281–296, 2005.
- [54] M. Granovetter. Threshold models of collective behavior. American journal of sociology, 83(6):1420–1443, 1978.
- [55] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and others. Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint arXiv:2005.08100, 2020.
- [56] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. CoRR, abs/1706.04599, 2017. URL <http://arxiv.org/abs/1706.04599>.
- [57] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In International Conference on Machine Learning, pages 1321–1330. PMLR, 2017.
- [58] P. Guélorget, B. Grilheres, and T. Zaharia. Deep active learning with simulated rationales for text classification. In Proceedings of the 2020 International Conference on Pattern Recognition and Artificial Intelligence, may 2020.
- [59] F. Hamborg, N. Meuschke, C. Breiting, and B. Gipp. news-please: A Generic News Crawler and Extractor. In Proceedings of the 15th International Symposium of Information Science, pages 218–223, Mar. 2017. doi: 10.5281/zenodo.4120316. event-place: Berlin.
- [60] N. Hassan, B. Adair, J. T. Hamilton, C. Li, M. Tremayne, J. Yang, and C. Yu. The quest to automate fact-checking. In Proceedings of the 2015 Computation+ Journalism Symposium, 2015.
- [61] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016.

- [62] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In Proceedings of the IEEE international conference on computer vision, pages 2961–2969, 2017.
- [63] J. Herzig, P. K. Nowak, T. Müller, F. Piccinno, and J. M. Eisenschlos. TAPAS: weakly supervised table parsing via pre-training. In ACL, 2020.
- [64] G. Hinton, N. Srivastava, and K. Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. Cited on, 14(8):2, 2012.
- [65] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580, 2012.
- [66] D. E. Ho, K. M. Quinn, et al. Measuring explicit political positions of media. Quarterly Journal of Political Science, 3(4):353–377, 2008.
- [67] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.
- [68] H. Hofmann, K. Kafadar, and H. Wickham. Letter-value plots: Boxplots for large data. Technical report, had.co.nz, 2011.
- [69] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. Neural networks, 2(5):359–366, 1989.
- [70] N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel. Bayesian active learning for classification and preference learning, 2011.
- [71] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.
- [72] F. Huang, X. Zhang, Z. Li, T. Mei, Y. He, and Z. Zhao. Learning Social Image Embedding with Deep Multimodal Attention Networks. Proceedings of the on Thematic Workshops of ACM Multimedia 2017 - Thematic Workshops '17, pages 460–468, 2017. doi: 10.1145/3126686.3126720. URL <http://arxiv.org/abs/1710.06582>. arXiv: 1710.06582.
- [73] F. Huang, X. Zhang, and Z. Li. Learning Joint Multimodal Representation with Adversarial Attention Networks. In Proceedings of the 26th ACM International Conference on Multimedia, MM '18, pages 1874–1882, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5665-7. doi: 10.1145/3240508.3240614. URL <http://doi.acm.org/10.1145/3240508.3240614>.
- [74] D. H. Hubel and T. N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. The Journal of physiology, 195(1):215–243, 1968.

- [75] B. Icard. Lying, deception and strategic omission: definition et evaluation. PhD thesis, Paris Sciences et Lettres, 2019.
- [76] P. Jain and A. Kapoor. Active learning for large multi-class problems. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pages 762–769. IEEE, 2009.
- [77] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov. Bag of Tricks for Efficient Text Classification. arXiv:1607.01759 [cs], Aug. 2016. URL <http://arxiv.org/abs/1607.01759>. arXiv: 1607.01759.
- [78] H. Jwa, D. Oh, K. Park, J. M. Kang, and H. Lim. exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). Applied Sciences, 9(19):4062, 2019.
- [79] J. Kahn, A. Lee, and A. Hannun. Self-training for end-to-end speech recognition. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 7084–7088. IEEE, 2020.
- [80] E. Kaiser and C. Wang. Packaging information as fact versus opinion: Consequences of the (information-) structural position of subjective adjectives. Discourse Processes, pages 1–25, 2021.
- [81] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pages 1–9, 2016.
- [82] M. Karan and J. Šnajder. Preemptive toxic language detection in wikipedia comments using thread-level context. In Proceedings of the Third Workshop on Abusive Language Online, pages 129–134, 2019.
- [83] A. Karpathy and L. Fei-Fei. Deep Visual-Semantic Alignments for Generating Image Descriptions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(4):664–676, Apr. 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2598339.
- [84] D. Katsaros, G. Stavropoulos, and D. Papakostas. Which machine learning paradigm for fake news detection? In 2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI), pages 383–387. IEEE, 2019.
- [85] A. Kendall, V. Badrinarayanan, and R. Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. arXiv preprint arXiv:1511.02680, 2015.
- [86] C. Kennedy. Vagueness and grammar: The semantics of relative and absolute gradable adjectives. Linguistics and philosophy, 30(1):1–45, 2007.
- [87] C. Kennedy. Two sources of subjectivity: Qualitative assessment and dimensional uncertainty. Inquiry, 56(2-3):258–277, 2013.

- [88] S. U. Khan, I. U. Haq, S. Rho, S. W. Baik, and M. Y. Lee. Cover the violence: A novel deep-learning-based approach towards violence-detection in movies. *Applied Sciences*, 9(22):4963, 2019.
- [89] Y. Kim. Convolutional Neural Networks for Sentence Classification. [arXiv:1408.5882 \[cs\]](https://arxiv.org/abs/1408.5882), Aug. 2014. URL <http://arxiv.org/abs/1408.5882>. arXiv: 1408.5882.
- [90] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. [arXiv preprint arXiv:1412.6980](https://arxiv.org/abs/1412.6980), 2014.
- [91] J. Kiros. neural-storyteller: A recurrent neural network for generating little stories about images, Aug. 2018. URL <https://github.com/ryankiros/neural-storyteller>. original-date: 2015-10-28T19:38:46Z.
- [92] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler. Skip-Thought Vectors. [arXiv:1506.06726 \[cs\]](https://arxiv.org/abs/1506.06726), June 2015. URL <http://arxiv.org/abs/1506.06726>. arXiv: 1506.06726.
- [93] A. Kirsch, J. van Amersfoort, and Y. Gal. BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning. [arXiv:1906.08158 \[cs, stat\]](https://arxiv.org/abs/1906.08158), Oct. 2019. URL <http://arxiv.org/abs/1906.08158>. arXiv: 1906.08158.
- [94] I. Krasin, T. Duerig, N. Alldrin, V. Ferrari, S. Abu-El-Haija, A. Kuznetsova, H. Rom, J. Uijlings, S. Popov, S. Kamali, M. Mallocci, J. Pont-Tuset, A. Veit, S. Belongie, V. Gomes, A. Gupta, C. Sun, G. Chechik, D. Cai, Z. Feng, D. Narayanan, and K. Murphy. OpenImages: A public dataset for large-scale multi-label and multi-class image classification. [Dataset available from https://storage.googleapis.com/openimages/web/index.html](https://storage.googleapis.com/openimages/web/index.html), 2017.
- [95] A. Krizhevsky, G. Hinton, and others. Learning multiple layers of features from tiny images. 2009. Publisher: Citeseer.
- [96] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.
- [97] K. Kundalia, Y. Patel, and M. Shah. Multi-label movie genre detection from a movie poster using knowledge transfer learning. *Augmented Human Research*, 5(1):1–9, 2020. Publisher: Springer.
- [98] K. Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339, 1995.
- [99] D. Lassiter and N. D. Goodman. Adjectival vagueness in a Bayesian model of interpretation. *Synthese*, 194(10):3801–3836, 2017.
- [100] A. Lecoutre, B. Negrevergne, and F. Yger. Recognizing art style automatically in painting with deep learning. In *Asian conference on machine learning*, pages 327–342. PMLR, 2017.

- [101] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.
- [102] D.-H. Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In Workshop on challenges in representation learning, ICML, volume 3, page 896, 2013.
- [103] D. D. Lewis and J. Catlett. Heterogeneous Uncertainty Sampling for Supervised Learning. In In Proceedings of the Eleventh International Conference on Machine Learning, pages 148–156. Morgan Kaufmann, 1994.
- [104] D. D. Lewis and W. A. Gale. A Sequential Algorithm for Training Text Classifiers. arXiv:cmp-lg/9407020, July 1994. URL <http://arxiv.org/abs/cmp-lg/9407020>. arXiv: cmp-lg/9407020.
- [105] X. Li and Y. Guo. Adaptive Active Learning for Image Classification. In 2013 IEEE Conference on Computer Vision and Pattern Recognition, pages 859–866, June 2013. doi: 10.1109/CVPR.2013.116.
- [106] M. Lin, Q. Chen, and S. Yan. Network in network. arXiv preprint arXiv:1312.4400, 2013.
- [107] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740–755. Springer, 2014.
- [108] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision, pages 2980–2988, 2017.
- [109] Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio. A Structured Self-attentive Sentence Embedding. arXiv:1703.03130 [cs], Mar. 2017. URL <http://arxiv.org/abs/1703.03130>. arXiv: 1703.03130.
- [110] L. Liu, X. Huang, J. Xu, and Y. Song. Oasis: Online analytic system for incivility detection and sentiment classification. In 2019 International Conference on Data Mining Workshops (ICDMW), pages 1098–1101. IEEE, 2019.
- [111] S. Liu, L. Guo, K. Mays, M. Betke, and D. T. Wijaya. Detecting frames in news headlines and its application to analyzing news framing trends surrounding us gun violence. In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 504–514, 2019.
- [112] J. R. Lott and K. A. Hassett. Is newspaper coverage of economic events politically biased? Public choice, 160 (1-2):65–108, 2014.
- [113] D. G. Lowe. Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2):91–110, 2004.

- [114] W. Luo, Y. Li, R. Urtasun, and R. Zemel. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. [arXiv:1701.04128 \[cs\]](https://arxiv.org/abs/1701.04128), Jan. 2017. URL <http://arxiv.org/abs/1701.04128>. arXiv: 1701.04128.
- [115] M.-T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. [arXiv preprint arXiv:1508.04025](https://arxiv.org/abs/1508.04025), 2015.
- [116] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning Word Vectors for Sentiment Analysis. In [Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies](#), pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/P11-1015>.
- [117] S. Madisetty and M. S. Desarkar. A neural network-based ensemble approach for spam detection in twitter. [IEEE Transactions on Computational Social Systems](#), 5(4):973–984, 2018.
- [118] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. van der Maaten. Exploring the limits of weakly supervised pretraining. In [Proceedings of the European Conference on Computer Vision \(ECCV\)](#), pages 181–196, 2018.
- [119] C. Maigrot, V. Claveau, and E. Kijak. Fusion-based multimodal detection of hoaxes in social networks. In [2018 IEEE/WIC/ACM International Conference on Web Intelligence \(WI\)](#), pages 222–229. IEEE, 2018.
- [120] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). [arXiv:1412.6632 \[cs\]](https://arxiv.org/abs/1412.6632), Dec. 2014. URL <http://arxiv.org/abs/1412.6632>. arXiv: 1412.6632.
- [121] L. McNally and I. Stojanovic. Aesthetic adjectives. In J. O. Young, editor, [The Semantics of Aesthetic Judgment](#), pages 17–37. Oxford University Press, 2017.
- [122] D. Merkel. Docker: lightweight linux containers for consistent development and deployment. [Linux journal](#), 2014(239):2, 2014.
- [123] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. [arXiv:1301.3781 \[cs\]](https://arxiv.org/abs/1301.3781), Sept. 2013. URL <http://arxiv.org/abs/1301.3781>. arXiv: 1301.3781.
- [124] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. [arXiv:1310.4546 \[cs, stat\]](https://arxiv.org/abs/1310.4546), Oct. 2013. URL <http://arxiv.org/abs/1310.4546>. arXiv: 1310.4546.
- [125] M. Mitsuhashi, H. Fukui, Y. Sakashita, T. Ogata, T. Hiraoka, T. Yamashita, and H. Fujiyoshi. [Embedding Human Knowledge into Deep Neural Network via Attention Map](#). 2019. [eprint: 1905.03540](https://arxiv.org/abs/1905.03540).

- [126] M. Munezero, C. S. Montero, M. Mozgovoy, and E. Sutinen. Emotwitter—a fine-grained visualization system for identifying enduring sentiments in tweets. In Computational Linguistics and Intelligent Text Processing, pages 78–91. Springer, 2015.
- [127] H. Nakayama and N. Nishida. Zero-resource Machine Translation by Multimodal Encoder-decoder Network with Multimedia Pivot. arXiv:1611.04503 [cs], Nov. 2016. URL <http://arxiv.org/abs/1611.04503>. arXiv: 1611.04503.
- [128] Y. NESTEROV. A method for unconstrained convex minimization problem with the rate of convergence $o(1/k^2)$. Doklady AN USSR, 269:543–547, 1983. URL <https://ci.nii.ac.jp/naid/20001173129/en/>.
- [129] N. Neverova, C. Wolf, G. W. Taylor, and F. Nebout. ModDrop: adaptive multi-modal gesture recognition. arXiv:1501.00102 [cs], Dec. 2014. URL <http://arxiv.org/abs/1501.00102>. arXiv: 1501.00102.
- [130] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal Deep Learning. pages 689–696, Jan. 2011.
- [131] A. Niculescu-Mizil and R. Caruana. Predicting good probabilities with supervised learning. In Proceedings of the 22nd international conference on Machine learning, pages 625–632, 2005.
- [132] S. J. Pan and Q. Yang. A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering, 22(10):1345–1359, Oct. 2010. ISSN 1558-2191. doi: 10.1109/TKDE.2009.191.
- [133] S. Paramesh and K. Shreedhara. Automated it service desk systems using machine learning techniques. In Data Analytics and Learning, pages 331–346. Springer, 2019.
- [134] E. Parzen. On estimation of a probability density function and mode. The annals of mathematical statistics, 33(3):1065–1076, 1962.
- [135] E. Pavlick, H. Ji, X. Pan, and C. Callison-Burch. The gun violence database: A new task and data set for nlp. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1018–1024, 2016.
- [136] X. Peng and C. Schmid. Encoding feature maps of cnns for action recognition. 2015.
- [137] J. W. Pennebaker, M. E. Francis, and R. J. Booth. Linguistic inquiry and word count: Liwc 2001. Mahway: Lawrence Erlbaum Associates, 71:2001, 2001.
- [138] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 1532–1543, 2014.

- [139] V. Pérez-Rosas, B. Kleinberg, A. Lefevre, and R. Mihalcea. Automatic detection of fake news. Proceedings of the 27th International Conference on Computational Linguistics, pages 3391–3401, 2018.
- [140] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. arXiv:1802.05365 [cs], Feb. 2018. URL <http://arxiv.org/abs/1802.05365>. arXiv: 1802.05365.
- [141] B. T. Polyak. Some methods of speeding up the convergence of iteration methods. Ussr computational mathematics and mathematical physics, 4(5):1–17, 1964.
- [142] R. Puglisi. Being the new york times: the political behaviour of a newspaper. The BE journal of economic analysis & policy, 11(1), 2011.
- [143] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.
- [144] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 779–788, 2016.
- [145] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.
- [146] F. Ribeiro, L. Henrique, F. Benevenuto, A. Chakraborty, J. Kulshrestha, M. Babaei, and K. Gummadi. Media bias monitor: Quantifying biases of social media news outlets at large-scale. In Proceedings of the International AAAI Conference on Web and Social Media, volume 12, 2018.
- [147] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford, and others. Okapi at TREC-3. Nist Special Publication Sp, 109:109, 1995.
- [148] R. Rojas. The backpropagation algorithm. In Neural networks, pages 149–182. Springer, 1996.
- [149] F. Rosenblatt. The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain. Psychological Review, pages 65–386, 1958.
- [150] V. L. Rubin, Y. Chen, and N. J. Conroy. Deception detection for news: three types of fakes. In Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community, page 83. American Society for Information Science, 2015.
- [151] S. Ruder. An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747, 2016.
- [152] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. nature, 323(6088):533–536, 1986. Publisher: Nature Publishing Group.

- [153] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. [arXiv:1409.0575 \[cs\]](https://arxiv.org/abs/1409.0575), Sept. 2014. URL <http://arxiv.org/abs/1409.0575>.
- [154] B. Russell. Vagueness. *The Australasian Journal of Psychology and Philosophy*, 1(2):84–92, 1923.
- [155] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. [arXiv:1801.04381 \[cs\]](https://arxiv.org/abs/1801.04381), Jan. 2018. URL <http://arxiv.org/abs/1801.04381>. arXiv: 1801.04381.
- [156] P. Schrodt. CAMEO Conflict and Mediation Event Observations. [arXiv:1509.01626 \[cs\]](https://arxiv.org/abs/1509.01626), Mar. 2012.
- [157] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.
- [158] O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. [arXiv preprint arXiv:1708.00489](https://arxiv.org/abs/1708.00489), 2017.
- [159] B. Settles. Active Learning Literature Survey. Technical Report, University of Wisconsin-Madison Department of Computer Sciences, 2009. URL <https://minds.wisconsin.edu/handle/1793/60660>.
- [160] B. Settles and M. Craven. An Analysis of Active Learning Strategies for Sequence Labeling Tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '08*, pages 1070–1079, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics. URL <http://dl.acm.org/citation.cfm?id=1613715.1613855>.
- [161] S. Shaar, N. Babulkov, G. D. S. Martino, and P. Nakov. That is a known lie: Detecting previously fact-checked claims. In *ACL*, pages 3607–3618, 2020.
- [162] C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- [163] M. Sharma, D. Zhuang, and M. Bilgic. Active Learning with Rationales for Text Classification. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 441–451, Denver, Colorado, June 2015. Association for Computational Linguistics. URL <http://www.aclweb.org/anthology/N15-1047>.
- [164] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36, 2017.
- [165] K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 395–405, 2019.

- [166] B. W. Silverman. Density Estimation for Statistics and Data Analysis. Chapman & Hall, London, 1986.
- [167] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.
- [168] S. Sinha, S. Ebrahimi, and T. Darrell. Variational adversarial active learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 5972–5981, 2019.
- [169] C. G. M. Snoek, M. Worring, and A. W. M. Smeulders. Early Versus Late Fusion in Semantic Video Analysis. In ACM International Conference on Multimedia. 2005. URL <https://ivi.fnwi.uva.nl/isis/publications/2005/SnoekICM2005>.
- [170] S. Solt. Multidimensionality, subjectivity and scales: Experimental evidence. In The Semantics of Gradability, Vagueness, and Scale Structure, pages 59–91. Springer, 2018.
- [171] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. Advances in neural information processing systems, 27, 2014.
- [172] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1–9, 2015.
- [173] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. arXiv preprint arXiv:1602.07261, 2016.
- [174] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning, pages 6105–6114. PMLR, 2019.
- [175] D. Tang, B. Qin, and T. Liu. Document modeling with gated recurrent neural network for sentiment classification. In Proceedings of the 2015 conference on empirical methods in natural language processing, pages 1422–1432, 2015.
- [176] R. Tapu, B. Mocanu, and T. Zaharia. TV News Retrieval Based on Story Segmentation and Concept Association. In 2016 12th International Conference on Signal-Image Technology Internet-Based Systems (SITIS), pages 327–334, Nov. 2016. doi: 10.1109/SITIS.2016.60.
- [177] A. Tchechmedjiev, P. Fafalios, K. Boland, M. Gasquet, M. Zloch, B. Zapilko, S. Dietze, and K. Todorov. ClaimsKG: A knowledge graph of fact-checked claims. In ISWC, pages 309–324, 2019.
- [178] J. B. Tenenbaum and W. T. Freeman. Separating Style and Content with Bilinear Models. Neural Computation, 12(6):1247–1283, June 2000. ISSN 0899-7667, 1530-888X. doi: 10.1162/089976600300015349. URL <http://www.mitpressjournals.org/doi/10.1162/089976600300015349>.

- [179] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. Journal of machine learning research, 2(Nov):45–66, 2001.
- [180] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. International journal of computer vision, 104(2):154–171, 2013.
- [181] L. G. Valiant. A Theory of the Learnable. In Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing, STOC '84, pages 436–445, New York, NY, USA, 1984. ACM. ISBN 978-0-89791-133-7. doi: 10.1145/800057.808710. URL <http://doi.acm.org/10.1145/800057.808710>.
- [182] K. van Deemter. Utility and language generation: The case of vagueness. Journal of Philosophical Logic, 38(6):607, 2009.
- [183] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. June 2017. URL <https://arxiv.org/abs/1706.03762v5>.
- [184] S. Verheyen, S. Dewil, and P. Égré. Subjectivity in gradable adjectives: The case of tall and heavy. Mind & Language, 33(5):460–479, 2018.
- [185] B. Verschuere, N. C. Köbis, Y. Bereby-Meyer, D. Rand, and S. Shalvi. Taxing the brain to uncover lying? meta-analyzing the effect of imposing cognitive load on the reaction-time costs of lying. Journal of applied research in memory and cognition, 7(3):462–469, 2018.
- [186] V. Vielzeuf, A. Lechervy, S. Pateux, and F. Jurie. CentralNet: a Multilayer Approach for Multimodal Fusion. arXiv:1808.07275 [cs], Aug. 2018. URL <http://arxiv.org/abs/1808.07275>. arXiv: 1808.07275.
- [187] S. Vosoughi, D. Roy, and S. Aral. The spread of true and false news online. Science, 359(6380):1146–1151, 2018. doi: 10.1126/science.aap9559.
- [188] N. T. Vu, H. Adel, P. Gupta, and H. Schütze. Combining recurrent and convolutional neural networks for relation classification. arXiv preprint arXiv:1605.07333, 2016.
- [189] V. Vukotić, C. Raymond, and G. Gravier. Bidirectional Joint Representation Learning with Symmetrical Deep Neural Networks for Multimodal and Crossmodal Applications. In ICMR, New York, United States, June 2016. ACM. URL <https://hal.inria.fr/hal-01314302>.
- [190] K. Wang, D. Zhang, Y. Li, R. Zhang, and L. Lin. Cost-Effective Active Learning for Deep Image Classification. IEEE Transactions on Circuits and Systems for Video Technology, 27:1–1, 2016. doi: 10.1109/TCSVT.2016.2589879.
- [191] F. M. F. Wong, C. W. Tan, S. Sen, and M. Chiang. Quantifying political leaning from tweets and retweets. In Proceedings of the International AAAI Conference on Web and Social Media, volume 7, 2013.

- [192] C. Wu, F. Wu, M. An, Y. Huang, and X. Xie. Neural news recommendation with topic-aware news representation. In Proceedings of the 57th Annual meeting of the association for computational linguistics, pages 1154–1159, 2019.
- [193] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. [arXiv:1502.03044 \[cs\]](https://arxiv.org/abs/1502.03044), Feb. 2015. URL <http://arxiv.org/abs/1502.03044>. arXiv: 1502.03044.
- [194] F. Yang, S. K. Pentylala, S. Mohseni, M. Du, H. Yuan, R. Linder, E. D. Ragan, S. Ji, and X. Hu. Xfake: explainable fake news detector with visualizations. In The World Wide Web Conference, pages 3600–3604, 2019.
- [195] L. Yang, Y. Zhang, J. Chen, S. Zhang, and D. Z. Chen. Suggestive Annotation: A Deep Active Learning Framework for Biomedical Image Segmentation. [arXiv:1706.04737 \[cs\]](https://arxiv.org/abs/1706.04737), June 2017. URL <http://arxiv.org/abs/1706.04737>. arXiv: 1706.04737.
- [196] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. Guibas, and L. Fei-Fei. Human action recognition by learning bases of action attributes and parts. In 2011 International Conference on Computer Vision, pages 1331–1338, Nov. 2011. doi: 10.1109/ICCV.2011.6126386. ISSN: 2380-7504.
- [197] W. Yin, K. Kann, M. Yu, and H. Schütze. Comparative study of cnn and rnn for natural language processing. [arXiv preprint arXiv:1702.01923](https://arxiv.org/abs/1702.01923), 2017.
- [198] S. Zagoruyko and N. Komodakis. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. [arXiv:1612.03928 \[cs\]](https://arxiv.org/abs/1612.03928), Feb. 2017. URL <http://arxiv.org/abs/1612.03928>. arXiv: 1612.03928.
- [199] O. F. Zaidan, J. Eisner, and C. D. Piatko. Machine Learning with Annotator Rationales to Reduce Annotation Cost. 2008.
- [200] T. Zaman, E. B. Fox, and E. T. Bradlow. A Bayesian approach for predicting the popularity of tweets. CoRR, abs/1304.6777, 2013. URL <http://arxiv.org/abs/1304.6777>.
- [201] B. Zhang, L. Li, S. Yang, S. Wang, Z.-J. Zha, and Q. Huang. State-relabeling adversarial active learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8756–8765, 2020.
- [202] C. Zhang and T. Chen. An active learning framework for content-based information retrieval. IEEE Transactions on Multimedia, 4(2):260–268, June 2002. ISSN 1520-9210. doi: 10.1109/TMM.2002.1017738.

- [203] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2921–2929, Las Vegas, NV, USA, June 2016. IEEE. ISBN 978-1-4673-8851-1. doi: 10.1109/CVPR.2016.319. URL <http://ieeexplore.ieee.org/document/7780688/>.
- [204] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis. Learning Rich Features for Image Manipulation Detection. arXiv:1805.04953 [cs], May 2018. URL <http://arxiv.org/abs/1805.04953>. arXiv: 1805.04953.
- [205] X. Zhou and R. Zafarani. Fake news: A survey of research, detection methods, and opportunities. arXiv preprint arXiv:1812.00315, 2018.

Titre: Apprentissage actif pour la détection d'objets d'intérêt opérationnel dans les contenus multimédia

Mots clés: apprentissage actif, apprentissage profond, données source ouverte

Résumé: Une profusion de contenus, acteurs et interactions en source ouverte sont ciblées par les analystes à des fins commerciales, politiques ou de renseignement. Analyser l'immensité de ces données requiert une assistance automatisée. Bien que les propositions récentes en matière d'architectures de réseaux de neurones aient montré de fortes capacités envers les modalités image et texte, leur entraînement exploite des jeux de données massifs, inexistant pour la majorité des classes d'intérêt opérationnel. Pour résoudre ce problème, l'apprentissage actif tire parti de la grande quantité de documents non annotés en sollicitant un *oracle* humain pour obtenir les labels des documents présumés les plus informatifs, afin d'améliorer la précision. Cependant, les justifications derrière les décisions du modèle sont opaques et sans lien avec celles de l'oracle. De plus, à cause de ses longues étapes successives, le déroulement de l'apprentissage actif nuit à ses performances en temps réel. Nos contributions dans cette thèse visent à analyser et résoudre ces prob-

lèmes à quatre niveaux. Premièrement, nous observons les justifications derrière les décisions d'un réseau de neurones. Deuxièmement, nous mettons ces justifications en perspective avec celles élaborées par des humains. Troisièmement, nous incitons un réseau de neurones à aligner ses justificatifs sur ceux d'un modèle professeur qui simule ceux d'un oracle humain, et améliorons sa précision. Finalement, nous mettons au point et exploitons un système d'apprentissage actif pour surmonter ses limitations usuelles. Ces études ont été menées sur des données monomodales texte ou image, ou sur des paires multimodales texte/image, principalement des articles de presse en anglais et en français. À travers les chapitres de cette thèse, nous traitons plusieurs cas d'utilisation, parmi lesquels la reconnaissance du vague et des fausses nouvelles, la détection du manque d'avis contradictoires dans les articles et la classification d'articles comme abordant des sujets arbitrairement choisis, tels que les manifestations ou la violence.

Title: Active learning for the detection of objects of operational interest in open-source multimedia content)

Keywords: active learning, deep learning, open-source data

Abstract: A profusion of openly accessible content, actors and interactions are targeted by analysts for intelligence, marketing or political purposes. Analysing the immensity of open source data requires automated assistance. Although recent propositions in neural network architectures have demonstrated strong capacities for image and text modalities, their training harnesses massive training datasets, non-existent for the majority of operational classes of interest. To address this issue, active learning takes advantage of the great amounts of unlabeled documents by soliciting from a human *oracle* the ground-truth labels of the presumed most informative documents, to improve accuracy. Yet, the model's decision-making rationales are opaque and might be unrelated to those of the oracle. Furthermore, with its time-consuming iterative steps, the active learning workflow is detrimental to its real-time performances. Our contributions in this thesis

aim to analyse and address these issues at four levels. Firstly, we observe the rationales behind a neural network's decisions. Secondly, we put these rationales into perspective with human rationales. Thirdly, we try and make the neural network align its decision-making rationales with those of a teacher model to simulate the rationales of a human oracle and improve accuracy in what is called active learning *with rationales*. Finally, we design and exploit an active learning framework to overcome its usual limitations. These studies were conducted with uni-modal text and image data, and multi-modal text and image associations, principally press articles in English and French. Throughout this work's chapters, we address several use cases among which fake news classification, vagueness classification, the detection of lack of contradiction in articles, the detection of arbitrary topics such as demonstrations and violence.