



HAL
open science

Harnessing the Power of Multimodal and Textual Data in Industry 4.0

Victor Pellegrain

► **To cite this version:**

Victor Pellegrain. Harnessing the Power of Multimodal and Textual Data in Industry 4.0. Artificial Intelligence [cs.AI]. Université Paris-Saclay, 2023. English. NNT : 2023UPAST093 . tel-04280319

HAL Id: tel-04280319

<https://theses.hal.science/tel-04280319v1>

Submitted on 10 Nov 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.

Harnessing the Power of Multimodal and Textual Data for Industry 4.0

*Exploitation de la Puissance des Données Multimodales et
Textuelles pour l'Industrie 4.0*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°573 : interfaces : matériaux, systèmes, usages
(INTERFACES)

Spécialité de doctorat: INFORMATIQUE

Graduate School : Sciences de l'ingénierie et des systèmes

Référent : CentraleSupélec

Thèse préparée à l'IRT SystemX et dans l'unité de recherche MICS (Université Paris-Saclay, CentraleSupélec), sous la direction de Céline Hudelot, Professeure des Universités, et le co-encadrement de Myriam Tami, Maître de Conférences, et Michel Batteux, Ingénieur-Chercheur.

Thèse soutenue à Paris-Saclay, le 4 juillet 2023, par

Victor PELLEGRIN

Composition du jury

Vincent Mousseau Professeur des Universités, Université Paris-Saclay	Président
Sébastien Lefèvre Professeur des Universités, Université Bretagne-Sud	Rapporteur & Examineur
Stefan Duffner Maître de Conférences, Université de Lyon	Rapporteur & Examineur
Cécile Capponi Professeure des Universités, Université Aix-Marseille	Examinatrice
Laurent Amsaleg Directeur de recherche, CNRS	Examineur

Titre: Exploitation de la Puissance des Données Multimodales et Textuelles dans l'Industrie 4.0

Mots clés: Apprentissage profond, Fusion de données multimodales, Traitement Automatique du langage, Few-shot learning, Diagnostic de défauts, Maintenance prévisionnelle

Résumé: Dans le paysage en constante évolution de l'Industrie 4.0, cette thèse aborde deux défis cruciaux visant à améliorer le diagnostic de défauts : une interprétation efficace des données multimodales provenant de divers capteurs et une exploitation intelligente des informations contenues dans des rares rapports de maintenance spécialisés.

Le premier défi implique la synthèse de flux de données de diverses modalités en une représentation expressive s'adaptant aux conditions dynamiques du système. Ceci nécessite le développement de stratégies innovantes pour traiter les données complexes efficacement en temps et en mémoire.

Le second défi concerne l'extraction d'informations précieuses à partir d'un nombre limité de rapports de maintenance rédigés par des experts. Cette tâche est rendue com-

plexe par le vocabulaire spécifique que ces rapports possèdent.

En réponse à ces défis, la thèse présente une architecture d'apprentissage profond unique qui gère habilement les longs flux de données multimodales non alignées. De plus, elle propose une méthode transductive innovante pour l'apprentissage à quelques exemples textuels, qui exploite les données étiquetées limitées disponibles pour améliorer les performances de prédiction, tout en assurant la confidentialité des informations sensibles.

Cette thèse est organisée en deux parties principales, la première traite de l'apprentissage multimodal pour le diagnostic des défauts, et la seconde cible l'apprentissage à quelques exemples en TAL pour l'analyse des données textuelles.

Title: Harnessing the Power of Multimodal and Textual Data in Industry 4.0

Keywords: Deep Learning, Multimodal Fusion, Natural Language Processing, Few-shot learning, Fault diagnosis, Predictive maintenance

Abstract: In the ever-evolving landscape of Industry 4.0, this thesis addresses two critical challenges aimed at enhancing fault diagnosis: effective interpretation of multimodal data from diverse sensors and smart exploitation of the information contained in scarce, specialized maintenance reports.

The first challenge involves the synthesis of data streams from various modalities into an expressive representation that can adapt to dynamic system conditions. This necessitates the development of innovative strategies to process complex data in a time and memory-efficient manner.

The second challenge focuses on extracting valuable information from the limited number of expert-written maintenance reports. This

task is made complex due to the highly specialized industry-specific vocabulary these reports possess.

In response to these challenges, the thesis presents a unique deep learning architecture that handles long, unaligned multimodal data streams. Furthermore, it proposes an innovative transductive method for textual few-shot learning, which leverages the limited available labeled data for improved prediction performance, while ensuring confidentiality of sensitive information.

Divided into two parts, the first addresses multimodal learning for fault diagnosis, and the second targets few-shot learning in NLP for textual data analysis.

À mes grands-pères, Jean-Claude et Roland.

ACKNOWLEDGEMENTS

A Ph.D. is not a straightforward nor easy path to follow, hence the quality of the interactions that one experiences along this journey are at utmost importance. In that sense and as I am deeply grateful to all the people that directly or indirectly contributed to the success of this Ph.D., I will not be brief.

First of all, I would like to extend my appreciation to the distinguished members of the jury: Sébastien Lefèvre and Stefan Duffner for the thorough reading and evaluation of my manuscript, Cécile Capponi and Laurent Amsaleg for your insights as examiners during the defense, and Vincent Mousseau for graciously chairing my defense. The quality discussions and remarks that you all brought to my attention allowed me to take a more critical look at some facets of my work, and contemplate new directions.

I remember my first visit to Saclay back in 2019, braving the snow for the first meeting with Céline and Michel which eventually paved the way for the internship preceding this thesis. From that day to the day of my defense, I was privileged to be supervised by a very supportive and empathetic team, which I guess is an invaluable asset in such a journey.

Céline, thank you for the trust you placed in me throughout this thesis, for your determination to prioritize the interests of doctoral students, and for always finding the time (even though I know it's a precious commodity in your schedule!) when necessary. I also greatly appreciated what we shared outside of work, be it about Loire wines, music, or riddles.

Myriam, thank you for co-supervising my thesis the way you did, especially for your invaluable support during the more challenging times when I began to question everything. It truly mattered in the successful completion of this thesis.

Michel, thank you for adeptly managing the occasional divergences between the academic side and industrial interests, and for granting me a certain freedom in research directions. Thanks also for the substantive advice related to the daily life of a doctoral student, especially during the lockdown periods.

I'm convinced that a thesis can only go well if well supervised and supported. In this sense, to all three of you, thank you again for the quality of your guidance.

Next, I'd like to thank all my colleagues. Especially Thomas, with whom I was closest during the second part of my thesis (I regret that our KNIVO collaboration didn't come to fruition); Yasmine for his ever-present availability and kindness; Victor for his initiative, ideas, and our volleyball discussions; and also Arthur, Etienne, and Manuel. Conversations with all of you were always engaging, whether work-related or not, and I hope to remain close to you all. Thank you, Pierre, for the collaboration that emerged towards the end of my thesis, and for your advice on my future career. Special thanks also to Fabienne, Vincent, Wassila, and Anaëlle for your positivity in the lab.

Within IRT SystemX, I remember all the doctoral students, especially Julien, Maria, Pascal, Clarisse, with whom we established a doctoral representation association, subsequently taken over perfectly by Tjark, Marina, Emmanuel. I also think of Simo, Kevin, Adrien, and Natkamon. Thanks to everyone for the kindness within this group and the *Exploding Kittens* and *Among Us* games during the lockdowns. Thanks also to Selma, Ibrahima, Junkai, and all MPO project partners for insightful scientific discussions and a great working atmosphere. Special thanks to Kristina for all the laughs, and also to Léa, Julie, Nicolas, and Flavien for the warm moments at lunch and regular favors.

Thanks to all my friends, who were there to take my mind off things during the toughest times, and who were understanding of my periods of social absence. Special thoughts to my three successive roommates: Dany, Thomas, and Vincent; my friends from Tours: Clément, Daivy, Pierre (both!), Quentin B., Tonin, Simon; and from Paris: Paul, Charlotte, Léa, Johan, Henri, Vincent W., Quentin P., Maxence, and many others.

I am also fortunate to have a tight-knit family, whom I would like to warmly thank. In particular, thanks to my parents for supporting all my decisions, and for giving me the education I received, that has led me here. Thanks to Théo, my brother, a steadfast supporter and lifelong loved one.

Last but not least, thank you, Sophiane, for accompanying, supporting, motivating, and being by my side throughout this experience. You have been my most precious ally, and I know that in some ways, you were also defending this thesis alongside me.

CONTENTS

LIST OF FIGURES	XI
LIST OF TABLES	XI
1 INTRODUCTION	1
1.1 Concrete motivations and context	1
1.2 Objectives and challenges	6
1.3 Outline and contributions	8
I PIONEERING MULTIMODAL LEARNING STRATEGIES FOR INDUSTRIAL FAULT DIAGNOSIS	11
2 BACKGROUND AND RELATED WORK	13
2.1 Addressing fault diagnosis with machine learning and deep learning	13
2.1.1 Background: Principles of the machine learning and deep learning paradigms	14
2.1.2 Machine learning and deep learning models for fault diagnosis	17
2.1.3 Fault diagnosis approaches using multimodal data	19
2.2 Multimodal learning	21
2.2.1 From multimodal perception to multimodal learning	21
2.2.2 Multimodal fusion: an overview	25
2.2.3 Deep neuronal architectures for multimodal representation learning . .	29
Conclusion	37
3 STREAMMUL T: A STREAMING MULTIMODAL TRANSFORMER FOR HETEROGENEOUS AND ARBITRARILY LONG SEQUENTIAL DATA	39
3.1 Introduction	39
3.2 Multimodal learning with heterogeneous and arbitrarily long sequential streams	42
3.2.1 Problem formalization	42
3.2.2 Positioning	44
3.3 Related work	44
3.3.1 Transformer architectures and unaligned modalities	44
3.3.2 Streaming input data	47
3.4 Proposed model	49
3.5 Experiments	52
3.5.1 Dataset and evaluation task	52
3.5.2 Experimental setting and results	53
3.5.3 Ablation study	56

3.6	Time and space complexities study	56
3.7	Implementation details	57
	Conclusion	60
4	THOUGHTS ON THE CHARACTERIZATION OF INFORMATION ACROSS MODALITIES	61
4.1	Introduction	61
4.2	Theoretical background	62
4.2.1	Multimodal learning provably performs better than unimodal	62
4.2.2	A brief recap on information theory	63
4.3	Maximizing redundant information	64
4.4	Characterizing complementary information	67
	Conclusion	70
II	TOWARDS REALISTIC FEW-SHOT TEXTUAL CLASSIFICATION	73
5	BACKGROUND AND RELATED WORK IN NLP: FROM SYMBOLIC METHODS TO FOUNDATION MODELS	75
5.1	Introduction	75
5.2	Early NLP methods	76
5.3	Word embeddings	79
5.4	Language models	82
5.5	Encoder-decoder architecture	85
5.6	Transformers	86
5.7	Foundation models	89
5.8	Few-shot learning in NLP	95
	Conclusion	99
6	A TRANSDUCTIVE APPROACH FOR PERFORMING FEW-SHOT CLASSIFICATION IN NLP	101
6.1	Introduction	101
6.2	Problem statement	102
6.2.1	Current methods limitations	102
6.2.2	Textual classification in few-shot setting	103
6.3	Transductive approaches for FSL in NLP	104
6.4	Experimental study of transductive few-shot inference for NLP classification	106
6.4.1	Limitations of existing benchmarks	106
6.4.2	Research questions and related results	107
	Conclusion	111
7	TEXTUAL FEW-SHOT CLASSIFICATION FOR API-BASED MODELS	113
7.1	Introduction	113
7.2	API based few-shot learning	116
7.2.1	Problem statement	116
7.2.2	Limitations of current methods	116

7.2.3	Proposed transductive approaches and baselines	117
7.2.4	A Fisher-Rao based regularizer	118
7.3	An enhanced experimental setting	118
7.3.1	Datasets	119
7.3.2	Model choice	119
7.3.3	Evaluation framework	120
7.4	Experiments	121
7.4.1	Overall results	121
7.4.2	Study under different data regimes	122
7.4.3	Ablation study on backbones	122
7.4.4	A dive into GPT-3.5 results	123
7.4.5	Multilingual experiment	124
7.4.6	Importance of model backbones on monolingual experiment	124
7.4.7	Importance of model backbones on multilingual experiment	125
7.4.8	Practical considerations	126
7.4.9	Links with the observations of Chapter 6	127
	Conclusion	127
8	CONCLUSION AND PERSPECTIVES	129
8.1	Summary of the contributions	129
8.2	Perspectives	131
8.2.1	Criticism and short-term perspectives	131
8.2.2	Long-term perspectives	132
9	APPENDIX	135
9.1	Proof of Theorem 1	135
9.2	Publication in the context of the MPO project	136
9.3	Résumé de la thèse en Français	144
	NOTATIONS	149
	ACRONYMS	149
	BIBLIOGRAPHY	153

LIST OF FIGURES

1.1	Pillars of Industry 4.0.	1
1.2	Example of a corrective maintenance report.	3
2.1	Example of multimodal perception of the environment.	22
2.2	Example of multimodal data acquirement in Industry 4.0 setting.	23
2.3	Illustration of the heterogeneity gap.	24
2.4	Different fusion techniques.	26
2.5	Multimodal representations mappings.	28
2.6	Joint and coordinated representations learning.	29
2.7	Early fusion techniques.	30
2.8	Multimodal autoencoder.	31
2.9	Intra-modality and inter-modality impacts of attention mechanism.	31
2.10	Visual Transformer architecture.	32
2.11	Multimodal segment embeddings.	33
2.12	Multimodal transformers variants.	34
2.13	VideoBERT architecture and pre-training.	36
2.14	CLIP Architecture.	36
3.1	Typical example of fault diagnosis task in the context of Industry 4.0.	40
3.2	Cross-modal attention block.	45
3.3	Cross-modal attention matrices.	46
3.4	Linearly growing receptive field linearly growing.	47
3.5	Forward step for the Augmented Memory Transformer.	48
3.6	Comparison of AM-TRF with Emformer.	49
3.7	Streaming Multimodal Transformer architecture.	50
3.8	Block processing for Multimodal learning in a streaming scheme.	51
3.9	Streaming Cross-modal Transformer module.	52
3.10	Training and inference flexible scheme.	53
3.11	Heatmap of StreaMulT attention weights.	55
4.1	Information diagram of two redundant modalities.	66
4.2	Information diagram of two mutually complementary modalities.	67
4.3	Information diagram of a modality-domination setting.	68
4.4	Information diagram of two modalities with no redundancy.	69
5.1	Comparison of CBoW and Skip-Gram approaches	81
5.2	Qualitative results for Word2Vec embeddings.	81
5.3	Neural language model architecture.	84

List of Figures

5.4	Sequence-to-sequence architecture.	85
5.5	Original transformer architecture.	87
5.6	Pre-training and fine-tuning paradigm.	90
5.7	Masked language modeling and similar pre-training objectives.	91
5.8	Timeline of released large language models.	93
5.9	In-context learning and instruction fine-tuning.	94
5.10	Few-shot learning paradigm.	96
5.11	Parameter-efficient tuning with adapter.	97
5.12	Prompt-based few-shot learning.	98
5.13	Inductive vs transductive settings.	99
6.1	N -shot K -way tasks example.	104
6.2	Comparison of cross-entropy-based and transductive-based approaches for different numbers of shots.	108
6.3	Comparison of cross-entropy-based and TIM-based approaches for BERT and RoBERTa backbones.	109
6.4	Comparison of BERT and RoBERTa backbones performances.	110
6.5	Comparison of cross-entropy-based and TIM-based approaches for different fine-tuning strategies.	111
7.1	API-based few-shot learning scenario.	116
7.2	Performance of the different pre-trained encoders on the monolingual datasets.	121
7.3	Effects of different ways and shots on test performance.	122
7.4	Impact of model size on performances.	123
7.5	Different losses when training a on GPT3.5 embeddings.	124
7.6	Performance of the different losses on multilingual datasets.	124
7.7	Performance of different pre-trained encoder on the monolingual datasets.	125
7.8	Performance of different pre-trained backbones on multilingual Amazon dataset.	126

LIST OF TABLES

2.1	Comparison of different approaches for multimodal representation learning. . .	35
3.1	StreaMulT results on CMU-MOSEI aligned.	54
3.2	StreaMulT results on CMU-MOSEI unaligned.	54
3.3	Ablation study on CMU-MOSEI aligned.	56
3.4	Time and space complexities.	56
3.5	Optimal hyperparameters for StreaMulT.	60
5.1	Summary of limitations of early NLP methods.	78
5.2	Summary of limitations of early NLP methods and word embeddings techniques.	82
5.3	Summary of advantages and limitations of general NLP methods and word embeddings techniques.	88
5.4	Pre-training objectives and their respective loss functions.	92
5.5	Overview of different transformer-based models.	93
6.1	Overview of the various datasets.	107
6.2	Comparison of CE-based and TIM-based approaches performances for different fine-tuning strategies.	111
7.1	Statistics of considered NLP datasets.	119
7.2	Preliminary experiment results.	120
7.3	Aggregated performances over the different datasets and considered backbones.	121
7.4	Global results for multilingual Amazon dataset.	126
7.5	Training time for 1 episode on a M1-CPU.	127

1 INTRODUCTION

1.1 CONCRETE MOTIVATIONS AND CONTEXT

PREDICTIVE MAINTENANCE AT THE AGE OF INDUSTRY 4.0

The Fourth Industrial Revolution, often referred to as *Industry 4.0*, is currently shaping our era with a new phase in the transformation of the industrial sector. Built on the digital revolution (the Third Industrial Revolution), Industry 4.0 is characterized by a fusion of technologies that blur the lines between the physical, digital, and biological spheres, leading to a systemic transformation of the entire value chain of the manufacturing sector (Schwab 2017). At the heart of Industry 4.0 lies a series of technological advancements, represented in Figure 1.1, such as the Internet of Things (IoT), Cyber-Physical Systems, Cloud Computing, Digital Twins, and Artificial Intelligence (AI). These advancements have resulted in a paradigm shift from traditional, linear manufacturing processes to complex, integrated systems where machinery and equipment can communicate and cooperate with each other and with humans in real time. This concept is commonly referred to as the *smart factory* (B. Chen et al. 2017). To illustrate, in a smart factory environment, an assembly line robot is capable of autonomously communicating with other machinery to adjust its production pace based on real-time demand or even preemptively order replacement parts when a failure is anticipated. Similarly, smart logistics systems in Industry 4.0 can dynamically reroute shipments based on real-time conditions, reducing delays and enhancing efficiency.

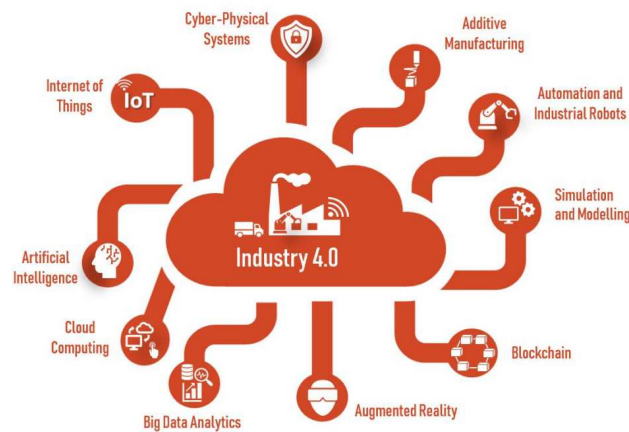



Figure 1.1: Pillars of Industry 4.0. Figure from (Ryalat et al. 2023).

1 Introduction

One of the most significant transformations in Industry 4.0 is the shift towards predictive maintenance. Traditional maintenance policies based on estimated lifetimes are giving way to systems that can predict failures and schedule maintenance in real time. Predictive maintenance, driven by real-time data from various sensors and machines, aims to prevent unplanned downtime, enhance efficiency, and increase the overall life span of the machinery (Mobley 2002). The emergence of predictive maintenance systems has been fueled by the massive availability of data from interconnected and intelligent automation systems that Industry 4.0 puts at the center of global production, particularly through the integration of smart sensors aiming to build global control systems such as Supervisory Control And Data Acquisition (SCADA). The induced challenge and opportunity - that motivate this thesis - lies in exploiting the vast data acquired by these sensors for fault monitoring, diagnosis, and more generally predictive maintenance.

Central to these developments is the role of data, that is the cornerstone of Industry 4.0. The interconnected sensors and devices nowadays generate an unprecedented amount of data that embodies a rich source of insight into the functioning, performance, and potential anomalies within the systems, but that are collected from various sources and in various forms, leading to the emergence of complex and often heterogeneous data sources. Consider the example of an automated production line in a smart factory. As part of its operation, it continuously generates multiple types of data through various sensors and systems. For instance, vibration sensors on the machinery provide data on the machine's physical state, indicating its stability or any unusual shaking that could signify a potential issue. Temperature sensors provide another form of data, offering insight into the machine's thermal conditions, and cameras installed in strategic locations capture real-time visual data of the machine's operation and the production process. Simultaneously, the system also generates textual data in the form of operational logs or maintenance reports that provide contextual information about the machine's operational status, historical issues, or previous performed maintenances.

The key advantage of considering this multimodal data is that it offers a comprehensive and detailed perspective of the system's state. Each modality, whether it be sensor readings, images, or textual reports, captures different facets of the system's condition, thereby enriching the information available for fault diagnosis or other predictive maintenance tasks. For instance, while real-time sensor data could provide immediate insights about the system's performance parameters such as temperature or vibration, image data could reveal physical anomalies or damages, and textual reports could offer context or detailed accounts of previous incidents or interventions. Besides, multimodal data introduces the capability for cross-verification of faults. An anomaly detected in one modality can be cross-checked and confirmed with information from another modality, adding a layer of redundancy and increasing the confidence of the fault detection process. This becomes particularly crucial when dealing with complex or subtle faults that may not be readily discernible in a single data modality, but become evident when multiple data types are analyzed collectively. But even more significantly, the integration of different modalities allows us to identify faults that might remain hidden when considering each modality in isolation. A minor anomaly in one modality, seemingly insignificant on its own, could be the critical piece of the puzzle when viewed in the context of other modalities, leading to the identification of a potential fault.



OMIS

Corrective Maintenance Report

Instrument:	RAMAN LIDAR
Location:	Central Facility
Date/Time (GMT)	12/15/2000 2000
Technician:	Chris Martin
Maint. Type:	Unscheduled CM
Problem Fixed:	No
Suspected Cause:	HW Failure
Component:	Laser
Problem Description:	The Raman Lidar laser energy had fallen to around 200 mJ. The lamps were due to be replaced, but the energy seemed to drop off rather quickly the past couple of days.
Action Performed:	Inspection of the optics during lamp replacement found that the Pockells Cell was burned. The Pockells cell had just been installed by Continuum on 10/19 and should be under warranty. Scheduled a Continuum service call for 12/18. System off line until then.

Figure 1.2: An example of a corrective maintenance report of a climate research facility. Figure from (Teske et al. 2001).

However, **the process of integrating these diverse data types presents a unique set of challenges, necessitating careful and innovative approaches for successful implementation.**

While the advantages of using multimodal data are apparent, the unique role of textual data must be emphasized. Textual data, often generated in the form of operational logs, or maintenance reports, provide a rich and contextualized understanding of system operations and past incidents (see Figure 1.2). Unlike numerical or visual data, textual data contains nuanced information that directly reflects the *expert* knowledge and interpretative insights of human operators, making it a valuable resource for fault diagnosis. For example, maintenance records can provide crucial insights into the system's historical problems, the repairs undertaken, and their effectiveness, aiding in the prediction of future faults. Even more, incident reports often describe the circumstances leading up to a fault, providing a narrative that can help identify patterns or triggers associated with system failures. Moreover, textual data can serve as a connecting bridge among different modalities, providing context and interpretive lens to raw numerical or visual data. A notation in a maintenance report might clarify, nuance or amplify an anomaly in the vibration data, mainly depending on the chosen words. This integration of textual data into the fault diagnosis process illuminates these connections and therefore enriches the analysis. However, the challenge lies in the fact that **this rich textual data is not abundant, making it harder to effectively leverage for our analyses.**

In essence, multimodal data, including textual data, are a cornerstone of Industry 4.0, providing a more comprehensive understanding of system operations. Each modality, with its distinct

perspective, enriches the information available for predictive maintenance. Particularly, textual data, encapsulating the richness of human language and expert insight, provides subtleties and nuanced patterns that sensor or image data may overlook. Through the integration of these diverse modalities, we aim to design a global representation of a system's state, enhancing the reliability of fault detection and predictive maintenance strategies. Ultimately, this leads to improved operational efficiency, reduced downtime, and optimized performance within the industry. However, the handling and interpretation of such complex data require advanced methods, which is where deep learning (DL) and other AI techniques come into play.

JOURNEY THROUGH ARTIFICIAL INTELLIGENCE: FROM EARLY ENDEAVORS TO THE ADVENT OF DEEP LEARNING

The field of AI has seen rapid development since its inception in the 1950s. The historic Dartmouth workshop (McCarthy et al. 2006), along with Alan Turing's groundbreaking paper *Computing Machinery and Intelligence* (Turing 1950), laid the foundations for this exciting field of study. While the original question - "Can machines think?" - and the pursuit of *strong* AI, including artificial consciousness, still remains elusive, it has nonetheless inspired the creation of autonomous systems that rival, and sometimes surpass, human performance in specific tasks. Thus, IBM Deep Blue literally beat chess world champion Gary Kasparov in 1997, while more recently Deepmind reinforcement learning models AlphaGo (Silver et al. 2016) and AlphaStar (Vinyals, Babuschkin, et al. 2019) achieved the same performance in more complex games, respectively Go and Starcraft 2. In addition to gaming, AI has been instrumental in transforming many industrial sectors. For instance, in healthcare, AI has not only been used for skin cancer detection (Esteva et al. 2017) but has also demonstrated promising results in diagnosing diabetic retinopathy (Gulshan et al. 2016). In biology, apart from revolutionizing protein-structure prediction with AlphaFold (Jumper et al. 2021), AI has been utilized in drug discovery and development (Stokes et al. 2020). The aeronautics sector has witnessed the conception of autonomous vehicles powered by AI (Grigorescu et al. 2020), while in Natural Language Processing (NLP), neural machine translation systems have significantly improved thanks to AI, notably in 2014 (Sutskever, Vinyals, et al. 2014), and more recently with the introduction of Transformer architecture (Vaswani et al. 2017). Furthermore, AI has made significant strides in predictive maintenance through machine health monitoring (Yiwei Cheng et al. 2019).

These advancements can be attributed largely to the success of Machine Learning (ML), and more recently, Deep Learning (LeCun et al. 2015; M. Raghu et al. 2020). Machine learning, a branch of subsymbolic AI, leverages past experiences, represented by annotated datasets, to build predictive models. This process involves an iterative optimization problem using the available data, placing significant importance on the representation of the input data. Unlike traditional ML, DL architectures use generic priors to learn a suitable representation of input data through non-linear transformations (Bengio, Courville, et al. 2013). This learned representation aims to extract salient features from the raw data structure, which is then used by a classifier to make relevant decisions. Over the past years, the community put a lot of emphasis on Representation Learning: the more expressive the representation is, the more effective and generalizable the model will be.

However, a key challenge that constrains DL models is the data requirement. These models typically require vast amounts of data to perform optimally, and this prerequisite often outstrips the available labeled data, especially in niche or sensitive domains. While the initial successes of Deep Learning were largely popularized by supervised learning approaches, where models were trained on large labeled datasets, the AI research community therefore quickly recognized the need for more versatile learning paradigms, especially for scenarios where labeled data is scarce or non-existent. This gave rise to the development of multiple learning paradigms to optimize data usage:

- **Unsupervised Learning:** These approaches, such as clustering (J. Xie et al. 2016) and dimensionality reduction (Geoffrey E Hinton et al. 2006), train models using unlabeled data, discovering hidden patterns and structures without guidance.
- **Semi-Supervised Learning:** As the name suggests, this technique utilizes a mix of labeled and unlabeled data for training (Zhu 2005). The idea is to leverage the unlabeled data to enhance the learning process, particularly when labeled data is limited.
- **Transfer Learning:** This paradigm revolves around the reuse of pre-trained models on new, related tasks. The principle is to leverage the knowledge acquired from one task to improve learning in another, reducing the need for extensive labeled data in the new task (S.J. Pan et al. 2010).
- **Domain Adaptation:** This approach aims to adapt models trained on one domain (source) to perform well on a different but related domain (target), especially when the target domain has limited labeled data (Ganin et al. 2016). It is a subset of transfer learning that addresses shifts in data distribution between tasks.
- **Few-Shot Learning (FSL):** FSL (Vinyals, Blundell, et al. 2016) focuses on training models to make accurate predictions with minimal labeled examples. It leverages techniques that emphasize generalization, enabling models to learn effectively from a small sample size.

More recently the AI community has turned towards self-supervised learning, a paradigm in which models are pre-trained on large amounts of unlabeled data and then fine-tuned on a smaller labeled dataset. This approach not only makes efficient use of the available data but also equips models with a better generalization capacity. The advent of self-supervised learning is complemented by the scaling paradigm (Kaplan et al. 2020; Rosenfeld et al. 2020), which posits that model performance can be improved by simply increasing the model size, data size, and the computational resources, given the right model architecture and learning algorithm. This has lately led to the rise of 'Foundation Models', such as GPT-4 (OpenAI 2023), which are large, general-purpose models trained on massive data from the internet. These models can be fine-tuned on specific tasks with relatively little data, redefining the state of the art in numerous AI applications. As we advance, the focus remains on harnessing these paradigms to build more effective, robust, and versatile AI systems.

1.2 OBJECTIVES AND CHALLENGES

In the evolving landscape of Industry 4.0, this thesis takes place in the project *Maintenance Prévisionnelle et Optimisation*¹ (MPO) of IRT SystemX. This project aims to overcome the technological and methodological barriers of predictive maintenance and the combination of maintenance policies in production systems, made possible by new technologies and artificial intelligence, and the computing power of the machines, in order to optimize their maintenance in operational condition. In the context of this project, the global objective of this thesis was to study predictive maintenance and more precisely **fault diagnosis** under the spectra of deep learning and multi-modal and heterogeneous data sources. It also includes some works on designing a specific use case, based on a three-tank system, aiming to illustrate the fault diagnosis on a simple applicative example and proposing baselines to tackle the challenges of predictive maintenance data and tasks. The related article (Pellegrain, Batteux, et al. 2022) was published in a national conference and is relegated to [Section 9.2](#). While situated within the highly applied context of Industry 4.0 and the MPO project, the ambition of this thesis extends beyond the development of models for specific applications. Instead, the main goal is to address the challenges methodologically, intending to introduce novel techniques for the general framework of harnessing multimodal and heterogeneous data. These newly proposed methods aim to unlock the potential of data diversity in Industry 4.0, thereby enabling enhanced fault diagnosis and other predictive maintenance tasks. As such, the focus of this thesis lies not in crafting a solution for a specific application, but rather in contributing methodological advancements that can be universally applied in the realm of data exploitation in Industry 4.0.

However, each of these ambitious goals also presents its unique set of challenges and considerations that requires careful and meticulous addressal.

- (i) A first objective deals with **the dynamic and real-time nature of industrial systems**. These systems generate data streams that are continuously acquired, often with heterogeneous acquisition frequencies. For instance, some sensors might collect data at millisecond intervals, while others might gather information every few minutes or even hours. The challenge here is to **manage these data streams effectively, in a time and memory-efficient manner**. Due to the real-time demands, it is crucial to devise strategies that are capable of rapidly adapting to changing conditions. These strategies must be able to provide meaningful insights for fault diagnosis while maintaining acceptable computational efficiency. Within this context, the task of revealing a strong diagnostic signal from potentially weak individual signals becomes even more critical. As an example, an immediate increase in temperature might be less alarming than a slower, yet consistent, increase over a period of time, which could indicate a potential failure or malfunction.
- (ii) The second objective emerges from the need to tackle the **complexity of integrating data with heterogeneous structures**. This data is frequently sourced from various sensors or systems, with each source providing a unique perspective on the system's condition. An example that illustrates this scenario could be a vibration sensor indicating an anomaly. However, when this data is coupled with additional information such as system's images

¹<https://www.irt-systemx.fr/en/projets/mpo/>

or noise, the diagnostic potential becomes far more precise and insightful. It is clear that considering such heterogeneity in data sources and their representations is crucial in improving the accuracy and reliability of fault diagnosis. This challenge is not unique to industrial systems but common to many domains, which has led to the introduction of multimodal learning and fusion paradigms. Many approaches have been proposed under these paradigms, aiming to capture the richness of these multiple perspectives and translate them into robust decision-making strategies. These strategies seek to consolidate data from different modalities, each contributing uniquely to the overall understanding of the system. However, a closer look reveals an under-explored aspect within these strategies: the interactions among features from different data sources. While these interactions can bring critical insights, they are often not explicitly considered in the fusion models. Therefore, we do not fully control how they influence the decision-making process. Further, when these interactions are taken into account, it is typically the redundant interactions that are most often considered. The complementary interactions, that amplify or refine the understanding of a system when considered together, are frequently overlooked. Therefore, the second objective is twofold: firstly, to better integrate multisource heterogeneous data; secondly, to reinforce our understanding and control the interactions among these data sources. This poses a broader question: **how can we design fusion models that not only effectively integrate data from multiple sources and modalities but also take advantage of the redundancy and complementarity among these features?**

- (iii) The third objective focuses on leveraging the wealth of information captured in textual data, particularly in maintenance reports. These documents, often written by experts, encapsulate rich, contextual information about the system's state, historical issues, and previous maintenance activities. The growing interest in exploiting all modalities for addressing Industry 4.0 tasks results in more open-access resources (Akhbardeh et al. 2020). However in real-world, the scarcity of such reports, combined with the highly specialized and industry-specific vocabulary, makes their processing and understanding a challenging task. Traditional methods of training DL models require many annotated data to understand and adapt to this specific language use. Given the rarity and specificity of these maintenance reports, applying usual supervised learning paradigms becomes unrealistic. Recent advancements in language models provide a promising direction for interpreting these reports, yet their application is not straightforward. **How can we effectively harness the expressiveness of human language encapsulated in these reports, especially when they are scarce? How can we adapt these advanced language models to the specific language used in these maintenance reports?** Moreover, the usage of these models should not compromise the privacy and confidentiality of sensitive information, adding another layer of complexity.

Addressing these challenges forms the core of this thesis. By exploring novel strategies and techniques, we aim to help in surmounting these obstacles and reveal the full potential of multimodal and textual data in predictive maintenance.

1.3 OUTLINE AND CONTRIBUTIONS

In line with the previously defined challenges, this thesis presents two distinct contributions, each devoted to a specific area of research: Multimodal Learning and FSL in NLP. This also defines the outline of the thesis, divided in two primary parts.

I. Exploiting multimodal data for fault diagnosis. The first part begins with a clear, pragmatic need from the industrial field to diagnose faults in complex, multimodal systems. This concrete motivation led us towards the development of a more abstract theoretical framework based on multimodal learning, which is inherently motivated by the multimodal nature of our real-world environment.

In [Chapter 2](#), we revisit related established concepts such as multimodal fusion and representation. We analyze the evolution of these paradigms, from their early stages to the advent of DL-based multimodal representations. This comprehensive review also includes an analysis of the few attempts that have applied ML for fault diagnosis, focusing on the pragmatic constraints of fault diagnosis that have not been addressed by previous multimodal approaches. Specifically, the challenges of handling arbitrarily long data streams in a memory and time-efficient manner, and performing inferences in streaming mode, are examined in depth.

Bridging the gap between theory and application, in [Chapter 3](#) we introduce "StreaMulT," a Streaming Multimodal Transformer. This innovative algorithm offers a unique solution to the challenges posed by Industry 4.0 systems' complexity. By employing cross-modal attention and a memory bank, StreaMulT is capable of processing arbitrarily long input sequences during training. Further, it operates in a streaming mode during inference, thereby managing the temporal unalignment of multimodal data and balancing the differences in data acquisition frequency. This contribution led to the article (Pellegrain, Tami, et al. 2022), published in the *Conférence Nationale d'Intelligence Artificielle 2022*.

[Chapter 4](#) extends the discussion to the theoretical realm, presenting an exploration of multimodal representation and fusion and highlighting the need for further research in datasets and architectures for effective multimodal learning.

II. Leveraging scarce and specific textual data in a realistic setting The second part of the thesis begins with [Chapter 5](#), offering an extensive overview of NLP methodologies, starting with early techniques centered on feature engineering and statistical word properties and transitioning towards DL approaches and recent Foundation models. Furthermore, the chapter examines associated works in Few-shot learning, shedding light on the latest progress and challenges in this research area.

In light of these developments, we notice a gap in the field when dealing with scenarios where labeled data are rare. Current FSL methods in NLP, mainly based on the prompting strategy, show limitations, especially for realistic classification tasks with a large number of classes. These limitations are primarily due to engineering efforts required to make these methods work effectively in such situations. To cope with these issues, in [Chapter 6](#) we revisit transductive learning in the NLP field, trying to reproduce the success encountered in computer vision. This paradigm, unlike inductive learning, enables the effective utilization of limited labeled data by taking advantage of the statistics of unlabeled data.

Then, we consider the increasing prevalence of proprietary and closed Application Programming Interfaces (APIs) for Large Language Models (LLM) in [Chapter 7](#). We introduce a new parameter-free regularizer based on the Fisher-Rao loss, which demonstrates its effectiveness and applicability in this setting. This differs from current methods and provides a novel way to tackle FSL problems. In such a scenario, our transductive approach enables fast and efficient predictions without the need to share sensitive label information, thus adapted data-privacy constraints. This not only paves the way for improved performance but also opens new research ideas for practical applications in the field of FSL. The article that emerged from this contribution is currently under review for publication in an international journal.

Finally, [Chapter 8](#) concludes this thesis and proposes perspectives for both parts.

PART I

PIONEERING MULTIMODAL LEARNING STRATEGIES FOR INDUSTRIAL FAULT DIAGNOSIS

2 BACKGROUND AND RELATED WORK

CHAPTER'S SUMMARY

In this chapter, we give the reader the background needed to motivate and understand the first part of this thesis. We start by presenting fundamentals of Fault diagnosis theory in [Section 2.1](#) and we review existing strategies to tackle this problem, focusing on ML approaches and exploring the few attempts that considered data from heterogeneous modalities. In [Section 2.2](#), we introduce the multimodal learning paradigm, with a particular emphasis on multimodal fusion. From there, we propose an overview of developed methodologies, beginning with older works relying on simple fusion strategies such as concatenation, and more focused on which level to realize the fusion. We then point out the advantages of building expressive data representations, which is mostly feasible by the mean of Deep-Learning-based architectures, and the closeness between multimodal fusion and multimodal representation. We therefore explore approaches on Multimodal Representation Learning, which are nowadays mainly based on the Transformer architecture.

2.1 ADDRESSING FAULT DIAGNOSIS WITH MACHINE LEARNING AND DEEP LEARNING

Due to plenty of causes - both internal and external - industrial machines are likely to suffer a fault at some point (e.g., corrosion). If not detected, these faults can lead to the incidence of failures (e.g., leakage). That is a major issue since it means a financial loss for the company and sometimes much more when human lives are at stake. To address this problem, it is common to perform fault diagnosis. Following (Isermann 2005) terminology, we properly define these previous terms.

DEFINITION

Definition 1. A *fault* is an uppermitted deviation of at least one characteristic property (feature) of the system from the acceptable, usual, standard condition.

A *failure* is a permanent interruption of a system's ability to perform a required function under specified operating conditions.

Fault monitoring refers to the detection of a fault occurrence.

Fault diagnosis consists in determining the type, size and location of the most possible fault, as well as its time of detection.

2 Background and Related Work

In practice though, in the literature it is common to write *fault diagnosis* to refer to both fault detection and its diagnosis as defined above.

The pioneer series of three articles of Venkatasubramanian et al. (Venkatasubramanian et al. 2003) is one of the first works to list and categorize the different methods of fault diagnosis; and therefore constitutes the starting point of our review. This series classifies fault diagnosis approaches depending on both the a priori knowledge one has on eventual faults, along with how they would be expressed through the acquired data of the system (*i.e.* fault symptoms). Two different kinds of strategies can be distinguished in the literature. A first family of approaches, named **model-based**, uses the a priori knowledge by the system by a physical model. On the other side, the approaches only relying on the history of acquired data are called **data-based**. While model-based methods can be well suited when one has a nice a priori understanding of physical laws governing the system, they become less relevant otherwise. Thus, when the considered system reaches a certain level of complexity, inter-components interactions can less easily be modelled. To address this, data-based approaches provide a viable alternative: the designed model aims to learn these components dependencies from the data history. We mostly focus on data-based works in our review, and more precisely on ML ones.

In the next section, we first introduce the paradigm of Machine Learning (ML) and Deep Learning (DL) through the lens of Statistical Learning theory and the popular supervised learning framework. In a second time, we review the different approaches that make use of ML and DL to tackle Fault diagnosis.

2.1.1 BACKGROUND: PRINCIPLES OF THE MACHINE LEARNING AND DEEP LEARNING PARADIGMS

Given a set of observations of a phenomenon, the aim of Statistical Learning (V. Vapnik 2000) is to build a model of this phenomenon that can then perform inference on new data, that is, make predictions. Machine Learning is a framework that tries to automate this learning process using algorithms to design a function that maps input observations to desired outputs. Based on statistics and optimization problems, this procedure selects the function that both best fits the observed data, and stays generalizable to unobserved data.

Formally, we consider an input space \mathcal{X} and an output space \mathcal{Y} . We consider tuples of observations $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$, that are viewed as realizations of the random variables X and Y respectively. Considering a dataset $\mathcal{D} = (\mathbf{x}^i; y^i)_{i=1}^n$ containing n independent and identically distributed data pairs sampled from a distribution density $p_{X,Y}$, unknown but such that:

$$p_{X,Y}(\mathbf{x}, y) = p_{Y|X}(y|f^*(\mathbf{x}))p_X(\mathbf{x})$$

we seek to address the related task, that is learning a function (*i.e.* a model) $f : \mathcal{X} \rightarrow \mathcal{Y}$ approaching f^* , the true unknown mapping of the task.

To assess for the quality of the model f , we consider a loss function $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}^+$, such that $\mathcal{L}(f(\mathbf{x}), y)$ measures the point-wise error when the model predicts $f(\mathbf{x})$ instead of y .

To fulfill the objective of learning a model close to the true mapping f^* , the usual learning paradigm is to minimize the population risk $R(f)$ defined in the following:

DEFINITION

Definition 2. Population Risk

Let $\mathcal{D} = (\mathbf{x}^i; y^i)_{i=1}^n$ a dataset of *i.i.d.* data pairs sampled from a distribution $p_{X,Y}$, $f : \mathcal{X} \rightarrow \mathcal{Y}$ a prediction function, \mathcal{L} a loss function. The population risk associated to f is defined as the expected loss:

$$R(f) = \mathbb{E}_{p_{X,Y}}[\mathcal{L}(f(X), Y)] \quad (2.1)$$

where $\mathbb{E}_{p_{X,Y}}$ is the expectation associated to distribution $p_{X,Y}$. As we usually cannot access the true distribution p , a common surrogate is to minimize the Empirical Risk.

DEFINITION

Definition 3. Empirical Risk

Let $\mathcal{D} = (\mathbf{x}^i; y^i)_{i=1}^n$ a dataset of *i.i.d.* data pairs sampled from a distribution p , $f : \mathcal{X} \rightarrow \mathcal{Y}$ a prediction function, \mathcal{L} a loss function. The empirical risk $\hat{R}_n(h)$ is defined as the empirical mean loss measured on the dataset:

$$\hat{R}_n(f) = \frac{1}{n} \sum_{i=1}^n \mathcal{L}(f(\mathbf{x}^i), y^i) \quad (2.2)$$

Hence, for a class of functions \mathcal{F} , the Empirical Risk Minimization (ERM) algorithm consists in finding $\hat{f} := \arg \min_{f \in \mathcal{F}} \hat{R}_n(f)$. The hypothesis space \mathcal{F} represents the family of models (for instance linear functions) on which to minimize the Empirical Risk, and is usually chosen by the learner in preamble of the procedure, following inductive biases regarding the input data and the task. From then, we note our model f_ψ , where $\psi \in \Psi$ are learnable parameters, and Ψ is the parameter space defined by the chosen family of models (for instance, vectors of weights and biases defining the linear models). The learning objective is now to find $\hat{\psi}$ that minimizes the empirical risk:

$$\hat{\psi} := \arg \min_{\psi \in \Psi} \hat{R}_n(f_\psi)$$

Finally, we can decompose f as $f = h \circ g$, in which $g : \mathcal{X} \rightarrow \mathcal{Z}$ represent a feature extraction module, that maps input observations to a latent space \mathcal{Z} and $h : \mathcal{Z} \rightarrow \mathcal{Y}$ a predictor that maps the latent representations to the output space. By writing g and h as parametric functions, and noting $\psi = (\theta, \phi) \in \Theta \times \Phi$, we note $f_\psi = h_\phi \circ g_\theta$. In the classical shallow ML setting, the manually designed feature extraction module g is fixed, and the ERM thus consists in optimizing only the predictor h_ϕ on parameter space Φ , that is, find $\hat{\phi}$ such that:

$$\hat{\phi} := \arg \min_{\phi \in \Phi} \hat{R}_n(h_\phi \circ g)$$

DEEP LEARNING

Over the last decade, the advent of DL architectures (LeCun et al. 2015), demonstrated their superiority over classical ML approaches in numerous application fields and their related tasks. At the heart of this paradigm: deep neural networks. A deep neural network of L layers is a function $f : \mathcal{X} \rightarrow \mathcal{Y}$ such that:

$$\forall x \in \mathcal{X}, f(x) = h \circ g(x) = h \circ g_1 \circ \dots \circ g_L(x)$$

Compared to previously formalized shallow models $f = h \circ g$ considered in classical ML approaches, deep neural networks' feature extraction modules g are composed of L stacked layers, that will also be optimized during the learning procedure, and thus not manually designed. These layers essentially characterize the whole network as they condition the learned representations of input data, which, if expressive enough, only needs a simple predictor h to effectively address a task.

A traditional neural network architecture is for instance the Multi-Layer Perceptron (MLP), that is composed of stacked linear functions, followed by non-linear activations, *i.e.* for $i = 1, \dots, L$,

$$g_i(x) = a_i(w_i^T x + b_i)$$

with w_i and b_i being the i^{th} -layer associated weights and bias, and a_i being the non-linear activation function, usually hyperbolic tangent, sigmoid function, softmax function, or rectified linear unit function (Goodfellow et al. 2016).

This breakthrough in AI quest is mainly explicable in DL ability to learn *good* representations from input data (Bengio, Courville, et al. 2013), compared to classical feature extraction modules. Their design consisting of stacked modules followed by non-linear activations offers the possibility to learn hierarchical and distributed representations in which last layers thus represent concepts of a higher abstraction, expressed as a combination of simpler components learned in first layers. These properties tend to facilitate the encoding of factors of variation of the input data, while being more invariant to meaningless noise. Therefore, many current works focus their energy on the design of representation learning algorithms that integrate such generic properties.

From there, we can rewrite the true mapping from inputs to outputs f^* such as $f^* = h^* \circ g^*$ with $g^* : \mathcal{X} \rightarrow \mathcal{Z}$ being the true mapping from input to latent space and $h^* : \mathcal{Z} \rightarrow \mathcal{Y}$ the true mapping from latent to target space. The sampling distribution $p_{X,Y}$ of the considered dataset \mathcal{D} can now be written as:

$$p_{X,Y} = p_{Y|X}(y|h^* \circ g^*(\mathbf{x}))p_X(\mathbf{x}) \tag{2.3}$$

and we now aim to find f that minimizes the associated empirical risk:

$$\hat{f} := \arg \min_{f \in \mathcal{F}} \hat{R}_n(f) \quad (2.4)$$

$$= \arg \min_{f \in \{h \circ g | g \in \mathcal{G}, h \in \mathcal{H}\}} \hat{R}_n(f) \quad (2.5)$$

with \mathcal{F} , \mathcal{G} , \mathcal{H} the class functions defining the hypothesis spaces. Using the parametric notation, we aim to find $(\hat{\theta}, \hat{\phi})$ such that

$$(\hat{\theta}, \hat{\phi}) = \arg \min_{(\theta, \phi) \in \Theta \times \Phi} \hat{R}_n(g_\theta \circ h_\phi)$$

When solving this optimization problem, we hope that the learned parameters $(\hat{\theta}, \hat{\phi})$ also minimize the population risk over unseen new samples, so that the inference function can be reliably used to solve the task of interest. To realize an effective learning procedure, the learner should then, based on priors regarding the task of interest and input data:

- Define an adequate hypothesis space defining the family of considered models (for instance Convolutional Neural Networks), through the parameter space $\Psi = \Theta \times \Phi$;
- Specify an adequate loss function \mathcal{L} to measure the pointwise prediction error of the model (for instance the Cross-Entropy loss);
- Select an appropriate learning procedure to solve the optimization problem induced by ERM (for instance, using Stochastic Gradient Descent algorithm or derivatives such as SGD with momentum (Sutskever, Martens, et al. 2013) or Adam optimizer (Kingma et al. 2015));
- Design a testing procedure to evaluate the model's performance on unseen data and therefore get insight on its generalization ability.

2.1.2 MACHINE LEARNING AND DEEP LEARNING MODELS FOR FAULT DIAGNOSIS

Several reviews (cited thereafter) list the different ML architectures designed for tackling the fault diagnosis problem. Some of these reviews adopt an industrial-domain-specific position: while (Nor et al. 2019) expose fault diagnosis methods that have been used for chemical process systems, (S. Zhang et al. 2020) focus on bearing faults, whereas (Rogers et al. 2019) only consider residential air conditioning systems. These studies mainly motivate their approach by the consequences of fault occurrences in their relative fields, such as the over-consumption of electricity and the induced economic costs (Rogers et al. 2019). Besides, the methods listed in these reviews are presented as relevant for dealing with data relative to these applicative fields. Therefore, (S. Zhang et al. 2020) essentially consider vibration and stator current data, as contained in the Paderborn dataset¹; whereas (Rogers et al. 2019) rather present models calibrated for thermostat and humidity data, with for each of these approaches an important and non-scalable work that consists in

¹Available online: <https://mb.uni-Paderborn.de/kat/forschung/datacenter/bearing-datacenter>

designing specific hand-crafted feature extractor g for the specific applications.

By contrast, other works adopt a more methodological position regarding their reviews of state-of-the-art algorithms for addressing fault diagnosis (Palade et al. 2006). This is more in adequation with our positioning. Most recent ones (Angelopoulos et al. 2019; Z. Li 2018; Reis et al. 2017) motivate their work by the emergence of new practical challenges induced by the arrival of Industry 4.0 era, such as notably the ability to handle massive and multi-sources data with a short-time response. These reviews qualify ML methods as more effective compared to model-based approaches when fault profiles are complex, such as (S. Zhang et al. 2020), which mention the limits of model-based approaches for the early detection of faults, due to symptoms that are untraceable by this kind of models. They also point out model-based approaches' difficulty to disentangle the simultaneous occurrences of different faults.

Although some articles only consider fault detection (Luo et al. 2018; Wen et al. 2019), the vast majority also considers fault isolation and identification². However as emphasized by (Reis et al. 2017), in practice two methodologies co-exist. On the one hand, Statistical Process Control community sequentially processes fault detection and fault isolation and identification. On the other hand, ML community often processes these two tasks in a simultaneous fashion, in the form of a $(C + 1)$ -classes classification, decomposed into one class of normal functioning mode and C distinct faulty functioning modes.

As presented in (Z. Li 2018), ML models used for fault diagnosis are generally composed of a feature-extraction module and a diagnosis module. In that configuration, the former feeds the latter relevant elements computed from raw data. Some feature-extraction modules focus on time domain to catch and characterize information contained within time series acquired from the system sensors, using for instance neural networks (Zarei et al. 2014). It is also common to use signal processing tools in order to exploit features from the time series in the frequency domain. (Yukun Liu et al. 2010) and (Taj et al. 2017) thus respectively use Fourier and Laplace transforms to this purpose. Finally, other approaches choose to work in the time-frequency domain, through the usage of wavelet transforms for instance (Z. Zhang et al. 2013). The choice of feature-extraction module is strongly influenced by the structure of input data and the subsidiary task, therefore by the a priori knowledge of its designer. The very diagnosis module is then composed of:

- either a first detection submodule aiming to perform fault monitoring, followed by a second classification submodule performing fault isolation and identification;
- either a unique classification module carrying out simultaneously both fault detection, and fault isolation and identification.

In a supervised setting, the unique classification module fed with extracted features is free to use any ML model: Support Vector Machine (Konar et al. 2009), Random Forest (B.-S. Yang et al. 2008), shallow neural networks (Jafar et al. 2010), Recurrent Neural Networks (RNN) (Yam et al. 2001), and so on. This scheme of performing simultaneously fault detection and classification has however been sometimes criticized (Reis et al. 2017), as it might lead to practical issues:

- fault occurrences that might lead to failures and dreaded event are often scarce in real datasets. This results in an imbalanced dataset problem, exacerbated the more faulty classes one considers.

²note that (Angelopoulos et al. 2019) sometimes use the word "diagnosis" to evoke fault detection though

- For this kind of tasks, a prediction error will have the same weight during the learning phase, regardless of which misclassification has been made. However, depending on the system criticality, one would like to put a lot more emphasis on the fault detection rather than on its proper identification.

To cope with these issues, a prior monitoring task can be realised using anomaly detection methods (Goldstein et al. 2016). Similarly to the architectures designed in Statistical Process Control community's works, these semi-supervised methods model the normal functioning mode of the system during the learning stage, and classify as fault the datapoints which deviate significantly from this model's prediction at test time. These approaches are more robust to imbalanced datasets and can then be coupled with a classification model to perform the isolation and identification task. Lastly, if the normal functioning mode conditions are unknown (*i.e.* in an unsupervised setting), it is also possible to design the diagnosis module by using clustering approaches (Diaz Rozo et al. 2017).

Similarly to model-based methods, classical ML approaches faced some limitations induced by growing complexity of industrial system data. As described in (Z. Li 2018; Y. Peng et al. 2010; S. Zhang et al. 2020), classical feature-extraction-based models based on a certain a priori knowledge on input data structure, may no longer be effective to perform a correct fault diagnosis. Indeed, with a growing complexity in studied systems, the manual feature engineering struggles in designing representations encompassing all the expressiveness and complexity of input data. As such, these approaches are less prone to model more abstract inter-dependencies between data signals and to be robust to noise. To answer these challenges, DL models are designed, as they integrate a representation learning part in the layers g_1, \dots, g_L . This part aims to automatically extract the most salient features for a subsidiary task (here the fault diagnosis), with no - or few - a priori knowledge on input data structure required (Bengio, Courville, et al. 2013; LeCun et al. 2015). Thus, numerous articles have shown the superiority of DL models over classical ML ones for fault diagnosis, using as representation learning algorithms either discriminative models (like Convolutional Neural Networks (CNN) (J. Pan et al. 2017; Wen et al. 2017; Xia et al. 2017), deep RNN (Abed 2015; L. Guo et al. 2017), Transformers (B. Wu et al. 2021), etc.) or generative models (like Probabilistic Graphical Models (PGM) (T. Liang et al. 2018; K. Yu et al. 2019), autoencoders (Jia et al. 2015; Shao et al. 2018; J. Sun et al. 2017), GANs (Han Liu et al. 2018; Y. Xie et al. 2018)). However, all these works consider unimodal data (namely sensors measurements), and therefore do not address the multimodal input challenge.

2.1.3 FAULT DIAGNOSIS APPROACHES USING MULTIMODAL DATA

The complexity of industrial systems and of the relative acquired datasets, reaches nowadays a new level, with sensors producing multimodal data. While some previous works tackled the challenge of fault diagnosis from various unimodal data such as thermal images (Choudhary et al. 2018; Janssens et al. 2015; Taheri-Garavand et al. 2015), x-ray data (Reid et al. 2013), photographs (J. Wang et al. 2019; Sen Wang et al. 2018) or textual maintenance reports (Sipos et al. 2014; F. Wang et al. 2016), the application of such models to multimodal data (*i.e.* of heterogeneous natures) is still in its infancy. Most previous works addressing the fault diagnosis task and mentioning "multimodal"

2 Background and Related Work

data actually refer to the different functioning modes of the considered system (such as an air conditioner functioning in eco-mode or in normal mode) (Sipple 2020). For (F. Zhou et al. 2018), the word "multimodal" refers to the different orders of derivatives of the input time series. To the best of our knowledge, only two articles properly consider multimodal data (as of heterogeneous natures) in a perspective of industrial maintenance. (Mian et al. 2022) fuse numerical time series of vibration signals with thermal images in order to improve classification performances in the context of bearing fault diagnosis of rotating machine. They use a classical ML approach, with an Hilbert transform module for feature extraction and a concatenation module for data fusion. Yang et al. (Zhe Yang et al. 2021) design a multimodal architecture to address failure prognostics, a related task. The aim of this challenge is to forecast the Remaining Useful Life (RUL) of a system, that is the duration before the system encounters failure. In that sense, the ultimate task is a regression, but the studied framework can be transferred to the one we consider. Their approach handle three modalities (sensors numerical measurements, images and texts) as three distinct blocks, learning respective unimodal representations using either convolutive layers (images and texts) or linear layers (numerical measurements). These unimodal representations are then concatenated and eventually fused using a regression layer. While these approaches are interesting and are close of our objective, they suffer some important limitations. A first limitation is the fact that they are focused on their specific application, rather than interesting in providing general methods for handling multimodal data in predictive maintenance related tasks. As a consequence their results are difficult to generalize to other systems. For instance, a strong limitation is related to their datasets. While in (Mian et al. 2022), the dataset is not publicly available thus preventing the community to compare one's work to theirs, in (Zhe Yang et al. 2021) the dataset is synthetic, which implies a lack of richness and diversity, especially for the textual modality. Indeed, the numerous appearances of the exact same sentences in different examples make the usually unstructured nature of raw text less prominent and representative in that case. Besides, the considered images are actually only curve plots corresponding to the acquired numerical measurements. Hence, they do not represent actual visual captures of the system, which have a much different local structure and would have brought additional information.

TAKEAWAYS

A large body of works has been proposed regarding ML learning approaches for fault diagnosis with two main strategies: sequentially processing fault detection then fault identification or processing the two tasks simultaneously. However, as in other domains these ML approaches have been limited by the hand-crafted feature engineering part and has open an avenue for DL models, that enable to automatically learn an expressive representation that can more easily and effectively be processed.

While the fault diagnosis in Industry 4.0 is multimodal by nature, only few approaches have taken interest in this challenge yet, handling either private or synthetic data. These observations emphasize an important and critical point for the study of multimodality in the context of industrial system monitoring: the unavailability of real multimodal dataset in the Industry 4.0 community. As for the MPO project, we did not either access multi-

modal datasets, as a main part of the objectives was rather to structure the data acquisition pipeline. As a consequence, in our study, while still motivated by the challenges that come from the predictive maintenance field, we will mostly consider datasets coming from alternative fields. Therefore, we hereby invite industrial actors to provide such public representative data, in order to encourage the development of future works on these high-stakes challenges.

Building upon this clear need for enhanced multimodal analysis in the realm of industrial systems, we delve deeper into the specific methodologies and potential applications of multimodal learning in the subsequent section.

2.2 MULTIMODAL LEARNING

2.2.1 FROM MULTIMODAL PERCEPTION TO MULTIMODAL LEARNING

Human beings perceive the world through a multimodal lens, integrating various sensory inputs to better understand and interact with their environment. Multimodal perception encompasses the following:

- Situating ourselves in space and navigating using sight to generate images of our surroundings,
- Communicating with one another through speech, thus producing and interpreting sounds,
- Smelling odors,
- Tasting flavors,
- Experiencing different temperatures and textures, and more.

Hence, from a cognitive perspective, the term "multimodal" here refers to the nature of different sensory stimulations that we, human beings, receive when engaging with the environment. The field of multisensory processing, also known as multisensory integration, investigates how distinct parts of the nervous system and brain process and combine these stimuli to form accurate beliefs about the environment. According to (Maragos et al. 2008), this whole process can be divided into three stages:

- **Sensation:** The electrical signal generated by a specific organ in response to a stimulus,
- **Perception:** The more complex process of filtering, aggregating, and organizing sensations,
- **Cognition:** The ultimate comprehension and decision-making component.

Although the boundaries between these stages are often blurred, the term *multimodal perception* is commonly employed to describe sensory-based reasoning about the environment, particularly the reverse path of inferring the world state from various stimuli. The accuracy and robustness of Human multimodal perception are either innate (determining the localization of a speaker

2 Background and Related Work

using sight and sound) or learned over time through repeated exposure to similar situations. To demonstrate the significance of this phenomenon, consider a person strolling alone on a cloudy beach. They can smell the aroma of meat cooking on a barbecue at a nearby restaurant. Suddenly, they hear a rumble of thunder. In this instance, the individual experiences three unimodal stimuli:

- The sight of the desolate, cloudy beach,
- The sound of thunder,
- The smell of barbecue.

These stimuli activate different sensory organs and their associated acquisition systems, namely the visual, auditory, and olfactory systems. Thus, the person's brain and nervous system will associate visual and auditory modalities as both indicate the presence of a storm (the sight of clouds and the sound of thunder), while filtering out irrelevant information, such as the smell of barbecue. Drawing from past experiences or learned information, the person will recognize this multimodal situation as dangerous by combining visual and acoustic complementary modalities (beaches are unsafe during thunderstorms due to the risk of lightning strikes). This example is illustrated in [Figure 2.1](#).

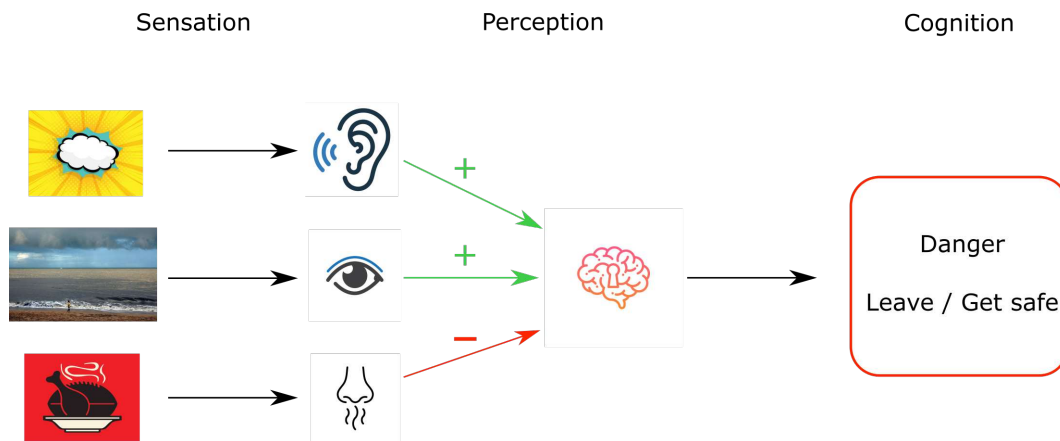


Figure 2.1: Example of multimodal perception of the environment.

As (Lachs 2017) highlight, multisensory integration not only aggregates relevant unimodal stimuli or filters out irrelevant ones but also enhances the strength of neural responses when processing multimodal events compared to unimodal ones. This phenomenon, known as *multimodal enhancement*, means that the measured response to a multimodal event exceeds the sum of measured responses when experiencing the same event unimodally. The enhancement capacity is even greater when the strongest response to unimodal stimuli is weak: this is the Principle of Inverse Effectiveness (Stein et al. 1993). (Lachs 2017) illustrate this principle by considering a task of speech comprehension in a crowded place. If the environment is excessively noisy, the auditory modality alone may not suffice for proper comprehension. Simultaneously, lipreading (the visual modality) can help decipher some words but is generally insufficient to understand an entire sentence. However, the combination of visual and acoustic cues can provide the listener with a general understanding of the conversation. Therefore, although both unimodal responses are relatively weak

for this task, the enhancement resulting from multisensory integration is substantial. Conversely, in a quiet environment, the listener only requires the auditory modality, which will generate a strong response, and the multimodal enhancement will be minimal. As a result, in the beach example, multisensory integration led the person to take the decision to seek safety. Meanwhile, if they had processed only unimodal signals independently, they would not have arrived at this conclusion, as none of the unimodal information (sight of a beach, sound of thunder, or smell of meat) typically suggests the need to urgently find shelter.

This scenario intuitively demonstrates the advantages of processing multisensory signals over unimodal ones: the human brain ingeniously gathers relevant modalities to exploit redundant and/or complementary information, resulting in an improved decision-making capacity. The primary motivation behind multimodal learning is to emulate the role of the human nervous system in its biological ability to aggregate pertinent data from different modalities in such a way that it enhances knowledge for a downstream task.

The parallel with our application is obvious. Indeed, in our industrial system case, the auditory stimuli can be replaced by sensors measurements that are continuously acquired, while the visual stimuli can be replaced by images of a part of the system that are regularly acquired (see Figure 2.2).

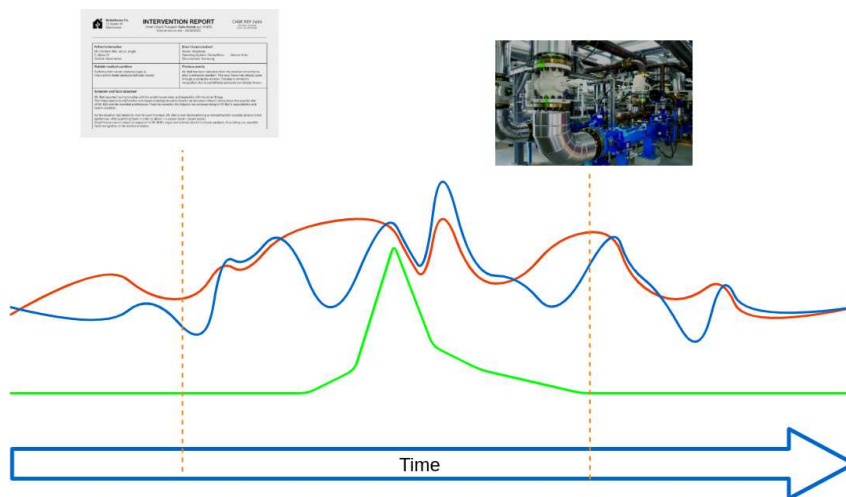


Figure 2.2: Example of multimodal data acquisition in Industry 4.0 setting.

The challenge lies in the fact that while the human nervous system naturally converts stimuli from various modalities into electrical signals through receptors from corresponding sensory organs and integrates them via multisensory neurons, numerical data from different modalities exist in distinct mathematical spaces and possess inconsistent distributions. For example, considered modalities can be either continuous (analog signals like audio recordings) or sparse and discrete (one-hot encoding vectors of raw text, *i.e.*, a symbolic modality). This issue is referred to as the **heterogeneity gap** and constitutes one of the main challenges of multimodal learning. In other words, as depicted in Figure 2.3, vectorial representations of semantical close concepts from

2 Background and Related Work

different modalities are generally also heterogeneous, which lead to the difficulty to measure the content similarity between different modalities.

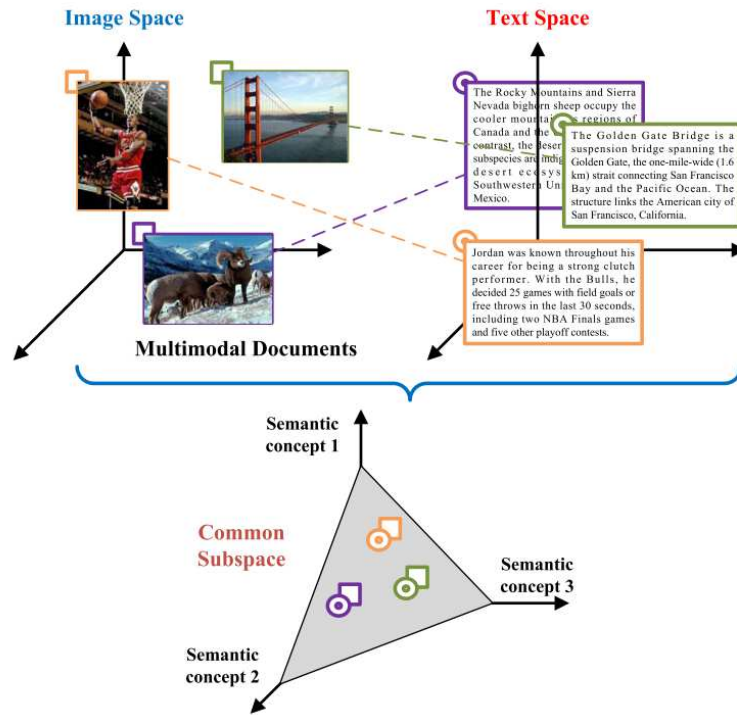


Figure 2.3: Illustration of the heterogeneity gap. Image from (W. Guo et al. 2019).

In particular, (Baltrusaitis et al. 2019) identify five primary challenges within the Multimodal Learning field:

- **Representation**, *i.e.* learning how to represent and summarize multimodal data in a way that exploits the complementarity and redundancy of multiple modalities
- **Translation**, *i.e.* mapping a data point from a source modality space to a corresponding point in a target modality space
- **Alignment**, *i.e.* identifying elements from different modalities related to the same semantic concepts or generative temporal events
- **Fusion**, *i.e.* determining the most effective and robust method of combining relevant information from different unimodal signals to enhance a decision-making procedure for a downstream task
- **Co-learning**, *i.e.* applying knowledge learned in one modality space to enhance inference in another modality with limited resources

From a pragmatic perspective, in the first part of this thesis we focus on **Multimodal Fusion for predictive maintenance downstream tasks such as fault diagnosis**. Successfully addressing this task is intrinsically linked to the challenges of Multimodal Representation and Alignment

challenges. Indeed, tackling these two challenges implicitly helps narrow the heterogeneity gap, making it a valuable preliminary step for Multimodal Fusion. In contrast, we do not explicitly prioritize Multimodal Translation, nor Multimodal Co-learning. These challenges are nonetheless once again ultimately dealing with the heterogeneity gap issue and very linked to the previous ones. For instance, the recent DALL-E 2 model (Ramesh et al. 2022), which addresses the popular challenge of text-to-image generation, relies on the CLIP (Radford, Kim, Hallacy, et al. 2021) pre-trained model, which aims to learn a joint text-image representation using contrastive learning (Bachman et al. 2019; Hjelm et al. 2019).

In the next section we essentially review previous works on Multimodal Fusion approaches, that sometimes thus also address other issues as stated above, and especially Multimodal Representation learning.

2.2.2 MULTIMODAL FUSION: AN OVERVIEW

We begin by formalizing the Multimodal Fusion framework by extending the setting introduced in [Subsection 2.1.1](#), in a similar way as (Y. Huang et al. 2021).

We now consider that a datapoint $\mathbf{x} = (x_1, \dots, x_M)$ is composed of M modalities and thus lives in a multimodal input space $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \times \mathcal{X}_M$, *i.e.* $\forall 1 \leq \alpha \leq M$, $x_\alpha \in \mathcal{X}_\alpha$, with \mathcal{X}_α the definition space of the modality α , with its specific dimension. Tuples $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ are viewed as realizations of the random variables $\mathbf{X} = (X_1, \dots, X_M)$ and Y respectively. We still consider a dataset $\mathcal{D} = (\mathbf{x}^i; y^i)_{i=1}^n$ containing independent and identically distributed data pairs sampled from a distribution $p_{\mathbf{X}, Y}$, unknown but factorizing as: $p_{\mathbf{X}, Y}(\mathbf{x}, y) = p_{Y|\mathbf{X}}(y|f^*(\mathbf{x}))p_{\mathbf{X}}(\mathbf{x})$, with f^* the true mapping from input to output space. We still seek to learn a function $f_\psi : \mathcal{X} \rightarrow \mathcal{Y}$, with $f_\psi = h_\phi \circ g_\theta$ approaching $f^* = h^* \circ g^*$. The main difference with unimodal framework is that functions g_θ and h_ϕ shall now be designed in a way such that they are able to effectively fuse information from input modalities.

MULTIMODAL FUSION IN EARLY ML APPROACHES: EARLY, LATE, HYBRID FUSION

Historically, ML research primarily focused on determining the optimal level for data fusion. Consequently, various approaches were categorized into three distinct groups: early fusion (or feature-level fusion), late fusion (or decision-level fusion), and hybrid fusion methods. [Figure 2.4](#) represents the different fusion strategies: **early** (d), **late** (e) and **hybrid** (f), with the help of Analysis Units (AU), Feature Fusion (FF) and Decision Fusion (DF) units, represented in schemas (a), (b) and (c), respectively. As their names suggests, FF and DF units represent the different fusion modules, while AU units aim to output a decision from an input vector. In the decomposition $f_\psi = h_\phi \circ g_\theta$, g_θ operates at a feature level, transforming raw inputs into exploitable features, that can then be exploited by h_ϕ to produce decisions. In that sense, we can see Feature-Fusion units as part of g_θ , rendering features in a well-suited structure, while Analysis Units and Decision-Fusion units as part of predictor h_ϕ , in charge of producing decisions.

2 Background and Related Work

In **early fusion** (d) (or feature-level fusion), the fusion mechanism operates within g_θ : input unimodal features are combined within the FF unit and sent to an AU to produce a final decision. Conversely, in the **late fusion** scheme (e), the fusion mechanism operates within h_ϕ : unimodal features are passed through unimodal AU to produce respective unimodal decisions, which are then fused within a DF unit to output the final decision. Lastly, the **hybrid fusion** technique (f) involves repeating either early or late fusion strategies on different sets of unimodal features, ultimately fusing intermediate decisions with a final DF unit, followed by a final AU to produce the ultimate decision. In that sense, in this strategy the fusion mechanism operates partly in g_θ and partly in h_ϕ .

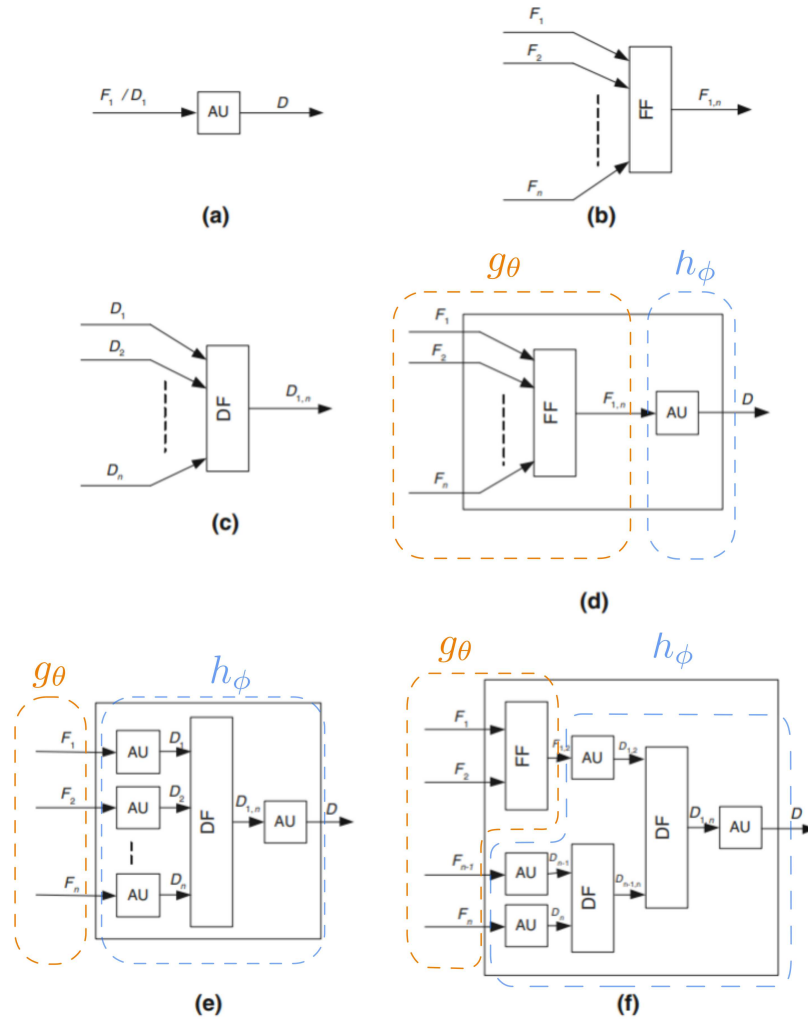


Figure 2.4: Different fusion techniques: early (d), late (e) and hybrid (f). Figure adapted from (Atrey et al. 2010). AU, FF and DF represent Analysis, Feature Fusion and Decision Fusion units, respectively.

Early fusion presents the opportunity to model inherent correlations between different modalities, expressed through low-level features, in order to capture inter-modality dependencies. However, due to the heterogeneity gap, modeling relationships between inconsistent distributions is a nontrivial task and is scarcely achievable when using standard feature fusion strategies, such as simple vector concatenation (Pérez-Rosas et al. 2013; Poria et al. 2016). Conversely, late fusion methods circumvent the heterogeneity gap problem by fusing unimodal decisions only, notably employing rule-based strategies like weighted combinations and majority votes, or by learning the fusion. For instance, (Neti et al. 2000) employ a rule-based strategy for addressing a audio/video Speaker recognition task using as prediction a combined score $D^i = \cos \alpha f_{v,i} + \sin \alpha f_{a,i}$, where $f_{a,i}$ and $f_{v,i}$ represent the scores from audio and video unimodal models, respectively. The weighting coefficient α is an hyperparameter chosen to minimize a cost function on the training set. (Fiérrez-Aguilar et al. 2003) propose to use an SVM to perform Biometric Verification from unimodal face, fingerprint, and signature recognition scores.

Overall, these early works on different fusion levels generally emphasized the advantages of late fusion over early fusion:

- it does not deal with heterogeneity gap between low-level features as it combines unimodal decision scores;
- it does not need to consider different acquisition times between modalities;
- it is usually more robust when one of the modalities is missing.

Nonetheless, late fusion does not exploit correlation at low-level features between modalities, and thus is not ideally suited for modeling multimodal complementarity. Additionally, from a practical standpoint, early fusion also requires to train only one model (for fusion), as opposed to late fusion.

To leverage the strengths of both approaches, some architectures adopt a hybrid strategy. (Z.-z. Lan et al. 2014) for instance address video event detection by first training $n + c + 1$ classifiers in a early-fusion scheme, in which n is the number of extracted features (individual classifiers), c is the number of categories for which features have been combined in an early-fusion fashion, and the last classifier is fed with all input features. After the training of these classifiers, their outputs (score vectors) are combined at test time to produce the final prediction. This double-fusion architecture hence benefits from the correlation modeled by early classifiers and robustness of late classifier to eventually provide better results than a single fusion method.

THE POWER OF REPRESENTATION LEARNING, AND ITS IMPACT ON MULTIMODAL FUSION

When considering deep neural networks, the feature-extraction module g_ϕ becomes a representation learning module composed of stacked layers: $g_\phi = g_1 \circ \dots \circ g_L$. When considering adequate priors to effectively design parameter space Θ , these layers learn deep, hierarchical, distributed representations that can easily be exploited subsequently by a simple predictor h_ϕ . Indeed, (Y. Huang et al. 2021) recently linked the performance of a multimodal algorithm to the quality of the learned latent representation:

DEFINITION

Definition 4. Latent Representation quality

Considering the framework previously defined, the latent representation quality $\eta(g)$ of a learned mapping $g \in \mathcal{G}$ is defined as:

$$\eta(g) = \inf_{h \in \mathcal{H}} [R(h \circ g) - R(h^* \circ g^*)] \quad (2.6)$$

with g^* and h^* the true mappings from input to latent space and from latent space to output space, respectively. Here $\inf_{h \in \mathcal{H}} R(h \circ g)$ is the best achievable population risk with the fixed latent representation g . Thus, to a certain extent, $\eta(g)$ measures the loss induced by the distance between g and g^* .

Using this definition of the representation quality and extending it to a multimodal framework as defined in the beginning of this Subsection, (Y. Huang et al. 2021) theoretically showed that the difference of population risks of models $\hat{f}_{\mathcal{N}} = \hat{h}_{\mathcal{N}} \circ \hat{g}_{\mathcal{N}}$ and $\hat{f}_{\mathcal{M}} = \hat{h}_{\mathcal{M}} \circ \hat{g}_{\mathcal{M}}$ learned on two different modalities subsets \mathcal{N} and \mathcal{M} was bounded by the difference of the corresponding latent representation qualities on these subsets. This directly suggests that an adequate proxy to ensure better performances on a multimodal learning task is to build a latent representation closer to the true mapping, as long as the sample size is sufficient. Besides, they also show that considering more modalities, with a sufficient sample size, implies a better latent representation quality, hence better learning performances. The intuition, depicted on Figure 2.5, is that for two subsets of modalities \mathcal{M} and \mathcal{N} such that $\mathcal{N} \subset \mathcal{M}$, the representation learning module $\hat{g}_{\mathcal{M}}$, minimizing empirical risk on \mathcal{M} has a more sufficient space to explore than $\hat{g}_{\mathcal{N}}$.

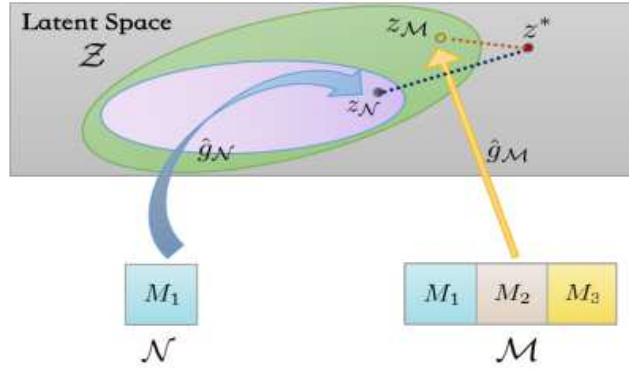


Figure 2.5: Different multimodal representations mappings $\hat{g}_{\mathcal{N}}$ and $\hat{g}_{\mathcal{M}}$ for relative modalities subsets $\mathcal{N} \subset \mathcal{M}$. These mappings produce respective images $z_{\mathcal{N}}$ and $z_{\mathcal{M}}$, for which the latter is closer to z^* , the image corresponding to the true mapping g^* . Figure from (Y. Huang et al. 2021).

As in other Deep Learning fields, focus is therefore on designing the most expressive and generalizable representation, thus on finding the best architecture for function class \mathcal{G} , while the classifiers considered when defining \mathcal{H} are often common architectures such as Multi-Layer Perceptrons. In that sense, the boundary between Multimodal Fusion and Multimodal Representation

Learning has become fuzzy. We thus focus on the following on multimodal representation learning.

MULTIMODAL REPRESENTATION LEARNING

Multimodal representation learning strategies are mainly divided into Joint Representation Learning and Coordinated Representation Learning. These two frameworks are illustrated in [Figure 2.6](#). The aim of the former is to embed unimodal representations together into a shared multimodal representation. Differently, Coordinated Representation Learning approaches learn distinct unimodal representations that are coordinated, using constraints during training, such as similarity maximization for close concepts. Contrastive approaches ([Bachman et al. 2019](#); [Hjelm et al. 2019](#); [Radford, Kim, Hallacy, et al. 2021](#)) are examples of strategies learning coordinated representations.

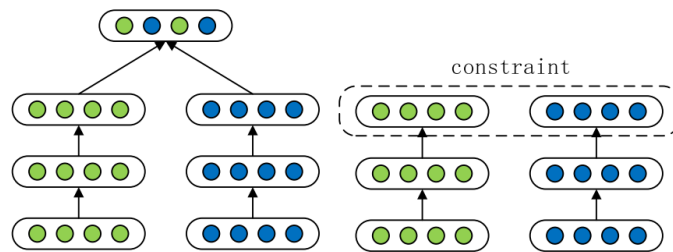


Figure 2.6: (left) Joint representation learning, (right) Coordinated representation learning. Figure from ([W. Guo et al. 2019](#)).

2.2.3 DEEP NEURONAL ARCHITECTURES FOR MULTIMODAL REPRESENTATION LEARNING

In this section, we present the different neural architectures that have been proposed in the literature to implement multimodal representation learning and the two strategies described in the previous section. These architectures can be divided into model-agnostic or specific architectures.

The most straightforward strategy for addressing deep multimodal representation learning is Model-agnostic approaches, like early additive or multiplicative fusion ([Bruni et al. 2012](#); [Zadeh, Minghai Chen, et al. 2017](#)). These methods design a shared subspace for joint representation learning using a shared hidden layer. Here, encoded data from various modalities are either concatenated, added, or multiplied before activation, thus enabling the fusion of semantics. [Figure 2.7](#) illustrates such concatenation and multiplication from different modalities.

In contrast to these architectures, typical models used for Deep Multimodal Representation Learning notably include Probabilistic Graphical Models (PGM), Autoencoders or Attention-based models. We recall the principles of these different models in the following.

2 Background and Related Work

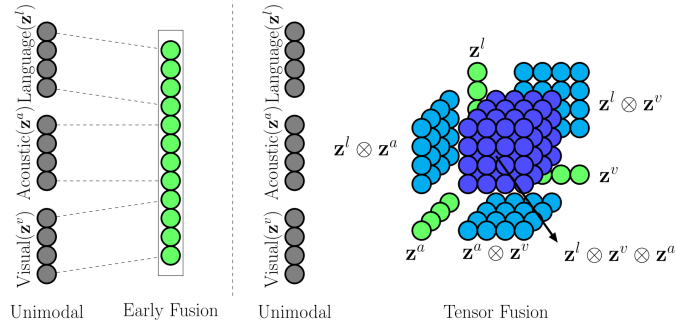


Figure 2.7: Early fusion techniques in neural networks: concatenation (left) and tensor multiplication (right). Figure from (Zadeh, Minghai Chen, et al. 2017).

PROBABILISTIC GRAPHICAL MODELS

PGM, like Multimodal Deep Boltzmann machines (Srivastava and R. R. Salakhutdinov 2012), or Multimodal Deep Belief Networks (Srivastava and R. Salakhutdinov 2012) are generative models, and thus learn a joint distribution over different modalities mainly using Maximum Likelihood Learning. Their main characteristic lies in their ability to handle missing modalities by generating them, permitting unsupervised training. Nevertheless, these models suffer from substantial drawbacks: the intractability of Maximum Likelihood Learning and the prohibitively expensive approximation inference algorithm. These limitations challenge the feasibility of employing these methods.

AUTOENCODERS

Similarly, autoencoders provide another unsupervised learning approach as they aim to encode input data in a condensed representation, while ensuring the preservation of essential semantic features through input reconstruction. Multimodal adaptations have been proposed (Ngiam et al. 2011; Silberer et al. 2014), with hidden representation layer taking as input both modalities, subsequently attempting to reconstruct them (see Figure 2.8). However, training solely depends on the reconstruction loss, which results in a task-agnostic representation. Constraints (such as the corruption of the input) or subsequent supervised objective need to be set up to add desired properties (like robustness) to the multimodal representation (Silberer et al. 2014).

ATTENTION-BASED MODELS AND TRANSFORMERS

Attention-based models are now well-known models that possess the ability to focus on a specific part of the input, depending on the context. They are widely used since apart from increasing performance, they bring some interpretability to decisions, evaluating the importance of features. Regarding multimodality, they have some interesting properties. Indeed, on the intra-modality level, they enable the selection of the most prominent features from each modality, guided by contexts from other modalities, like in Visual Question Answering (Zichao Yang et al. 2016). At the inter-modality level, they balance the contribution from different modalities, assigning more

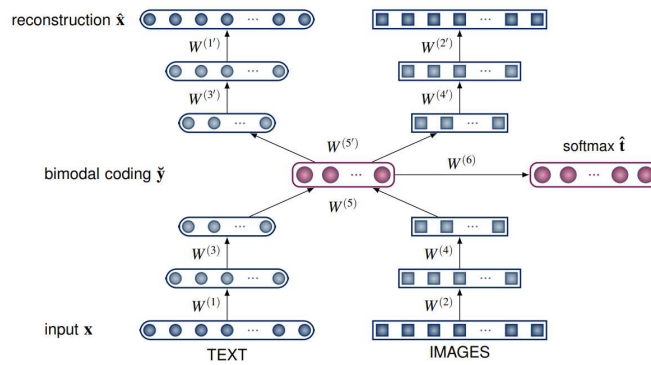


Figure 2.8: Multimodal autoencoder. The model aims to encode the bi-modal input into a joint compressed representation before reconstructing the two modalities’ inputs from this representation. Figure from (Silberer et al. 2014).

weight to the ones with greater importance (Long et al. 2018). This two-level impact is illustrated in Figure 2.9.

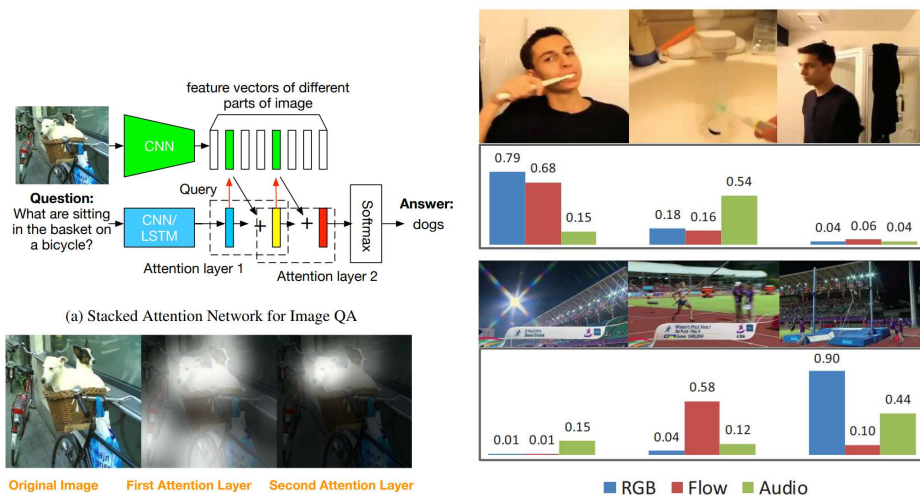


Figure 2.9: Intra-modality (left) and inter-modality (right) impacts of attention mechanism. On the left-side picture, the attention mechanism forces the model to focus on specific parts of the image input, conditioned by the textual input to perform VQA. On the right-side picture, the attention mechanism balances the weight of each input modality depending on their relevances for identifying a scene. Figures from (Zichao Yang et al. 2016) (left) and (Long et al. 2018) (right).

The advent of the Transformer architecture (Vaswani et al. 2017) marked a significant shift in this domain. This type of encoder-decoder model has gained a massive interest, with numerous derivatives and impressive performances on applicative tasks across different modalities, *e.g.* in NLP (Devlin et al. 2019) or in computer vision (Dosovitskiy et al. 2021). Their building block, Multi-head Self-Attention mechanism, aims to learn a contextual representation Z of an input sequence X .

2 Background and Related Work

Each attention head first maps the input X to a set of key K , value V and query Q matrices. The queries and keys are combined through a matrix product to produce attention weights (through a softmax function), representing the contextual interdependencies of the input elements. The values elements are finally multiplied by these weights to produce the output representation Z . Formally:

$$Z = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.7)$$

$$= \text{softmax}\left(\frac{XW_QW_K^TX^T}{\sqrt{d_k}}\right)XW_V \quad (2.8)$$

Here, d_k denotes the dimension of queries and keys, while W_Q, W_K, W_V represent weight matrices. Transformer encoder blocks are commonly used to learn contextual representations that can afterward be used for subsidiary tasks.

Remark (Preprocessing). It is essential to note that the variable X in Equation 2.8 is not typically raw input data, but rather the initial embedding of tokenization of X :

$$X = E(T(X)) \quad (2.9)$$

where E is an embedding block and T a tokenizer. The considered input data hence does not need to be initially a sequence (as for textual data) to be processed by the Transformer: this sequential formatting is the task of a designed tokenizer. For instance, Visual Transformer (ViT) (Dosovitskiy et al. 2021) uses small patches as tokens to represent an image (see Figure 2.10).

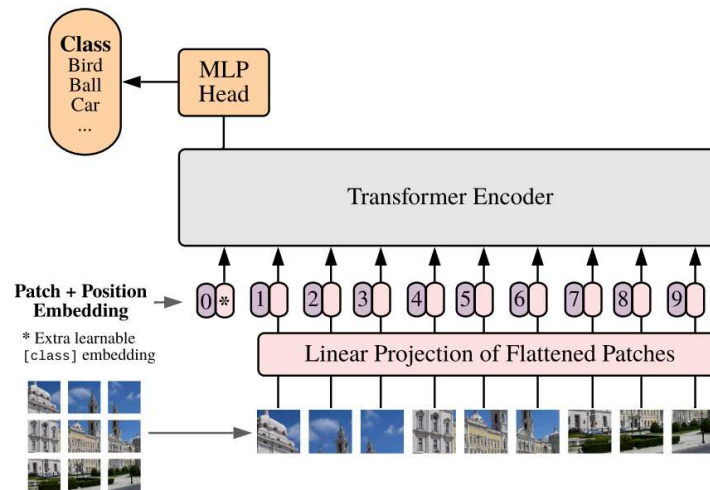


Figure 2.10: Visual Transformer architecture. Input images are tokenized in small patches that are encoded with positional embeddings. Figure from (Dosovitskiy et al. 2021).

The intent of the embedding block is to map the sequence into an initial expressive latent space, typically achieved through a linear projection (Dosovitskiy et al. 2021; Vaswani et al. 2017). It is commonplace to fuse several types of embeddings at the token level, thereby injecting pertinent information into the model. For example, the Self-Attention mechanism, being invariant to the positioning of tokens within the sequence, can utilize absolute positional embeddings added to the initial token embeddings as an inductive bias for positional relevance information. Numerous works have sought to ascertain the most effective and efficient methodologies for computing these positional embeddings. The original transformer proposed in (Vaswani et al. 2017) employs either learnable vectors or sinusoidal functions to offer absolute embeddings, with little noticeable variation in performance outcomes. The approach of absolute positional encoding through a learnable vector has been adopted widely in subsequent works (Devlin et al. 2019; Z. Lan et al. 2020; Radford, Narasimhan, et al. 2018; Radford, J. Wu, et al. 2019). (Shaw et al. 2018) proposed to encode relative positions, predicated on the intuition that the distance between two tokens holds more significance than their absolute positions. In that case, learned relative positional embeddings based on the token distances are added to keys and values matrices during attention calculation (Equation 2.7). This methodology has been replicated in subsequent studies (He et al. 2021; Z. Huang et al. 2020; Ke et al. 2021; Raffel, Shazeer, et al. 2020).

For the multimodal framework, segment encoding may be incorporated at the token level in a similar manner to positional encoding, thereby informing the model of the token modality (G. Li et al. 2020; L. H. Li et al. 2019). This embedding fashion is illustrated in Figure 2.11.

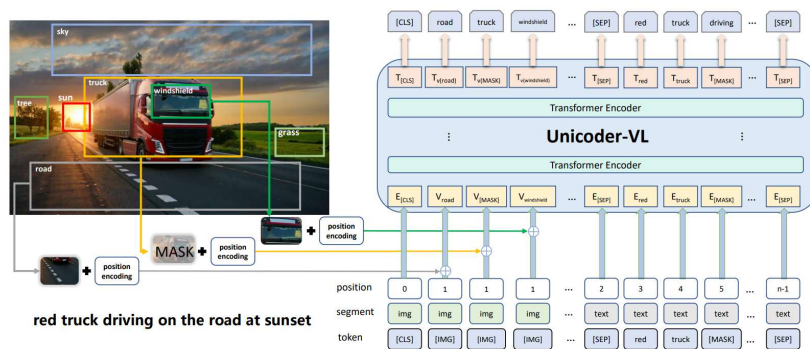


Figure 2.11: Multimodal segment embeddings. Besides the positional embeddings, each image patch and textual input is encoded with a special segment embedding indicating its input modality. Figure from (G. Li et al. 2020).

Transformers have since then been used to handle multimodal data in many variants of the Vanilla architecture, as exemplified in Figure 2.12. Much like traditional approaches discussed in Subsubsection 2.2.2, architectural decisions are primarily guided by the fusion timing—early (a, b, d), late (c), or throughout the model (e and f)—and the aspiration to generate either joint or coordinated representations. For instance, VideoBERT (C. Sun, Myers, et al. 2019) applies early concatenation of visual-text sequences to learn high-level joint features (b) in a self-supervised

2 Background and Related Work

fashion. (Tsai et al. 2019) employ cross-attention-to-concatenation (f) on three modalities (audio, visual and textual) to model cross-modal interactions in all modality pairs. In contrast, (R. Li et al. 2021) proceed to a later fusion (c) by initially encoding intermediate modality-specific representations for music pieces and 120-frames seed motion sequences, which are subsequently fused in a cross-modal transformer to learn the correspondence between both modalities and generate the future motion sequences. Advantages and limitations of studied approaches are summarized in Table 2.1.

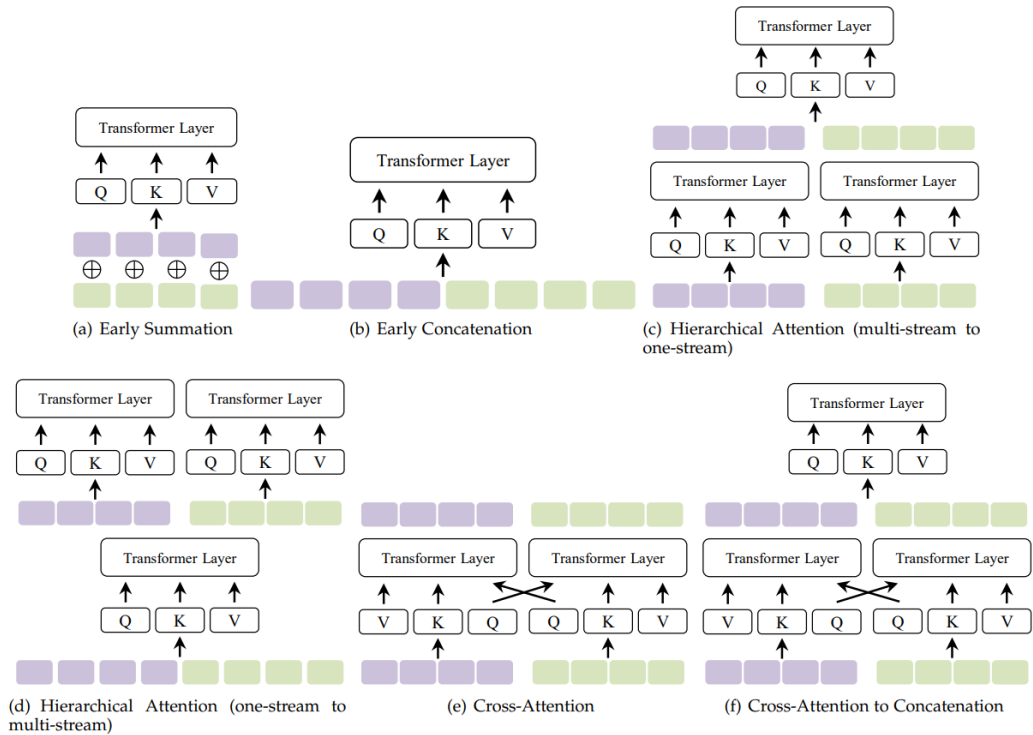


Figure 2.12: Different variants of Transformer for processing multimodal data. Colors represent modalities. Figure from (P. Xu et al. 2022).

PRE-TRAINING AND SELF-SUPERVISED LEARNING

Apart from initial encodings at the token level and architecture design (Figure 2.12), effective modeling of cross-modal interactions can be achieved through the pre-training objective. Originally developed for NLP and related tasks, transformers have revolutionized the field by enabling effective modeling of contextual dependencies within sequences. The trend of pre-training transformer-based models on vast quantities of unlabeled data using self-supervised objectives to learn general language knowledge has led to the development of large foundation models (see Section 5.7). These models demonstrate impressive performance across diverse tasks and exhibit robust generalization ability.

Table 2.1: Comparison of different approaches for multimodal representation learning.

Approach	Advantages	Drawbacks
Model-Agnostic	Simple and flexible.	May miss complex dynamics.
PGM	Handles missing modalities. Unsupervised training.	Computationally expensive.
Autoencoders	Unsupervised. Captures essential semantics.	Task-agnostic representation.
Attention-based Models	Interpretable. Balance modalities.	Performance varies with task complexity.
Transformers	High performance. Sometimes interpretable through attention maps. Offers many architecture variants.	Requires careful tokenizer and embedding layer designs. Requires consequent computational power.

Self-supervised learning is a learning setting in which the objective is defined by the data themselves. Aside from having the advantage not to need labeled data, these methods force the models to learn representations that leverage the structure of the data, as it constrains the learning objective. For instance, it is common to try to predict a hidden part of the input:

- The seminal work (Devlin et al. 2019) introduced the Masked Language Modelling loss (MLM), that needs the model to predict a masked token in a sentence.
- That approach has been adapted to other modalities. (Dosovitskiy et al. 2021) hence explored a Masked Patched Prediction pre-training objective, consisting in predicting the mean color of corrupted image patches, while (Junkun Chen et al. 2020) similarly mask some frames of speech inputs, and tries to reconstruct the initial sequence from the corrupted data.

These modality-specific learning objectives have been quite straightforwardly extended to the multimodal framework, especially for unlabeled and unaligned datasets. Although the corresponding losses remain unimodal, the associated learning process leverages the cross-modal dependencies between multimodal inputs to gain information from the other modalities. It is also frequent to consider a general loss composed of the sum of modality-specific losses. For instance, the VideoBERT (C. Sun, Myers, et al. 2019) model’s training consists in encoding textual and visual sequences and to predict masked tokens (either textual or visual) using modality-specific input sequences and the MLM objective. Besides the text-only and video-only objectives, a third cross-modal objective is tackled: after encoding a bimodal sequence formed by the concatenation of textual and video sequences (see Figure 2.13), the model shall predict if the two sequences are temporally aligned using as input the *CLS* token. The global pre-training objective is composed of the sum of the three objectives.

In the case in which we possess aligned modalities however, we can use this alignment as a self-supervised objective. A popular framework of SSL that is suited for aligned modalities is contrastive learning, which encourages representations of input data and their augmented views to be

2 Background and Related Work

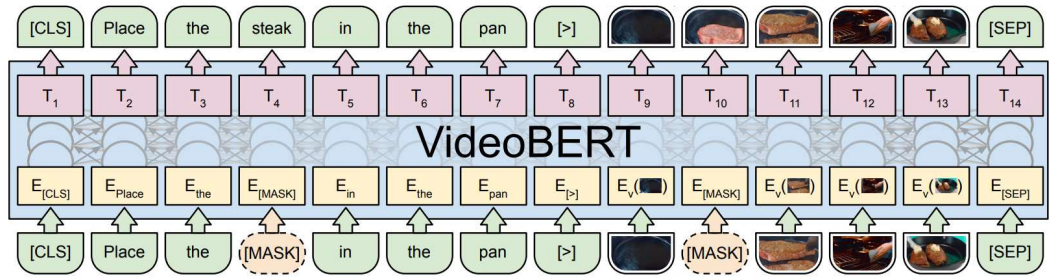


Figure 2.13: VideoBERT architecture and pre-training. Figure from (C. Sun, Myers, et al. 2019).

close in latent space, while pushing apart representations of different inputs. This way, the learned representations should be invariant to small perturbations, while encoding salient features. Given a context vector c , the popular InfoNCE loss (Oord et al. 2018) uses categorical cross-entropy to identify the positive sample \mathbf{x} drawn from the distribution $p(\mathbf{x}|c)$ from unrelated noises x' . This loss optimizes the negative log probability of classifying the positive sample correctly:

$$\mathcal{L}_{\text{InfoNCE}} = \mathbb{E} \left[\log \frac{f(\mathbf{x}, c)}{\sum_{x' \in X} f(x', c)} \right] \quad (2.10)$$

where $f(\mathbf{x}, c)$ estimates the density ratio $\frac{p(\mathbf{x}|c)}{p(\mathbf{x})}$.

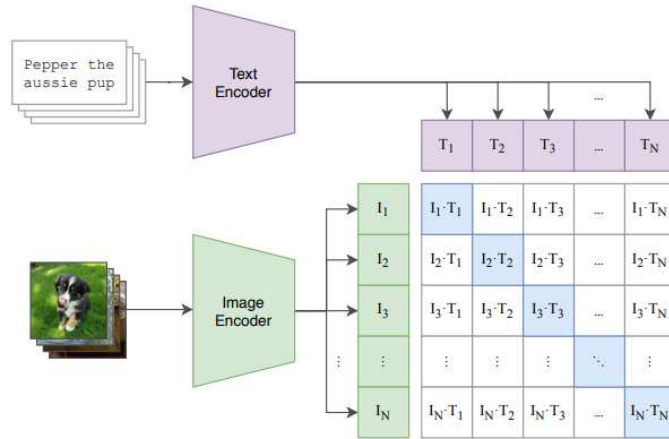


Figure 2.14: CLIP Architecture. Figure from (Radford, Kim, Hallacy, et al. 2021).

This loss inspired Radford et al. to design CLIP (Radford, Kim, Hallacy, et al. 2021) architecture, illustrated in Figure 2.14. The model consists of mapping related image and text embeddings in a common subspace by simple linear projections. For a batch of N image-text pairs, all N^2 cosine similarities are then computed. The first term of the loss then fixes each image as the context c and minimizes the corresponding $\mathcal{L}_{\text{InfoNCE}}$. The second term of the loss replicates the same procedure by fixing each text as context.

Despite the simplicity of the architecture, this self-supervised setting enabled to pre-train CLIP on 400 million unlabeled image-text pairs. This results in impressive results, the model achieving for instance the same performances in Zero-Shot setting on ImageNet as a fully supervised ResNet 50. Besides CLIP, many works have explored the contrastive framework to pre-train multimodal architectures (Alayrac, Rezacens, et al. 2020; Bachman et al. 2019; J. S. Chung et al. 2016; Hjelm et al. 2019; Miech et al. 2020; C. Sun, Baradel, et al. 2019).

In summary, the popularized paradigm consisting in pre-training transformers on self-supervised objectives has also been explored intensively in the multimodal paradigm. Taking inspiration from the BERT introduced Masking Language Modeling loss, these architectures' pre-training objectives mainly consist in reconstructing masked tokens from inputs, in cross-modal or modality-conditional fashions (J. Lu et al. 2019; C. Sun, Myers, et al. 2019). Moreover, in these approaches the alignment between different modalities also constitutes an interesting supervision. This alignment is even the main self-supervised objective of contrastive methods.

CONCLUSION

In conclusion, there has been extensive work in the realm of machine learning and deep learning for unimodal fault diagnosis. However, the area of multimodal diagnosis has been less thoroughly explored, reflecting a significant gap in research that is waiting to be addressed. Notably, the field of fusion and multimodal representation learning has witnessed considerable development. These advancements are mostly driven by challenges in text/image tasks, with recent trends highlighting the supremacy of transformer-based architectures. However, the general framework for these multimodal transformer architectures can still be improved and adapted for new scenarios, such as multimodal fault diagnosis. Despite these advancements, the distinctive properties of the corresponding data in industrial applications, including the presence of unaligned and long temporal streams, add a layer of complexity. Furthermore, the multimodal nature of these streams has not been considered enough within the realm of fault diagnosis, hence there is a rich opportunity for exploration and development. In addressing this complex problem, it is vital to note that the heterogeneity gap remains a significant challenge. Given this, there is a pressing need to define a new setting for this kind of data, an undertaking we will focus on in the subsequent chapter. In parallel, we will also introduce a new architecture, named StreaMulT, specifically designed to confront these emerging challenges.

3 STREAMMUL T: A STREAMING MULTIMODAL TRANSFORMER FOR HETEROGENEOUS AND ARBITRARILY LONG SEQUENTIAL DATA

CHAPTER'S SUMMARY

In this chapter, we tackle the new challenges posed by the rising complexity of Industry 4.0 systems, and their relation to fault detection and diagnosis tasks. We explore these challenges in a realistic environment that involves multi-source data streams from various modalities, including time series sensor measurements, machine images, and textual maintenance reports. These heterogeneous multimodal streams also differ in their acquisition frequency, may embed temporally unaligned information and can be arbitrarily long, depending on the considered system and task. Building on the previous chapter, wherein we examined principal approaches to multimodal fusion, we broaden our scope to this setting. We consider arbitrarily long multimodal streams in conjunction with related tasks, such as prediction across time. To tackle this challenge, we propose StreaMulT, a Streaming Multimodal Transformer. StreaMulT employs cross-modal attention and a memory bank to process arbitrarily long input sequences during training and operates in a streaming mode during inference. Our findings indicate that StreaMulT elevates the state-of-the-art metrics on the CMU-MOSEI dataset for the Multimodal Sentiment Analysis task. Remarkably, it outperforms other multimodal models in managing considerably longer inputs. Finally, the experiments conducted underscore the criticality of the textual embedding layer, leading us to question recent advancements in Multimodal Sentiment Analysis benchmarks. This chapter, therefore, offers a comprehensive exploration of the challenges and potential solutions associated with the application of multimodal learning in streaming settings.

3.1 INTRODUCTION

As explained in the previous chapters, the availability of massive amounts of data, coupled with recent ML breakthroughs offers great potential in numerous domains and particularly for the industry. More specifically, in Industry 4.0 era, one major challenge is to exploit all information sources related to a system in order to perform data-driven diagnosis for corrective and predictive maintenances. To represent a typical example of studied industrial system, we consider an aircraft engine that is continuously running and from which we acquire feedback data of different modalities (numerical time series, raw text, images, sound, etc.) over time. For example, these modalities

can correspond to sensors measurements, textual maintenance reports, system photographs, system audio recordings, and so on. From these data, our goal is, depending on the task, either to detect if the system is in a faulty mode (fault monitoring) or to determine which fault is occurring (fault diagnosis). This setting is illustrated in [Figure 3.1](#).

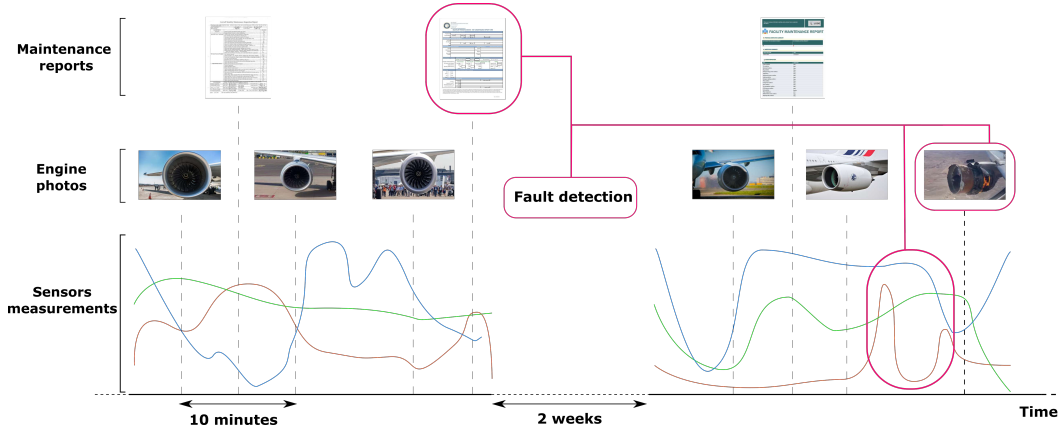


Figure 3.1: Typical example of fault diagnosis task in the context of Industry 4.0: case of an aircraft engine. Each modality present fault symptoms through acquired data (red circles), that, if fused together, can enable the fault detection (and identification).

This paradigm comes with different challenges, from which we decide to consider the following:

- **Heterogeneous modalities:** The different sources of acquired data can come in different modalities, hence resulting in a heterogeneity gap issue when combining them. It is therefore relevant to develop methods aiming to narrow this gap, to exploit redundant and complementary information across modalities (see [Section 2.2](#)).
- **Heterogeneous acquisition frequencies:** Despite their heterogeneous nature (and therefore structures), different sources of data generally possess their own acquisition frequencies. For instance on illustrated [Figure 3.1](#), numerical sensors measurements of physical quantities such as temperature, pressure, vibration or current signals, can be acquired at a regular high frequency, in the order of few seconds. On the other hands, system photographs, are also obtained at a regular acquisition frequency but with a greater period (say hours). Eventually, textual maintenance reports are acquired only every time following a maintenance, that is at a low and sporadic frequency.
- **Unaligned modalities:** The different acquired streams are generally not aligned on the temporal axis. Indeed, as illustrated on [Figure 3.1](#), a fault occurring at a specific time step may be highly correlated with very recent sensors measurements or system photographs, while the related relevant information for the textual modalities would be contained in a much former report.
- **Arbitrarily long input sequences:** As introduced in the previous point, depending on the fault, the relevant part of the input data to perform fault monitoring or diagnosis can

be located far back in time for one modality, relatively to another one. Thus we consider that the acquired streams can be arbitrarily long and we do not bound them.

- **Streaming mode:** Depending on the level of criticality of the system, it can be imperative to perform the fault monitoring/diagnosis task with a relatively short response time. Plus, we can also imagine industrial systems that have to run uninterruptedly. In both cases it is either not desirable or not feasible to wait for the system to stop before executing the diagnosis module. Consequently, we consider as mandatory the ability to the designed approach to work in a streaming fashion, that is processing the input streams as they are acquired over time.

These different challenges have been tackled in the literature but separately to the best of our knowledge. If a large avenue of research exists in multimodal learning, and from now on recently mainly based on the Transformer architecture (see [Subsection 2.2.3](#)), the quadratic dependency of space and time complexities of the architecture with the input length limits its use for arbitrarily long inputs or streaming inference. By the mean of StreaMulT, we thus propose to tackle these five problems jointly.

CHAPTER'S CONTRIBUTIONS

In this chapter, our contributions are threefold:

- Motivated by this industrial application and its key challenges, we formally define a new applicative paradigm, in which one aims to solve a prediction task across time, from heterogeneous (by nature and acquisition frequency) multimodal sequential data and in a streaming fashion, hence handling arbitrarily long input data at both training and inference time.
- We then introduce StreaMulT, a Streaming Multimodal Transformer architecture based on cross-modal attention and conveying a memory bank to tackle these issues and deal with unaligned input streams.
- Due to the lack of a either public or private (within the MPO project) datasets adapted to our task, we propose to evaluate our model with the CMU-MOSEI dataset on a multimodal sentiment analysis task, in order to compare StreaMulT performances with previous approaches. It includes both multimodal and unaligned streams. We show that our model can deal with arbitrarily long sequences without suffering from performance loss. When improving the textual pre-trained embedding, we even improve the state-of-the-art metrics on this dataset.

In [Section 3.2](#) we formalize the multimodal setting with arbitrary long sequential data and we define the positioning we decided to adopt to tackle the task of industrial diagnosis in this setting. We then review the connected works that brought us to develop the architecture of StreaMulT in [Section 3.3](#). We introduce the StreaMulT model in [Section 3.4](#), and we finally conduct experiments on CMU-MOSEI dataset and ablation study in [Section 3.5](#).

3.2 MULTIMODAL LEARNING WITH HETEROGENEOUS AND ARBITRARILY LONG SEQUENTIAL STREAMS

3.2.1 PROBLEM FORMALIZATION

In order to avoid confusion between modality, sample, feature dimension and time indices, we use greek letters to index the modalities.

Let $M \in \mathbb{N}$ be the number of considered modalities. For each modality α , with $\alpha \in \llbracket 1, M \rrbracket$, we consider the corresponding time series X_α , indexed by time according to its own acquisition times and lying in its own definition space:

$$X_\alpha := (X_\alpha(t))_{t \in \mathcal{T}_\alpha} \text{ and } \forall t \in \mathcal{T}_\alpha, X_\alpha(t) \in \mathcal{X}_\alpha$$

where \mathcal{T}_α and \mathcal{X}_α are respectively the countable (possibly not finite) set containing acquisition times of modality α and its associated definition space. We can for instance suppose real components without loss of generality, i.e. $\mathcal{X}_\alpha = \mathbb{R}^{d_\alpha}$ with d_α the feature dimension.

Let \mathcal{X} be the set defined as:

$$\mathcal{X} := \{X(t), t \in \mathbb{R}\} \text{ where } X(t) := (X_1(s_1))_{\substack{s_1 \leq t \\ s_1 \in \mathcal{T}_1}} \times \dots \times (X_M(s_M))_{\substack{s_M \leq t \\ s_M \in \mathcal{T}_M}} \quad (3.1)$$

The elements of \mathcal{X} are basically M-tuples whose the α^{th} term is composed of the elements of the sub-sequence X_α up to a specific time step t that is common to all modalities.

A label space \mathcal{Y} and the corresponding set of ground truth time steps \mathcal{T}_y are defined depending on the considered specific task and on the input data. From these elements, one can construct a dataset $\mathcal{D} = (\mathbf{x}^i, y^i)_{i=1, \dots, n}$ composed of realizations of previously introduced random variables:

$$\forall i \in \llbracket 1, n \rrbracket \begin{cases} t_i := \mathcal{T}_y[i], \text{ where } \mathcal{T}_y[i] \text{ denotes the } i^{\text{th}} \text{ element of } \mathcal{T}_y \\ \mathbf{x}^i := X(t_i) \\ y^i := y(t_i) \end{cases}$$

The global objective of this setting is thus to perform a supervised prediction task (classification or regression) on this dataset. Hence, given $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ a loss function and \mathcal{F} a function class, we aim to find $f \in \mathcal{F}$ minimizing the associated empirical risk (see Equation 3):

$$f^* = \arg \min_{f \in \mathcal{F}} \hat{R}_n(f) = \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \mathcal{L}(f(\mathbf{x}^i), y^i) \quad (3.2)$$

Example 1. Ideal fault diagnosis

In an ideal setting of fault diagnosis, one would like to be able to give a prediction of the state of the system in real time, that is, every time one acquires a new data point, from any modality. Hence in that case,

$$\mathcal{Y} = \begin{cases} \{0, 1\} \text{ for fault monitoring} \\ \llbracket 1, C \rrbracket \text{ for fault diagnosis} \end{cases}, \quad \mathcal{T}_y = \bigcup_{1 \leq \alpha \leq M} \mathcal{T}_\alpha$$

The elements of the dataset $\mathcal{D} = (\mathbf{x}^i, y^i)$ are thus the M-tuples composed of subsequences of all modalities up to a time step t_i , associated with $y(t_i)$, the state of the system at time t_i , where t_i takes all possible values of data acquisition times among all modalities.

Example 2. Fault diagnosis with resources or user constraint

The ideal setting of fault diagnosis described above is nonetheless in general not realistic for inference, as the acquisition frequencies and the available resources can make the diagnosis module impossible to run in real time. In such a case, or if the user wants to put a specific constraint on the time to output a prediction, \mathcal{T}_y can be constructed iteratively:

Algorithm 1 Creation of custom \mathcal{T}_y with constraints.

```

 $t_i \leftarrow 0$ 
while data_acquisition do
     $t_i \leftarrow t_i + 1$ 
    if condition_on_ $t_i$  then
         $\mathcal{T}_y \leftarrow \mathcal{T}_y \cup \{t_i\}$ 
    end if
end while

```

In **Algorithm 1**, the condition "data_acquisition" refers to the state of the data acquisition process and can be seen as the upper bound of the value of t_i . Namely, this condition is set to *True* while t_i has not reached the last time step of acquired time series for the training set, and can indefinitely be set to *True* for inference in streaming.

The "condition_on_ t_i " includes all different constraints defined by the task or the user. For instance, in the current case of industrial diagnosis, with a resource constraint imposing a minimum of 10 time steps between two predictions:

$$\text{"condition_on_}t_i\text{"} = \text{"}t_i \in \left\{ \bigcup_{1 \leq \alpha \leq M} \mathcal{T}_\alpha \right\} \text{"} \wedge (\text{"}\mathcal{T}_y = \emptyset\text{"} \vee \text{"}t_i - \mathcal{T}_y[-1] \geq 10\text{"})$$

in which \wedge represents the logical AND and \vee represents the logical OR.

Remark. When the inference is realized in a streaming fashion, the construction of \mathcal{T}_y and the execution of the diagnosis module are simultaneous.

Example 3. Multimodal Sentiment Analysis

If we consider now a sentiment analysis task in which the objective is to assign a score from -3 to 3 to each sentence contained in a long sequence (keeping past sentences in input), then for a sequence of s multimodal sentences, the associated ground truth time steps are the last acquisition time steps of each sentence:

$$\mathcal{Y} = [-3, 3], \quad \mathcal{T}_y = \left\{ \max \left(\bigcup_{\alpha=1}^M \mathcal{T}_\alpha^j \right), 1 \leq j \leq s \right\}$$

where j is the sentence index and \mathcal{T}_α^j are the acquisition time steps of modality α for sentence j .

We now describe the positioning we decided to adopt regarding this global problem.

3.2.2 POSITIONING

To the best of our knowledge, the previous framework has never been introduced as such, hence the related task of Multimodal Fault diagnosis (addressing the five challenges) has never been dealt with. Therefore, there is no existing and available public dataset to evaluate models on this task. There exist many multimodal datasets, for other different applications fields, like Visual Question Answering (Microsoft COCO (Lin et al. 2014)), Affective Computing (CMU-MOSEI (Zadeh, P. P. Liang, et al. 2018)), Healthcare (MIMIC-iii (Johnson et al. 2016)) and so on, but none of these datasets possess the five desired properties of our challenge. Recently, (P. P. Liang, Y. Lyu, et al. 2021) proposed a unified benchmark spanning 15 datasets, 10 modalities, 20 prediction tasks, and 6 research areas. Among these datasets, the ones related to the affective computing field appeared to us as the closest to our problem, as they present a sequential setting, with unaligned streams from different modalities and with different acquisition frequencies. We thus chose to conduct experiments on the CMU-MOSEI dataset, addressing a Multimodal Sentiment Analysis task, as introduced in [Example 3](#).

This decision - by lack of dataset considering Multimodal Fault Diagnosis task - is compatible with the lens we see our challenge through: by seeing the tasks of fault monitoring and diagnosis as a unique classification in $C + 1$ classes as in [Subsection 2.1.1](#), and by writing the function f in [Equation 3.2](#) as $f = h \circ g$, we now seek \hat{f} that minimizes the related empirical risk as defined in [2.5](#):

$$\hat{f} = \arg \min_{f \in \{h \circ g | g \in \mathcal{G}, h \in \mathcal{H}\}} \hat{R}_n(f) \quad (3.3)$$

Following the discussion pointing out the importance of the quality of a multimodal representation in [Subsubsection 2.2.2](#), and its link to the subsidiary prediction performances, we mainly focused our research work on finding an architecture dealing with data presented in [Section 3.2](#) and maximizing their multimodal representation, in a task-agnostic manner. The only assumption we make on the considered task is that we are in a supervised setting.

3.3 RELATED WORK

While Transformers architectures have been widely used on numerous applicative tasks, we show in [Subsection 3.3.1](#) that their complexity prevents them to cope with long inputs or to run in a streaming fashion as such. We present some variants addressing this limitation in [Subsection 3.3.2](#).

3.3.1 TRANSFORMER ARCHITECTURES AND UNALIGNED MODALITIES

Classical approaches dealing with multimodal sequential data, such as RNN-based architectures, do not tackle the unalignment issue (Zadeh, P. P. Liang, et al. 2018; Zadeh, Poria, et al. 2018), and

hence consider the input multimodal data are temporally aligned. Furthermore, the autoregressive nature of these architectures generally implies to consider same acquisition timesteps along different modalities.

Multimodal Transformer (Tsai et al. 2019) addresses both these issues, taking advantage of its cross-modal transformer modules, that aims to learn a contextual and cross-modal representation of unaligned input sequences as depicted in Figure 3.2. At the heart of this module, cross-modal attention blocks indeed express a target modality α with raw features from a source modality β . Formally, considering our input sequences X_α and X_β from modalities α and β , the cross-modal attention for X_α attending to X_β , denoted $X_{\beta \rightarrow \alpha}$ is computed as:

$$X_{\beta \rightarrow \alpha} := \text{softmax} \left(\frac{Q_\alpha K_\beta^T}{\sqrt{d_k}} \right) V_\beta \quad (3.4)$$

$$= \text{softmax} \left(\frac{X_\alpha W_{Q_\alpha} W_{K_\beta}^T X_\beta^T}{\sqrt{d_k}} \right) X_\beta W_{V_\beta} \quad (3.5)$$

with Q_α the query matrix for modality α , K_β, V_β the key and value matrices for modality β ; $W_{Q_\alpha}, W_{K_\beta}, W_{V_\beta}$ being learned weights, and d_k being the common embedding dimension for query and key matrices.

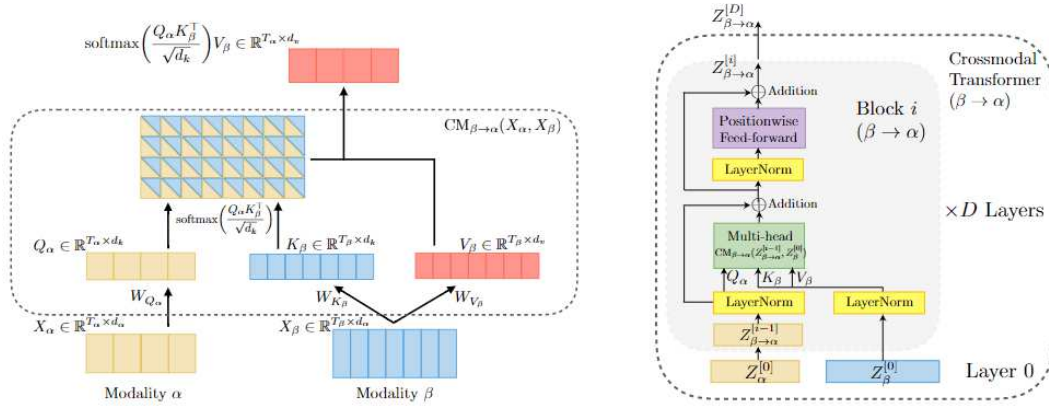


Figure 3.2: Cross-modal attention block between sequences X_α, X_β from distinct modalities (left) and cross-modal transformer module (right). Figures found in (Tsai et al. 2019).

Unalignment is mainly handled by the matrices product $\left(Q_\alpha K_\beta^T \right)$ which sets the receptive field of cross-modal interactions to the entire input sequences X_α and X_β , hence enabling long-range dependencies modeling, whereas prior works first realign multimodal sequences with the same length and then use autoregressive nature of a model (such as RNN) to iteratively fuse cross-modal information. This makes these approaches inadequate for asynchronous modalities, and less effective, as intermodal interactions are only computed through a compressed hidden state, resulting in a loss of information for long-range dependencies. This cross-modal alignment can be viewed as a step diagonal activation in the cross-modal attention matrix, as pictured in Fig-

Figure 3.3, hence as a temporal monotonic attention. Another drawback of these models lies in their autoregressive nature, making it difficult to parallelize.

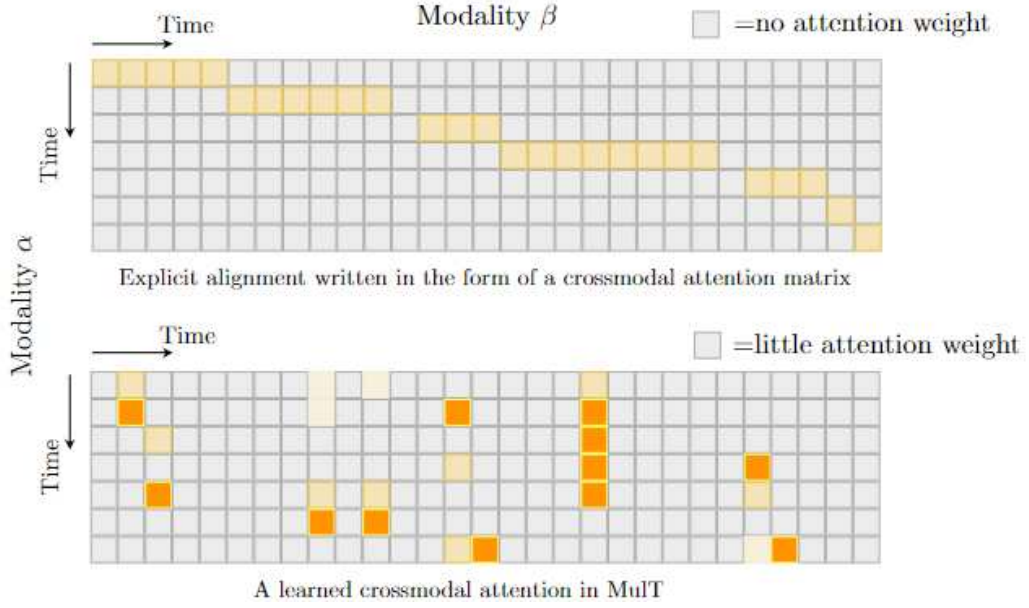


Figure 3.3: Examples of cross-modal attention matrices: explicitly aligned data on top and unaligned data on bottom. Orange boxes correspond to cross-modal pairs the model attends to, with higher weights on brighter boxes. Figure from (Tsai et al. 2019).

However, due to the arbitrarily long size of input sequences in our setting, Multimodal Transformer architecture faces two main issues. Training is intractable due to its space and time complexities, and inference cannot be done in a streaming way, as the vanilla model needs the whole sequence as input to compute the relative matrix product. The construction of efficient transformers is actually a well studied subject, as stated in a recent survey (Tay et al. 2022). Indeed, the self-attention module essentially implies to compute the product of two $l \times l$ matrices (where l is the length of the input sequence), and hence has a complexity in $\mathcal{O}(l^2)$. Thus, many works try to reduce this quadratic complexity, up to a linear one, in order to speed up computation time or to enable longer history for input data.

These approaches approximate the full quadratic-cost attention matrix by adding some sparsity to it, using essentially either Low-rank methods (Sinong Wang et al. 2020), fixed (Child et al. 2019) or learned (Kitaev et al. 2020) sparsification patterns, side memory modules (Zaheer et al. 2020), kernelization (Katharopoulos et al. 2020), or recurrence (Dai et al. 2019).

Remark. From this point, and until the end of the chapter, for the sake of clarity we adopt new notations:

- X_i will denote the i^{th} segment of input X , whereas $X_{i,\alpha}$ will refer to the i^{th} segment of modality α of input X

- X^l will refer to the value of variable X at the layer l
- M_α will be used to refer to the memory bank of modality α , whereas M still refers to the number of modalities
- n will be used to refer to the number of segments, rather than the number of samples

3.3.2 STREAMING INPUT DATA

To our knowledge, none of the previous papers (Tay et al. 2022) (so-called Efficient Transformers) yet considered arbitrary long or streaming data frameworks. This is an issue, as even a matrix whose computation complexity is linear in the input length becomes intractable for very long sequences. In the same way, for input sequences acquired on the fly, modeling inter-modalities dependencies with a cross-modal Transformer requires to recompute the whole attention matrix, which is also intractable. On the other side, some prior works focus on dealing with streaming scenarios, although unimodal. That is the case of papers addressing Automatic Speech Recognition (ASR), or Simultaneous Machine Translation (SMT) tasks, for which there is a need to keep a relevant temporal information flow, coupled with a necessary low latency at inference. This results in a quality-latency trade-off, in which the model needs to produce an output with only a partially available input sequence to ensure low latency. If some works choose to mask previous and future contexts using a sliding window (Moritz et al. 2020; Tripathi et al. 2020; Q. Zhang et al. 2020), a strategy so called time-restricted attention, other ones segment input sequences in smaller chunks before performing self-attention on those chunks (Z. Tian et al. 2020; C. Wang et al. 2020). The main drawback of the former strategy is that the receptive span of the self-attention is linearly growing with the number of transformer layers (see Figure 3.4), implying more latency; while the issue of the latter strategy is on the contrary that the relation between different chunks is lost, undermining the performances of the model as long-range dependencies cannot be computed.

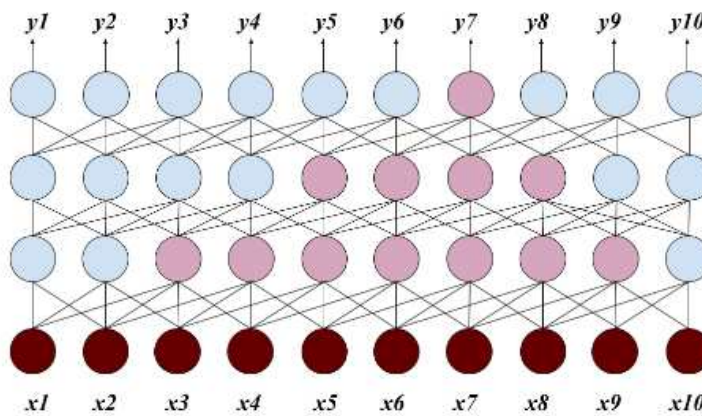


Figure 3.4: Examples of receptive field linearly growing with the number of layers: context masking for the y_7 position (left=2, right=1). Figure from (Q. Zhang et al. 2020).

To alleviate the issue of chunk-wise methods, (C. Wu et al. 2020; Yeh et al. 2021) add a memory bank of multiple slots to this architecture, aiming to store salient history from long-range history. Thus, whereas a recurrent-connection-based approach such as Transformer-XL (Dai et al. 2019) can only attend to a segment that is k steps away after $\mathcal{O}(k)$ steps, Augmented-memory Transformer (AM-TRF) (C. Wu et al. 2020) can already attend to previous segments embeddings through attention performed on memory bank.

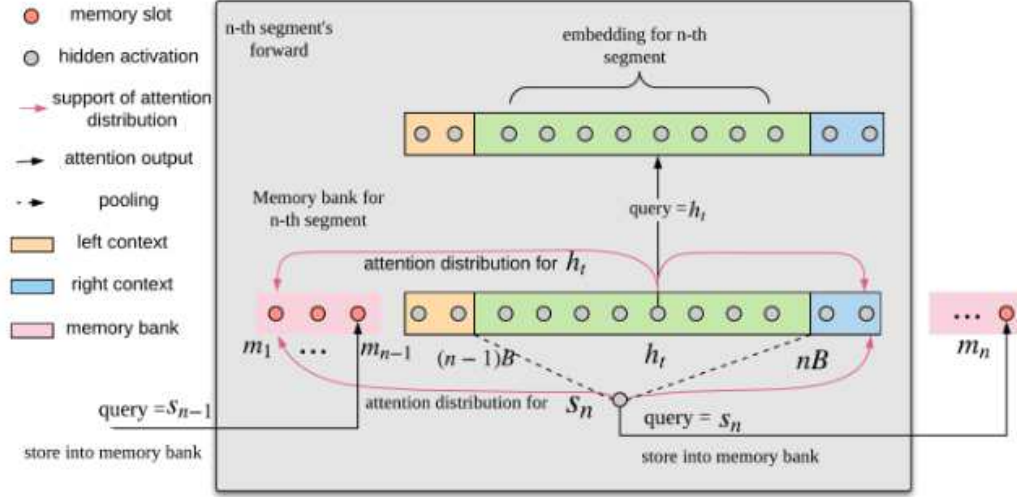


Figure 3.5: Illustration of one forward step for the augmented memory transformer on the n -th segment. B refers to the segment length. Figure from (C. Wu et al. 2020).

Formally, input sequence is first chunked into non-overlapping smaller segments $(C_i)_{i \geq 0}$, which are concatenated with left L_i and right R_i context blocks to prevent boundary effects, hence forming contextual segments $X_i := [L_i : C_i : R_i]$. Considering a contextual segment X_i and a memory bank $\mathbf{M}_i = [m_1, \dots, m_{i-1}]$ containing compressed information from previous segments, the output X_i^{n+1} of the n -th layer is computed as:

$$\begin{aligned}
 \hat{X}_i^n &= \text{LN}(X_i^n) \\
 K_i^n &= W_k[\mathbf{M}_i^n, \hat{X}_i^n] \\
 V_i^n &= W_V[\mathbf{M}_i^n, \hat{X}_i^n] \\
 Q_i^n &= W_Q \hat{X}_i^n \\
 [Z_{L,i}^n : Z_{C,i}^n : Z_{R,i}^n] &:= \text{Attn}(Q_i^n, K_i^n, V_i^n) + X_i^n \\
 \hat{X}_i^{n+1} &= \text{FFN}(\text{LN}([Z_{L,i}^n : Z_{C,i}^n : Z_{R,i}^n])) \\
 X_i^{n+1} &= \text{LN}(\hat{X}_i^{n+1} + [Z_{L,i}^n : Z_{C,i}^n : Z_{R,i}^n]) \\
 m_i^n &= \text{Attn}(W_Q s_i^n, K_i^n, V_i^n)
 \end{aligned}$$

where s_i^n is the mean of C_i^n and LN, FFN, Attn respectively correspond to Layer Normalization, Feed-Forward and Attention layers. After passing through all N layers, outputs corresponding to left and right contexts are discarded to keep only center segments representations $(C_i^N)_{i \geq 0}$. Figure 3.5 illustrates this architecture.

Emformer architecture (Shi et al. 2021) is an improved version of AM-TRF, in the sense that it performs attention on the memory bank from the lower layer, hence dumping its autoregressive nature and becoming parallelizable during training. Besides, it considerably reduces the amount of computation by caching Key and Value matrices from previous segments, and optimizes global performance by cutting off some dependencies during self-attention computation. Figure 3.6 sums up the main differences between both architectures.

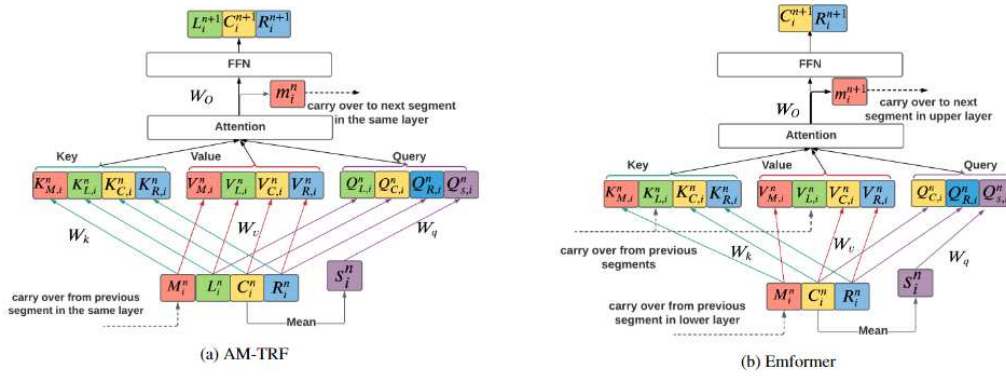


Figure 3.6: Comparison of AM-TRF (left) with Emformer (right). The two approaches mainly differ in the content of M_i^n , that contains summarized information from lower layer in Emformer (enabling to parallelize the computations on all layers); and in the cached keys and values from previous segments. All these optimized changes render the architecture more efficient and prone to work in a streaming scheme. Figure from (Shi et al. 2021).

3.4 PROPOSED MODEL

We propose StreaMulT, a Streaming Multimodal Transformer architecture, taking advantages of both a cross-modal attention mechanism and a block processing approach to tackle the different challenges of this framework. Finally, we optimize the training scheme of the model to lower space complexity, training time and enabling inference short-time response at the same time.

Our global end-to-end architecture combines benefits from block processing and cross-modal attention. The architecture is illustrated in Figure 3.7. We describe here the processing of the data of modality α , with $1 \leq \alpha \leq M$.

X_α is first passed through a 1D convolutional layer aiming to model some local temporal structure, and map all modalities to a common feature dimension d . Segment bounds are then fixed. Extending the block processing method to input data with heterogeneous sampling rates, we define hard segment bounds with respect to the temporal axis, hence producing shared segments across modalities, as illustrated in Figure 3.8. Thus, following block processing approach, every

3 StreaMulT: A Streaming Multimodal Transformer For Heterogeneous and Arbitrarily Long Sequential Data

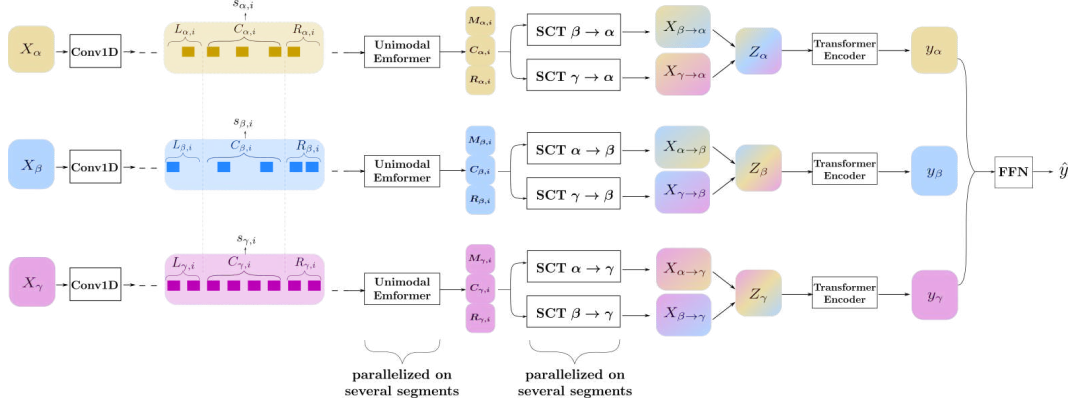


Figure 3.7: Streaming Multimodal Transformer architecture. SCT stands for Streaming Cross-modal Transformer. Different colors represent heterogeneity nature of different modalities, and shadings represent cross-modal features. Each modality-specific time series is passed through a 1D-convolutional layer, and then through a unimodal Emformer block to initialize its modality-specific memory bank. Cross-modal interactions are then captured through SCT blocks, that express a target modality with the help of source modalities' features and memory banks. Target modalities representations computed from different source modalities are then concatenated and passed through modality-specific Transformer encoders, that output contextual cross-modal representations, summarizing the whole sequences. These outputs are then processed by a final FFN to produce the prediction.

contextual segments $X_{\alpha,i} = [L_{\alpha,i} : C_{\alpha,i} : R_{\alpha,i}]$ are processed in a parallel way. They are first given to a modality-specific Emformer module to initialize its own modality memory bank \mathbf{M}_{α} . Then, each source modality / target modality (β / α) pair ($\beta \neq \alpha$) is processed by its own Streaming Cross-modal Transformer (SCT) module. Specifically, each segment from the target modality $X_{\alpha,i}$ is expressed using the same temporal segment from the source modality $X_{\beta,i}$ along with the source modality memory bank $\mathbf{M}_{\beta,i}$. For each layer n :

$$\left[\hat{C}_{\alpha,i}^n, \hat{R}_{\alpha,i}^n \right] = \text{LN}([C_{\alpha,i}^n, R_{\alpha,i}^n]) \quad (3.6)$$

$$\left[\hat{C}_{\beta,i}^n, \hat{R}_{\beta,i}^n \right] = \text{LN}([C_{\beta,i}^n, R_{\beta,i}^n]) \quad (3.7)$$

$$K_{\beta,i}^n = [K_{\mathbf{M},\beta \rightarrow \alpha,i}^n, K_{L,\beta \rightarrow \alpha,i}^n, K_{C,\beta \rightarrow \alpha,i}^n, K_{R,\beta \rightarrow \alpha,i}^n] \quad (3.8)$$

$$V_{\beta,i}^n = [V_{\mathbf{M},\beta \rightarrow \alpha,i}^n, V_{L,\beta \rightarrow \alpha,i}^n, V_{C,\beta \rightarrow \alpha,i}^n, V_{R,\beta \rightarrow \alpha,i}^n] \quad (3.9)$$

$$Z_{C,\beta \rightarrow \alpha,i}^n = \text{Attn}(Q_{C,\beta \rightarrow \alpha,i}^n, K_{\beta,i}^n, V_{\beta,i}^n) + C_{\beta \rightarrow \alpha,i}^n \quad (3.10)$$

$$Z_{R,\beta \rightarrow \alpha,i}^n = \text{Attn}(Q_{R,\beta \rightarrow \alpha,i}^n, K_{\beta,i}^n, V_{\beta,i}^n) + R_{\beta \rightarrow \alpha,i}^n \quad (3.11)$$

$$\left[\hat{C}_{\alpha,i}^{n+1}, \hat{R}_{\alpha,i}^{n+1} \right] = \text{FFN}(\text{LN}([Z_{C,\beta \rightarrow \alpha,i}^n, Z_{R,\beta \rightarrow \alpha,i}^n])) \quad (3.12)$$

$$[C_{\alpha,i}^{n+1}, R_{\alpha,i}^{n+1}] = \text{LN}([\hat{C}_{\alpha,i}^{n+1}, \hat{R}_{\alpha,i}^{n+1}] + [Z_{C,\beta \rightarrow \alpha,i}^n, Z_{R,\beta \rightarrow \alpha,i}^n]) \quad (3.13)$$

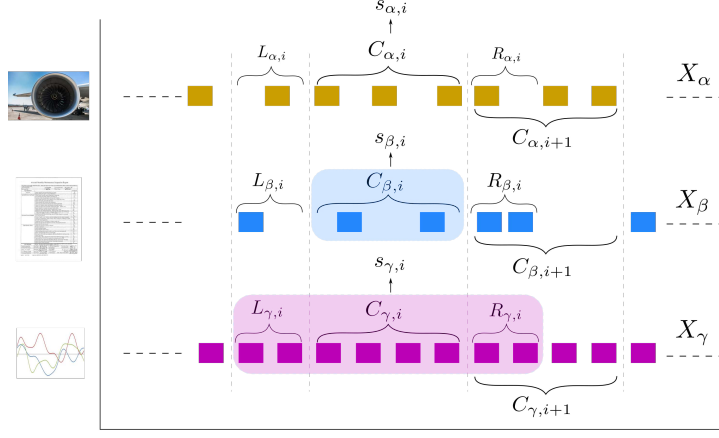


Figure 3.8: Block processing for Multimodal learning in a streaming scheme. For modality α : $X_\alpha, C_{\alpha,i}, L_{\alpha,i}$ and $R_{\alpha,i}$ respectively correspond to the full input sequence, the initial i -th block, and the left and right contexts associated to this block to form the contextual i -th segment. $s_{\alpha,i}$ corresponds to the mean of current segment $C_{\alpha,i}$. Blue area represents an initial block for modality β while the pink one represents a contextual segment for modality γ .

where,

$$[K_{M,\beta \rightarrow \alpha,i}^n, K_{C,\beta \rightarrow \alpha,i}^n, K_{R,\beta \rightarrow \alpha,i}^n] = W_{k,\beta \rightarrow \alpha} [M_{\beta,i}, \hat{C}_{\beta,i}^n, \hat{R}_{\beta,i}^n] \quad (3.14)$$

$$[V_{M,\beta \rightarrow \alpha,i}^n, V_{C,\beta \rightarrow \alpha,i}^n, V_{R,\beta \rightarrow \alpha,i}^n] = W_{v,\beta \rightarrow \alpha} [M_{\beta,i}, \hat{C}_{\beta,i}^n, \hat{R}_{\beta,i}^n] \quad (3.15)$$

$$[Q_{C,\beta \rightarrow \alpha,i}^n, Q_{R,\beta \rightarrow \alpha,i}^n] = W_{q,\beta \rightarrow \alpha} [C_{\beta \rightarrow \alpha,i}^n, R_{\beta \rightarrow \alpha,i}^n] \quad (3.16)$$

and $(K_{L,\beta \rightarrow \alpha,i}^n, V_{L,\beta \rightarrow \alpha,i}^n)$ are the key and value copies (cached) corresponding to previous segments, up to left context size. This module is illustrated in Figure 3.9.

After the last layer N , right contexts representations $(R_{\beta \rightarrow \alpha,i}^N)_{i>0}$ are discarded. $(C_{\beta \rightarrow \alpha,i}^N)_{i>0}$ are concatenated to form the final cross-modal representation $X_{\beta \rightarrow \alpha}$. We then concatenate along the feature dimension all cross-modal outputs corresponding to the same target modality α in a

vector $Z_\alpha := \begin{pmatrix} X_{1 \rightarrow \alpha} \\ \dots \\ X_{\alpha-1 \rightarrow \alpha} \\ X_{\alpha+1 \rightarrow \alpha} \\ \dots \\ X_{M \rightarrow \alpha} \end{pmatrix}$, that is given as input to a Transformer Encoder exploiting sequential

nature of data, to produce modality output y_α . All modality outputs are eventually concatenated and passed through a final fully-connected layer to output prediction \hat{y} .

TRAINING SCHEME: BALANCING SPACE AND TIME COMPLEXITIES

The main motivation to design StreaMulT architecture is to handle the arbitrarily long nature of considered multimodal input sequences. In that sense, the block processing mechanism we use

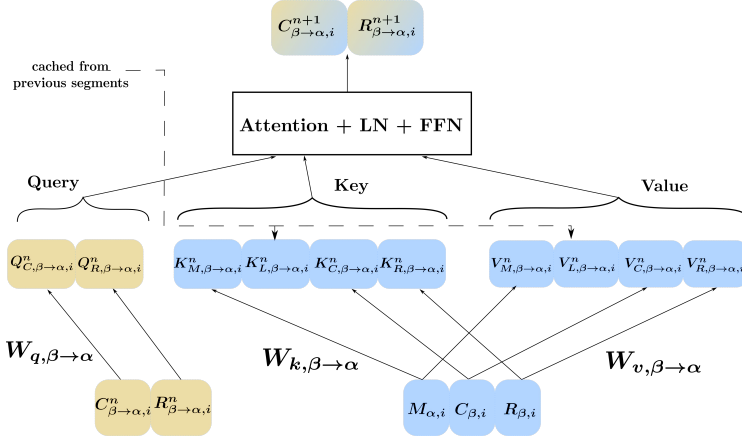


Figure 3.9: Streaming Cross-modal Transformer module.

aims to alleviate the quadratic complexity of cross-modal attention modules, similarly to several speech recognition works (Shi et al. 2021; C. Wu et al. 2020). However, these applications focus on getting short-time response at inference to perform simultaneous speech translation or recognition and hence essentially differ from our framework. Indeed, to handle very long sequences we are at least as concerned about space complexity as time complexity. We thus cannot train our model in the same fashion as these approaches, that is by parallelizing on all input segments the cross-modal attention computation. This indeed still implies a quadratic space complexity to store cross-modal attention weights matrix.

To fulfill both space capacity and efficient training time constraints, we introduce a flexible training scheme. This is illustrated in Figure 3.10. More specifically, at training time we parallelize operations of Memory bank initialization and Streaming Cross-modal Transformer modules on subsequences of h consecutive segments. h is chosen in an empiric way, as the highest integer enabling one’s memory capacity to run the model. This training scheme enables StreaMulT to run arbitrarily long sequences by only storing limited-size matrices, while still benefiting from simultaneous computations through parallelization. Space and time complexities for different layer types are derived in Section 3.6. Note that we do not change the segment length but rather concatenate them in a single matrix product. This enables to keep short segments at inference and thus still work in a short-time response for streaming application.

3.5 EXPERIMENTS

3.5.1 DATASET AND EVALUATION TASK

Despite having a public or private dataset compatible with the Streaming Multimodal Learning challenge, involving long, heterogeneous and unaligned input sequences, we conduct experiments on CMU-MOSEI dataset (Bagher Zadeh et al. 2018), to empirically evaluate the StreaMulT architecture and compare it with existing approaches handling sequential unaligned multimodal data. CMU-MOSEI dataset consists of 23,454 movie review video clips on YouTube, from which are extracted audio and video features using Facet (based on CERT (Littlewort et al. 2011)) and CO-

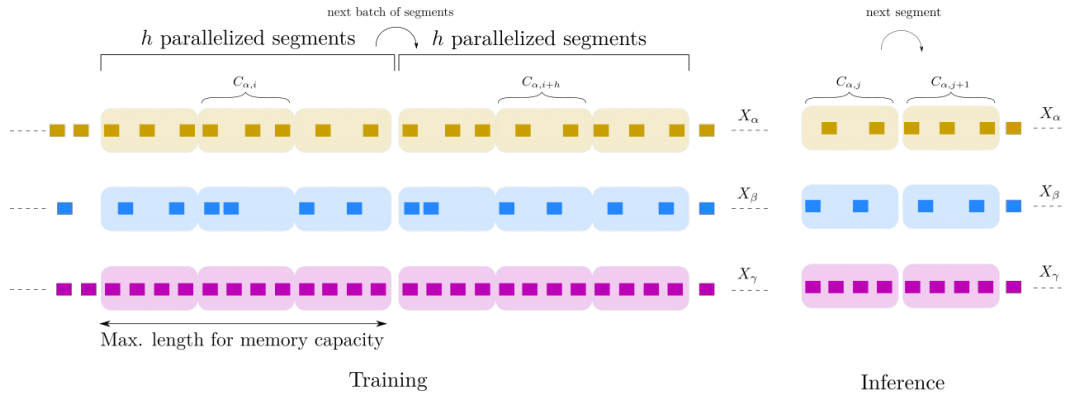


Figure 3.10: Flexible scheme. At training time (left), subsequences of h consecutive segments are created to parallelize cross-modal attention operations. At inference (right), one can still process segments one by one to obtain a short-time response.

VAREP (Degottex et al. 2014). Textual features are also extracted from words transcripts, using GloVe (Pennington et al. 2014) pre-trained embeddings. This produces an unaligned version of the dataset, which is used to create a word-aligned version, using P2FA algorithm (J. Yuan et al. 2008). All aligned sentences are padded to a fixed length of 50 time steps.

The related task aims to perform sentiment analysis on these clips, labeled by human annotators with a sentiment score from -3 to 3. As in (Tsai et al. 2019) and previous works, we evaluate model performances using various metrics: 7-class-accuracy, binary accuracy (positive or negative statements), F1-Score, MAE and correlation between model’s predictions and labels.

3.5.2 EXPERIMENTAL SETTING AND RESULTS

To highlight StreaMulT added value, we conduct experiments in different settings. We first consider input video clips as our whole input sequences, and observe StreaMulT performances when dividing these clips into smaller segments. As we need to define hard segment temporal bounds, which are not given in the unaligned version of CMU-MOSEI, we conduct this experiment with the aligned version of the dataset. For StreaMulT, we choose to divide the input sentences into 5 segments of length 10.

We compared StreaMulT performances with Multimodal Transformer (MulT) and other models addressing Multimodal Sentiment Analysis challenge, among which the recent SOTA methods (W. Han et al. 2021; W. Yu et al. 2021). We strongly emphasize that the added value of StreaMulT is its ability to deal with arbitrarily long unaligned multimodal inputs, and that it does not intend to address Multimodal Sentiment Analysis specific task. Hence at first we only reported Multimodal Transformer metrics scores given in (Tsai et al. 2019) for a fair comparison, as both approaches use GloVe embeddings for text modalities whereas most recent works (W. Han et al. 2021; W. Yu et al. 2021) use BERT embeddings. We also used the available official code¹ for Multimodal Transformer architecture to run the experiments, with hyperparameters given in (Tsai et al. 2019).

¹<https://github.com/yaohungt/Multimodal-Transformer>

3 StreaMulT: A Streaming Multimodal Transformer For Heterogeneous and Arbitrarily Long Sequential Data

We could not reproduce the results shown in the paper, hence we present the results we obtained, that are not as good as the given ones. All scores from our experiments are averaged on 5 runs. The corresponding results are represented in the upper part of following Table 3.1. This shows that our architecture globally reproduces the results of Multimodal Transformer (even performs a little bit better on some metrics), which highlights the availability of its memory bank to properly convey salient information through time, as StreaMulT receptive field only attends to segments of length 10, while MulT attends to whole sequence of length 50.

We then decided to use contextual pre-trained embedding layers for textual modality, namely BERT (Devlin et al. 2019) and BART (Lewis et al. 2020). The corresponding results are described in the lower part of Table 3.1, with a significant improvement in all metrics, StreaMulT-BART achieving now the best results on the aligned version of CMU-MOSEI dataset.

Metric	MAE ^l	Corr ^h	Acc ₇ ^h	Acc ₂ ^h	F1 ^h
MulT‡	0.580	0.703	51.8	82.5	82.3
MulT*	0.615	0.666	49.32	81.05	81.42
StreaMulT*	0.608	0.671	50.08	81.08	81.01
MulT-BERT*	0.563	0.771	50.85	85.59	85.63
StreaMulT-BERT*	0.551	0.764	52.04	85.46	85.56
MulT-BART*	0.543	0.782	53.83	86.28	86.29
StreaMulT-BART*	0.523	0.786	54.54	86.97	86.97

Table 3.1: Results on CMU-MOSEI aligned. Best results are marked in bold. ‡: results from (Tsai et al. 2019). *: Own implementation or reproduced from official code with provided hyper-parameters.

We then trained the Multimodal Transformer and StreaMulT architectures on unaligned version of CMU-MOSEI dataset and reported the results in Table 3.2.

Metric	MAE ^l	Corr ^h	Acc ₇ ^h	Acc ₂ ^h	F1 ^h
TFN‡	0.593	0.700	50.2	-/82.5	-/82.1
LMF‡	0.623	0.677	48.0	-/82.0	-/82.1
MFM‡	0.568	0.717	51.3	-/84.4	-/84.3
ICCN‡	0.565	0.713	51.6	-/84.2	-/84.2
MulT‡	0.591	0.694	50.7	-/81.6	-/81.6
MISA‡	0.568	0.724	-	82.59/84.23	82.67/83.97
MAG-BERT‡	0.539	0.753	-	83.8/85.2	83.7/85.1
Self-MM‡	0.530	0.765	-	82.81/85.17	82.53/85.30
MMIM‡	0.526	0.772	54.24	82.24/85.97	82.66/85.94
MulT-BERT	0.544	0.776	52.86	82.85/85.95	83.18/85.97
MulT-BART	0.532	0.792	54.17	84.11/86.9	84.51/86.95
StreaMulT-BERT	0.570	0.774	50.89	82.31/85.98	82.71/86.13
StreaMulT-BART	0.531	0.778	53.89	83.30/86.35	83.74/86.39

Table 3.2: Results on CMU-MOSEI unaligned. Best results are marked in bold. ‡: results from (W. Han et al. 2021). †: results from (Tsai et al. 2019).

Once again, the usage of a contextual pre-trained embedding layer significantly improves performances. The Multimodal Transformer architecture coupled with a BERT embedding layer now equals the performances of SOTA MMIM model on several metrics, questioning the real

improvement on the Multimodal Sentiment Analysis task over the last three years. Besides, it emphasizes the power of language models, which is supported by the performances of MulT-BART, defining a new SOTA for several metrics on this dataset.

We finally simulated arbitrarily long sequences by concatenating all video clips related to the same speaker and considering these as inputs streams. In this setting, StreaMulT architecture successfully parallelizes its training along segments and handles long sequences at inference in a streaming way. On the other side, Multimodal Transformer faces memory issue.

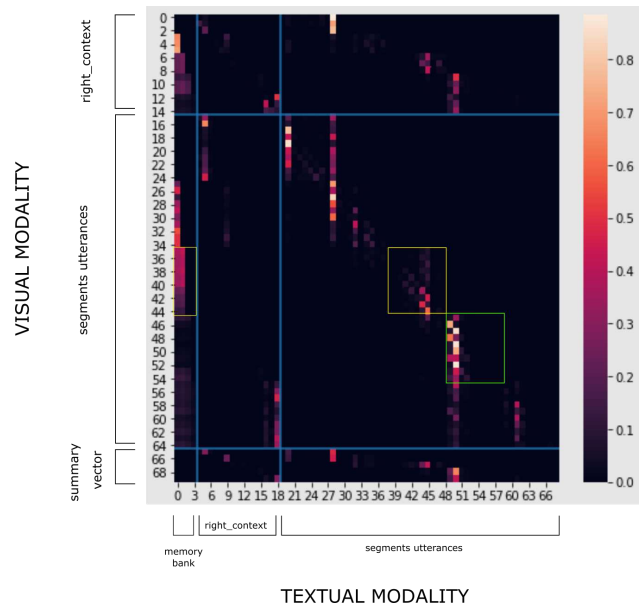


Figure 3.11: Heatmap of StreaMulT attention weights for the Visual/Textual cross-modal module. The sequence of length 50 is chunked into segments of length 10, with left and right contexts of respectively lengths 10 and 3.

To qualitatively validate our architecture, we also plot the heatmap of the different attention weights of the model in Figure 3.11.

This plot represents the different attention weights of the Streaming Cross-modal Transformer related to the visual/textual modalities, for a multimodal sequence of length 50. For consistence with previous notations, we call α the visual modality and β the textual modality. On the x-axis, the key matrix K_β is organized as: [memory bank; right contexts; segments utterances]. On the y-axis, the query matrix Q_α is organized as: [right contexts; segments utterances; summary vectors]. Different blocks are delimited on Figure 3.11 by vertical and horizontal blue lines.

This figure first reminds us, as stated in (Tsai et al. 2019), that language sequences are unaligned across modalities. This is indeed shown by the several activations on vertical lines (differing from a temporal monotonic diagonal line), corresponding to specific word embeddings correlated to many visual frames.

If some of these unalignments remain in the scope of the same temporal segment, as illustrated in

the fourth segment by the green box, the access to the memory bank enables the model to attend beyond the current segment and to catch unalignments at longer range, as illustrated in the third segment by the yellow boxes. The yellow box on the right witnesses the unaligned dependencies within the third segment, while the left yellow box illustrates that some textual features of the past history activate the visual frames of the current segment.

These different behaviors show the ability of the StreaMulT architecture to adapt its strategy depending of the context, attending to unaligned data from past history via memory bank when necessary.

3.5.3 ABLATION STUDY

We conducted some ablation experiments to assess for the importance of specific parts of the model or of the data. The results of these experiments are displayed in Table 3.3. Specifically, we tried to highlight the importance of each modality for the considered MSA task by sequentially leaving it out.

While omitting sound or images streams does not affect much the performances of the model (less than 1% loss in binary accuracy and $F1$ -score), the absence of textual modality results in an impressive drop of more than 15% in binary accuracy and $F1$ -score, that cannot be compensated by visual and audio modalities.

Metric	MAE ^l	Corr ^h	Acc ₇ ^h	Acc ₂ ^h	F1 ^h
(audio, visual)	0.826	0.274	41.11	65.36/67.59	66/68.69
(audio, textual)	0.546	0.775	53.07	81.92/86.13	82.46/86.11
(textual, visual)	0.542	0.784	53.59	82.32/86.61	83.21/86.76
StreaMulT-BART	0.523	0.786	54.54	82.99/86.97	83.46/86.97

Table 3.3: Ablation study on CMU-MOSEI aligned. Best results are marked in bold.

3.6 TIME AND SPACE COMPLEXITIES STUDY

Layer Type	Time Complexity by layer	Space Complexity by layer	Sequential Operations
Self-Attention	$\mathcal{O}(n^2 \cdot d)$	$\mathcal{O}(n^2 + n \cdot d)$	$\mathcal{O}(1)$
Cross-modal Attention	$\mathcal{O}(n_\alpha \cdot n_\beta \cdot d)$	$\mathcal{O}(n_\alpha \cdot n_\beta + n_\alpha \cdot d + n_\beta \cdot d)$	$\mathcal{O}(1)$
Streaming Cross-modal Attention (regular training scheme)	$\mathcal{O}(n_\alpha \cdot n_\beta \cdot d)$	$\mathcal{O}(n_\alpha \cdot n_\beta + n_\alpha \cdot d + n_\beta \cdot d)$	$\mathcal{O}(1)$
Streaming Cross-modal Attention (flexible training scheme)	$\mathcal{O}(n_\alpha \cdot h \cdot C_\beta \cdot d)$	$\mathcal{O}(h^2 \cdot C_\alpha \cdot C_\beta + h \cdot C_\alpha \cdot d + h \cdot C_\beta \cdot d)$	$\mathcal{O}(\frac{n_\alpha}{h C_\alpha})$

Table 3.4: Time and space Complexities for different layer types.

Table 3.4 derives the different time and space complexity classes for different types of layers, along with the number of sequential operations. Vanilla self-attention layers have a quadratic complexity both in time and in space, which is problematic for handling long sequences. Similarly, cross-modal attention, as defined in (Tsai et al. 2019) also has a quadratic complexity in the sequence length. More precisely, the complexity class depends of the product of the two modalities lengths n_α, n_β , as they can differ.

Streaming Cross-modal Attention modules trained in regular fashion for blocks processing (as in

(Shi et al. 2021)) have the same space and time complexity classes, which make them intractable for arbitrarily long sequences. This indeed requires to compute the matrix product of $Q_\alpha \in \mathbb{R}^{d_{q_\alpha} \times d}$ and $K_\beta \in \mathbb{R}^{d_{k_\beta} \times d}$, with $d_{q_\alpha} = n_{\text{seg}} \cdot (R_\alpha + C_\alpha + 1)$ and $d_{k_\beta} = n_{\text{seg}} \cdot (R_\beta + C_\beta + l_{\text{mem}})$. n_{seg} is the number of segments of the input sequence, R_α and R_β correspond to the length of right contexts for modalities α, β , and C_α and C_β to the length of their central segments. Last, l_{mem} corresponds to the length of a memory cell. We suppose that R and l_{mem} are negligible before C , and noting that $n_{\text{seg}} = \frac{n_\alpha}{C_\alpha} = \frac{n_\beta}{C_\beta}$, one obtains the results mentioned above.

If we train this layer in the flexible scheme as described in Section 3.4, for each subsection of h consecutive segments we need to handle the product of matrices $Q_\alpha \in \mathbb{R}^{d_{q_\alpha} \times d}$ and $K_\beta \in \mathbb{R}^{d_{k_\beta} \times d}$, with now $d_{q_\alpha} = h \cdot (R_\alpha + C_\alpha + 1)$ and $d_{k_\beta} = h \cdot (R_\beta + C_\beta + l_{\text{mem}})$, which has a time complexity class of $\mathcal{O}(h^2 \cdot C_\alpha \cdot C_\beta \cdot d)$. As mentioned in the third column, to process the whole sequence we need to perform $\frac{n_\alpha}{h C_\alpha}$ sequential operations, which also derives the whole time complexity class. Note that the space complexity now only depends on h, C and d , as we only need to store a sub-sequence at a time.

At inference, one can thus choose $h = 1$ to process the input sequence in streaming, enabling a short-time response with time and space complexity classes being respectively $\mathcal{O}(C_\alpha \cdot C_\beta \cdot d)$ (for one segment) and $\mathcal{O}(C_\alpha \cdot C_\beta + C_\alpha \cdot d + C_\beta \cdot d)$.

3.7 IMPLEMENTATION DETAILS

We now describe the different parts of the implementation of the StreaMulT algorithm for the example of Multimodal Sentiment Analysis.

Algorithm 2 StreaMulT Training loop.

Require: train_loader, model, text_encoder, optimizer, criterion

```

for  $i = 1, \dots, \text{nb\_sequences\_batches}$  do
  sequences, labels  $\leftarrow$  iterate(train_loader)
  raw_text, audio, vision  $\leftarrow$  sequences
  text  $\leftarrow$  text_encoder(raw_text)
  segments_batches  $\leftarrow$  sequence_to_segments_batches(text, audio, vision, labels,
                                                    segment_size, memory_batch_size,
                                                    left_context, right_context)

  state  $\leftarrow$  None
  for  $j = 1, \dots, \text{nb\_segments\_batches}$  do
    text, audio, vision, labels  $\leftarrow$  segments_batches[j]
    preds, state  $\leftarrow$  model(text, audio, vision, state)
    loss  $\leftarrow$  MAE(preds, labels)
    model  $\leftarrow$  backward_propagation(loss, model)
    model  $\leftarrow$  update(model, optimizer)
  end for
end for

```

Algorithm 3 StreaMulT forward loop.

Require: text, audio, vision, state
 $X_t, X_a, X_v \leftarrow [\text{Conv1D}_t(\text{text}), \text{Conv1D}_a(\text{audio}), \text{Conv1D}_v(\text{vision})]$
state, $X_t, X_a, X_v \leftarrow \text{Emformer}_t(X_t), \text{Emformer}_a(X_a), \text{Emformer}_v(X_v)$
 $Z_{a \rightarrow t}, \text{state} \leftarrow \text{SCT}_{a \rightarrow t}(X_t, X_a, \text{state})$
 $Z_{v \rightarrow t}, \text{state} \leftarrow \text{SCT}_{v \rightarrow t}(X_t, X_v, \text{state})$
 $Z_t \leftarrow [Z_{a \rightarrow t} : Z_{v \rightarrow t}]$
 $Z_t \leftarrow \text{resize_segments}(Z_t)$
 $Z_t \leftarrow \text{TransformerEncoder}_t(Z_t)$
 $Z_{t \rightarrow a}, \text{state} \leftarrow \text{SCT}_{t \rightarrow a}(X_a, X_t, \text{state})$
 $Z_{v \rightarrow a}, \text{state} \leftarrow \text{SCT}_{v \rightarrow a}(X_a, X_v, \text{state})$
 $Z_a \leftarrow [Z_{t \rightarrow a} : Z_{v \rightarrow a}]$
 $Z_a \leftarrow \text{resize_segments}(Z_a)$
 $Z_a \leftarrow \text{TransformerEncoder}_a(Z_a)$
 $Z_{t \rightarrow v}, \text{state} \leftarrow \text{SCT}_{t \rightarrow v}(X_v, X_t, \text{state})$
 $Z_{a \rightarrow v}, \text{state} \leftarrow \text{SCT}_{a \rightarrow v}(X_v, X_a, \text{state})$
 $Z_v \leftarrow [Z_{t \rightarrow v} : Z_{a \rightarrow v}]$
 $Z_v \leftarrow \text{resize_segments}(Z_v)$
 $Z_v \leftarrow \text{TransformerEncoder}_v(Z_v)$
 $Z \leftarrow [Z_t[-1] : Z_a[-1] : Z_v[-1]]$
preds $\leftarrow \text{projection_layer}(Z)$
return preds, state

In [Algorithm 2](#), the function `sequence_to_segments_batches` splits the input batches of long sequences into smaller segments batches whose size is controlled by the parameter `memory_batch_size`, depending on the available memory of the hardware (this is illustrated by the batches of h parallelized segments in [Figure 3.10](#)). The variable "state" is initialized as None and will contain the different memory banks, along with the cached left contexts (keys and values). The forward loop of the model is detailed in [Algorithm 3](#).

In [Algorithm 3](#), the different unimodal `segment_batches` are passed through unimodal Emformers to initialize memory banks and get a first intramodal representation. All cross-modal representations $Z_{\alpha \rightarrow \beta}$ are then obtained through related SCT modules, which also update the content of the variable "state". The function "resize_segments" splits the different `segments_batches` into segments, from which contextual representations are learned thanks to a modality-specific Transformer Encoder. A last projection module composed of feed-forward layers with residual connections and dropout regularization (for training) produces the final representations, from which the predictions related to these segments are obtained thanks to an usual classifier (a linear layer).

Algorithm 4 Streaming Cross-modal Transformer ($\beta \rightarrow \alpha$) forward loop.**Require:** $X_\alpha, X_\beta, \text{state}, \text{rpe}$ $X_{\beta \rightarrow \alpha} \leftarrow X_\alpha$ **for** $i = 1, \dots, \text{nb_layers}$ **do** $\text{rc_blocks}_{\beta \rightarrow \alpha}, \text{central_segments}_{\beta \rightarrow \alpha} \leftarrow X_{\beta \rightarrow \alpha}$ $\text{summary}_{\beta \rightarrow \alpha} \leftarrow \text{summarize}(\text{rc_blocks}_{\beta \rightarrow \alpha}, \text{central_segments}_{\beta \rightarrow \alpha})$ $\text{rc_blocks}_\beta, \text{central_segments}_\beta \leftarrow X_\beta$ $X_{\beta \rightarrow \alpha} \leftarrow [\text{LN}([\text{rc_blocks}_{\beta \rightarrow \alpha} : \text{central_segments}_{\beta \rightarrow \alpha}])] : \text{summary}_{\beta \rightarrow \alpha}$ $X_\beta \leftarrow [\text{memory}_\beta : \text{LN}([\text{rc_blocks}_\beta : \text{central_segments}_\beta])]$ $Q_{\beta \rightarrow \alpha} \leftarrow X_{\beta \rightarrow \alpha} W_{Q_{\beta \rightarrow \alpha}}$ $K_\beta, V_\beta \leftarrow \text{split}(X_\beta W_{KV_\beta})$ $K_\beta \leftarrow [K_\beta[: \text{mem_size} + \text{rc_size}] : \text{cached_K}_\beta : K_\beta[-\text{central_segments_size} :]]$ $V_\beta \leftarrow [V_\beta[: \text{mem_size} + \text{rc_size}] : \text{cached_V}_\beta : V_\beta[-\text{central_segments_size} :]]$ $Q_{\beta \rightarrow \alpha}, K_\beta, V_\beta \leftarrow \text{reshape_multihead_scaling}(Q_{\beta \rightarrow \alpha}, K_\beta, V_\beta)$ $a_{\text{weights}} \leftarrow \text{attention_mask}(Q_{\beta \rightarrow \alpha} (K_\beta + \text{rpe}_k)^T)$ $a_{\text{probs}} \leftarrow \text{dropout}(\text{softmax}(a_{\text{weights}}))$ $\text{output} \leftarrow a_{\text{probs}} (V_\beta + \text{rpe}_v)$ $X_{\beta \rightarrow \alpha}, \text{state} \leftarrow \text{after_attention_operations}(\text{output})$ **end for** $Z_{\beta \rightarrow \alpha} \leftarrow X_{\beta \rightarrow \alpha}$ **return** $Z_{\beta \rightarrow \alpha}, \text{state}$

In [Algorithm 4](#), a cross-modal representation $Z_{\beta \rightarrow \alpha}$ is computed from unimodal input streams X_α, X_β , along with the variable "state" that contains global information such as memory banks or cached keys and values from previous segments (used as left context). Therefore, at the beginning of each layer, right context blocks and central segments are extracted from the input streams used for queries and keys/values. Summary vectors are then computed for query stream as a temporal average pooling of each segment. The queries and keys/values input streams are then reordered on temporal axis, respectively as [right_context_blocks, central_segments, summary] and [memory_bank, right_context_blocks, central_segments], in order to compute all attention weights in a single matrix product. Matrices $Q_{\beta \rightarrow \alpha}, K_\beta$ and V_β are thus computed thanks to linear projection layers (K and V are computed as once and split in two halves), and cached keys and values are concatenated at the relevant time steps on temporal axis. As its name suggests, the function "reshape_multihead_scaling" reshapes these matrices along the feature dimension axis to perform multihead-attention, and rescales their corresponding elements by the factor $\sqrt{d_k}$ (see [Equation 3.4](#)). The queries/keys matrix product is then computed, with an additive term "rpe_k" in the key matrix corresponding to relative positional embeddings, implemented in the same way as in (Shaw et al. 2018). "rpe_k" and "rpe_v" are obtained as linear projections of a distance matrix "rpe", global for the whole StreaMult architecture. An attention mask is also applied to ensure the fact that queries attend to relevant keys. At the end, the output representation of $X_{\beta \rightarrow \alpha}$ is fed to several output layers (feed-forward layers, residual connections, layer normalizations; see [Equation 3.12](#) and [Equation 3.13](#)) and is given as input for the next SCT layer. The

"after_attention_operations" function also contains update steps for the variable "state".

We realized an hyperparameters tuning when training the StreaMulT model, evaluating different parameters configurations on a validation set. The optimized hyperparameters are listed in Table 3.5, alongside with their different values.

Hyperparameter	Value
batch size	16
nb layers Emformer	3
nb layers SCT	4
nb heads attention	8
embedding dimension	40
segment size	5
memory size	5
left context	5
right context	3
keep raw	False
fine-tune text encoder	True
learning rate	1e-3
learning rate text encoder	1e-5

Table 3.5: Optimal hyperparameters configuration for StreaMulT on CMU-MOSEI aligned.

The training has also been realized with Adam optimizer (Kingma et al. 2015), early stopping procedure and gradient clipping. Dropout is frequently used throughout the network, mostly with a weight of 0.1.

CONCLUSION

In this chapter, we introduced StreaMulT, a model that merges the power of cross-modal attention for multimodal representation and the efficiency of the block processing approach to manage arbitrarily long sequences in a streaming manner. In doing so, StreaMulT effectively responds to the novel challenges of Multimodal Learning with heterogeneous and arbitrarily long sequential streams—a task that previous approaches have struggled with. Experiments carried out on the CMU-MOSEI dataset demonstrated promising results, with a notable enhancement in state-of-the-art metrics and a demonstrated capacity to handle arbitrarily long data during training and process sequences in a streaming manner during inference. The paradigm has numerous applications such as Industrial Monitoring, which necessitates an adapted dataset for benchmarking future related works. A main drawback of StreaMulT and similar multimodal architectures though, is that we do not control how the cross-modal interactions are captured through the learned representations. Thus, in the next chapter we present some thoughts on the characterization of relevant information across modalities.

4 THOUGHTS ON THE CHARACTERIZATION OF INFORMATION ACROSS MODALITIES

CHAPTER'S SUMMARY

This chapter presents a discussion of diverse multimodal interactions, rather than advancing a specific contribution. It begins by decomposing the relevant content into redundant and complementary types of information. Subsequently, it delves into the exploration of research focused on maximizing redundant information, predominantly within the multi-view setting, and the frameworks employed therein. The final section attempts to broaden these approaches to encapsulate the characterization of complementary information, and articulates critiques of both existing methodologies and the deficit of evaluation benchmarks. This analysis offers a comprehensive understanding of the ongoing challenges and potential paths forward in the field of multimodal learning.

4.1 INTRODUCTION

The preceding chapter delved into the development and understanding of StreaMulT, a streaming multimodal transformer capable of managing arbitrarily long, unaligned heterogeneous data streams. This innovative model, like its multimodal counterparts, attempts to model relationships between different modalities. It does so in a supervised manner, **employing the powerful backpropagation algorithm to devise meaningful and insightful latent multimodal representations**. The pragmatic capability of the StreaMulT architecture has been underscored, particularly in relation to handling voluminous, unaligned, and diverse data streams. However, as we turn the page onto this chapter, our focus shifts subtly, yet significantly. While previous models, including StreaMulT, have offered valuable contributions to multimodal learning, an under-explored area has emerged – the lack of models that rely on well-defined theoretical tools and assumptions, such as mutual information losses, to leverage and control complementary information between modalities.

In a multimodal setting, various modalities often bring forward information that may appear redundant on the surface. Many multimodal models, accordingly, tend to focus primarily on multi-view settings where the redundant information is the primary target. This, indeed, is a valid and essential task, as redundant information is assumed to essentially be relevant for downstream prediction tasks. However, in doing so, we should not lose sight of another equally crucial aspect – **the potential complementarity that exists between different modalities**. Indeed, differently to multi-view settings in which inputs generally consist in variations of a same scene (such as data augmentations or different point of views), we may also be interested in different multimodal

settings, in which input may describe a scene at different scales or times. In that configuration, different modalities may share less (down to none) information, making the exploitation of complementary information crucial.

Complementarity, in this context, refers to the unique and supplementary information that different modalities may bring to the table, which could be key to building a more comprehensive understanding of the data at hand. Complementary information, when effectively utilized, may not only enhance the richness of multimodal representations but also bring insights that could potentially be overlooked otherwise.

The significance of harnessing the potential of complementarity in multimodal learning is clear. However, how to integrate such a notion into our existing models in a theoretically robust and practical way is a challenge yet to be fully tackled.

This chapter reflects our exploration into this very challenge, without delving into specific experiments. It encapsulates an important question that has persistently emerged throughout the course of this thesis work: **How can we effectively integrate complementarity into multimodal learning, and how should we measure the performance of such endeavors?** This exploration is crucial and deserves to be highlighted here, as it forms the groundwork for future investigations and implementations.

4.2 THEORETICAL BACKGROUND

4.2.1 MULTIMODAL LEARNING PROVABLY PERFORMS BETTER THAN UNIMODAL

For the sake of clarity, we limit the scope of this chapter to a setting with only two modalities. However, the discussions and conclusions outlined here are applicable to any number of modalities. Thus, we restrict the setting introduced in [Subsection 2.2.2](#) to $M = 2$ modalities, where data points are represented as $\mathbf{x} = (x_{(1)}, x_{(2)})$ and modeled by random variables $(X_1, X_2) \in \mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2$. We denote $\mathcal{F}_i, \mathcal{G}_i, \mathcal{H}_i$ as the restriction of the classes of functions $\mathcal{F}, \mathcal{G}, \mathcal{H}$ (respectively) to the unimodal input space \mathcal{X}_i , for $i \in \{1, 2\}$.

The multimodal fusion framework is motivated by the fundamental assumption:

$$\min_{f \in \mathcal{F}} R(f) \leq \min \left(\min_{f_1 \in \mathcal{F}_1} R(f_1), \min_{f_2 \in \mathcal{F}_2} R(f_2) \right) \quad (4.1)$$

that is, that considering more modalities is beneficial for a task. Noting (\hat{h}, \hat{g}) and (\hat{h}_i, \hat{g}_i) the empirical risk minimizers learned on $(\mathcal{H}, \mathcal{G})$ and $(\mathcal{H}_i, \mathcal{G}_i)$ for $i \in \{1, 2\}$, respectively, (Y. Huang et al. 2021) show that :

$$R(\hat{h} \circ \hat{g}) \leq \min_{i=1,2} \left[R(\hat{h}_i \circ \hat{g}_i) + \eta(\hat{g}) - \eta(\hat{g}_i) + \mathcal{O}\left(\sqrt{\frac{1}{n}}\right) \right]. \quad (4.2)$$

where $\eta(g)$ is the latent representation quality introduced in [Equation 4](#), and n is the sample size of the training dataset. As $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2$, for $i \in \{1, 2\}$ any candidate $g_i \in \mathcal{G}_i$ can be retrieved in \mathcal{G} , and thus $\eta(\hat{g}) - \eta(\hat{g}_i) \leq 0$. In essence, this proposition argues that for a sufficient sample size n , the inclusion of more modalities enhances performance on learning tasks,

and this enhancement is measured by the quality of its latent representation. This hypothesis appears quite intuitive: augmenting the number of modalities leads to an increased data availability, which can help the model in refining its predictions. The additional modality either reinforces the model's current belief (thereby increasing its predictive confidence) or introduces a novel element to the input data. This new element, coupled with information from other modalities, may alter the model's belief, reducing its confidence in the prior prediction and perhaps even producing a change in the prediction itself. Therefore, we can classify this flux of information as either *redundant* or *complementary*. The information theory, as proposed by (Shannon 1948), provides a solid framework to formalize these concepts.

4.2.2 A BRIEF RECAP ON INFORMATION THEORY

Considering two random variables X and Y , the mutual information between X and Y is defined by :

$$I(X; Y) = H(X) - H(X|Y)$$

as the difference between the entropy of X and the conditional entropy of X given Y . In the context of information communication, $I(X; Y)$ quantifies the average reduction in bits required to encode X given knowledge of Y , compared to the scenario where Y is unknown. As entropy measures the uncertainty of a random variable's value, $I(X; Y)$ can also be interpreted as the reduction in uncertainty about one variable's value when the other is observed. In our multi-modal context, we use the mutual information operator to measure the interdependencies between modalities X_1 and X_2 , and between these modalities and the prediction task at hand represented by the random variable Y .

From there, one can define redundancy between modalities, as in (Federici et al. 2020):

DEFINITION

Definition 5 (Redundancy). X_1 is redundant with respect to X_2 for Y if and only if $I(Y; X_1|X_2) = 0$. If we also have $I(Y; X_2|X_1) = 0$, we say that X_1 and X_2 are mutually redundant.

The redundancy between modalities X_1 and X_2 for Y can thus be measured by $I(X_1; X_2; Y)$. It corresponds to the quantity of predictive information shared by both modalities.

Inversely, we define the complementarity of one modality relative to another as follows:

DEFINITION

Definition 6 (Complementarity). X_1 is complementary with respect to X_2 for Y if and only if $I(Y; X_1|X_2) > 0$.

The complementarity between modalities X_1 and X_2 for Y can thus be measured by $I(X_1, X_2; Y) - I(X_1; X_2; Y) = I(Y; X_1|X_2) + I(Y; X_2|X_1)$. It corresponds to the

quantity of predictive information that is modality-specific, hence not shared by both modalities.

In the rest of the chapter, we focus on these two parts of the information. We first review works that focused on maximizing the redundancy across modalities, and then discuss the limitations of current multimodal approaches when characterizing the complementarity.

4.3 MAXIMIZING REDUNDANT INFORMATION

Numerous studies in the field of multimodal learning have aimed to exploit redundant information across modalities to construct more expressive latent representations. This is particularly the case of all works concentrating on multi-view scenarios, where redundancy is inherently assumed between the two views. In the multi-view learning paradigm, the input variable is partitioned into two different views X_1 and X_2 and there is a target variable Y of interest. As a consequence, it is highly connected to our multimodal setting in which X_1 and X_2 are two different modalities of a same observed phenomenon.

As formulated by (Sridharan et al. 2008):

Assumption (Multi-view assumption). There exists an $\epsilon_{info} > 0$ such that:

$$I(Y; X_2 | X_1) \leq \epsilon_{info} \quad \text{and} \quad I(Y; X_1 | X_2) \leq \epsilon_{info}$$

The Multi-view assumption states that (on average) if we already know X_1 , then there is little more information that we could gain about Y from observing X_2 (and vice-versa). This small potential gain is quantified by ϵ_{info} .

This hypothesis is however generally not assumed (for a small ϵ_{info}) in the multimodal setting, as different modalities, compared to different views of a same scene, may contain a non-negligible quantity of modality-specific information that is of use for prediction.

Building on this assumption, various frameworks have been developed to capitalize on this information without requiring supervision. Many studies, for instance, employ the self-supervised paradigm and particularly the contrastive learning framework, conjecturing that "a powerful representation is one that models view-invariant factors" (Y. Tian, Krishnan, et al. 2020). These works, driven by the InfoMax principle (Linsker 1988), aim to bring representations of different views closer to each other and hence maximize mutual information between them (Bachman et al. 2019; Henaff 2020; Ji et al. 2019; Y. Tian, Krishnan, et al. 2020). Similarly, (Alayrac, Recasens, et al. 2020) extend this framework to the multimodal setting, using Inf_{NCE} loss (Oord et al. 2018) between the modality representations of videos (audio, visual, textual modalities) in a shared latent space.

Concurrently, alternative strategies have been proposed to refine this approach by discarding superfluous information. These strategies mainly build on the concept of a sufficient representation (Achille et al. 2018):

DEFINITION

Definition 7 (Sufficient representation). A representation Z of X is sufficient for Y if and only if $I(X; Y|Z) = 0$.

The mutual information between X and its representation Z can then be decomposed as follows:

$$I(X; Z) = I(Y; Z) + I(X; Z|Y) \quad (4.3)$$

Proof. Using the multivariate mutual information chain rule (Cover 1999), we have:

$$\begin{aligned} I(X; Z|Y) &= I(X; Z) - I(X; Y; Z) \\ &= I(X; Z) - I(Y; Z) - I(Y; Z|X) \end{aligned}$$

As Z is a representation of X , we have $I(Y; Z|X) = 0$, which concludes the proof. \square

The first term represents the predictive information we seek to preserve for effective prediction, while the second term, devoid of predictive power, is considered as superfluous for the task at hand. The information bottleneck principle (Tishby et al. 2000) provides a suitable approach to construct expressive representations in a supervised manner. This principle seeks to minimize $I(X; Z)$, while simultaneously maximizing $I(Y; Z)$. In other words, it constraints Z to be a minimal sufficient statistics (Soatto et al. 2016) of X to predict Y . Given the complexities associated with computing mutual information, proxies such as variational lower bounds are often used (Alemi et al. 2017).

The information bottleneck principle was further adapted to the multi-view setting by (Qi Wang et al. 2019), and to an unsupervised framework by (Federici et al. 2020). The key theoretical contribution of their work is [Corollary 1](#).

Corollary 1. Let X_1 and X_2 be two mutually redundant views for a target Y and let Z_1 be a representation of X_1 . If Z_1 is sufficient for X_2 (i.e. $I(X_1; X_2|Z_1) = 0$) then Z_1 is as predictive for Y as the joint observation of the two views ($I(X_1, X_2; Y) = I(Z_1; Y)$). In that case:

$$I(X_1; Z_1) = I(X_2; Z_1) + I(X_1; Z_1|X_2)$$

In the latter equation, the first term is predictive for X_2 , while the second term represents superfluous information for the task (because of the mutual redundancy of the views). This result suggests an unsupervised learning objective: to maximize the first term while simultaneously minimizing the second one. By doing that, we force the representation Z_1 to be sufficient for X_2 (hence conserving its predictive power following the corollary), while discarding superfluous information to make the representation more robust. The global objective simultaneously optimizes the same tradeoff by symmetrically decomposing $I(X_2; Z_2)$. These quantities can be approximated using lower bounds on mutual information (Hjelm et al. 2019; Oord et al. 2018; Poole et al. 2019).

The hypothesis that the learned representation should contain the minimal sufficient information is supported by (Y. Tian, C. Sun, et al. 2020), that focus on identifying *good* views for contrastive

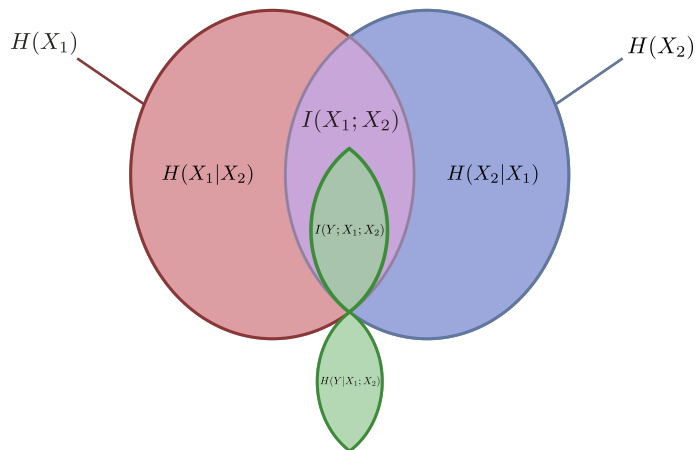


Figure 4.1: Information diagram of two modalities X_1, X_2 that are mutually redundant for a given target Y . The amount of information conveyed by X_1 and X_2 are represented by red and blue areas, respectively, while the purple area represents the amount of information share by both modalities. The amount of predictive information, conveyed by the variable Y , is represented by the green area. The only amount of predictive information that is accessible is $I(Y; X_1; X_2)$. This piece of information is shared by both modalities (mutual redundancy), hence its representation area on the diagram is encapsulated in the purple area, representing $I(X_1; X_2)$. It is worth noting that we generally lack access to the entirety of the information conveyed by Y ; this unavailable quantity is $H(Y|X_1; X_2)$.

learning in a multi-view setting.

The goal of maximizing redundant information between views in the latent representations is largely driven by the mutual redundancy assumption intrinsic to the multi-view scenario. Indeed, from a multi-view standpoint, where the same object is observed from different angles, view-specific data is often treated as noise that does not contribute to prediction. Figure 4.1 illustrates a setting of total mutual redundancy between the modalities X_1 and X_2 for a target Y , with the help of an information diagram, a type of Venn diagram.

As a result of this assumption, methods that learn representations to maximize this information perform well on multi-view downstream tasks (Tosh et al. 2021). This framework has been extended to the multimodal setting, in which the modalities are considered as the different views. The related works essentially rely on contrastive methods to tackle the multimodal coordinated representation learning in a self-supervised manner (Alayrac, Recasens, et al. 2020; J. S. Chung et al. 2016; Miech et al. 2020; Radford, Kim, Hallacy, et al. 2021; C. Sun, Baradel, et al. 2019), where modalities are for instance videos, text or sound. The contrastive framework implicitly rely on the same assumption, as several works have shown the parallel between used contrastive losses and the maximization of mutual information between views (Y. Tian, C. Sun, et al. 2020; M. Wu et al. 2020). However, by focusing solely on shared factors, these approaches neglect the complementary part of the information, thereby failing to harness all synergies between modalities.

4.4 CHARACTERIZING COMPLEMENTARY INFORMATION

When considering modalities that comprise complementarity, the general setting is the one depicted in Figure 4.2, where the predictive information available is distributed across both modalities, with some information being shared and some being specific to each modality.

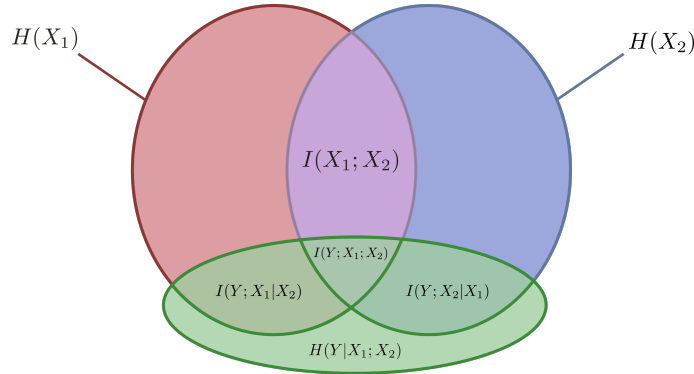


Figure 4.2: Information diagram of two modalities (X_1, X_2) that are mutually complementary for a given target Y . The amounts of information conveyed by X_1 , X_2 and Y are still represented by red, blue and green areas, respectively, while the purple area still represents the amount of information shared by both modalities. While some predictive information is shared by both modalities, *i.e.* $I(Y; X_1; X_2) > 0$, there is modality-specific predictive information, *i.e.* $I(Y; X_1|X_2) > 0$ and $I(Y; X_2|X_1) > 0$, represented by the intersections of red/green and blue/green areas that are outside of the purple area.

In an attempt to preserve modality-specific information, (Y.-C. Liu et al. 2021) propose to directly contrast multimodal input tuples describing the same scene, as opposed to learning a cross-modal embedding space by contrasting distinct modalities. This strategy enables the model to retain unique information associated with each modality. This approach is further refined by (Yunze Liu et al. 2021), who enhance negative sampling and positive sample generation, ensuring equal weight is given to each modality during the process of learning representations. Subsequent research, such as (W. Han et al. 2021; W. Yu et al. 2021), seek to exploit modality-specific information in conjunction with shared information. Nevertheless, these studies primarily rely on backpropagation to leverage this information, rather than exploiting theoretical insights to develop representations that accurately depict and manage the complementarity between modalities.

These work hence aim to leverage modality-specific information through supervision, assuming that predictive information $I(Y; X_1|X_2)$ and $I(Y; X_2|X_1)$ will be retained in the learned representation, facilitated by backpropagation. However, the acquisition of substantial annotated data is costly and not always feasible. In such scenarios, an unsupervised approach is more suitable to effectively and affordably leverage predictive information. The task becomes more challenging when we relax the redundancy assumption, as it becomes harder to differentiate relevant information from noise and superfluous information within modality-specific content.

Besides, by using only backpropagation to guide the learning, there is no real control over the type of information embedded in that representation that aims to leverage modality-specific content,

for instance whether redundant information is also included (W. Han et al. 2021; Y.-C. Liu et al. 2021; Yunze Liu et al. 2021; Wan et al. 2021; W. Yu et al. 2021).

To address this latter limitation, and taking inspiration from (M. Lee et al. 2021) we tried to implement an architecture that aims to build shared and private (*i.e.* modality-specific) representations that are also disentangled. The proposed model was based on a Variational AutoEncoder (VAE) architecture that produced for each bimodal input a shared representation and modality-specific (private) representations. A global learning objective aimed to simultaneously minimize the reconstruction and disentanglement losses. The framework was appealing:

- it leveraged self-supervised framework through reconstruction loss;
- it was motivated by theoretical assumptions, using mutual information estimators for disentangling shared and private representations;
- it would have provided an accessible latent space (to observe learned patterns) and easy sampling process.

Unfortunately we never succeeded in training the model, either the representations did not carry relevant information or they were not disentangled.

The balanced setting of Figure 4.2 might also be unrealistic. There could be situations where one modality significantly influences the prediction due to possessing more information relevant to the target Y . Contrary to the assumption in (Yunze Liu et al. 2021), which aims to give more weight to weaker modalities, the ideal model should prioritize the dominant modality. If we go one step further, we can imagine a setting in which all the predictive information is contained in a single modality, as illustrated in Figure 4.3. In an extreme case, all predictive information could be modality-specific, rendering the other modality superfluous and approaches based on maximizing redundant information ineffective.

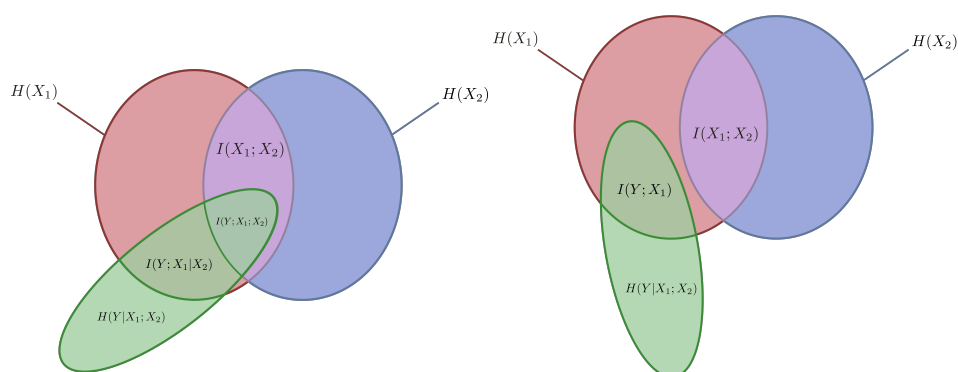


Figure 4.3: Modality-domination setting. In that setting, modality X_1 has a much bigger impact than modality X_2 , which does not encompass any modality-specific predictive information, *i.e.* $I(Y; X_2|X_1) = 0$ (left). In the extreme case, all predictive information is made unavailable from the perspective of X_2 view, that is $I(Y; X_2) = 0$ (right).

On the other hand, we can also envisage a setting in which both modalities are predictive but do not share any predictive information, as illustrated in Figure 4.4.

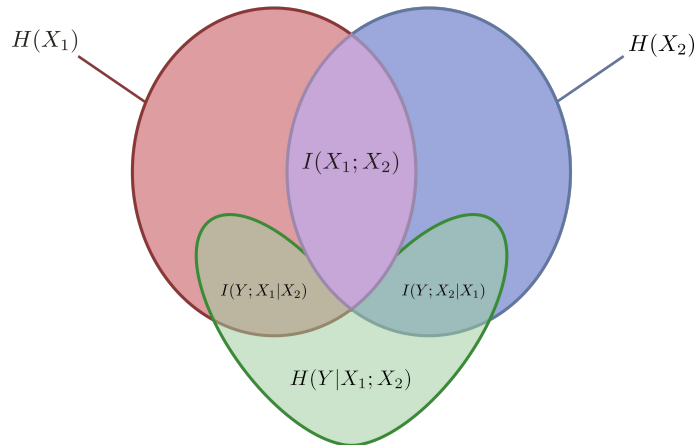


Figure 4.4: Information diagram of two modalities (X_1, X_2) that do not share predictive information for a target Y , *i.e.* $I(Y; X_1; X_2) = 0$. The model shall thus combine modality-specific contents to provide correct prediction.

Observations drawn from these scenarios reveal a potential problem in the approach to multi-modal learning tasks, which might originate from an ill-defined problem statement. While some research has started to leverage modality-specific information, they mostly create representations that are developed through supervised backpropagation. However, the global complementary setting actually encompasses many different configurations, therefore the objective seems ambitious as to design representations that are robust to these different situations only using supervision. This problem is dual, as the considered tasks and related public datasets do not always represent the different situations depicted above. If some recent datasets aims to address tasks that require to combine modality-specific information, such as sarcasm detection (Castro et al. 2019), or multi-modal disambiguation (Talmor et al. 2021), there is no (to our knowledge) public dataset or benchmark that focuses on explicitly evaluating models on their ability to design representations that leverage complementary information across modalities and that are robust to specific configurations depicted above. (P. P. Liang, Y. Lyu, et al. 2021) however show a promising direction by gathering many datasets and related tasks, that for some require the model to have an ability to leverage a certain level of complementarity to perform well.

Finally, the interaction between modalities can fluctuate based on the specific requirements of a task. Some classification tasks only necessitate the additive interaction of data from multiple modalities, as the labeling is dependent on elements that are only jointly available in these modalities. For instance, in the case of identifying a "green pencil" one modality might provide the visual representation of a pencil, while another may furnish the color information, namely, green. Conversely, certain tasks demand a more sophisticated integration of the modalities, *i.e.* a proper

reasoning step. These tasks require the combination of elements from different modalities in an insightful manner that utilizes the content from both. For instance, consider a medical diagnostic AI system that uses three modalities: medical imaging (like CT scans), patient medical history, and real-time vital sign data. The medical imaging modality offers visual evidence of potential physical abnormalities. The patient's medical history provides context on past health issues, family history, etc. The real-time vital sign data delivers immediate health information, like heart rate, blood pressure, and oxygen levels. Diagnosing a complex condition like a lung disease might involve reasoning across all three modalities. A CT scan might reveal a lung nodule, the patient's medical history could indicate a long history of smoking, and the real-time vital sign data might show low oxygen levels in the blood. The AI system must then reason that the lung nodule might be cancerous, potentially exacerbated by the patient's smoking history, and the low oxygen levels could be due to impaired lung function from the cancer. This reasoning process creates a possible diagnosis like "Lung Cancer - Identified through CT scan, corroborated by smoking history and low oxygen levels". This diagnosis involves a nuanced understanding and combination of data across all three modalities.

Very recent research attempts to tackle this complex challenge. For example (P. P. Liang, Yun Cheng, et al. 2023) propose to decompose multimodal interactions into redundancy, uniqueness and synergy. This approach acknowledges the varying complexity of tasks and the different types of interplay that may exist between modalities. The future of multimodal learning research will likely involve further exploration of these dynamics, working towards more sophisticated models that can adaptively handle a range of scenarios and tasks.

CONCLUSION

In this chapter, we delved into the different natures of multimodal interactions, distinguishing the distinct redundant and complementary information. Through a review of contemporary works, we have underscored the importance of maximizing redundant information within the multiview setting, while concurrently highlighting the important role that complementary information plays in multimodal landscape. Nevertheless, current methodologies overwhelmingly rely on backpropagation as the central tool for learning modality-specific representations. This strategy, while effective in certain contexts, tends to undermine the development of truly robust and versatile multimodal representations that can adapt to a wide array of scenarios. The lack of evaluation benchmarks stresses this issue, preventing accurate assessments of models' proficiency in leveraging complementary information.

As a conclusion, the task of adequately leveraging and understanding multimodal interactions remains a formidable challenge. The redundancy-complementarity dichotomy provides a useful lens through which to approach the problem, but it is clear that more sophisticated methods and robust evaluation measures are needed to tackle the diverse and complex nature of multimodal interactions.

From the experiments conducted on StreaMulT architecture in [Subsection 3.5.3](#), textual modality appeared to be the most informative one, as its ablation leads to the biggest per-

performances drop. This observation endorses our hypothesis that the semantics of a precise and detailed textual maintenance report, coupled with the expressive power of high-dimension pre-trained textual encoders, can place the text as the predominate modality for a fault diagnosis task. Thus, in the second part we decide to put a special emphasis on text. In the following chapter, we give the reader some background on NLP research directions, from the classic tasks and architectures, up to recent interest for large foundation models and their application to FSL tasks.

PART II

TOWARDS REALISTIC FEW-SHOT TEXTUAL CLASSIFICATION

5 BACKGROUND AND RELATED WORK IN NLP: FROM SYMBOLIC METHODS TO FOUNDATION MODELS

CHAPTER'S SUMMARY

This chapter offers an overview of Natural Language Processing (NLP) methodologies, up to the development of recent large Foundation Models, and then transitions towards Few-shot learning, a strategy for learning from limited labeled data, before culminating in a discussion of FSL applied to NLP.

The initial section of this chapter outlines the progression of NLP research in understanding human language. This includes early rule-based or feature engineering methods, the utilization of word embeddings to create distributed, meaningful representations, and the development of various architectures for effective Language Models. In [Section 5.7](#), we investigate the prevailing approach to addressing NLP tasks, which involves large pre-trained transformer-based Language Models and their subsequent evolution towards creating versatile central models capable of handling a diverse range of tasks, despite their distinct nature. Finally, we explore in [Section 5.8](#) the realm of Few-Shot Learning, examining its principal techniques and intersection with current NLP paradigms, while shedding light on the latest progress and challenges in this research area.

5.1 INTRODUCTION

Natural Language Processing is a crucial subdomain of computer science and AI, focused on enabling computers to comprehend, interpret, and generate human languages. NLP methods have evolved over the years to handle the messiness of textual data. The primary challenges in NLP indeed stem from the inherent complexity of natural language, which is often ambiguous, context-dependent, and unstructured (Manning and Schütze 2001). To tackle these challenges, NLP encompasses a wide range of tasks:

- low-level tasks, such as tokenization (K. Church et al. 2021), filtering (Manning, Raghavan, et al. 2008), and stemming (Porter 1980), which prepare and process raw text,
- intermediate-level tasks, like part-of-speech tagging (Marcus et al. 1993) and named entity recognition (Bunescu et al. 2005), which analyze and label the data,

- and high-level tasks, including machine translation (Sutskever, Vinyals, et al. 2014), sentiment analysis (Pang et al. 2002), and question answering (Woods 1977), that draw from this analysis to perform complex language understanding and generation.

This section traces the history of NLP advances, beginning with early rule-based methods and feature engineering techniques. We then explore the development of word representation methods, focusing on word embeddings, which have become a crucial component in modern NLP systems. The next part delves into language models, from count-based approaches such as N -grams to neural language models based on RNN, encoder-decoder architectures, and attention mechanisms. Finally, we discuss the recent emergence of transformer-based models and large Foundation models, which bridge the gap between word embeddings and language models by leveraging contextual word representations.

5.2 EARLY NLP METHODS

RULE-BASED AND FEATURE ENGINEERING

Rule-based methods, which originated in the early days of NLP and AI in the 1950s, relied on manually crafted rules and expert knowledge to process and analyze text. These methods were based on a set of predefined linguistic rules or patterns that were applied to the text to extract or manipulate information (William John Hutchins 1986). Some popular rule-based NLP techniques included phrase structure grammars, and context-free grammars (Chomsky 1956). Techniques such as regular expressions and finite-state automata (Mohri 1997) were also used to identify patterns and perform basic text processing tasks, such as tokenization and stemming. Rule-based methods were widely used in early machine translation systems, such as the 1954 Georgetown-IBM experiment (W. John Hutchins 2004), and natural language interfaces (Androutsopoulos et al. 1995; Woods 1977). Rule-based methods have limitations, such as scalability and adaptability to new languages or domains, that respectively require the developments of new and complex rules. The manual creation of rules is time-consuming and requires significant domain knowledge, making these methods less efficient compared to more recent data-driven approaches.

Feature engineering is a process of extracting relevant features from raw data that can be used to build effective ML models. In the context of NLP, feature engineering often involved using expert knowledge to design features based on linguistic properties and domain-specific knowledge (Jurafsky 2000). Part-of-speech (POS) tagging was used as a preprocessing step in early NLP systems, identifying the grammatical role of each word in a sentence (Marcus et al. 1993). This information could then be used as input for other NLP tasks, such as parsing or information extraction. Named entity recognition (NER) is another example of feature engineering in early NLP systems, where the goal is to identify and classify proper nouns, such as people, organizations, and locations, within a text (Bunescu et al. 2005). Dependency parsing extracts the syntactic structure of a sentence by identifying the relationships between words (i.e., subject, object, modifiers). Like POS tagging and NER, dependency parsing was used as a feature in other NLP tasks (Y. Zhang et al. 2011). Similarly to rule-based methods, feature-engineering-approaches face several major limitations, such as the need for time-consuming expert knowledge for designing effective feature

extraction methods, that may not be generalizable to new tasks and do not scale efficiently to large datasets or long sequences. While feature engineering and expert knowledge played a significant role in early NLP tasks, another approach that emerged for handling unstructured textual data was the use of vector space models.

VECTOR SPACE MODELS: BAG-OF-WORDS AND TF-IDF

In these models based on linear algebra, documents and words are represented as vectors (Salton et al. 1975) with the aim of leveraging some similarity between them. Bag of Words (BoW) (Harris 1954) is a simple and widely-used method for representing text data in NLP tasks. BoW converts text into a fixed-size vector by counting the frequency of words in a document and disregarding the order of words. BoW represents each document as a vector with the same length as the vocabulary size. Each element in the vector corresponds to a word in the vocabulary and contains the frequency of that word in the document. The main limitation of BoW is that it ignores word order and contextual information, making it less effective for capturing semantic relationships between words. Additionally, BoW can lead to high-dimensional and sparse representations, which can be computationally expensive for large vocabularies.

Term Frequency-Inverse Document Frequency (TF-IDF) is a technique that extends the BoW approach by incorporating the importance of words in a document relative to their importance in the entire corpus. TF-IDF is calculated as the product of the term frequency (TF) (Luhn 1957), which is the number of times a word appears in a document, and the inverse document frequency (IDF) (Sparck Jones 1972), which is the logarithm of the ratio of the total number of documents in the corpus to the number of documents containing the word. Hence, for a word w and a document d from a corpus C :

$$\begin{aligned} \text{TF-IDF}(w, d, C) &= \text{TF}(w, d) \times \text{IDF}(w, C) \\ &= \text{Card}(\{x \in d \mid x = w\}) \times \log \frac{\text{Card}(C)}{\text{Card}(\{c \in C \mid w \in c\})} \end{aligned}$$

where $\text{Card}(C)$ denotes the cardinality of set C . The IDF weighting scheme assigns higher weights to words that are less frequent in the entire corpus, effectively reducing the impact of common words and emphasizing the importance of more informative words for a given document. Although TF-IDF provides a more sophisticated representation of text data compared to the BoW approach, it still has limitations. Similar to BoW, TF-IDF does not capture word order or contextual information.

While vector space models such as BoW and TF-IDF have proven effective in capturing document-level information and enabling the application of ML techniques to textual data without requiring engineering or expert knowledge, they do not inherently account for the sequential and structured nature of language. To address this shortcoming, researchers have turned to probabilistic frameworks that can model the dependencies and relationships between words in a sequence.

PROBABILISTIC FRAMEWORKS

Probabilistic frameworks, such as Conditional Random Fields (CRFs) and Hidden Markov Models (HMMs), have been widely used in early NLP tasks to model sequences and dependencies between elements in a text. HMMs are generative probabilistic models that represent the joint probability distribution of observed and hidden variables (Rabiner 1989). CRFs, on their side, are discriminative probabilistic models that directly model the conditional probability of the hidden variables given the observed variables (Lafferty et al. 2001). These probabilistic models have been used in tasks like POS tagging, NER, and shallow parsing, among others (Finkel et al. 2005; Sha et al. 2003). While they have proven to be effective in capturing relationships and dependencies in sequential data, some limitations remain, such as their lack of scalability (when dealing with long sequences or datasets, CRF are computationally expensive, whereas HMM struggle in capturing long-range dependencies due to the Markov assumption) or the lack of semantic representation (these models operate at the level of individual words), preventing them to leverage the deep semantic structure of natural language.

Methods	Scalability to large datasets	Adaptability	Expert Knowledge	Robustness to Unknown Words	Dependencies Between Words	Long Sequences Scalability	Semantic Representation
Rule-based	-	-	+	-	-	-	-
Feature-Engineering Based	-	+/-	+	+/-	-	-	-
Vector Space Models (BoW, TF-IDF)	+/-	+	-	-	-	+/-	-
Probabilistic Frameworks (HMMs, CRFs)	-	-	+	+/-	+	-	-

Table 5.1: Summary of limitations of early NLP methods. "+" denotes significant presence/requirement of the criterion, "-" denotes significant lack/limitation, and "+/-" denotes moderate presence/requirement.

TAKEAWAYS

Despite the success of early NLP methods in addressing various language processing tasks, these early techniques struggle in capturing the rich semantic and syntactic information present in natural language. The BoW and TF-IDF models, for example, lack the ability to represent the semantic relationships between words and fail to account for word order, which is crucial for understanding the meaning of a text. Similarly, while probabilistic frameworks like HMMs and CRFs offer a way to model sequences and dependencies,

they still rely on hand-crafted features and do not scale well to large vocabularies or complex dependencies. The limitations of each of these methods are synthesized in Table 5.1. As the field of NLP evolved, researchers recognized the need for better word representations that could capture both the syntactic and semantic information in text. The development of word embeddings, which are continuous vector representations of words, emerged as a promising solution to address these limitations. In the next section, we delve into the world of word embeddings, exploring the various techniques that have been proposed to learn these representations, from count-based to prediction-based methods, and how they have significantly advanced the state-of-the-art in NLP.

5.3 WORD EMBEDDINGS

The limitations of early NLP methods led to the development of word embeddings as a way to better represent and capture semantic and syntactic information about words. Word embeddings are continuous and dense vector representations that map words from a large vocabulary into a lower-dimensional space. These embeddings are based on the distributional hypothesis, which states that words that occur in similar contexts tend to have similar meanings (Firth 1957; Harris 1954)¹. They can be generated using various techniques, broadly categorized into count-based and prediction-based methods.

COUNT-BASED WORD EMBEDDINGS

Count-based word embedding techniques take the idea to put information about contexts into word vectors literally, by manually designing a word-context matrix M in which columns represent potential contexts and rows represent words. In a second step, a dimension reduction technique is applied to the matrix to produce dense embeddings. As their name suggests, these approaches are based on global corpus statistics, and in that sense share some similarities with BoW and TF-IDF. However, those latter methods are not considered as count-based word embeddings because they represent documents rather than individual words and produce sparse vectors instead of dense embeddings.

From there, the different count-based word embeddings strategies differ in the way to consider what is context (hence defining what represent the matrix columns) and how to compute matrix elements. A simple co-occurrence-based approach is for instance to consider as contexts the surrounding words contained in a fixed-size sliding window, and to define M as a word-word matrix with M_{ij} being the number of times word w_i appears in context w_j (Lund et al. 1996). Based on the same definition of contexts, information theoretic measures such as Pointwise Mutual Information (PMI) (K. W. Church et al. 1990) and Positive Pointwise Mutual Information (PPMI) (Bullinaria et al. 2007) have been used to define word representations in matrix M . PMI of a words pair (w_i, w_j) is defined as the log ratio between joint probabilities and product of marginal probabilities: $PMI(w_i, w_j) = \log \frac{P(w_i, w_j)}{P(w_i)P(w_j)}$. Intuitively, designing the matrix M such that

¹Also found as "You shall know a word by the company it keeps"

$M_{ij} = PMI(w_i, w_j)$ will associate positive values to word pairs (w_i, w_j) that appear more frequently in a same context than if they were independent, and negative values to word pairs that appear less frequently than being independent. (Bullinaria et al. 2007) extend this idea by considering only positive values, that is defining $M_{ij} = PPMI(w_i, w_j) = \max(PMI(w_i, w_j), 0)$. Finally, a popular count-based word embedding technique is Latent Semantic Analysis (LSA) (Deerwester et al. 1990). Alternatively, LSA considers different documents from a corpus C as contexts, and hence designs matrix M as a word-document matrix, with $M_{ij} = TF-IDF(w_i, d_j, C)$. The second step is then to reduce the dimensionality of the term-document matrix through a singular value decomposition (SVD) to capture latent semantic relationships between words and documents. By doing so, LSA can identify and represent synonyms, polysemes, and other linguistic relationships in the reduced-dimensional space.

While count-based word embeddings capture dependencies between words and semantic relationships through their term-context matrix, constructing and factorizing such large matrices may undermine their scalability. Besides, count-based models generally struggle with out-of-vocabulary words since they are based on direct observation of the training corpus.

PREDICTION-BASED WORD EMBEDDINGS

Prediction-based word embeddings are generated by training models to predict words or their contexts based on the local context information, which is generally a sliding window surrounding the target word. This approach aims to learn word representations that can effectively capture semantic and syntactic information while exploiting the co-occurrence patterns of words in their local contexts. Two popular prediction-based word embedding techniques are Word2Vec and FastText.

Word2Vec, developed by Mikolov et al., is a highly influential prediction-based word embedding method. Word2Vec consists of two main model architectures: Continuous Bag of Words (CBoW) (Tomás Mikolov, K. Chen, et al. 2013) and Skip-Gram (Tomas Mikolov et al. 2013). CBoW aims to predict the target word based on the surrounding context words, while Skip-Gram focuses on predicting context words given a target word (see Figure 5.1). For both architectures, word vectors are model parameters that are updated along the training through Maximum Likelihood Estimation (MLE) when moving the sliding window along the training corpus and predicting either target word or context words at each position. Word2Vec embeddings have been shown to produce state-of-the-art results on various NLP tasks when released.

FastText (Bojanowski et al. 2017) is an extension of the Word2Vec algorithm that focuses on learning representations for subword units. By representing words at the character scale, FastText can efficiently learn embeddings for rare and out-of-vocabulary words. FastText has been shown to improve performance on a range of NLP tasks, such as text classification (Dharma et al. 2022). Finally, **GloVe**, a popular hybrid approach between count-based and prediction-based techniques has been developed in 2014 (Pennington et al. 2014). GloVe combines the benefits of matrix factorization techniques, like LSA, and local context window-based methods, such as Word2Vec. It constructs a word co-occurrence matrix from a large corpus and uses a weighted least squares objective function to learn word vectors that can effectively capture semantic and syntactic information. It hence captures both global and local context information, allowing for a more com-

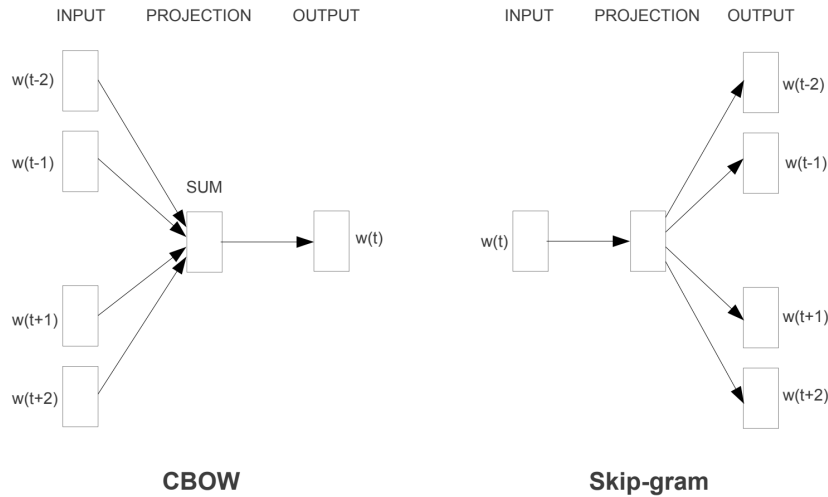


Figure 5.1: Comparison of CBoW and Skip-Gram approaches. CBoW projects context words to predict a central word (left), while Skip-Gram inversely projects a unique word to predicts its context (right). Figure from (Tomás Mikolov, Le, et al. 2013).

prehensive representation of word meaning. However, it requires explicit construction of the co-occurrence matrix, which can be computationally expensive for larger corpora, and it can be sensitive to the choice of hyperparameters, such as the window size and weighting scheme.

Interestingly, using similarity to build rich word representations is not reflected only in quantitative metrics of subsidiary tasks. (Tomas Mikolov et al. 2013) indeed qualitatively analyzed the learned vector space and pointed out geometrical patterns based on meanings similarity (see Figure 5.2). Thus, the difference between the representation vectors of many country/capital pairs seem to produce the same vector. Another example (Tomás Mikolov, Le, et al. 2013) shows the similar distribution of embedding vectors from a language to another one, suggesting a simple linear mapping for translation.

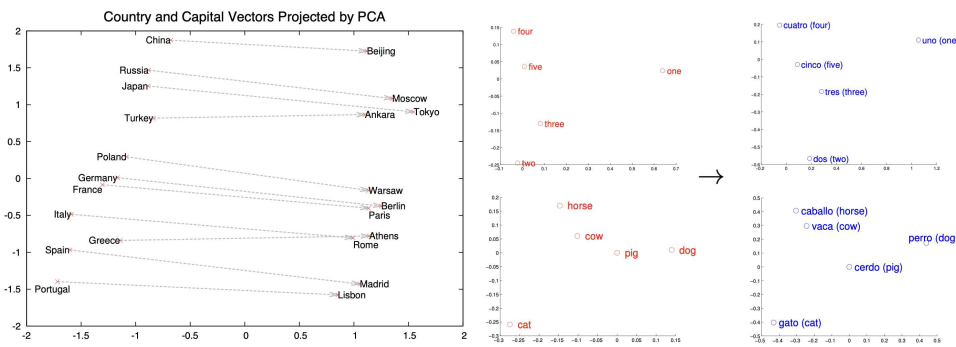


Figure 5.2: Qualitative results for Word2Vec embeddings. Subtracting capital vector to its related country vector produces similar vector among all country/capital pairs (left). Learned embeddings of number and animal words have very similar spatial distribution in English and Spanish (right).

Methods	Scalability to large datasets	Adaptability	Expert Knowledge	Robustness to Unknown Words	Dependencies Between Words	Long Sequences Scalability	Semantic Representation	Context-dependent representations
Rule-based	-	-	+	-	-	-	-	-
Feature-Engineering Based	-	+/-	+	+/-	-	-	-	-
Vector Space Models (BoW, TF-IDF)	+/-	+	-	-	-	+/-	-	-
Probabilistic Frameworks (HMMs, CRFs)	-	-	+	+/-	+	-	-	-
Count-Based Word Embeddings	+/-	+	-	-	+	+/-	+	-
Prediction-Based Word Embeddings	+	+	-	+	+	+/-	+	-

Table 5.2: Summary of limitations of early NLP methods and word embeddings techniques. "+" denotes significant presence/requirement of the criterion, "-" denotes significant lack/limitation, and "+/-" denotes moderate presence/requirement.

TAKEAWAYS

Word embeddings have become an essential tool in NLP, capturing semantic and syntactic relationships between words and providing a foundation for more advanced techniques. However, despite their ability to capture word relationships, word embeddings have limitations, particularly in representing context-dependent word meanings. Indeed, these representations are pre-computed in a static corpus, which may not be convenient when using a word in a different context afterwards (this is notably the case for polysemous words that have in this framework only one representation). Besides, long sequences can be handled well as the window size can be varied, but distant dependencies might be missed. The comparison of approaches is thus updated in [Table 5.2](#).

We now delve into language models, which offer a comprehensive approach to capture the structure and context of language. Their development have led to powerful and versatile models capable of handling complex linguistic phenomena and significantly improving performance on a wide range of tasks, such as machine translation, speech recognition, and text generation.

5.4 LANGUAGE MODELS

Language models play a critical role in various NLP tasks by predicting the likelihood of a sequence of words, represented as a probability distribution over words. Given a sequence of words

(w_1, w_2, \dots, w_n) , a language model assigns a probability $\mathbb{P}(w_1, w_2, \dots, w_n)$ to this sequence. This can be used for numerous applications such as machine translation (Bahdanau et al. 2015; Koehn et al. 2003; Sutskever, Vinyals, et al. 2014), speech recognition (G. Hinton et al. 2012; Jelinek 1991), and text generation (Graves 2013). In this section, we explore the evolution of language modeling techniques, from early count-based approaches to more sophisticated neural models that have driven significant advances in the field of NLP.

COUNT-BASED LANGUAGE MODELS

The early days of language modeling were dominated by count-based methods, with N -gram models being one of the most widely-used approaches (Jelinek 1991). N -grams are simply contiguous sequences of N words, where N is a fixed integer. An N -gram language model predicts the probability of a word given its preceding $N - 1$ words by estimating the frequency of N -grams in a large corpus. Thus, an N -gram model makes a Markov assumption, which states that the probability of a word depends only on the previous $N - 1$ words:

$$\mathbb{P}(w_n | w_1, \dots, w_{n-1}) \approx \mathbb{P}(w_n | w_{n-N+1}, \dots, w_{n-1})$$

N -gram probabilities $\mathbb{P}(w_n | w_{n-N+1}, \dots, w_{n-1})$ can be estimated by counting in a corpus the occurrences of N -gram $(w_{n-N+1}, \dots, w_{n-1}, w_n)$ and normalizing by the number of occurrences of $(w_{n-N+1}, \dots, w_{n-1})$.

Despite their simplicity, N -gram models suffer from several limitations, such as data sparsity, which occurs when certain N -grams do not appear in the training corpus, leading to inaccurate probability estimates. To overcome this issue, various smoothing techniques have been proposed (S. F. Chen et al. 1996). Other drawbacks of N -gram models are their inability to capture long-range dependencies, as they only consider a fixed number of preceding words to predict the next word, or the curse of dimensionality they may face when considering large vocabulary (Bengio, Ducharme, et al. 2000).

While count-based language models have provided a foundation for early NLP research, their limitations have led to the development of more advanced techniques such as neural language models (Bengio, Ducharme, et al. 2000), that afterwards leveraged the power of deep learning to better understand and represent natural language.

NEURAL LANGUAGE MODELS

Neural language models aim to provide a continuous representation of words and capture semantic and syntactic information in dense vector space. They have demonstrated their ability to overcome some of the limitations of count-based language models, such as the curse of dimensionality and the sparsity of N -grams. One of the first neural language models was a feedforward neural network (FFN) language model (Bengio, Ducharme, et al. 2000). This model aimed to predict the next word in a sequence by concatenating word embeddings of previous words and feeding them into the FFN. The output models the word probability given a context. The model's architecture is illustrated in Figure 5.3.

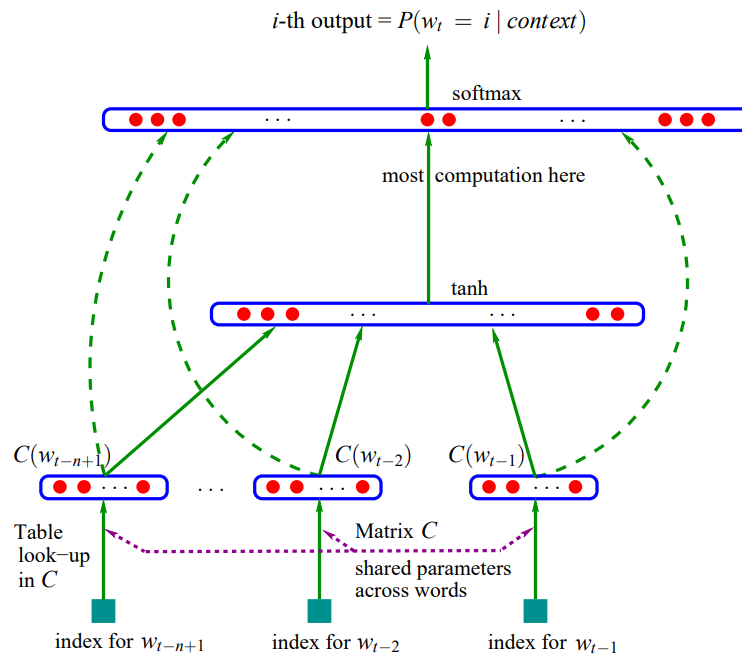


Figure 5.3: Neural Language Model architecture. The input sentence $(w_{t-n+1}, \dots, w_{t-1})$ is converted to feature vectors stored in a matrix C , which are then fed to a neural network g represented by the green plain lines. The output of g estimates the probability of each word in the vocabulary, conditioned the input context. Figure from (Bengio, Ducharme, et al. 2000).

Recurrent Neural Networks were introduced as an extension to feedforward neural language models to better capture long-range dependencies in natural language data (Elman 1990). RNNs are designed to process sequences of variable length by maintaining a hidden state that can store information from previous time steps (Tomás Mikolov, Karafiát, et al. 2010). However, RNNs have some limitations, such as the vanishing gradient problem that makes learning long-range dependencies difficult (Hochreiter, Bengio, et al. 2001). To overcome the vanishing gradient problem in RNNs, **Long Short-Term Memory (LSTM)** networks were proposed (Hochreiter and Schmidhuber 1997). LSTMs introduce a gating mechanism that helps to maintain and propagate information over long sequences, making them more effective for learning long-range dependencies. LSTMs have thus been used as building blocks for Language Models (Sundermeyer et al. 2012). Finally, **Gated Recurrent Units (GRU)** are another variant of RNNs that simplify the LSTM architecture while retaining its ability to model long-range dependencies (Cho et al. 2014). GRUs use update and reset gates to control the flow of information in the hidden state, making them computationally more efficient than LSTMs, however they may not capture long-term dependencies as well as LSTM.

5.5 ENCODER-DECODER ARCHITECTURE

Many NLP tasks require not only an understanding of the input text but also the generation of a meaningful output sequence, such as in neural machine translation and text summarization. To tackle these challenges, a new class of models has emerged: encoder-decoder architectures, also known as **sequence-to-sequence models** (Sutskever, Vinyals, et al. 2014). The encoder-decoder architecture is composed of two main components: the encoder and the decoder. The encoder processes the input sequence and generates a fixed-length context vector that encapsulates the essential information of the input. The decoder, in turn, takes this context vector and generates an output sequence, conditioned on the input sequence. These architectures split the model into two parts, with one component (the encoder) focusing on processing the input sequence and the other (the decoder) generating the output sequence (Cho et al. 2014). In the early encoder-decoder models, both the encoder and decoder were typically implemented as RNNs, LSTMs, or GRUs. The encoder processes the input sequence one token at a time, updating its hidden state at each step. The final hidden state of the encoder is then used as the initial hidden state of the decoder, which generates the output sequence one token at a time. An illustration of this family of architectures is given in Figure 5.4.

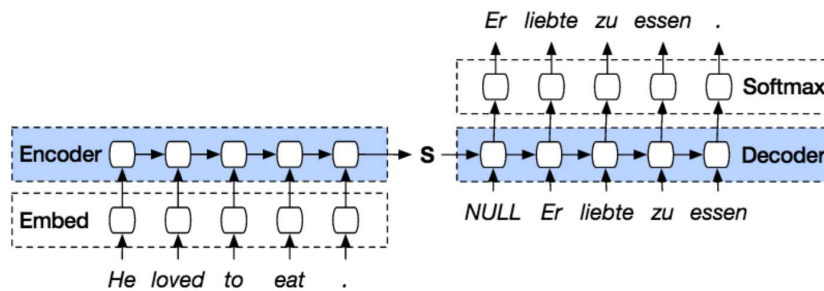


Figure 5.4: Sequence-to-Sequence architecture². Every words of the input sentence are embedded and then sequentially fed to the encoder module, that stores the input information in a context S . Using this context and the previous generated token (starting with a special token), the decoder module sequentially generates the output.

While the encoder-decoder architecture was a significant improvement over the previous models, it still faced some limitations. One of the main challenges was that the encoder had to compress the entire input sequence into a single fixed-size context vector, which could result in loss of information, especially for long input sequences (Bahdanau et al. 2015). This limitation prompted researchers to explore more sophisticated ways to better capture and leverage the information in the input sequence, leading to the development of attention mechanism.

ATTENTION MECHANISM

The key idea behind attention mechanism (Bahdanau et al. 2015) is that the decoder should be able to focus on different parts of the input sequence at different time steps, rather than relying

²Figure from <https://www.guru99.com/seq2seq-model.html>

solely on a single context vector. This allows the model to weight the importance of different input tokens and selectively retrieve information from the input sequence. In an attention-based encoder-decoder model, the encoder produces a sequence of hidden states, one for each input token. The decoder, at each time step, computes a weighted sum of these hidden states, where the weights are determined by the attention mechanism. These weights, also known as attention scores, indicate how much the decoder should "attend" to each input token when generating the output token at a given time step. The attention mechanism computes attention scores using a scoring function that takes as input the current hidden state of the decoder and the hidden states of the encoder. There are several variants of the scoring function, such as dot product, additive, and multiplicative attention (T. Luong et al. 2015). The introduction of attention mechanisms significantly improved the performance of encoder-decoder models on a wide range of NLP tasks, including neural machine translation (Bahdanau et al. 2015), text summarization (Rush et al. 2015), and speech recognition (Chorowski et al. 2015). The success of attention mechanisms in these tasks paved the way for further advancements in NLP, such as the development of transformers.

5.6 TRANSFORMERS

Despite the success of attention mechanisms in improving the performance of encoder-decoder models, researchers continued to explore ways to further enhance the capabilities of NLP models. One significant drawback of the RNN-based models was their sequential nature, which makes it difficult to parallelize the computations and exploit the full potential of modern hardware, such as GPUs. In response to this challenge, (Vaswani et al. 2017) introduced the Transformer architecture, which replaces the recurrent layers in encoder-decoder models with self-attention mechanisms. This groundbreaking innovation has become the foundation for many state-of-the-art models in NLP, including BERT (Devlin et al. 2019), GPT (Radford, Narasimhan, et al. 2018), and their variants, as well as in other domains (vision (Dosovitskiy et al. 2021), speech (Radford, Kim, T. Xu, et al. 2022), etc.).

The self-attention mechanism is at the core of the Transformer architecture. Unlike the attention mechanism used in encoder-decoder models, self-attention operates within a single sequence, allowing each token to attend to all other tokens in the sequence. This mechanism enables the model to capture long-range dependencies more effectively and allows for parallel computation across tokens. See [Subsubsection 2.2.3](#) for a more detailed overview of the self-attention mechanism. The Transformer architecture is built upon a stack of self-attention layers and feed-forward layers, with residual connections and layer normalization applied throughout the model. The original Transformer model proposed in (Vaswani et al. 2017) consists of an encoder and a decoder, similar to the earlier encoder-decoder models. The encoder is composed of a stack of identical layers, each containing a multi-head self-attention mechanism followed by a position-wise feed-forward network. The decoder has a similar structure, with an additional layer of cross-attention that attends to the encoder's output. The global architecture is presented in [Figure 5.5](#).

Transformers can also be designed as standalone encoders or decoders for various NLP tasks, depending on the nature of the problem and the desired model architecture. For instance, BERT (Devlin et al. 2019) is built upon a stack of Transformer encoder layers, while GPT (Radford, Narasimhan, et al. 2018) uses a stack of Transformer decoder layers. Using only the encoder part of the Trans-

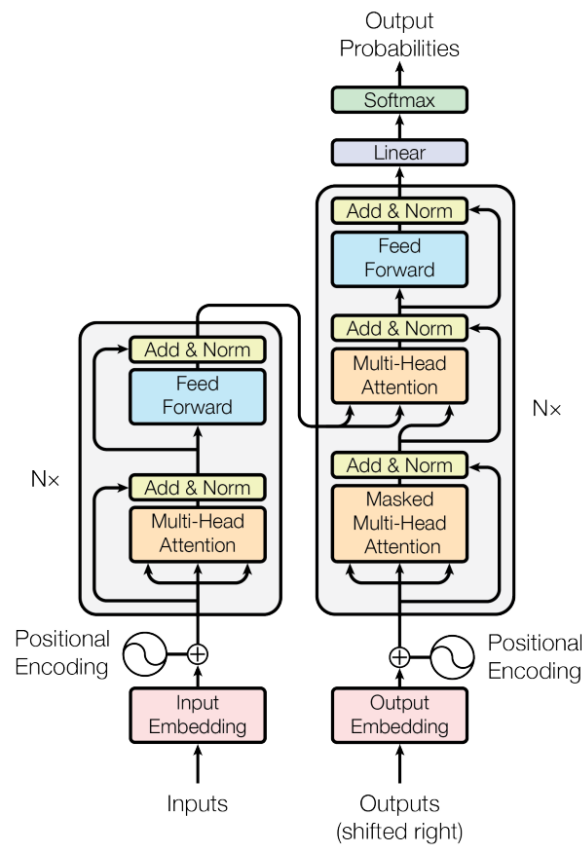


Figure 5.5: Original Transformer architecture. Similarly to encoder-decoder models, the embedded input is first encoded in a specific module before the decoder module generates the output autoregressively. The main difference is the use of Self-attention modules that make possible to model contextual dependencies between all parts of the sequences. The masking process in the decoder modules enables to parallelize the training. Figure from (Vaswani et al. 2017).

former architecture can be more suitable for tasks that require a fixed-length representation of the input sequence, such as sentence classification. The Transformer encoder processes the input sequence and produces a contextualized representation for each token, which can be aggregated or pooled to generate a fixed-length vector. On the other hand, using only the decoder part of the Transformer can be advantageous for tasks that involve generating text or predicting the next token in a sequence, such as language modeling, text generation, and summarization. The Transformer decoder is designed to handle autoregressive decoding, where the model generates one token at a time and feeds the generated tokens back as input for the subsequent steps. This architecture enables the model to leverage the self-attention mechanism for capturing dependencies between generated tokens, while still benefiting from the parallelizability and efficient handling of long-range dependencies offered by the Transformer architecture.

Methods	Scalability to large datasets	Adaptability	Expert Knowledge	Robustness to Unknown Words	Dependencies Between Words	Long Sequences Scalability	Semantic Representation	Context-dependent representations
Rule-based	-	-	+	-	-	-	-	-
Feature-Engineering Based	-	+/-	+	+/-	-	-	-	-
Vector Space Models (BoW, TF-IDF)	+/-	+	-	-	-	+/-	-	-
Probabilistic Frameworks (HMMs, CRFs)	-	-	+	+/-	+	-	-	-
Count-Based Word Embeddings	+/-	+	-	-	+	+/-	+	-
Prediction-Based Word Embeddings	+	+	-	+	+	+/-	+	-
Count-based Language Models	+/-	+/-	-	-	+/-	-	-	-
Recurrents Neural Networks (LSTM,GRU)	+	+	-	+	+	+/-	+	+
Transformers	+	+	+	+	+	+	++	++

Table 5.3: Summary of advantages and limitations of general NLP methods and word embeddings techniques. "+" denotes significant presence/requirement of the criterion, "-" denotes significant lack/limitation, and "+/-" denotes moderate presence/requirement.

TAKEAWAYS

Driven by the diverse requirements of NLP tasks and the inherent pursuit of comprehending and generating human language automatically, numerous frameworks and methodologies have been pursued and refined, successively diminishing the constraints of preceding methods (see Table 5.3). The advent of word embedding methods marked a significant milestone, providing dense, vector-based semantic representations that proved invaluable for a multitude of downstream tasks.

Recurrent Neural Networks, particularly LSTM, advanced this paradigm by capturing distributed, contextually-dependent representations via their hidden state. They led to the introduction of a new architectural framework: the Encoder-Decoder model. This approach is exceptionally suitable for tasks requiring contextual generation, such as machine translation.

The colossal breakthrough came with the advent of Transformer models, inspired by the Encoder-Decoder architecture and the introduction of the Attention Module. These models offer outstanding semantic and context-aware representations through their self-attention module, directly capturing all types of dependencies across sequence elements, rather than compressing pertinent information within a hidden state as is the case with

LSTM. Furthermore, the ability of Transformer models to parallelize efficiently permits impressive scaling, aligning seamlessly with the capabilities of modern hardware. This has resulted in Transformers becoming the cornerstone for the vast majority of today’s architectural designs in NLP and other applications of Deep Learning.

5.7 FOUNDATION MODELS

Transformers have significantly impacted the field of NLP, and their introduction came with a change of paradigm in the field. Rather than using an end-to-end supervised framework composed of task-specific neural networks, most works in the recent years follow the pre-training and fine-tuning paradigm to achieve state-of-the-art performance across a wide range of NLP tasks. This has today led to the Foundation models era, that aim to unify all kind of NLP tasks within a single architecture.

Remark. Following the Center for Research on Foundation Models of Stanford University³, we refer to Foundation models (Bommasani et al. 2021) as the following: “In recent years, a new successful paradigm for building AI systems has emerged: Train one model on a huge amount of data and adapt it to many applications. We call such a model a foundation model.”. These models are based on Pre-trained Language Models (PLMs) architectures (see thereafter), and as they become larger and larger, are often referred to as Large Language Models. The interchange of these terms is hence frequent in the literature.

PRE-TRAINING AND FINE-TUNING PARADIGM

The pre-training and fine-tuning paradigm has emerged as a successful approach for building Pre-trained Language Models in NLP. The idea is to first train a large neural network (mainly transformer-based one) on a massive amount of unsupervised text data (such as the C4 dataset (Raffel, Shazeer, et al. 2020)), and then fine-tune the pre-trained model on a specific supervised task (Howard et al. 2018; Peters et al. 2018). This approach leverages the ability of DL models to learn rich and meaningful representations from large-scale data, which can then be adapted to specific tasks with relatively small amounts of labeled data (see Figure 5.6).

Transfer learning is a key concept underlying the pre-training and fine-tuning paradigm. It refers to the process of transferring knowledge learned in one task or domain to another, usually related, task or domain (S. J. Pan et al. 2010). In NLP, transfer learning has been shown to be highly effective, as the knowledge learned from large-scale unsupervised text data can be generalized to a wide range of tasks (Ruder et al. 2019). The benefits of transfer learning in NLP are numerous. Firstly, it allows for more efficient learning and better generalization, as the pre-trained model has already learned meaningful language representations (Bengio, Courville, et al. 2013). Secondly, it reduces the need for labeled data in the target task, as the pre-trained model can be fine-tuned with relatively small amounts of labeled data (Peters et al. 2018). Finally, it leads to faster convergence

³<https://crfm.stanford.edu/>

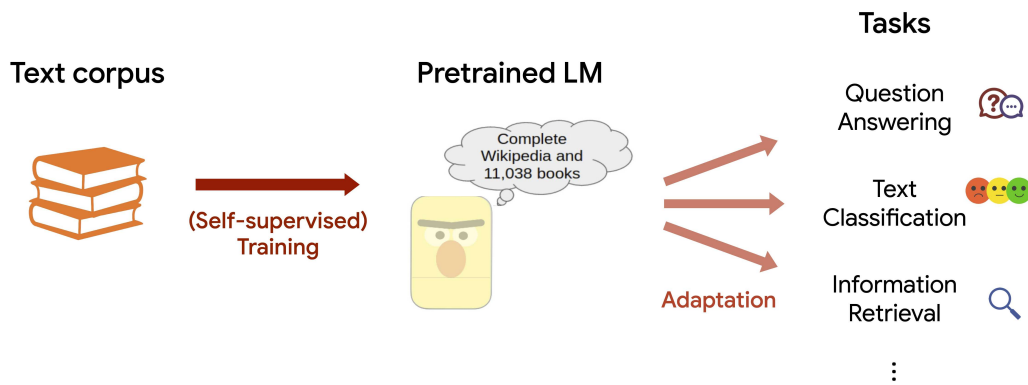


Figure 5.6: Pre-training and fine-tuning paradigm.⁴ Large Language Models are first trained in an unsupervised fashion on massive textual corpora, and then fine-tuned on a specific supervised dataset for a related task.

and improved performance, as the model can leverage the knowledge learned during pre-training (Howard et al. 2018; Ruder et al. 2019).

PIONEERING WORKS: PRE-TRAINED LANGUAGE MODELS TO PRODUCE CONTEXTUAL WORD REPRESENTATIONS

As we discussed, Foundation models aim to acquire a vast amount of knowledge by pre-training on massive unsupervised corpora. The choice of pre-training tasks and associated losses is therefore crucial in enabling these models to gain the general linguistic knowledge necessary for effective downstream task performance. By carefully designing the pre-training objective, we can encourage the model to learn valuable patterns, structures, and relationships within the data that can be effectively transferred to a wide range of downstream tasks. In this context, pre-training losses play a pivotal role in guiding the learning process of foundation models and shaping their ability to generalize and adapt to various NLP challenges.

In the initial stages, **ELMo** (Peters et al. 2018) was developed to obtain context-sensitive word representations by first pre-training a bidirectional LSTM (biLSTM) network (rather than acquiring fixed word representations). Subsequently, the biLSTM network was fine-tuned to cater to particular downstream tasks.

BERT (Devlin et al. 2019) is a powerful model based on the Transformer encoder architecture. BERT is pre-trained on a large corpus of text using a **Masked Language Modeling** (MLM) objective, which enables it to learn bidirectional contextual representations. In this objective, a certain percentage of the input tokens are randomly masked (literally replaced by a MASK token), and the model is trained to predict the original token based on the context provided by the surrounding unmasked tokens. The MLM loss is calculated by comparing the predicted probabilities for the masked tokens with the true tokens using cross-entropy. This objective allows BERT to learn

⁴Figure from <https://ai.stanford.edu/blog/linkbert/>

deep bidirectional representations, capturing both the left and the right context of each token. BERT is also pre-trained using a Next Sentence Prediction (NSP) loss, in which the model shall predict if a sequence is subsequent to another one (but the NSP loss appeared to have low impact on performance). (Yamaguchi et al. 2021) explored other cheaper pre-training objectives, similar to MLM, and showed comparable performance (see Figure 5.7). Context-aware word representations of BERT and its variants (such as RoBERTa (Yinhan Liu et al. 2019)) have demonstrated state-of-the-art performance on a wide range of NLP predictive tasks, such as sentiment analysis, named entity recognition, and question-answering. Fine-tuning BERT on task-specific datasets allows it to adapt its powerful pre-trained representations to the target task, often with minimal additional training.

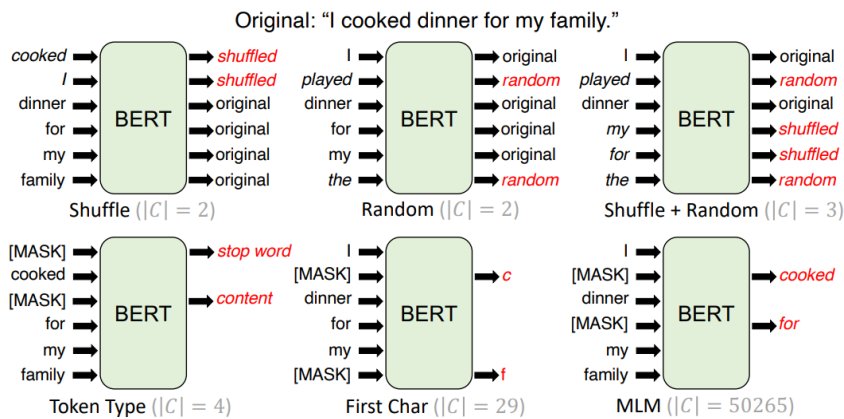


Figure 5.7: Masked Language Modeling and similar pre-training objectives. In each scenario, $|C|$ represents the number of classes of the pre-training objective, which considerably impacts computational efficiency. Figure from (Yamaguchi et al. 2021).

GPT (Radford, Narasimhan, et al. 2018) is another significant milestone in contextual word representations. GPT models are based on the Transformer decoder architecture and are pre-trained using a unidirectional autoregressive **Language Modeling** (LM) objective. The primary goal of GPT is to predict the next token in a sequence given its preceding context. The LM loss is computed by comparing the predicted probabilities for the next token in the sequence with the true next token using cross-entropy. The unidirectional nature of GPT allows it to learn powerful contextual representations, capturing the left context of each token. However, due to their autoregressive loss, these models are especially suitable for generative tasks such as dialogues and document summarization. There have been several iterations of the GPT model, with GPT-2 (Radford, J. Wu, et al. 2019) and GPT-3 (Brown et al. 2020), especially differing by their sizes, both in number of parameters and training corpora. More recently, GPT-4 (OpenAI 2023) was released, once again crushing its previous version size with now 1 trillion (10^{12}) parameters, and now being multimodal, as it can process both text prompts and images as input. Like BERT, GPT models can be fine-tuned on task-specific datasets to adapt their pre-trained representations to the target tasks.

Conditional Language Modeling (CLM) objective is another type of pre-training loss used in some foundation models. Unlike standard LM loss used in GPT, which focuses on predicting the next word in a sequence given the previous words, or the MLM loss used in BERT that concentrates on predicting randomly masked words within a sentence, the CLM loss aims at reconstructing the input sequence after a specific kind of perturbations. A prominent encoder-decoder architecture that employs CLM objective is T5 (Raffel, Shazeer, et al. 2020), that adopts a text-to-text transfer learning approach, where both input and output sequences are represented as text strings. It is pre-trained on a denoising autoencoder task, which involves reconstructing the original text from a corrupted version. During pre-training, T5 introduces noise to the input text by applying transformations such as token masking or deletion. The model then learns to recover the original input sequence from the perturbed version. By learning to reconstruct the original sequence, T5 captures bidirectional context and adapts well to various NLP tasks. Another notable architecture that uses CLM loss is BART (Lewis et al. 2020). BART also adopts a denoising autoencoder setup, applying transformations such as token masking, token deletion, or text shuffling. The combination of bidirectional context and autoregressive nature allows both T5 and BART to excel in a wide range of tasks, taking advantage of both LM and MLM frameworks.

The different pre-training objectives are listed in Table 5.4. For each objective, the considered network aims to model the conditional probability p . It can be trained with maximum likelihood estimation.

Objective	Loss
MLM	$\mathcal{L}_{MLM} = - \sum_{\tilde{w} \in m(\mathbf{w})} \log p(\tilde{w} \mathbf{w}_{\setminus m(\mathbf{w})})$
LM	$\mathcal{L}_{LM} = - \sum_{t=1}^T \log p(w_t \mathbf{w}_{<t})$
CLM	$\mathcal{L}_{CLM} = - \sum_{t=1}^T \log p(w_t \tilde{\mathbf{w}}, \mathbf{w}_{<t})$

Table 5.4: Pre-training objectives and their respective loss functions for a sentence $\mathbf{w} = (w_1, \dots, w_T)$. $\mathbf{w}_{<t} := (w_1, \dots, w_{t-1})$, while $m(\mathbf{w})$ designs masked words of \mathbf{w} , $\mathbf{w}_{\setminus m(\mathbf{w})}$ designs the unmasked elements of \mathbf{w} and $\tilde{\mathbf{w}}$ designed corrupted sentence.

In summary, the introduction of PLM have revolutionized the field of NLP, providing general-purpose contextual word representations that have significantly improved performance across various tasks. Building on this success, following works developed larger architectures to still improve performances on downstream tasks.

LARGE LANGUAGE MODELS

Several studies (Hoffmann et al. 2022; Kaplan et al. 2020; Rosenfeld et al. 2020) have demonstrated the advantages of scaling up language models in terms of model size, dataset size, and computational resources, by introducing scaling laws in terms of loss reduction. This led to the emergence of **Large Language Models** (LLMs). LLMs, typically composed of Transformer-based architec-

Model	Architecture	Pre-training Loss	Corpus
ELMo	LSTM	biLM	WikiText-103
GPT	Transformer Decoder	LM	BookCorpus
BERT	Transformer Encoder	MLM & NSP	WikiEn+BookCorpus
RoBERTa	Transformer Encoder	MLM	BCOS
BART	Transformer	CLM	BCOS
T5	Transformer	CLM	C4

Table 5.5: Overview of different Transformer-based models. BCOS stands for BookCorpus+CCNews+OpenWebText+STORIES. biLM is a bidirectional LM loss.

tures with hundreds of billions or more parameters, are trained on extensive text datasets. These scaled-up models, despite adopting similar Transformer architectures and pre-training objectives as smaller PLMs, benefit significantly from increased model size, data size, and computational power. Over the last years, several tech resource-rich organizations launched their own LLM, with for instance Google’s PaLM (Chowdhery et al. 2022) and LaMDA (Thoppilan et al. 2022), OpenAI’s GPT-4 (OpenAI 2023), DeepMind’s Chinchilla (Hoffmann et al. 2022), or Meta’s LLaMA (Touvron et al. 2023). In parallel, a team of researchers released BLOOM (Scao et al. 2022), a 176B-parameter open-access language with the aim to make this kind of models publicly accessible. Figure 5.8 provides an overview of the main LLM released over the last years.

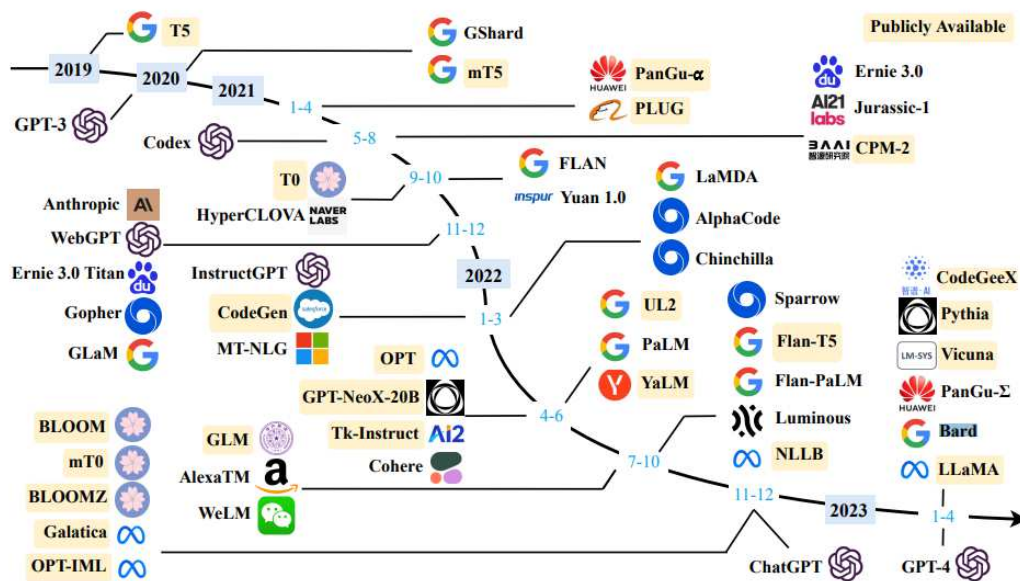


Figure 5.8: Timeline (left-to-right) of the released LLMs (bigger than 10B parameters) over the last years. The models marked in yellow are the ones made available for public use. The figures along the timeline represent the month of release. Figure from (W. X. Zhao et al. 2023).

Interesting learning abilities

LLMs exhibit strong capacities to understand natural language, generate text, and display emergent abilities, that "are not present in small models but arise in large models" (Wei, Tay, et al. 2022). These abilities include In-context learning (ICL) and Instruction formatting.

Introduced by GPT-3 (Brown et al. 2020), ICL allows language models to generate outputs at test time, given demonstrations of a task, without requiring additional fine-tuning or gradient updates. While the 175B GPT-3 model exhibits strong ICL abilities, the GPT-1 and GPT-2 models do not.

Besides, when fine-tuned on multi-task datasets using instructions (natural language descriptions), LLMs show considerable performance on unseen tasks that are also described by instructions (Ouyang et al. 2022; Sanh et al. 2022), without necessarily giving the model explicit examples, improving generalization abilities. Some studies (H. W. Chung et al. 2022; Wei, Bosma, et al. 2022) showed that this phenomenon induced by instruction-formatting essentially appears once a sufficient size has been reached. Some models such as Galactica (R. Taylor et al. 2022) even include Instruction formatting within the pre-training stage to achieve superior performance and better generalization capacity.

These emergent abilities are illustrated in Figure 5.9.

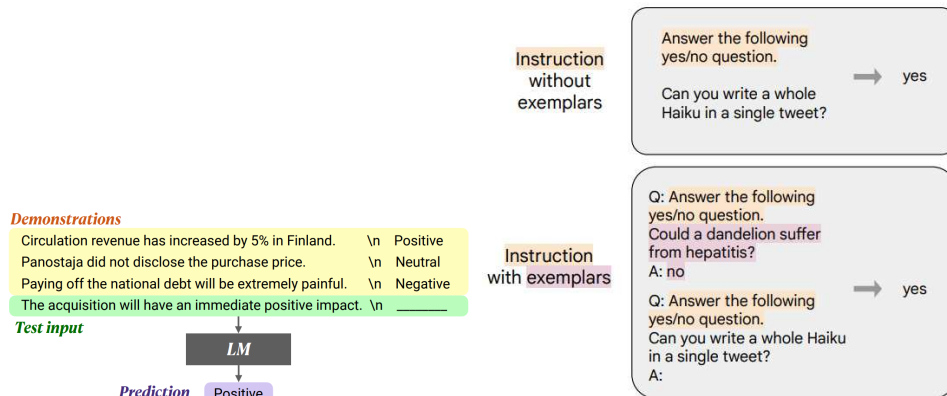


Figure 5.9: In-context learning (left): The model is given a prompt containing k input-label pairs (here $k = 3$) alongside with a test input (in the same prompt), and is asked to predict in response the test label. The model leverages the information contained in the demonstrations to effectively generate the label with no gradient update. Figure from (S. Min, X. Lyu, et al. 2022).

Instruction fine-tuning (right): The model is fine-tuned by providing Natural language descriptions of the task in preamble. It can also contain labeled examples in the prompt (bottom). Figure from (H. W. Chung et al. 2022).

Some limitations

Whereas LLMs have demonstrated impressive performance across a broad spectrum of NLP tasks, they sometimes produce unexpected outputs, or hallucinations, that may cause harm or mislead the user. To prevent this behavior, the concept of human alignment has been introduced to ensure LLMs outputs align with human expectations (Glase et al. 2022; Ouyang et al. 2022). Reinforcement learning from human feedback (RLHF) (Christiano et al. 2017; Ziegler et al. 2019) for instance uses a policy-gradient RL algorithm to adjust LLMs based on human feedback. The integration

of human preferences via instructions, combined with training on both code and natural text segments, resulted in the development of the GPT-3.5 series. After undergoing a conversation-like training process, the widely-adopted chatbot ChatGPT was introduced, significantly influencing future AI research and underscoring the potential of human-like AI systems. Google similarly then released their chatbot BARD, aligned on human preferences with their own instruction fine-tuning method FLAN (Wei, Bosma, et al. 2022). Anthropic’s Claude chatbot has on its side been aligned with human moral behavior using a technique called Constitutional AI (Bai et al. 2022), providing a principle-based approach to produce harmless outputs.

TAKEAWAYS

Large Language Models have made remarkable strides in the field of NLP by employing the **pre-training and fine-tuning paradigm**. This approach has enabled these models to achieve impressive results on a wide range of NLP tasks, even though the tasks themselves are quite diverse. While these models are yet subject to **hallucinations**, human **alignment** appeared as first step to ensure more control on their output. However, the fine-tuning process needs sizable labeled datasets for adapting the model to a new task, given the significant number of parameters involved. The challenge of gathering annotated data is amplified by the expenses involved and the scarcity of such data across different languages and domains. Consequently, there is a pressing need to develop effective methods for learning with limited annotated data. In parallel, LLMs show **emergent abilities**, such as **In-Context Learning**, that may be suitable for addressing this challenge. This leads us to the next section, which focuses on Few-Shot Learning (FSL) techniques for NLP.

5.8 FEW-SHOT LEARNING IN NLP

FEW-SHOT LEARNING PARADIGM

Few-Shot Learning (FSL) refers to the ability to learn tasks with limited annotated examples. This ability of humans, that are able to use their previous experience to adapt fastly to new context, has been largely studied recently in the context of machine learning algorithms (Lake et al. 2015). As illustrated in Figure 5.10, it can concern many tasks: classification, generation, etc.

Historically, **Meta-learning** -or learning to learn (Thrun et al. 1998)- approaches have for quite long stood as the *de-facto* paradigm for FSL (K. Lee et al. 2019; A. Raghu et al. 2020; A. Rusu et al. 2019; Snell et al. 2017; Q. Sun et al. 2019; Sung et al. 2018). Meta-learning refers to the process of improving a learning algorithm with multiple learning episodes (**episodic training**). These learning episodes are a distribution of tasks and not data samples. This improved learning ability has then been applied to the FSL realm. For instance, MAML (Antoniou et al. 2019; Finn et al. 2017), arguably the most popular meta-learning method, tries to train a model such that it can be fine-tuned end-to-end using only a few supervised samples while retaining high generalization ability.

Meta-learning approaches are mainly divided into **optimization-based**, **model-based**, or **metric-based**. **Optimization-based meta-learning** methods focus on finding an optimal initialization

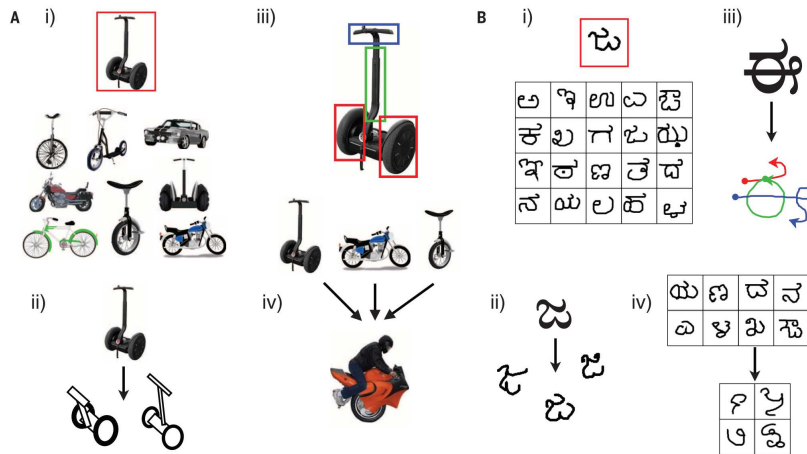


Figure 5.10: Few-shot learning paradigm. The objective is to leverage information from one or few annotated examples in order to perform many downstream tasks such as classification (i), generation of new examples (ii), segmentation and parsing (iii), new concepts generation (iv). Figure from (Lake et al. 2015).

of model parameters, such that they can be fine-tuned efficiently with minimal supervision data (Finn et al. 2017; Ravi et al. 2017). **Model-based approaches** involve learning a model that can generate or adapt parameters for new tasks with the help of limited examples, often by using memory-augmented networks or modular architectures (Graves et al. 2014; N. Mishra et al. 2017). Lastly, **metric-based methods** rely on learning a similarity metric between instances, such that classification can be performed by comparing the relationships between few-shot examples and new instances in a latent space (Snell et al. 2017; Vinyals, Blundell, et al. 2016). Semi-supervised learning methods with few annotations also contribute to the FSL landscape, combining a small amount of labeled data with a larger pool of unlabeled data to improve performance on specific tasks (Oliver et al. 2018; Rasmus et al. 2015).

The majority of these methodologies have primarily been developed and tested within the realm of computer vision. Nonetheless, certain articles have shown that straightforward techniques rooted in transfer learning can competently compete with meta-learning approaches (Jiaxin Chen et al. 2020; Y. Tian, Yue Wang, et al. 2020). As a result, a significant number of modern investigations are centered around the **pre-training and efficient fine-tuning paradigm** as a means of developing effective methods for FSL (Jiaxin Chen et al. 2020). Similarly, in state-of-the-art NLP, FSL is predominantly executed through strategies that harness the power of Pre-trained Language Models.

FEW-SHOT LEARNING FOR NLP TASKS USING LARGE LANGUAGE MODELS

A significant body of research has addressed the challenge of FSL in NLP by leveraging Pre-trained Language Models (PLMs) (Devlin et al. 2019; Yinhan Liu et al. 2019; Radford, J. Wu, et al. 2019; Zhilin Yang et al. 2019). These approaches can be broadly categorized into three primary groups: **parameter-efficient tuning**, **prompt-based learning**, and **in-context learning**. Parameter-efficient tuning aligns with methods in the field of computer vision, introduced at the end of

previous paragraph, drawing heavily on the principles of transfer learning. On the other hand, the approaches of prompt-based learning and in-context learning are specific to the domain of NLP. They innovatively restructure tasks into natural language "prompts" and take advantage of Pre-trained Language Models (PLMs) to fill in these prompts.

Parameter-efficient tuning: These methods, such as adapters (Houlsby et al. 2019) have emerged as a promising solution for transfer learning and FSL in NLP tasks. These approaches involve adding lightweight, task-specific adapter layers to pre-trained transformer models, which allow for fine-tuning on limited labeled data while keeping the majority of the pre-trained model's parameters fixed (see Figure 5.11). Examples of such methods include AdapterHub (Pfeiffer et al. 2020), a framework for adapting transformers, and (D. Guo et al. 2021), referred to as "Diff-Pruning", accomplishing a similar objective by incorporating a sparse, task-specific difference vector to the original parameters. Moreover, in some cases, fine-tuning just a small fraction of the pre-trained model has proven to be effective. For instance, BitFit (Ben Zaken et al. 2022) only fine-tunes the bias parameters, which account for less than 1% of the total model parameters, yet it achieves competitive results on downstream tasks. More recently, T-FEW (Haokun Liu et al. 2022) proposed an approach consisting in adding learned vectors that rescale the network's internal activations.

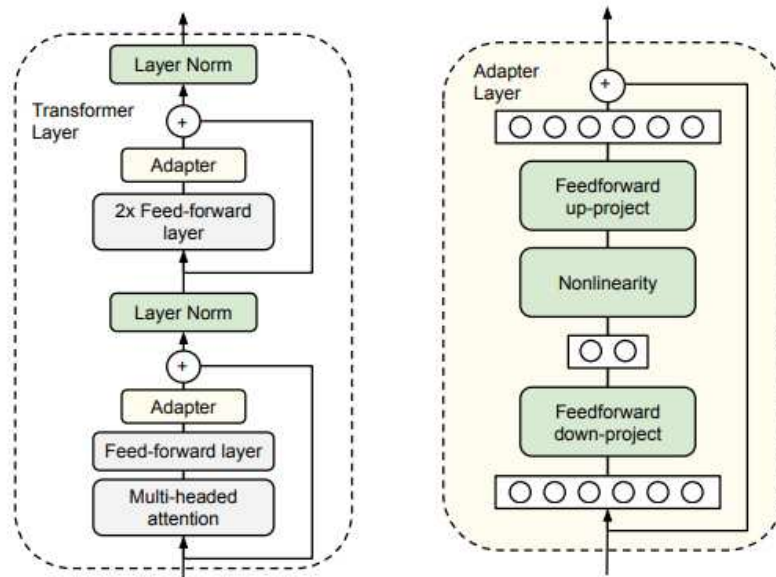


Figure 5.11: Adapter architecture (right) and its integration in Transformer (left). The Adapter consists in few-parameter modules that are inserted after Transformer FFN. When fine-tuning the modified architecture on a downstream tasks, only green modules (within Adapter and Layer Normalization) are updated. Figure from (Houlsby et al. 2019).

Prompt-Based Few-Shot Learning: In recent years, Pre-trained Language Models (PLMs) have been used to solve FSL tasks in NLP, notably using a prompting strategy. The idea is to frame the task as a language modeling problem by designing a template that guides the model towards generating a desired output. The seminal work (Schick et al. 2020) formalizes the prompt setting

by defining the template as pattern-verbalizer pairs, in which the pattern is a function mapping a set of input sentences to a cloze question. Verbalizers, on the other hand, are injective functions that map discrete labels into natural language phrases or tokens. This association leverages the generation capability of PLMs to perform classification tasks using a template, allowing the classification task to be formatted in a way that is intelligible to the PLM (Ding et al. 2022; P. Liu et al. 2023). This framework is illustrated in Figure 5.12. By varying the patterns and verbalizers, it is then possible to annotate a larger unlabeled dataset with soft labels, on which a classic classifier will be fine-tuned.

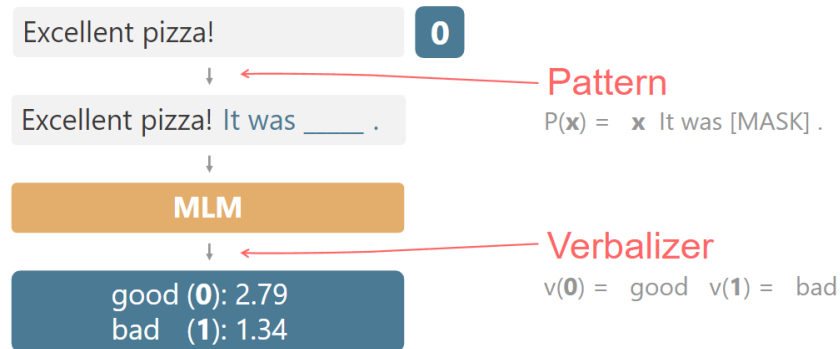


Figure 5.12: Prompt-based few-shot learning. The considered objective is to classify the input sentence "Excellent pizza!" as good or bad. The pattern P is first transforming the input as a cloze question $P(x)$. $P(x)$ is then fed to a PLM that outputs prediction scores for the masked word. Eventually, the verbalizer v converts the token prediction scores as classification logits.⁵

In-Context Learning: GPT3 (Brown et al. 2020), GPT4 (OpenAI 2023) and related chatbot ChatGPT based on InstructGPT model (Ouyang et al. 2022) showed that PLMs were also efficient for in-context FSL tasks. In this setting, the prompt is composed of the task description, but also some support input examples with their corresponding outputs and a query input with the objective to predict the query output (Wei, Xuezhi Wang, et al. 2022). ICL hence requires no parameter update, produces a new prediction model for each new prompting, and therefore quickly adapts to a new task (see Figure 5.9).

INDUCTIVE VS TRANSDUCTIVE FEW-SHOT LEARNING

Learning an inductive classifier on embeddings generated by a pre-trained model, as proposed by (Snell et al. 2017), is a common baseline for performing FSL. This approach is prevalent in NLP, where a parametric model is trained on data to infer general rules that are applied to label new, unseen data (known as inductive learning (V. N. Vapnik 1999)). However, in FSL scenarios with limited labeled data, this approach can be highly ambiguous and lead to poor generalization. Transduction offers an attractive alternative to inductive learning (Sain 1996). Unlike inductive learning, which infers general rules from training data, transduction involves finding rules that work specifically for the unlabeled test data. By utilizing more data, such as unlabeled test instances, and

⁵Figure from <http://timoschick.com/explanatory%20notes/2020/10/23/pattern-exploiting-training.html>

aiming for a more localized rule rather than a general one (see Figure 5.13), transductive learning has shown promise and practical benefits in FSL for computer vision (Dhillon et al. 2020; Y. Guo et al. 2020; R. Hou et al. 2019; S. X. Hu et al. 2020; Y. Hu et al. 2021; J. Liu et al. 2020; Yanbin Liu et al. 2019; Yaoyao Liu et al. 2020; Qiao et al. 2019; Veilleux et al. 2021; Yikai Wang et al. 2020; Ling Yang et al. 2020; Ziko et al. 2020).

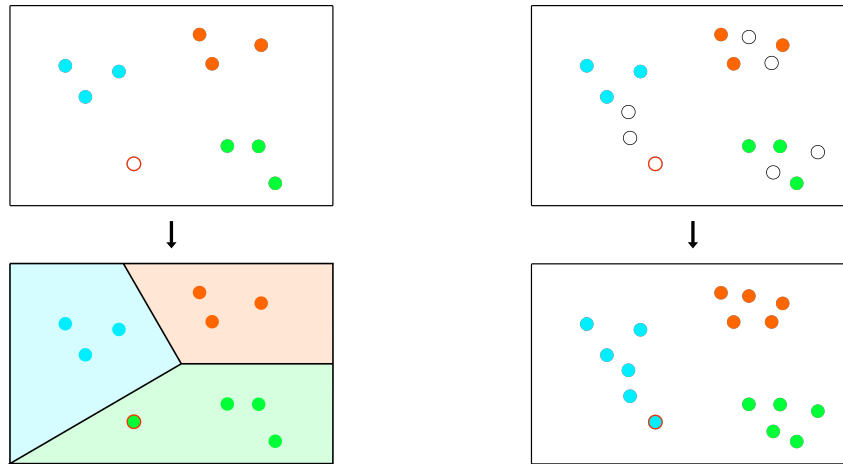


Figure 5.13: Inductive vs transductive settings. In the inductive setting (left), the model aims to learn general rules from labeled data, that will then serve to classify all unlabeled test samples, one by one. In the transductive setting (right), the model leverages information from both labeled data and all available unlabeled samples to adapt its classification to these samples. In this example, the same datapoint represented by a red circle is not classified the same way by the two approaches.

Transductive methods yield substantially better performance than their inductive counterparts by leveraging the statistics of the unlabeled data (such as batch normalization statistics (Nichol et al. 2018)). While (R. Hou et al. 2019; Yanbin Liu et al. 2019) use graphs or cross-attention modules to perform label propagation from support to query samples, other main strategies consist in minimizing the entropy of query samples predictions (Dhillon et al. 2020), using prototype rectification (J. Liu et al. 2020), Laplacian regularization (Ziko et al. 2020), optimal transport (Y. Hu et al. 2021), or maximizing Mutual Information measures (Boudiaf et al. 2020; Y. Guo et al. 2020; Veilleux et al. 2021). However, despite their success experienced in the vision community, this framework has not yet been explored in the context of textual data.

CONCLUSION

In conclusion, this chapter provided a comprehensive overview of the evolution and current state of NLP, delving into the various methodologies and techniques that have shaped the field. We began with early NLP approaches, including rule-based methods, vector space models, and probabilistic frameworks, before moving on to the groundbreaking de-

velopment of word embeddings that significantly advanced the state-of-the-art. The chapter then explored the emergence of language models and the attention mechanism, which have led to the transformative introduction of transformer architectures.

Large PLMs have revolutionized NLP by providing general-purpose contextual word representations that have greatly improved performance across a wide range of tasks. The pre-training and fine-tuning paradigm has proven highly successful, and has further pushed the boundaries of what is possible in NLP. However, these advancements based on the scaling paradigm require huge computational resources and available annotated data for fine-tuning. To handle this challenge, an interest in Few-shot Learning for NLP has grown. If universal efficient transfer-learning-based have been explored, new NLP-specific FSL paradigms have been developed, based on natural language prompts, and leveraging PLMs generation ability. Yet, they may not be suitable for realistic assumptions. A possible solution could be the use of transductive paradigm, that has not been explored in NLP. This is the main focus of [Chapter 6](#).

6 A TRANSDUCTIVE APPROACH FOR PERFORMING FEW-SHOT CLASSIFICATION IN NLP

CHAPTER'S SUMMARY

In this chapter, we explore the potential of transductive methods for textual classification in the context of few-shot learning, aiming to address the limitations of current FSL methods in NLP, specifically the engineering efforts required for realistic classification tasks with a large number of classes. We first discuss the limitations of current FSL methods, such as prompt-based strategies or in-context learning. Then, in [Section 6.3](#) we explore the application of transductive approaches, which have shown promising results in computer vision, to NLP classification. Finally, in [Section 6.4](#) we evaluate the performance of traditional transductive regularizers in comparison to inductive techniques on textual few-shot classification tasks and investigate the impact of different factors, such as the number of backbone parameters and fine-tuning strategies, on the performance of transductive methods. The results indicate that transductive methods have difficulty outperforming inductive cross-entropy-based fine-tuning when there is some flexibility in the pre-trained feature extractor parameters. However, by fixing all parameters of the feature extractor, the transductive approach finally rivals the inductive one.

6.1 INTRODUCTION

As discussed in previous chapter, Few-Shot Learning (FSL) has gained significant attention in the field of NLP due to its ability to rapidly adapt to new tasks using limited labeled data. Current FSL methods, such as prompting and ICL, have demonstrated promising results in a wide range of NLP tasks. However, as the complexity of the classification problem grows, especially in cases with a large number of classes, these methods are confronted with inherent limitations, such as the need for extensive engineering to achieve practical results. This chapter aims to address these limitations by exploring the potential of transductive methods for textual classification in the context of few-shot learning. Transductive methods, which have been successfully applied in other domains, offer a promising alternative to traditional FSL techniques by leveraging the structure of the input data to make predictions for the unseen data points. By adapting these methods for textual tasks, we seek to harness their potential to tackle the challenges posed by the ever-increasing complexity and scale of classification problems in NLP, hence meeting more realistic assumptions.

CHAPTER'S CONTRIBUTIONS

The primary contributions of this chapter are three-fold:

- We provide an analysis of the limitations of current FSL methods in NLP, specifically in terms of the engineering efforts required for realistic classification tasks with a large number of classes, and we formulate the textual few-shot classification problem.
- We propose a novel adaptation of transductive methods for textual classification in the context of FSL, enabling effective utilization of limited labeled data.
- We present a series of research questions and their related experiments conducted to validate or rebut the effectiveness of our proposed methods, comparing their performance to the inductive techniques in FSL for NLP.

6.2 PROBLEM STATEMENT

The main assumption of FSL in modern NLP paradigm supposes the availability of a large pre-trained backbone model. The objective is to leverage this model's learned representations to adapt to a novel classification task when only a handful of annotated samples are at our disposal.

6.2.1 CURRENT METHODS LIMITATIONS

While previous works on NLP-FSL present promising results, they mainly focus on datasets with a reduced number of classes (*i.e.* always less than 10 classes and often less than 5 classes) (Mahabadi et al. 2022; Perez et al. 2021). However, when considering realistic setting, a few-shot classifier shall be able to classify among much more unseen classes, or to have a generalization ability that makes it prone to quickly adapt to a new set of classes. Under this consideration, current NLP-FSL strategies face practical limitations:

- Using a prompt-based approach demands a cumbersome handcraft engineering to design every Pattern-Verbalizer pairs. Thus, recent studies have questioned the benefits of prompt-based learning due to the high variability in performance caused by the choice of prompt (Haokun Liu et al. 2022). As the number of classes increases, crafting appropriate prompts and verbalizers becomes increasingly difficult, and the resulting prompts may not be equally effective for all classes. This can lead to a performance degradation in complex classification problems. Besides, this engineering is mainly validated on held-out labeled examples, which could not be available in general (Perez et al. 2021). The prompting setting is therefore hardly scalable for tasks with realistic settings. To cope with these limitations, recent NLP-FSL approaches try to alleviate the importance of template design (Logan IV et al. 2022), or to break with prompt paradigm (Fei et al. 2022).
- Several works have shown that in-context-learning design, along with the choice and ordering of training samples, is highly sensitive and not robust to the choice of PLM (Y. Lu et al.

2022; Z. Zhao et al. 2021). Second, as the number of classes increases, the need for longer contexts to provide sufficient examples for all classes can exceed the maximum input length of the models. This can result in the truncation of important information or the inability to adequately represent the full range of classes. These drawbacks prevent the usage of such strategy for realistic NLP-FSL tasks.

- Finally, parameter-efficient tuning methods shall be considered on a case-by-case basis. While T-FEW (Haokun Liu et al. 2022) additionally requires a set of manually created prompts for each dataset making it hard to use in practice, Diff-Pruning (D. Guo et al. 2021) considers an inconsistent set of parameters that change values across different tasks, which may prevent us to use it on highly variable number of test classes for hardware practical reasons. Nonetheless, some approaches such as (Houlsby et al. 2019), or BitFit (Ben Zaken et al. 2022) (consisting in fine-tuning only bias terms in transformer-encoder layers) seem not to present specific drawback for our setting, hence we will compare the latter with transductive approaches in the conducted experiments.

6.2.2 TEXTUAL CLASSIFICATION IN FEW-SHOT SETTING

In response to the constraints inherent in NLP-specific methodologies such as prompt-based and ICL strategies, we propose using the episodic framework popularized by meta-learning and mostly used to formalize few-shot learning setting in computer vision, and we adapt it to the NLP paradigm.

Let Ω be the considered vocabulary, we denote Ω^* its Kleene closure. The Kleene closure corresponds to sequences of arbitrary size written with tokens in Ω , *i.e.*, $\Omega^* = \bigcup_{i=0}^{\infty} \Omega^i$. Given an input space \mathcal{X} with $\mathcal{X} \subseteq \Omega^*$, a latent space \mathcal{Z} and a label space \mathcal{Y} , we consider a pre-trained backbone model $g_{\theta} : \mathcal{X} \rightarrow \mathcal{Z} = \mathbb{R}^d$, where $\theta \in \Theta$ represents the parameters of the encoder and d is the embedding dimension size.

The objective of few-shot classification is to learn a classifier $h_{\phi} : \mathcal{Z} \rightarrow \mathcal{Y}$ from limited labeled data and generalize to new, unseen tasks or classes. To accomplish this, we consider transfer-learning-based strategies that are evaluated on an **episodic testing** setting. In such setting, randomly sampled few-shot tasks are created from a test dataset $\mathcal{D}_{test} := \{(x^i, y^i)\}_{i=1}^{N_{test}}$ that has a set of classes \mathcal{Y}_{test} , unseen by the backbone during pre-training. To follow the nomenclature of the FSL literature, each few-shot classification task is defined by the number of targeted classes K and is composed of a support set S and a query set Q .

For each class $1 \leq k \leq K$, N_S labeled samples from the class k are randomly sampled from \mathcal{D}_{test} to compose S , while N_Q different and unlabeled samples from the class k are randomly sampled from \mathcal{D}_{test} to compose Q . Thus, $S = \{x^i, y^i\}_{i \in \mathcal{I}_S}$ with $Card(S) = N_S \times K$, and $Q = \{x^i\}_{i \in \mathcal{I}_Q}$, with $Card(Q) = N_Q \times K$. \mathcal{I}_S and \mathcal{I}_Q represent the drawn indices during the sampling process for support set and query set, respectively. The task is thus named a N_S -shot K -way task. Pre-trained models use few-shot techniques and the labeled support sets to adapt to

the tasks at hand and are evaluated based on their performances on the unlabeled query sets. This setting is illustrated in Figure 6.1 for a computer vision application.

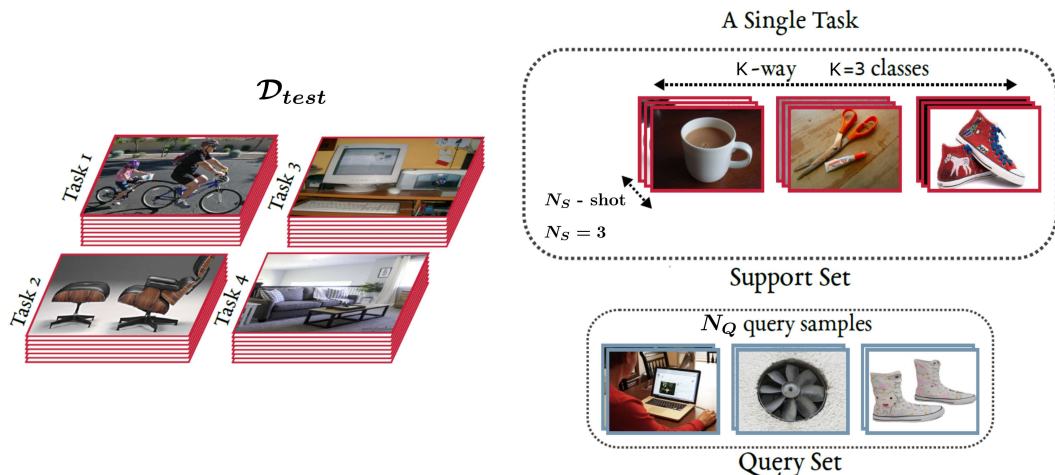


Figure 6.1: 3–shots 3–ways tasks example for a computer vision task. Figure from (Ouali 2023).

Remarks.

- Contrary to the works of computer vision, there is no necessary distinction between the dataset used to pre-train the backbone g_θ and the test dataset \mathcal{D}_{test} . Indeed, as the current pre-training corpora are mostly composed of pages of the entire internet (or a large part of it), it seems difficult to check that the model did not see test samples during pre-training stage. However, in NLP the backbone is pre-trained using self-supervised objectives (rather than supervised tasks), therefore there is no risk of overlap between pre-training and testing tasks.
- Episodic testing is slightly different than the original episodic training introduced in meta-learning approaches. In the latter, a single model is incrementally trained or fine-tuned on the different tasks, improving its robustness and generalization task after task. Differently, we use the episodic setting as an evaluation protocol, meaning that a different model is initialized for each generated few-shot task, and all tasks are compiled independently in parallel. This approach allows to compute more reliable performance statistics by evaluating the generalization capabilities of each method on a more diverse set of tasks. Finally, as we want to evaluate the performance of NLP-FSL approaches for larger number of classes, in this very chapter we fix the number of ways to be equal to the number of classes of the test dataset, *i.e.* $K = Card(\mathcal{Y}_{test})$.

6.3 TRANSDUCTIVE APPROACHES FOR FSL IN NLP

To alleviate the drawbacks of few-shot approaches using prompting strategies, and especially the extensive manual engineering needed for designing all verbalizers for multiclass classification, we

explore transductive approaches that achieved promising results in computer vision community such as TIM (Boudiaf et al. 2020), and their application to NLP classification.

Specifically, we train a classification head $h_\phi : \mathcal{Z} \rightarrow \mathbb{R}^K$ mapping the representations features to the posterior distribution space to perform prediction. To simplify the equations for the rest of the paper, we use the following notations for the posterior predictions of each $i \in \mathcal{I}_S \cup \mathcal{I}_Q$ and for the class marginals within Q :

$$p_k^i = h_\phi(g_\theta(x^i))_k = \mathbb{P}(Y = k | X = x^i; \theta, \phi) \quad \text{and} \quad \hat{p}_k = \frac{1}{|Q|} \sum_{x^i \in Q} p_k^i = \mathbb{P}(Y_Q = k; \theta, \phi)$$

where X and Y are the random variables associated with the raw features and labels, respectively, and where Y_Q means restriction of the random variable Y to set Q .

The global classifier $f_{\phi^*, \theta^*} = h_{\phi^*} \circ g_{\theta^*}$ is obtained by simultaneously training the classification head and fine-tuning the feature extractor such that they solve the following objective:

$$(\phi^*, \theta^*) = \arg \min_{\phi, \theta} \text{CE} - \lambda \times R_Q \quad (6.1)$$

with $\text{CE} := -\frac{1}{|S|} \sum_{i \in \mathcal{I}_S} \sum_{k=1}^K y_k^i \log(p_k^i)$ being the cross-entropy supervision on the support set (in

which y_k^i is the k^{th} coordinate of the one-hot encoded label vector associated to sample i) and R_Q being a transductive loss on the query set Q . The exact definition of R_Q depends on the transductive approach. It is worth noting that transductive regularization has been introduced in literature, grounded in the InfoMax principle (Cardoso 1997; Linsker 1988). In the upcoming paragraph, we provide an overview of the transductive techniques presented in prior works.

Entropic Minimization An effective regularizer for transductive FSL can be derived from the field of semi-supervised learning, drawing inspiration from the approach introduced in (Grandvalet et al. 2004). This regularizer, proposed in (Dhillon et al. 2020), utilizes the conditional Shannon Entropy (Cover 1999) of forecast results from query samples during testing to enhance model generalization. Formally:

$$R_Q^H = \frac{1}{|Q|} \sum_{i \in \mathcal{I}_Q} \sum_{k=1}^K p_k^i \log(p_k^i) \quad (6.2)$$

Mutual Information Maximization A promising alternative to the entropic minimization for addressing the challenges of transductive FSL is to adopt the Info-max principle. (Boudiaf et al. 2020) extended this idea, introduced in (W. Hu et al. 2017), and proposed as regularizer a surrogate of the mutual-information $R_Q^I(\beta)$:

$$R_Q^I(\beta) = -\sum_{k=1}^K \hat{p}_k \log \hat{p}_k + \beta \frac{1}{|Q|} \sum_{i \in \mathcal{I}_Q} \sum_{k=1}^K p_k^i \log(p_k^i) \quad (6.3)$$

$$= \hat{\mathcal{H}}(Y_Q) + \beta(-\hat{\mathcal{H}}(Y_Q | X_Q)) \quad (6.4)$$

where $\mathcal{H}(\hat{Y}_Q)$ and $-\mathcal{H}(Y_Q|X_Q)$ are Monte-Carlo estimators of the marginal entropy of the query set and the negative conditional entropy over labels given features on the query set, respectively. Hence the maximization of the second term (when minimizing $-R_Q^I(\beta)$) in Equation 6.1 makes the classifier more confident, making its posterior distribution more spiky, while the maximization of the first term prevents the model to degenerate by always predicting the same class. The balance between the two terms of the loss is controlled by the hyperparameter β .

The α -TIM method (Veilleux et al. 2021) extends the TIM setting by considering imbalanced datasets, hence non-uniform labels distributions. The corresponding $R_Q^{I_\alpha}$ loss is in that sense based on empirical Tsallis α -entropy \hat{H}_α rather than on Shannon entropy:

$$R_Q^{I_\alpha} = \frac{1}{\alpha - 1} \left(\frac{1}{|Q|} \sum_{i \in \mathcal{I}_Q} \sum_{k=1}^K (p_k^i)^\alpha - \sum_{k=1}^K \hat{p}_k^\alpha \right) \quad (6.5)$$

$$= \hat{H}_\alpha(Y_Q) - \hat{H}_\alpha(Y_Q|X_Q) \quad (6.6)$$

(Veilleux et al. 2021) empirically show that using estimators of Tsallis entropy is indeed better suited to handle imbalanced classes than Shannon entropy.

We finally compare these methods with an inductive baseline:

Linear probing The inductive baseline loss can be obtained by assigning $\lambda = 0$. We refer to this approach as Linear Probing: fine-tuning a linear head on top of a pre-trained model is a popular approach to learn a classifier for various classification tasks and was originally proposed in (Devlin et al. 2019).

6.4 EXPERIMENTAL STUDY OF TRANSDUCTIVE FEW-SHOT INFERENCE FOR NLP CLASSIFICATION

In this section we describe the experimental protocol and results to compare the performances of these different transductive methods for the task of few-shot text classification in realistic settings.

6.4.1 LIMITATIONS OF EXISTING BENCHMARKS

Previous studies on textual few-shot classification (Gao, Fisch, et al. 2021; Mahabadi et al. 2022; Schick et al. 2021; Schick et al. 2022; Tam et al. 2021) have predominantly assessed their algorithms on classification tasks with a restricted number of labels (typically less than five). The statistics of mostly used datasets in these works are depicted in Table 6.1. Real-world problems yet often comprise larger multi-class classification tasks, which could undermine current FSL methods due to the

significant required handcraft engineering. We take a step forward and consider datasets that are more representative of real-world scenarios. Hence, we decided to run our tests on the following datasets:

- Tweet eval (Barbieri et al. 2020) contains english tweets annotated with 20 different emojis.
- Banking77 (Casanueva et al. 2020) contains online banking customer service queries annotated with their intents, distributed among 77 classes.

Dataset	Task Description	Number of Classes
BoolQ	Binary Classification	2
CB	Natural Language Inference	3
COPA	Choice of Plausible Alternatives	2
WiC	Word-in-context	2
WSC-DistilBERT	Coreference Resolution	2
SST-2	Sentiment Analysis	2
SST-5	Sentiment Analysis	5
MR	Sentiment Analysis	2
CR	Sentiment Analysis	2
MPQA	Opinion Polarity Detection	2
Subj	Subjectivity/Objectivity Analysis	2
TREC	Question Classification	6
CoLA	Linguistic Acceptability	2
MNLI	Natural Language Inference	3
SNLI	Natural Language Inference	3
QNLI	Question Answering/Natural Language Inference	2
RTE	Natural Language Inference	2
MRPC	Paraphrase Detection	2
QQP	Duplicate Question Detection	2
AG's News	News Category Classification	4
Yelp Reviews Full Star	Sentiment Analysis	5
Yahoo Questions	Topic Classification	10
Tweet_eval	Emoji prediction	20
Banking77	Customer queries Classification	77

Table 6.1: Overview of the various datasets.

6.4.2 RESEARCH QUESTIONS AND RELATED RESULTS

RQ1: Do transductive methods improve few-shot classification performances over classic transfer learning?

To answer this question, we trained different transductive methods presented in Section 6.3, and we compare their performances with the linear probing inductive baseline (by setting $\lambda = 0$

in Equation 6.1). The different plots in Figure 6.2 represent the classification accuracy on test set for different values of N_S , consider $K = 20$ classes. These specific plots correspond to the Tweet_eval dataset, with BERT as the pre-trained backbone, and a classification head composed of two linear layers (768×768 and 768×20) separated by a *relu* activation. For each bar, the accuracy is averaged on 5 different seeds and a 95% confidence interval is given.

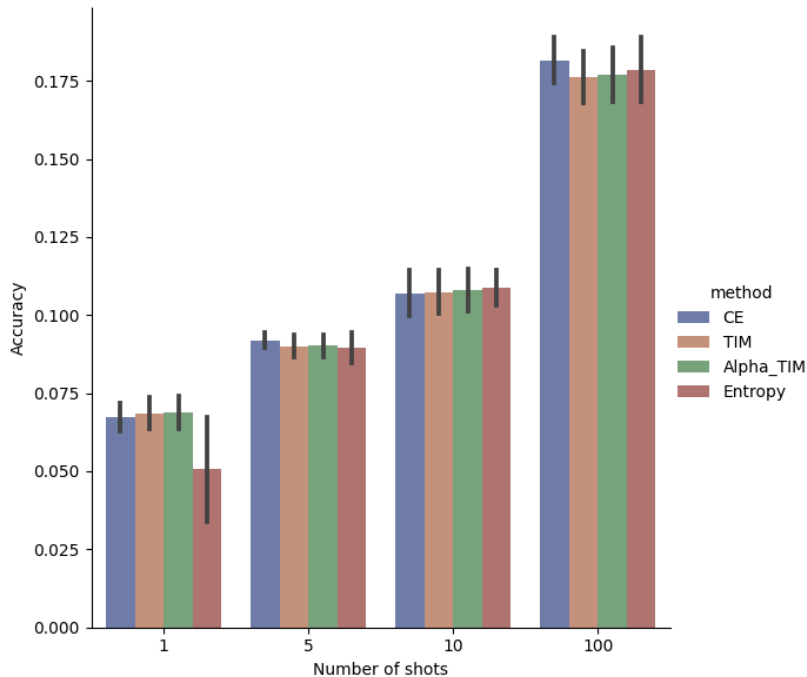


Figure 6.2: Comparison of cross-entropy-based and transductive-based approaches for different N_S values on Tweet_eval dataset. We consider $K = 20$ classes.

While the method consisting in minimizing Shannon entropy for conditional output distributions struggles to compete with other strategies for one-shot setting, none of the presented approaches clearly has an edge over the other ones and especially not significantly on the inductive baseline consisting in fine-tuning a classification head with cross-entropy (CE). From there, we try to explore the different reasons that could explain the inefficiency of transductive methods over inductive ones on NLP tasks, as the performance improvement claimed on vision tasks was promising. Specifically, we focused on comparing the inductive baseline only with the TIM approach, as it was proven to be effective on the vision tasks.

RQ2: Does the number of parameters of backbone have an impact ?

A possible way to explain the fact that transductive methods struggle to beat inductive fine-tuning on few-shot textual classification may reside in the quality of representations learned by the pre-trained backbone. Thus, we try here to compare the difference of performances between a pre-trained BERT-base architecture (110M parameters) and a RoBERTa-large architecture (354M

parameters). In the meantime, we focus on the Banking⁷⁷ dataset for evaluation, as its test set is balanced with 40 samples per class. Indeed, the Tweet_eval set is unbalanced, which may undermine TIM performances, as the intuition of this approach is to push the label distribution towards a uniform distribution.

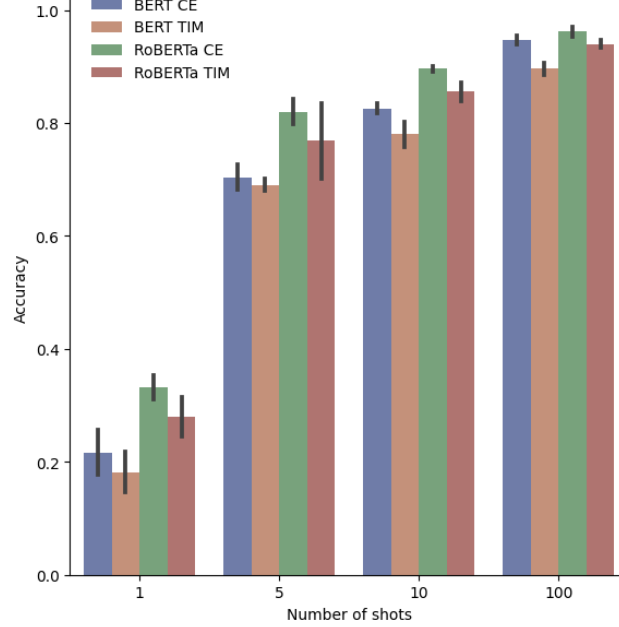


Figure 6.3: Comparison of cross-entropy-based and TIM-based approaches for BERT and RoBERTa backbones on the Banking⁷⁷ dataset, considering $K = 77$ classes.

The results are illustrated in [Figure 6.3](#). This plot clearly denies this hypothesis: if improving the initial representation by increasing the capacity of the pre-trained backbone clearly results in a performance improvement, the transductive method does not compete with its inductive counterpart.

Finally we also compare the performances of such architecture with a different classification head. Namely, as in (Boudiaf et al. 2020), we suppose that:

$$p_k^i \propto \exp\left(-\frac{\tau}{2}\|\phi_k - z^i\|^2\right) \quad (6.7)$$

where $\Phi := [\phi_1, \dots, \phi_K]$ denotes learnable classifier weights, $z^i = \frac{g_\theta(x^i)}{\|g_\theta(x^i)\|^2}$ are the normalized representations produced by pre-trained backbone and τ is a temperature parameter. In this setting, classification head weights Φ are initialized as the prototypes of the support set, as introduced in (Snell et al. 2017):

$$\phi_k^{(0)} = \frac{\sum_{i \in \mathcal{I}_S} y_k^i z^i}{\sum_{i \in \mathcal{I}_S} y_k^i}$$

The results of the experiment are illustrated in [Figure 6.4](#).

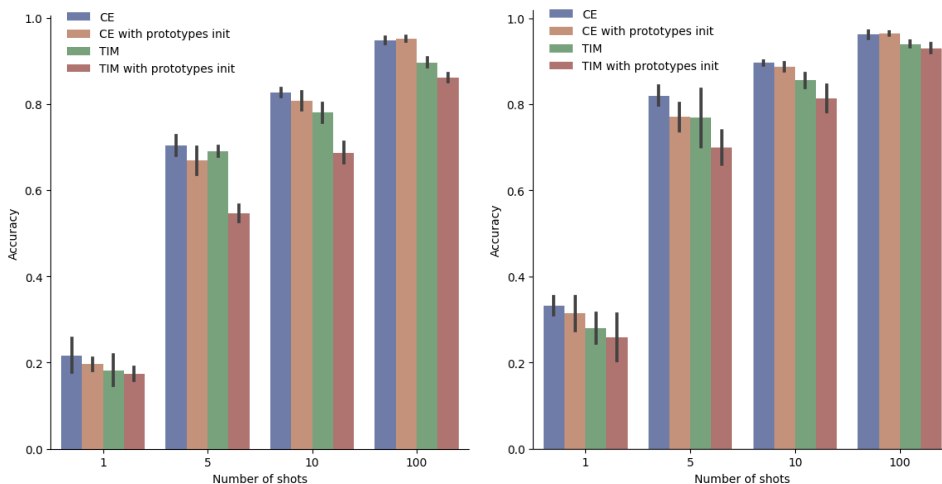


Figure 6.4: Comparison of BERT (left) and RoBERTa (right) backbone performances on Banking77 when initializing classification head as support set prototypes. We consider $K = 77$ classes.

As we can see, initializing the weight matrix of the classification head according to the prototypes of the support set also does not help the transductive method, which faces a decrease in performance across all few-shot regimes (with only an accuracy similar to CE for $N_S = 100$).

RQ3: Which fine-tuning strategy improve results?

Eventually, we try different fine-tuning strategies to improve accuracy on the few-shot classification task:

- Freezing all the weights of the pre-trained backbone, and only fine-tuning the classification head. This strategy is referred as "Frozen LM" on the plots.
- Freezing all the weights of the pre-trained backbone except the parameters controlling the layer normalization procedures, and the classification head. This strategy is referred as "LayerNorm" on the plots.
- Freezing all the weights of the pre-trained backbone except the bias parameters, and the classification head. This strategy is referred as "BitFit" (Ben Zaken et al. 2022) on the plots.
- Fine-tuning all parameters of the model. This strategy is referred as "Complete" on the plots.

The detailed results are reported in Table 6.2 with relative gains of TIM regularizer over inductive-based method, while Figure 6.5 illustrates them as bar plots.

Our analysis reveals that exhaustive fine-tuning of all model parameters does not necessarily guarantee superior outcomes when juxtaposed with alternative strategies like BitFit or LayerNorm. Interestingly, these strategies offer a more cost-effective approach to fine-tuning, and in certain data regimes, they even surpass the performance of complete fine-tuning. It is noteworthy (but

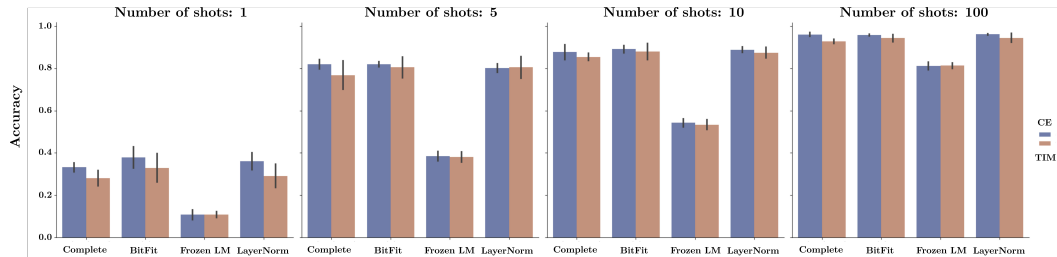


Figure 6.5: Comparison of cross-entropy-based and TIM-based approaches for different fine-tuning strategies on the Banking77 dataset ($K = 77$).

	1			5			10			100		
	CE	TIM	Gain	CE	TIM	Gain	CE	TIM	Gain	CE	TIM	Gain
C	33.18	27.99	↓-5.19	82.01	76.87	↓-5.15	87.74	85.52	↓-2.22	96.17	92.80	↓-3.37
BF	37.92	32.92	↓-5.00	82.08	80.52	↓-1.56	89.16	88.05	↓-1.10	95.84	94.42	↓-1.43
FLM	10.71	10.84	↑0.13	38.38	38.09	↓-0.29	54.22	53.37	↓-0.85	81.23	81.36	↑0.13
LN	36.04	29.09	↓-6.95	80.26	80.52	↑0.26	88.90	87.53	↓-1.36	96.23	94.55	↓-1.69

Table 6.2: Results of the different fine-tuning methods for the Banking77 dataset ($K = 77$), along with the relative gain of TIM against CE method: Complete fine-tuning (C), BitFit (BF), Frozen LM (FLM), LayerNorm (LN).

not surprising) that maintaining frozen weights for the pre-trained feature extractor g_θ consistently resulted in inferior performance, as the model ability to adapt to unseen classes is restricted to the classification head parameters. However, if we focus on the Gain column in Table 6.2, we observe that this fine-tuning configuration is the one in which TIM regularizer most competes (and sometimes slightly surpasses) CE-based fine-tuning. This is more coherent with the results obtained in original TIM work (Boudiaf et al. 2020), for which the parameters of the visual feature extractor are frozen.

CONCLUSION

In this chapter, we delved into the utilization of transductive losses as supplementary objectives for textual few-shot classification, aiming to address the limitations of prompting-based and in-context-learning-based approaches in real-world few-shot scenarios with a vast number of classes. Throughout our experiments, we evaluated the performance of traditional transductive regularizers applied to textual few-shot classification. We discovered that transductive methods have difficulty outperforming inductive cross-entropy-based fine-tuning when there is some flexibility in the pre-trained feature extractor g_θ parameters, regardless of g_θ 's capacity or the classification head h_ϕ 's initialization. Last but not least, we found that by fixing all parameters of g_θ , the transductive approach finally rivals the inductive one. Building on this insight, the next chapter will focus on examining textual few-shot classification in an API-based setting.

7 TEXTUAL FEW-SHOT CLASSIFICATION FOR API-BASED MODELS

CHAPTER'S SUMMARY

In this chapter, we address the increasing prevalence of proprietary and closed APIs for large language models like GPT-4 and ChatGPT, which have significant implications for practical applications of NLP, including few-shot classification. Few-shot classification entails training a model to execute a new classification task with minimal labeled data. Our investigation presents three key contributions. Firstly, we introduce a situation in which a pre-trained model is made accessible through a gated API, taking into account compute-cost and data-privacy constraints. Secondly, we delve deeper into the application of transductive inference, a learning paradigm that has been relatively underexplored within the NLP community. As opposed to traditional inductive learning, transductive inference takes advantage of the statistics of unlabeled data. In this context, we also introduce a new parameter-free transductive regularizer based on the Fisher-Rao loss, demonstrating its applicability and effectiveness in the gated API embedding setting. This approach fully leverages unlabeled data, avoids sharing any label information with third-party API providers, and could serve as a baseline for future research. Finally, we propose an enhanced experimental setting and compile a benchmark of eight datasets encompassing multiclass classification in four different languages, with up to 151 classes. We evaluate our methods using eight backbone models and an episodic evaluation across 1,000 episodes, which demonstrate the superiority of transductive inference over the standard inductive setting.

7.1 INTRODUCTION

Recent advances in NLP have been largely driven by the scaling paradigm (Kaplan et al. 2020; Rosenfeld et al. 2020), where larger models with increased parameters have been shown to achieve state-of-the-art results in various NLP tasks (Radford, J. Wu, et al. 2019; Touvron et al. 2023). This approach has led to the development of foundation models such as ChatGPT (Kocoń et al. 2023; Lehman et al. 2023), GPT-4 (OpenAI 2023), GPT-3 (Brown et al. 2020), T5 (Raffel, Shazeer, et al. 2020), and BERT (Devlin et al. 2019), which have achieved unprecedented performance in text classification (Yinhan Liu et al. 2019), language modeling, machine translation (Fan et al. 2021), and coding tasks (Mark Chen et al. 2021).

Despite the success of the scaling paradigm, significant challenges still exist especially when the many practical constraints of real-world scenarios have to be met: labeled data can be severely limited (*i.e.*, few-shot scenario (Y. Song et al. 2022; Ye et al. 2021)), data privacy is critical for many industries and has become the subject of increasingly many regulatory pieces (Commission 2016; Com-

mission 2020), compute costs need to be optimized (Strubell et al. 2019). Furthermore, these challenges are made even more complex as stronger foundation models are now available only through APIs (e.g., OpenAI’s GPT-3, GPT-4 or ChatGPT, Anthropic’s Claude or Google’s PaLM (Chowdhery et al. 2022)) which has led to some of their parameters being concealed, presenting new challenges for model adaptation (Solaiman 2023). This chapter is still centered on the fundamental task of few-shot text classification, but with a specific focus on cloud-based/API access, as their ease of integration, reduced infrastructure overhead, and the ability to leverage cutting-edge models is likely to become the standard approach for numerous enterprises looking to implement few-shot NLP classification tasks. Specifically, we formulate three requirements for API-based FSL (see Figure 7.1):

- (R1) Black-box scenario.** We focus on learning from models that are opaquely deployed in production to the end-user, who only has access to the end-point of the encoder, *i.e.*, the resulting text embedding produced by the final layer of the network.
- (R2) Low resources / computation time.** AI systems are often required to make rapid predictions at high frequencies in various real-world applications. Therefore, any few-shot classifier used in such scenarios should have a low training and inference time, as well as require minimal computational resources.
- (R3) Limited Data Sharing.** When utilizing API models, data sharing becomes a major concern. In the current landscape, providers are increasingly offering less transparent procedures for training their networks. As a result, users prefer sharing as little information as possible, such as labeling schema and annotated data, to safeguard their data privacy.

While numerous previous studies have addressed the popular *few-shot* classification setting, to our knowledge no existing line of work adequately satisfies the three API requirements described above. In particular, prompt-based FSL (Schick et al. 2020) and parameter-efficient fine-tuning FSL (Houlsby et al. 2019) both require access to the model’s gradients, while In-Context learning scales poorly with the task’s size (e.g. number of shots, number of classes) (Brown et al. 2020; Y. Chen et al. 2022; S. Min, Lewis, et al. 2022; S. Min, X. Lyu, et al. 2022) and requires full data sharing. Instead, in this work, we focus on methods that can operate within API-based constraints.

Under **R1**, **R2**, and **R3** requirements, the standard inductive learning (Haokun Liu et al. 2022) may be quite limiting. To mitigate the labeled data scarcity while retaining API compliance, we once again explore transduction (V. N. Vapnik 1999) in the context of textual few-shot classification. Specifically, in the context of few-shot learning, transductive FSL (Yanbin Liu et al. 2019) advocates leveraging unlabeled test samples of a task as an additional source of information on the underlying task’s data distribution in order to better define decision boundaries. Such additional source essentially comes for free in many *offline* applications, including sentiment analysis for customer feedback, legal document classification, or text-based medical diagnosis.

For this API-based setting, our findings corroborate the recent findings in computer vision (Boudiaf et al. 2020; Y. Hu et al. 2021; Lichtenstein et al. 2020; Yanbin Liu et al. 2019; Ziko et al. 2020), that substantial gains can be obtained from using transduction over induction, opening new avenue of research for the NLP community. This is in adequation with the last findings of Chapter 6,

when we considered a frozen backbone. We discuss the links between the two chapters in [Subsection 7.4.9](#).

However, the transductive gain usually comes at the cost of introducing additional hyperparameters, and carefully tuning them. Motivated by Occam’s razor principle, we propose a novel hyperparameter-free transductive regularizer based on Fisher-Rao distances and demonstrate the strongest predictive performances across various benchmarks and models while keeping hyperparameter tuning minimal, thereby emphasizing its effectiveness and practicality in the current context. We believe that this parameter-free transductive regularizer can serve as a baseline for future research.

CHAPTER’S CONTRIBUTIONS

In this chapter, our contributions are threefold:

A new textual few-shot scenario: We present a new scenario for FSL using textual API-based models that accurately captures real-world constraints. Our novel scenario opens up new research avenues and opportunities to address the challenges associated with FSL using API-based models, paving the way for improved performance and practical applications in the field. We show that current NLP FSL approaches all face limitations to tackle classification in this setting.

A novel transductive baseline: We propose a transductive FSL algorithm that utilizes a novel parameter-free Fisher-Rao based loss. By leveraging only the network’s embedding (**R1**), our approach enables fast and efficient predictions (**R2**) without the need to share the labeling schema or the labels of few-shot examples making it compliant with (**R3**). This innovative method marks a significant step forward in the field of few-shot learning, offering improved performance and practicality for real-world applications.

A truly improved experimental setting: Previous studies on textual few-shot classification (Gao, Fisch, et al. 2021; Mahabadi et al. 2022; Schick et al. 2021; Schick et al. 2022; Tam et al. 2021) have predominantly assessed their algorithms on classification tasks with a restricted number of labels (typically less than five). In line with the previous chapter, we take a step forward and create a benchmark that is more representative of real-world scenarios. Our benchmark relies on a total of eight datasets, covering multiclass classification tasks with up to 151 classes, across four different languages. Moreover, we further enhanced the evaluation process by not only considering 10 classifiers trained with 10 different seeds (Logan IV et al. 2022; Mahabadi et al. 2022), but also by relying on episodic evaluation on 1,000 episodes (Hospedales et al. 2021). Our results clearly demonstrate the superiority of transductive methods.

7.2 API BASED FEW-SHOT LEARNING

7.2.1 PROBLEM STATEMENT

As in the framework defined in [Subsection 6.2.2](#), we consider a vocabulary Ω , an input space \mathcal{X} with $\mathcal{X} \subseteq \Omega^*$ and a latent space \mathcal{Z} . We then seek to learn a classifier from limited labeled data and generalize to new, unseen tasks or classes by adapting a pre-trained backbone model $g_\theta : \mathcal{X} \rightarrow \mathcal{Z}$, by the mean of few-shot tasks created from a test dataset \mathcal{D}_{test} . Each task has a support set S composed of $N_S \times K$ labeled examples and a query set Q composed of $N_Q \times K$ unlabeled examples, sampled between K unseen classes.

Setting the values of N and K in textual FSL is not standardized. Therefore, in all of our experiments, we have relied on setting $(N, K) \in \{5, 10\}^2$. In the API-based setting, the main difference is that we assume that we are unable to access the exact structure of g_θ as mentioned in **R1**. However, we do have access to the last embedding of the encoder which is available for our use (see **R1**). The other desiderate **R2** and **R3** are represented in the schema of the API-based FSL setting depicted in [Figure 7.1](#).

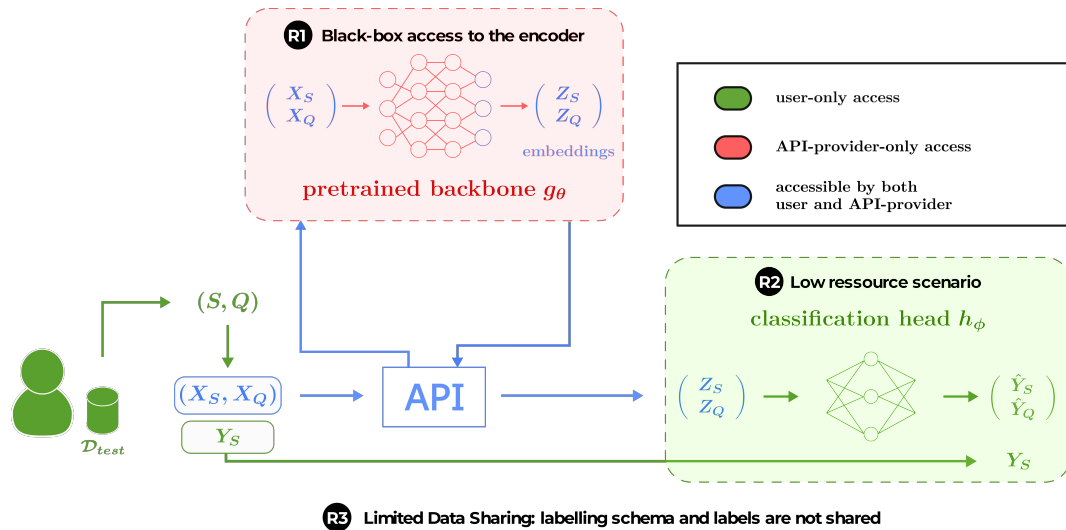


Figure 7.1: API-based few-shot learning scenario. The black-box API is providing embeddings from the pre-trained encoder g_θ . The black-box scenario discards existing inductive approaches and ICL methods due to inaccessible of model’s parameters (**R1**) and privacy concerns (**R3**). This scenario allows to tune a classification head h_ϕ (using induction or transduction) at low computational cost (**R2**), while retaining all support labels locally.

7.2.2 LIMITATIONS OF CURRENT METHODS

Besides the drawbacks of current NLP FSL techniques for large number of classes (explored in [Subsection 6.2.1](#)), including important engineering or poor generalization ability, new limitations to these strategies are pointed out by the considered API-based setting.

Prompt-based few-shot learning: These approaches (Ding et al. 2022; P. Liu et al. 2023; Schick et al. 2020) face limitations when learning from API: (i) encoder access for gradient computation is infeasible (as in **R1**), (ii) prompting requires to send data and label which raises privacy concerns (as in **R3**), and (iii) labeling new points is time-consuming (see in **R2**) and expensive due to the need to send all shots for each input token¹. **Parameter-efficient fine-tuning.** Relying on parameter-efficient fine-tuning methods (Ben Zaken et al. 2022; Houlsby et al. 2019; Haokun Liu et al. 2022; Pfeiffer et al. 2020) with an API is not possible due to the need to compute gradients of the encoder (as per **R1**) and the requirement to send both the labeling schema and the labels, which violates **R3**. **In Context Learning.** A significant drawback of this approach (Wei, Xuezhi Wang, et al. 2022) is that the user must supply the input, label examples, and task description, which is both slow (Haokun Liu et al. 2022) (**R2**) and raises data privacy concerns (as highlighted in **R3**). Additionally, the inability to reuse text embeddings for new tasks or with new labels without querying the model’s API limits practicality and scalability, making reusable encoding unfeasible for ICL models. **Meta-learning.** Unlike the three previous lines of work, meta-learning methods operate by modifying the pre-training procedure and therefore assume access to both the training data and the model, which wholly breaks both **R1** and **R3**.

7.2.3 PROPOSED TRANSDUCTIVE APPROACHES AND BASELINES

As in Chapter 6, our goal is to learn a classifier $f_{\phi^*, \theta^*} = h_\phi \circ g_\theta$. However, as in the API-based setting we cannot access backbone parameters θ , we aim to train only the classification head h_ϕ by solving the related objective:

$$\phi^* = \arg \min_{\phi} \text{CE} - \lambda \times R_Q \quad (7.1)$$

with $\text{CE} = -\frac{1}{|S|} \sum_{i \in \mathcal{I}_S} \sum_{k=1}^K y_k^i \log(p_k^i)$ being the cross-entropy supervision on the support set

and R_Q being a transductive loss on the query set Q . In the conducted experiments, we chose to compare the transductive methods based on Entropic Minimization (**H**) and TIM algorithm (**I**), associated to respective regularizers R_Q^H and $R_Q^I(\beta)$ (as introduced in Equation 6.2 and Equation 6.3) with the Linear probing inductive baseline (**CE**). All three methods were introduced in Chapter 6. We finally consider another inductive baseline: Prototypical Networks (**PT**). Prototypical Networks learn a metric space where the distance between two points corresponds to their degree of similarity. During inference, the distance between the query example and each class prototype is computed, and the predicted label is the class with the closest prototype. Prototypical networks have been widely used in NLP and are considered as a strong baseline (Gao, X. Han, et al. 2019; Snell et al. 2017; S. Sun et al. 2019).

Limitation of existing transductive strategies: Despite its effectiveness, the TIM method implies the need to fine-tune the weight of different entropies using the hyperparameter β . This hyperparameter-tuning process can be time-consuming and may require extensive experimentation to achieve optimal results. Additionally, recent studies have shown that relying solely on the

¹The cost of API queries is determined by the number of input tokens that are transmitted.

first entropic term, which corresponds to the Entropic Minimization scenario in Equation 6.2, can lead to suboptimal performance in FSL.

7.2.4 A FISHER-RAO BASED REGULARIZER

In the FSL scenario, minimizing parameter tuning is crucial. Motivated by this, in this section we introduce a new parameter-free transductive regularizer which fits into the InfoMax framework. Additionally, our loss inherits the attractive properties of the recently introduced Fisher-Rao distance between soft-predictions $\mathbf{q} := (q_1, \dots, q_K)$ and $\mathbf{p} := (p_1, \dots, p_K)$, which is given by (Picot et al. 2023) and (Gomes et al. 2022):

$$d_{\text{FR}}(\mathbf{q}, \mathbf{p}) := 2 \arccos \left(\sum_{k=1}^K \sqrt{q_k \times p_k} \right). \quad (7.2)$$

The proposed transductive regularizer denoted by R_Q^{FR} , for each single few-shot task, can be described as measuring the Fisher-Rao distance between pairs of query samples:

$$R_Q^{\text{FR}} := \frac{1}{|Q|} \sum_{i \in Q} -\log \sum_{j \in Q} \sum_{k=1}^K \sqrt{p_k^i \times p_k^j} = \frac{1}{|Q|} \sum_{i \in Q} -\log \sum_{j \in Q} \cos \left(\frac{d_{\text{FR}}(\mathbf{p}^i, \mathbf{p}^j)}{2} \right), \quad (7.3)$$

where $d_{\text{FR}}(\mathbf{p}^i, \mathbf{p}^j)$ is the Fisher-Rao distance between pairs of soft-predictions $(\mathbf{p}^i, \mathbf{p}^j)$. Furthermore, it is shown that expression (7.3) yields a surrogate of the Mutual Information as shown by the following proposition. This result to the best of our knowledge is new, as far as we can tell.

Proposition 1. (Fisher-Rao as a surrogate to maximize Mutual Information) Let $(\mathbf{q}_i)_{i \in Q}$ be a collection of soft-predictions corresponding to the query samples. Then, it holds that:

$$R_Q^{\text{FR}} + \log |Q| \leq R_Q^I(1) \leq R_Q^I(\alpha), \quad \forall 0 \leq \alpha \leq 1. \quad (7.4)$$

Proof: Further details are relegated to Section 9.1.

Advantage of R_Q^{FR} over $R_Q^I(\beta)$: Similarly to $R_Q^I(\beta)$, R_Q^{FR} can be exploited to maximize the Mutual Information. However, R_Q^{FR} is parameter free and thus, it does not require to tune β .

7.3 AN ENHANCED EXPERIMENTAL SETTING

In this chapter, we put a special emphasis on the experimental setting, which builds upon the limitations from prior studies and observations outlined in Chapter 6. Specifically, we underscore the diversity in our evaluation datasets. These datasets are characterized by a broad range of classes and varied label distributions, further enhancing their robustness. Moreover, drawing from the performance disparities observed between the BERT and RoBERTa backbones in the previous chapter, we initiate an in-depth exploration involving multiple pre-trained backbones, spanning both monolingual and multilingual scopes. Finally, we direct our attention towards the capacity for generalization and adaptability in this chapter. As such, we integrate a greater number of tasks that contain fewer sampled classes per task.

7.3.1 DATASETS

Benchmarking the performance of FSL methods on diverse set of datasets is critical to evaluate their generalization capabilities in a robust manner as well as their potential on real-world applications. As mentioned in [Subsection 6.4.1](#), previous work on FSL (Mahabadi et al. 2022; Perez et al. 2021) mainly focus on datasets with a reduced number of classes (*i.e.*, $K < 5$). Motivated by practical considerations we choose to build a new benchmark composed of datasets with a larger number of classes.

Dataset	Number of classes
Multilingual Amazon Reviews Corpus (Keung et al. 2020)	32
Go Emotion (Demszky et al. 2020)	22
Tweet_eval (Barbieri et al. 2020)	20
Banking77 (Casanueva et al. 2020)	77
Clinic (Larson et al. 2019)	151

Table 7.1: Statistics of the considered datasets.

Specifically, besides Tweet_eval (Barbieri et al. 2020), Banking77 (Casanueva et al. 2020) studied in [Chapter 6](#), we consider:

- Multilingual Amazon Reviews Corpus (MARC) (Casanueva et al. 2020), that consists of reviews extracted from different Amazon marketplaces. The reviews comprise six languages: English, German, French, Spanish, Japanese, and Chinese.
- Go Emotion (Demszky et al. 2020), which contains Reddit comments extracted from popular English-language subreddits and labeled with emotion categories.
- Clinic (Larson et al. 2019), that consists of thousands of annotated examples of natural language queries and responses, covering 150 intent classes over 10 domains, and one out-of-scope class.

These datasets cover a wide range of text classification scenarios and are of various difficulty. A summary of the datasets used can be found in [Table 7.1](#). They are all available in Dataset (Lhoest et al. 2021).

7.3.2 MODEL CHOICE

The selection of an appropriate backbone model is a critical factor in achieving high performance in few-shot NLP tasks. To ensure the validity and robustness of our findings, we have included a diverse range of transformer-based backbone models in our study, including:

- Three different sizes of RoBERTa-based models (Yinhan Liu et al. 2019). Similar to BERT, RoBERTa is pre-trained using the cloze task (W. L. Taylor 1953). We consider two different sizes of the RoBERTa model, namely RoBERTa (B) with 124M parameters and RoBERTa (L) with 355M parameters and DistilRoBERTa, a lighter version of RoBERTa trained through a distillation process (Geoffrey E. Hinton et al. 2015), for a total of 82M parameters.

- Three sentence-transformers encoder (Reimers et al. 2019). Following the recommendation of (Muennighoff et al. 2023), we consider MPNET-base (K. Song et al. 2020) (109M parameters), MiniLM (33M parameters) (W. Wang et al. 2020), and Albert Small V2 (11M parameters) (Z. Lan et al. 2020).
- Multilingual models. To address realistic scenarios, we do not restrict our study to the English language. We rely on three sizes of XLM-RoBERTa (Conneau et al. 2020): base (B) with 124M, large with 355M (L) and XL (XL) with 3.5B of parameters.
- GPT-3.5 model: to mimic the typical setting of API-based models, we also conduct experiments on GPT-3.5 (Brown et al. 2020), only accessible through OpenAI’s API.

Preliminary Experiment. In our experiments, the backbone models are of utmost importance. Our objective in this preliminary experiment is to assess the efficacy of these models when fine-tuning **only** the model head across a variety of datasets. Through this evaluation, we aim to gain insight into their generalization abilities and any dataset-specific factors that may influence their performance. This information is used to analyze the performance of different models in the few-shot scenario, as described in Section 7.4. We present the results of this experiment in Table 7.2, noting that all classes were considered, which differs from the episodic approach detailed in Section 7.4.

Model	Params	Emotion	Twitter	Clinic	Banking77	Amazon			
		en	en	en	en	en	fr	es	de
Albert Small V2 (XS)	11M	25.2	18.3	67.0	88.1	33.5	X	X	X
MiniLM (S)	33M	30.2	19.3	67.1	92.3	39.5	X	X	X
MPNET-base (B)	109M	30.2	22.5	67.4	94.3	41.3	X	X	X
DistilRoBERTa (S)	82M	23.3	26.0	68.5	90.9	40.0	X	X	X
RoBERTa (B)	124M	21.0	25.5	66.7	91.4	39.2	X	X	X
RoBERTa (L)	355M	15.0	23.0	64.5	90.0	38.1	X	X	X
XLM-RoBERTa (B)	278M	21.0	22.1	66.5	87.0	40.1	19.2	17.5	18.3
XLM-RoBERTa (L)	559M	14.0	18.0	64.5	86.2	38.2	17.5	15.6	18.1
XLM-RoBERTa (XL)	3.48B	25.4	19.0	68.9	95.0	41.0	18.9	17.9	22.0
GPT-3.5	175B	38.9	35.3	70.4	98.7	48.4	30.4	34.0	33.5

Table 7.2: Preliminary experiment results. Accuracy of the different backbone trained on each training set.

7.3.3 EVALUATION FRAMEWORK

Prior research in textual FSL typically involves sampling a low number of tasks, generally less than 5, of each dataset. In contrast, we utilize an episodic testing framework that generates a large number of N-shots K-ways tasks. To account for the model’s generalization ability, we average the results for each dataset over 1000 episodes, with the K considered classes varying in every episode. For each experiment, we consider the F1-Score as the evaluation metric.

7.4 EXPERIMENTS

7.4.1 OVERALL RESULTS

Global results: To evaluate the effectiveness of various few-shot methods, we conducted a comprehensive analysis of their classification performance across all datasets, all backbones, and all considered N-way/K-shot scenarios. Results are reported in [Table 7.3](#).

An interesting observation is that transductive approaches based on TIM (I) and Fisher-Rao (FR) regularizers outperform their inductive counterparts based on Linear Probing (CE) and Prototypical Networks (PT) strategies. Notably, we found that vanilla entropy minimization, on which solely relies H, consistently underperforms in all considered scenarios. Our analysis revealed that FR surpasses traditional fine-tuning based on cross-entropy by a margin of 3.7%.

Mono-lingual experiment:

In order to thoroughly analyze the performance of each method, we conducted a per-dataset study, beginning with a focus on the mono-lingual datasets. [Figure 7.2](#) reveals that the global trends observed in [Table 7.3](#) remain consistent across datasets of varying difficulty levels. Notably, we observed consistent improvements achieved by transductive regularizers (such as I or FR) over CE. However, the relative improvement is highly dependent on the specific dataset being evaluated. Specifically, FR achieves +6.5% F1-score on Banking77, but only a shy +1.5% on Tweet_eval. A strong baseline generally suggests highly discriminative features for the task, and therefore a strong upside in leveraging additional unlabeled features, and vice versa. Therefore, we hypothesize that the potential gains to be obtained through transduction correlate with the baseline’s performance. Additional results can be found on [Subsection 7.4.5](#) multilingual experiments (*i.e.*, on es, de, fr) which exhibit the same behavior.

Table 7.3: Aggregated performance over the different datasets and considered backbones.

K-shots	10		5	
	10	5	10	5
FR	52.09	61.99	48.71	56.55
I	<u>50.07</u>	59.17	<u>46.42</u>	55.74
H	15.07	27.39	15.33	25.84
CE	48.31	56.87	45.27	53.94
PT	47.29	56.05	44.32	53.20

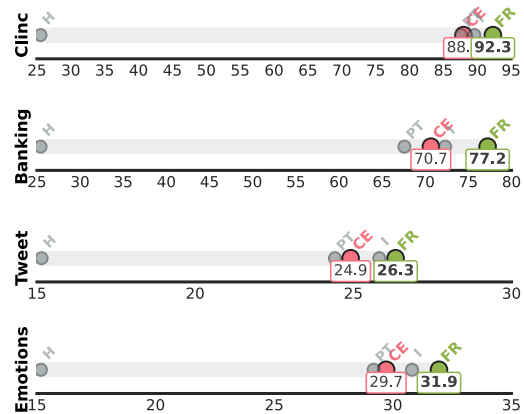


Figure 7.2: Performance of the different pre-trained encoders on the monolingual datasets.

7.4.2 STUDY UNDER DIFFERENT DATA REGIMES

In this experiment, we investigated the performance of different loss functions under varying conditions of 'ways' and 'shots'. As shown in Figure 7.3, we observed that increasing the number of classes ('ways') led to a decrease in F1-score while increasing the number of examples per class ('shots') led to an improvement in F1-score. This can be explained by the fact that having more data enables the classifier to better discern the unique characteristics of each class.

Interestingly, the relationship between the number of shots and classification F1-score may not be the same for all classes or all loss functions. Figure 7.3 shows that different loss functions (e.g. FR on Banking77) benefited greatly from adding a few shots, while others did not show as much improvement. However, this variability is dependent on the specific dataset and language being used, as different classes may have different levels of complexity and variability, and some may be inherently easier or harder to classify than others.

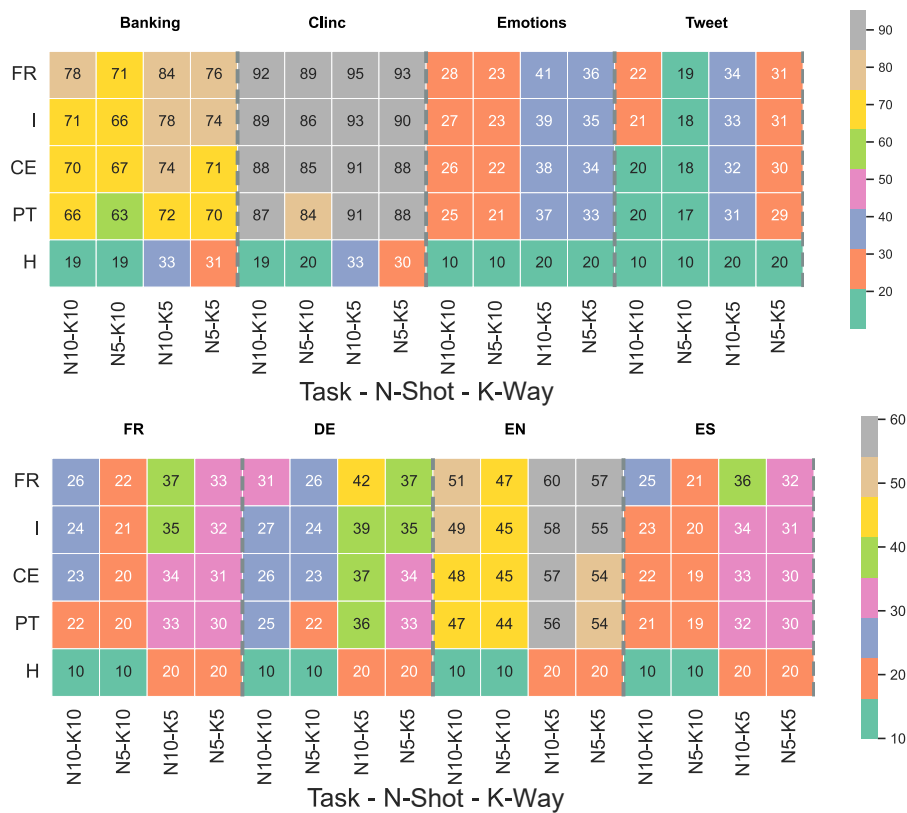


Figure 7.3: The effect of different ways and shots on test performance. Monolingual experiments are shown on top, and multilingual experiments on bottom.

7.4.3 ABLATION STUDY ON BACKBONES

In this experiment, we examined how different loss functions perform when increasing the number of parameters in various models. The results, presented in Figure 7.4, show the average perfor-

mance across the experiments (with multilingual datasets on the left, without on the right) and are organized by loss function. We observed an *inverse scaling law* for both the RoBERTa and XLM-RoBERTa family of models, where increasing the number of parameters led to a decrease in performance for the losses we tested. However, within the same family, we observe that the superiority of FR remains consistent. An interesting finding from Figure 7.4 is that the transduc-

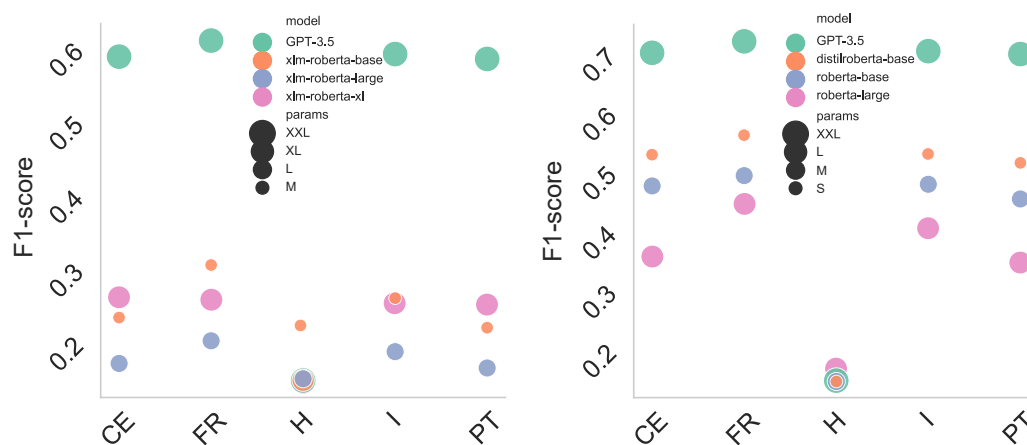


Figure 7.4: Impact of backbone’s size on performances. Both multilingual (left) and monolingual (right) model families show some inverse scaling laws and a superiority for the FR regularizer over other methods.

tive regularization technique using FR outperforms other methods on GPT-3.5. This highlights the effectiveness of FR in improving the performance of the model and suggests that transductive regularization may be a promising approach for optimizing language models.

7.4.4 A DIVE INTO GPT-3.5 RESULTS

GPT-3.5 appears to be the backbone providing the most informative a priori embeddings in Table 7.2 and could be considered as the prime model for API-based FSL, showcasing the current requirements in this area. It is thus a typical candidate for application uses that must meet the following criteria **(R1)** - **(R3)**. Therefore, we put a special emphasis on its related results.

Figure 7.5 (left) details the GPT-3.5 results of the experiments conducted on the mono-lingual datasets. These plots highlight the consistency of the tendencies emerged in Table 7.2, Table 7.3 and Figure 7.2, namely: the superiority of transductive approaches (FR and I) over inductive ones (CE and PT), the underperformance of the entropic-minimization-based strategy (H), and the higher amount of information conveyed by GPT-3.5 learned embeddings over other backbones, resulting in higher F1-scores on all datasets.

These phenomena still occur in the multi-lingual setting, as illustrated in Figure 7.5 (right), stressing the superiority of transductive (and especially FR) over other approaches for presumably universal tasks, beyond english-centered ones, and without the need of using language-specific engineering as for prompting-based strategies.

Note that for both of these settings, the entropic-minimization-based strategy (H) seems to be capped at a 15% F1-score, thus with no improvement over other backbone embeddings, and independently of the dataset difficulty.

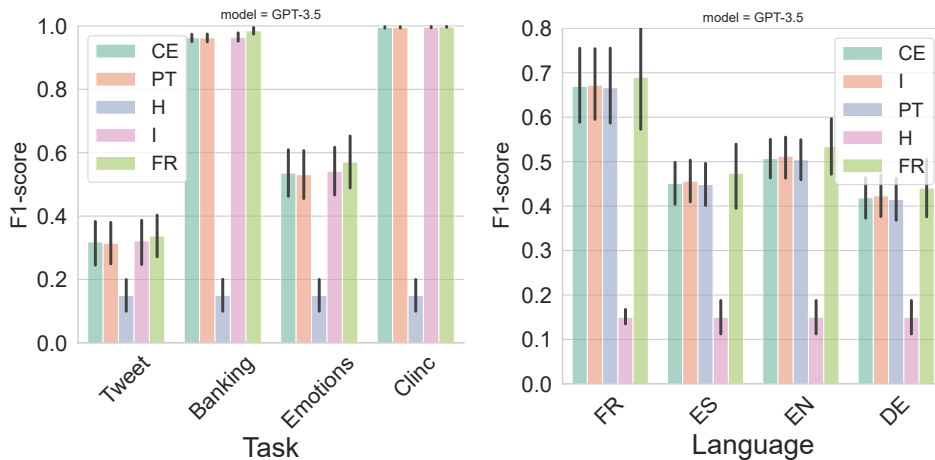


Figure 7.5: The different losses when training a on GPT3.5 embeddings.

7.4.5 MULTILINGUAL EXPERIMENT

To provide an exhaustive analysis, we report the same experiment that is made in [Subsection 7.4.1](#) for multi-lingual models on the MARC dataset. The observations made in [Subsection 7.4.4](#) are not specific to GPT-3.5 backbone and extend to the other multi-lingual encoders (that is XLM-RoBERTa-based ones). While both latin languages (French and Spanish) share almost identical results, with a trend very similar to the one of English language (an F1-score gain of around 4% for FR over CE), the results on German language exhibit an F1-score increased by more than 6% when switching from inductive CE to transductive FR, flirting with performances obtained on English tasks.

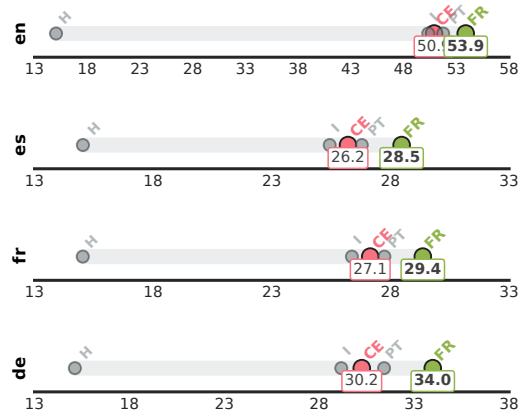


Figure 7.6: Performance of the different losses on multilingual datasets.

7.4.6 IMPORTANCE OF MODEL BACKBONES ON MONOLINGUAL EXPERIMENT

In this section, we report the results of our experiment aggregated per backbone. The goal is to understand how the different losses behave on the different backbone. The results are presented in [Figure 7.7](#). While the trends observed in

the previous charts are retrieved for the majority of backbones, some of these models are exceptions. For example, while transductive methods perform generally better than inductive methods, the CE-based method seems to perform slightly better than I for XLM-RoBERTa-xl. Additionally, while FR is the most effective method for the majority of backbones, it is surpassed by I for the all-distilroberta-v1 model. Furthermore, the inverse-scaling-law details are found for the RoBERTa(B/L) and XLM-RoBERTa (B/L) models per dataset. In general, it is interesting to note that although model performance is constrained by dataset difficulty, the performance order of each method is consistent across all 4 datasets for each considered backbone.

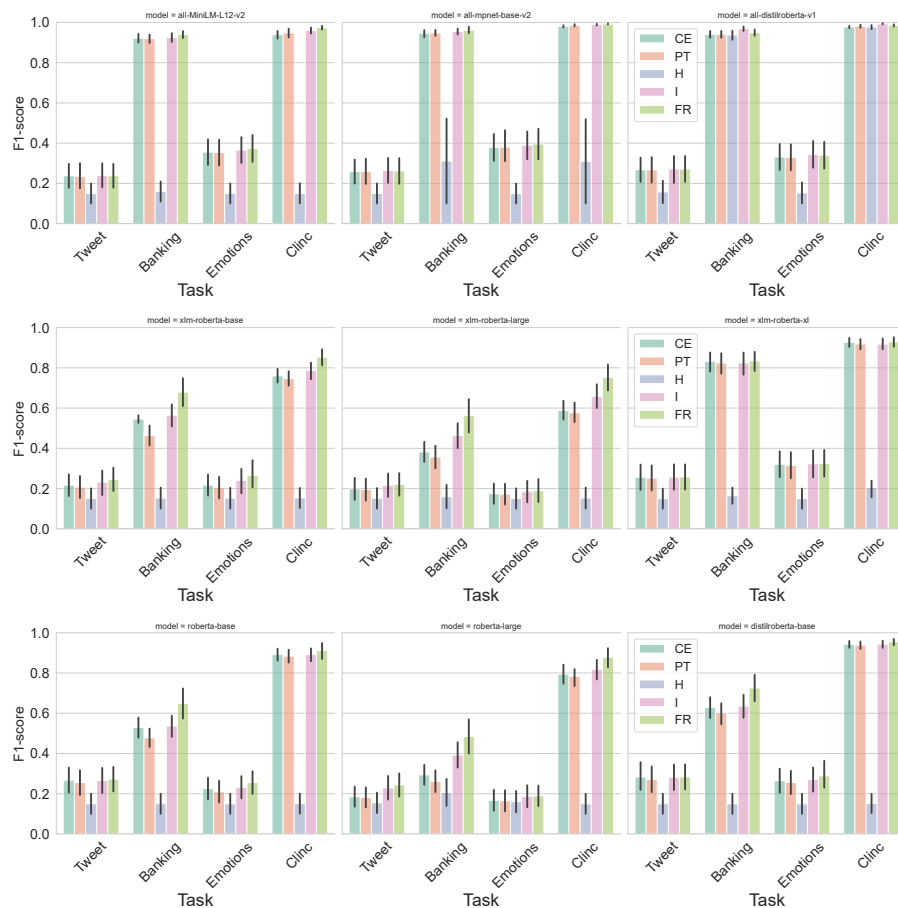


Figure 7.7: Performance of different pre-trained encoder on the monolingual datasets.

7.4.7 IMPORTANCE OF MODEL BACKBONES ON MULTILINGUAL EXPERIMENT

In this experiment, we report the performance of different losses on the Amazon dataset by averaging the results over the number of shots, ways for the different losses. The results are presented in [Figure 7.8](#). Our observations indicate that the transductive regularization, both for I and FR, consistently improves the results for different models, including base and large models, as well as

GPT-3.5. Similar to the findings reported in the main paper, we observe an inverse scaling law, with XLM-RoBERTa-base outperforming the larger versions.

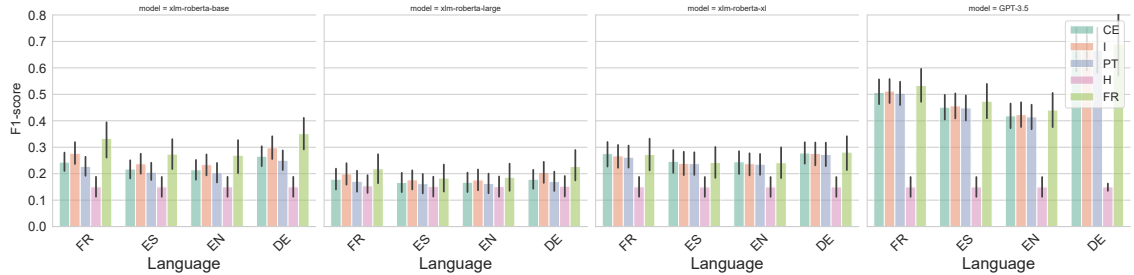


Figure 7.8: Performance of different pre-trained backbones on multilingual Amazon dataset.

RESULTS PER LANGUAGE

In this experiment, we report the performance of different losses on the Amazon dataset by averaging the results over the number of shots, ways, and model backbones. The results are presented in Table 7.4. Our observations indicate that the transductive regularization improves the results for two languages over the inductive baseline (i.e., CE). Additionally, we note that the observed improvements for FR are more consistent. This further demonstrates that the transductive loss can be useful in few-shot NLP.

	fr	de	en	es
FR	29.36	33.98	53.89	28.47
I	<u>27.74</u>	<u>31.41</u>	<u>51.75</u>	<u>26.79</u>
H	15.04	15.13	15.04	15.04
CE	27.15	30.24	50.89	26.21
PT	26.37	29.16	50.34	25.44

Table 7.4: Global results for multilingual Amazon dataset.

7.4.8 PRACTICAL CONSIDERATIONS

In this experiment, we adopt a practical standpoint and aim to evaluate the effectiveness of an API model, specifically GPT-3.5. In Subsection 7.4.8, we report the training speed of one episode on a MAC with CPU. Overall, we observed that the transductive loss is slower as it necessitates the computation of the loss on the query set, whereas PT is faster as it does not involve any optimization. Furthermore, we note that FR is comparable in speed to I. To provide a better understanding of these results, we can compare our method with existing approaches (in the light of R2). For instance, PET (Schick et al. 2020) entails a training time of 20 minutes on A100, while ADAPET (Tam et al. 2021) necessitates 10 minutes on the same hardware.

Loss	CPU Time
CE	0.45s
FR	0.83s
H	0.75s
I	0.83s
PT	0.01s

Table 7.5: Training time for 1 episode on a M1-CPU.

7.4.9 LINKS WITH THE OBSERVATIONS OF CHAPTER 6

In the previous chapter, our comparison between transductive and inductive methods under different fine-tuning conditions yielded mixed results. When the backbone parameters were accessible for fine-tuning, our experiments indicated an advantage for inductive methods. However, when the backbone parameters were frozen, the performance of the two methods was comparable, with a slight advantage for transductive methods in some data regimes, although the performance difference was not statistically significant.

The results obtained in the current chapter confirm and expand upon these initial promising observations. We find that transductive methods not only perform at least as well as the inductive ones when the backbone parameters are frozen, but also exhibit even better performance in this setting. One potential explanation for this superior performance of transductive methods in the API-based setting is the adoption of episodic evaluation, where we consider a fixed number of classes during inference. This evaluation approach differs from the one used in the previous chapter, where all classes were considered simultaneously. Indeed, the fixed number of classes during inference reduces the complexity of the problem, allowing transductive methods to better exploit the structure and relationships among the few-shot examples, which is one of the key strengths of transductive learning. Furthermore, episodic evaluation does not discredit the generalization ability of the studied approach, as the reported performances are averaged over 1000 parallel episodes, with different classes sampled for each episode.

In summary, our findings in this chapter provide strong empirical evidence that transductive methods are a serious candidate for few-shot classification in an API-based setting, where the backbone parameters are unavailable.

CONCLUSION

In this chapter, we have presented a novel few-shot learning framework that effectively leverages API models while adhering to the critical constraints of real-world applications (i.e., **R1**, **R2**, **R3**). The **R1** constraint is particularly relevant and crucial, as current competitive models are only accessible via API, preventing access to model parameters. Our approach is especially appealing as it shifts the computational requirements **R2**, eliminating the need for heavy computations for the user, and enables training classifiers on-the-fly in web browsers without sharing labels of the data **R3**.

Building upon the mixed results from the previous chapter, we have demonstrated the significant advantages of using transductive losses to perform NLP FSL in this API-setting,

that exhibit better performances than inductive ones when the backbone parameters are frozen, with a significant power of generalization across a large number of new classes and at a consequently affordable cost. The regularizer based on the Fisher-Rao distance provides a promising candidate, which is parameter-free and could serve as a straightforward baseline in future studies. In conclusion, in this chapter we successfully addressed the initial motivations for developing an efficient and effective FSL framework that meets real-world constraints. By shedding light on the potential of transductive losses and demonstrating their practicality in various use-cases, we hope to inspire further exploration and refinement of these methods, ultimately contributing to the advancement of FSL in the field of NLP.

8 CONCLUSION AND PERSPECTIVES

8.1 SUMMARY OF THE CONTRIBUTIONS

The first part of this thesis addresses **the challenge of exploiting multimodal data for fault diagnosis in the context of Industry 4.0 systems**. Motivated by a tangible need in the industrial field, we dove into the exploration of complex multimodal systems. Our journey begins in [Chapter 2](#) with the development of a theoretical framework based on multimodal learning, motivated by the intricate multimodal nature of our real-world environment. In this framework, we examined established concepts such as multimodal fusion and representation, taking a comprehensive view of the evolution of these paradigms from their early stages to the advent of DL-based multimodal representations. Our analysis, enriched by a focus on previous few attempts to apply Machine Learning for fault diagnosis, highlights the practical constraints of this application that have been overlooked by previous multimodal approaches.

In [Chapter 3](#), our investigation led to the identification of five significant challenges arising from the considered setting. In response, we developed **StreaMulT, a Streaming Multimodal Transformer**. This architecture employs cross-modal attention and a memory bank to process arbitrarily long input sequences during training and operate in a streaming mode during inference. This approach uniquely addresses the complexity posed by Industry 4.0 systems, efficiently managing the temporal unalignment of multimodal heterogeneous data and differences in data acquisition frequency. Despite an access to an adapted industrial dataset, its evaluation on the connected multimodal sentiment analysis task revealed that our model can manage arbitrarily long sequences without a loss in performance. With a carefully selected textual embedding module, StreaMulT surpassed existing methods, setting a new state-of-the-art performance on the CMU-MOSEI dataset. Coupled with the ablation study, this underscored the significant influence of the textual modality, thus justifying the emphasis placed on it in the second part of the thesis.

In [Chapter 4](#), we investigated the various interactions within multimodal data, which are categorized into redundant and complementary information types. We underscored the crucial role of complementary information, while simultaneously noting the scarcity of robust methodologies and benchmarks to evaluate the capacity of models to exploit this type of information.

The second part of the thesis is dedicated **to harnessing the unique value of textual data in the realm of Industry 4.0, offering a rich, contextual understanding of system operations, past incidents and expert knowledge**. Such insights are crucial for fault diagnosis and predictive maintenance. However, these reports are scarce and use industry-specific language, presenting challenges in processing and interpretation. To overcome this challenge, we adopt the few-shot learning paradigm. As detailed in [Chapter 5](#), our exploration dives into the realm of Natural Language Processing, investigating its progression from early methodologies to DL approaches and large Foundation Models. We further highlight the function of FSL and the primary frameworks

that facilitate the application of large PLMs within this paradigm.

In **Chapter 6**, the limitations of current FSL methods, specifically the engineering efforts required for realistic classification tasks with a large number of classes, are explored. In response, **we propose a novel adaptation of transductive techniques for textual classification**. The study demonstrates that transductive methods rival inductive ones when all parameters of the feature extractor are fixed.

Finally, in **Chapter 7** we take into consideration the increasing prevalence of proprietary and closed APIs for LLMs. **A new scenario for FSL using textual API-based models is presented, highlighting the constraints related to computation cost and data privacy**. The chapter introduces a **new parameter-free transductive regularizer based on the Fisher-Rao loss**, demonstrating its effectiveness in the gated API embedding setting. Moreover, it proposes an enhanced experimental setting for compiling a benchmark of datasets encompassing multi-class classification in different languages.

GENERAL TAKEAWAYS

In summary, in this thesis we have provided two methodological contributions in two major areas:

1. the proposal of the StreaMulT architecture (Pellegrain, Tami, et al. 2022), a multi-modal approach that serves as a pivotal contribution to the evolution of fault diagnosis methodologies,
2. the introduction of novel transductive methods for Few-Shot Learning in Natural Language Processing, framed in a realistic and API-based context (under review for publication in an international journal).

Apart from these methodological contributions, we have also proposed:

- a significantly expanded state-of-the-art in fault diagnosis, with an illustrative designed case study (Pellegrain, Batteux, et al. 2022).
- important discussions regarding the formalization and characterization of multi-modal interactions, particularly the roles of redundancy and complementarity in multimodal representation learning.

This thesis, while offering significant advancements, is a stepping stone in a continually evolving field of research. As we move forward, it is important to remember that the applications and methodologies described here will need to be tested further and refined in response to the challenges and opportunities presented by new developments in Industry 4.0. In the following section, we critically examine the contributions of this thesis, and propose possible directions for future research, further strengthening the impact of this work on the broader landscape of Industry 4.0 systems.

8.2 PERSPECTIVES

8.2.1 CRITICISM AND SHORT-TERM PERSPECTIVES

While StreaMulT presents an effective framework in handling multi-modal data processing, it falls short in certain areas that demand closer examination. We thereby list its limitations, focusing on the lack of robustness in performance, issues with missing modalities and imbalanced datasets, and the fully supervised approach it embodies.

Performance Limitations First and maybe most importantly, we did not conduct an exhaustive study of running and latency time. Even though the architecture theoretically allows the handling of arbitrarily long inputs during training and can be deployed in a streaming fashion at inference, the latency becomes a crucial factor. In the case that the system’s speed is less than optimal, it undermines the capability of real-time deployment, which is a crucial aspect of the streaming aspect of StreaMulT. This oversight is a significant drawback, particularly for industrial applications, where timeliness often equals efficiency. Our choice has been to use a chunk-wise approach with augmented memory, but different strategies exist, such as monotonic attention (Arivazhagan et al. 2020; X. Ma et al. 2020; Raffel, M. T. Luong, et al. 2017), in which one should alternate between reading the input and writing the output.

Robustness and Handling of Missing Modalities Further, StreaMulT does not address the issue of missing modalities which can impact the functionality of cross-modal attention modules. It also does not explicitly tackle problems related to imbalanced datasets and concept drifts. Consequently, the model does not account for the adaptation of training in streaming, a critical requirement for maintaining the system’s robustness. By not taking these aspects into account, the model could potentially be unsuitable for deployment in a dynamic and ever-evolving industrial setting where data is rarely perfect or consistent. A straightforward perspective is therefore to consider strategies that tackle concept drifts (Souza et al. 2020), for instance leveraging continual learning (Kirkpatrick et al. 2017).

Fully Supervised Approach and Its Implications The assumptions made when designing StreaMulT raise some critical questions as well. The model presumes a fully supervised approach and relies heavily on the availability of numerous datapoints from all modalities. This assumption is somewhat contradictory with the assertions made in Part 2 of the thesis, where we advocate that textual data are scarce. In addition, StreaMulT relies heavily on supervision and backpropagation. These methods are the precise limitations pointed out in Chapter 4. StreaMulT hence does not exploit complementary information effectively to provide control over representation. Therefore, it is essential to acknowledge that StreaMulT, while being a promising tool, remains bound by the constraints and limitations characteristic of current approaches. Weakly supervised and unsupervised settings could be handled using anomaly detection approaches, if the fault occurrences are rare in the considered setting.

In conclusion, while StreaMulT presents a step forward in multimodal data processing, it faces several critical areas, which must be addressed for its successful implementation in real-world scenarios. Further research is necessary to address these limitations and explore possible solutions that would allow StreaMulT to fully fulfill its potential.

Our transductive approach introduced in [Chapter 7](#) to perform textual classification in FSL paradigm with API-based language models introduces an innovative methodology. Nevertheless, there are several key criticisms that should be addressed in order to truly judge the applicability and efficacy of this approach.

Method Specificity and Inference Latency One primary concern is that this method performs optimally for API-based LLMs. The efficacy of transductive methods as compared to their inductive counterparts, in settings not based on API, was shown to be significantly lower in [Chapter 6](#). This creates a strong limitation on the scope and utility of this method. Moreover, the inherent latency of transductive methods surpasses that of inductive methods. While the training time has been studied thoroughly, the inference time, a critical factor for real-world applications, was not analyzed exhaustively. For this methodology to be applicable in an industrial setting, where real-time responses are often crucial, this aspect needs to be studied in depth.

Dependence on Annotations and Application to Multi-Source Data This approach, even though it employs few annotations, still relies on them. In an industrial setting, one might access a labeled multimodal dataset of faults, but the alignment of these labels with the associated maintenance reports is not guaranteed. Consequently, the challenge arises of how to ensure an appropriate annotation scheme for these reports. Similarly, this approach concentrates solely on textual data. Given the initial goal of incorporating data from multiple sources, a question arises: how can this methodology be implemented in a broader multimodal framework? Some recent works such as (Alayrac, Donahue, et al. 2022) extend the paradigm of pre-trained LLMs to other modalities, such as images, to perform multimodal FSL with In-Context learning paradigm. Exploring an adaptation of transductive framework to these architecture thus constitutes an interesting perspective.

Privacy Concerns and Dependence on Contemporary Framework Finally, while this approach addresses privacy concerns by leveraging API-based models, the underlying assumption may seem unrealistic. It assumes that labels carry the most sensitive data, rather than the input data. In practice, the input data are often also sensitive and thus of higher concern from a privacy perspective, and anonymization of textual documents might undermine expressive content that is already scarce. As a result, the approach might not be as effective in scenarios where privacy of the input data is crucial.

In summary, while our transductive approach offers a promising solution for NLP few-shot classification with API-based LLM, several criticisms highlight the areas where further work is needed. This includes the inference latency, reliance on annotations, suitability for multimodal data integration, and privacy concerns related to input data. Addressing these issues could potentially expand the applicability and usefulness of this approach in varied settings.

8.2.2 LONG-TERM PERSPECTIVES

At the time of AI is preponderant and completely questioning the perspectives of the society, mainly through the Large Language Models breakthrough, the multimodal quest strikes back as a mean of grounding world concepts (Girdhar et al. 2023). As the last section of this manuscript, we chose to discuss what can appear as a philosophical yet central question: "What is a modality?".

Despite the many previous works studying the best ways to perform multimodal data fusion, mainly through representation learning, there is no formal definition of what is called a modality, and therefore how two different sources of data can be considered as coming from either same or different modalities. Indeed, (Baltrusaitis et al. 2019) informally define a *modality* as "the way in which something happens or is experienced" and adds that "a research problem is characterized as multimodal when it includes multiple such modalities". Besides, most of previous works follow this kind of informal definitions, and often give as examples of multimodal data the human experience of the world, through human multisensory integration (see [Subsection 2.2.1](#)).

In light of the *heterogeneity gap* paradigm, a first attempt to define the modalities α, β of two data sources X_α and X_β could be through their definition domain. For instance, a RGB picture (of height h and width w) of a dog in $\mathbb{R}^{h \times w \times 3 \times 256}$ and a text describing a dog, encoded as l different d -length one-hot vectors, lie in vastly different spaces even though they share redundant information and can be semantically close (they both embed the concept of a dog). However, this definition seems insufficient, as two images of different resolutions exist in different spaces, yet intuitively do not exhibit heterogeneity gap. On the other hand, some studies treat similar structured inputs, such as RGB and LIDAR images, as distinct modalities under a multimodal framework. Yet, no framework provides tools to determine if these modalities are closer to each other than an image and a text of a dog are.

The heterogeneity gap paradigm supports the previous informal definitions of modality and multimodality, in the sense that all these considerations are human-centered. The challenge it presents is that a prediction model designed for a modality α may not perform efficiently when applied to a different modality β , as the data structure differs. However, this model design choice is determined according to the assumptions the human learner makes on the input data: namely, an *inductive bias*. In that sense, we propose to define a modality through the lens of inductive biases.

Considering the multimodal fusion framework of [Subsection 2.2.2](#), aside from training algorithm selection, optimization, and loss selection, the model choice is essentially determined by the hypothesis space \mathcal{F} . This parameter space is where an inductive bias can be added, particularly in response to the nature and structure of input data, hence their modality. For example, assuming spatial structures in images such as locality or translation invariance, CNNs are a popular choice due to their ability to share weights locally in space. Similarly, RNN are employed to manage the presumed recurrent structure of text.

By writing $\mathcal{F} = \mathcal{H} \circ \mathcal{G}$ where \mathcal{H} is the representation's hypothesis space and \mathcal{G} is the classifier's hypothesis space, one can define the notion of modality in a relational way: two sources of data X_α and X_β are said to be from the same modality if and only if they are processed with the same representation's hypothesis spaces, *i.e.* $\mathcal{G}_\alpha = \mathcal{G}_\beta$. This means that the learner applies the same bias when encoding them prior to classification. With this consideration, we could define a distance between two modalities using a distance between their representation's hypothesis spaces.

Eventually, if this section is just a discussion and a proposition, we truly believe that the definition and characterization of a modality, and understanding its distance with other modalities

8 Conclusion and Perspectives

could benefit the multimodal learning field by offering frameworks to address the heterogeneity gap challenge more effectively.

9 APPENDIX

9.1 PROOF OF THEOREM 1

In this Appendix, we prove the inequality (Equation 7.4) provided in Proposition 1. The right-hand side of (Equation 7.4) follows straightforwardly from the definition of $R_Q^I(\beta)$ and the non-negativity of the Shannon entropy. In order to prove the first inequality, we need to introduce the following intermediate result.

For any arbitrary random variable X and countable random variable Y , and any real number β , let

$$I_\beta(X; Y) := -\mathbb{E}_{X^*Y} \log \mathbb{E}_X \left[\frac{P(Y|X)}{P(Y|X^*)} \right]^\beta,$$

where the random variable X^* follows the same distribution than X . Notice that it is obvious that $I_1(X; Y) = I(X; Y)$, where $I(X; Y)$ is Shannon Mutual Information.

Lemma 1. For any arbitrary random variable X and countable random variable Y , we have

$$I(X; Y) \geq I_\beta(X; Y), \text{ for } 0 \leq \beta \leq 1.$$

Proof of the lemma: We must show that the different of $I(X; Y) - I_\beta(X; Y)$ is nonnegative. To this end, we write this difference as:

$$I(X; Y) - I_\beta(X; Y) = -\mathbb{E}_{X^*Y} \log \frac{P^{1-\beta}(Y|X^*) \mathbb{E}_X P(Y|X)}{\mathbb{E}_X P^\beta(Y|X)} \quad (9.1)$$

$$\geq -\log \mathbb{E}_{X^*Y} \frac{P^{1-\beta}(Y|X^*) \mathbb{E}_X P(Y|X)}{\mathbb{E}_X P^\beta(Y|X)} \quad (9.2)$$

$$= -\log \sum_{y \in \mathcal{Y}} \mathbb{E}_{X^*} P(y|X^*) \frac{P^{1-\beta}(y|X^*) \mathbb{E}_X P(y|X)}{\mathbb{E}_X P^\beta(y|X)} \quad (9.3)$$

$$= -\log \sum_{y \in \mathcal{Y}} \frac{\mathbb{E}_{X^*} P^\beta(y|X^*) \mathbb{E}_X P(y|X)}{\mathbb{E}_X P^\beta(y|X)} \quad (9.4)$$

$$= -\log \sum_{y \in \mathcal{Y}} \mathbb{E}_X P(y|X) \quad (9.5)$$

$$= 0, \quad (9.6)$$

where the first inequality follows by applying Jensen's inequality to the function $t \mapsto -\log(t)$.

Proof of Proposition 1: From Lemma 1, using Jensen's inequality, we have

$$I(X; Y) = -\mathbb{E}_{X^*Y} \log \mathbb{E}_X \left[\frac{P(Y|X)}{P(Y|X^*)} \right], \quad (9.7)$$

$$\geq -\mathbb{E}_{X^*Y} \log \mathbb{E}_X \left[\frac{P(Y|X)}{P(Y|X^*)} \right]^\beta \quad (9.8)$$

$$\geq -\mathbb{E}_{X^*} \log \mathbb{E}_X \mathbb{E}_{Y|X^*} \left[\frac{P(Y|X)}{P(Y|X^*)} \right]^\beta \quad (9.9)$$

$$= -\mathbb{E}_{X^*} \log \mathbb{E}_X \sum_{y \in \mathcal{Y}} P^\beta(Y|X) P^{1-\beta}(Y|X^*), \quad (9.10)$$

where inequality (9.8) follows by applying Lemma 1 and inequality (9.9) follows by exploiting the convexity of the function $t \mapsto -\log(t)$ for any $0 \leq \beta \leq 1$. Finally, it is not difficult to check from the definition of the Fisher-Rao distance given by expression (7.2) that

$$\cos \left(\frac{d_{\text{FR}}(P(y|X = x), P(y|X = x^*))}{2} \right) = \sum_{y \in \mathcal{Y}} \sqrt{P(y|X = x)P(y|X = x^*)}. \quad (9.11)$$

Using the identity given by (9.11) in expression (9.10) setting $\beta = 1/2$, we obtain the desired inequality

$$I(X; Y) \geq -\mathbb{E}_{X^*} \log \mathbb{E}_X \cos \left(\frac{d_{\text{FR}}(P(y|X), P(y|X^*))}{2} \right). \quad (9.12)$$

The inequality (7.4) immediately follows by replacing the distribution of the random variable X with the empirical distribution on the query and $P(y|x)$ with the soft-prediction corresponding to the feature x , which concludes the proof of the proposition.

9.2 PUBLICATION IN THE CONTEXT OF THE MPO PROJECT

In the next pages, we provide the reader the publication (Pellegrain, Batteux, et al. 2022).

Démonstration de surveillance de défaillances sur un exemple applicatif

Fault monitoring demonstration on an applicative example

PELLEGRAIN Victor
IRT SystemX
2, boulevard Thomas
Gobert
91120 Palaiseau
victor.pellegrain@irt-
systemx.fr

BATTEUX Michel
IRT SystemX
2, boulevard Thomas
Gobert
91120 Palaiseau
michel.batteux@irt-
systemx.fr

LAIR William
EDF R&D
7 *BD Gaspard Monge*
91120 Palaiseau
william.lair@edf.fr

KACZMAREK Michel
Airbus Protect
1 Bd Jean Moulin
CS 70562
78996 Elancourt Cedex
Michel.Kaczmarek@apsys-
airbus.com

Résumé — La gestion de la maintenance d'installations industrielles de production est un facteur important de compétitivité. Différentes techniques existent afin d'assurer au mieux les stratégies de maintenance, par exemple la surveillance et le diagnostic permettant de détecter et d'identifier une défaillance à la suite de son occurrence. Les travaux présentés dans cette publication consistent à montrer l'application d'un algorithme de surveillance pour détecter des occurrences de défaillances sur un exemple applicatif virtuel du projet de recherche MPO, pour Maintenance Prévisionnelle et Optimisation, de l'IRT SystemX. L'exemple est le système 3-Réservoirs, déjà présenté dans une précédente communication, et nous y avons appliqué un algorithme d'apprentissage automatique afin de construire un outil de surveillance de défaillances.

Mots-clefs — *Surveillance, Diagnostic, Réseau de neurones récurrents, LSTM*

Abstract — Managing the maintenance of industrial plants is an important factor of competitiveness. Different techniques can be used to ensure maintenance strategies: fault monitoring and diagnosis, for instance, to detect and identify a failure after it occurs. Works presented within this publication show the application of a monitoring algorithm to detect occurrences of failures on an applicative example. These works are realized within the MPO project (Predictive maintenance and Optimization) at IRT SystemX. The example is the 3-Tanks system, already presented in previous works. A machine learning algorithm was implemented, based on data generated by simulation.

Keywords — *Monitoring, Diagnosis, Recurrent neural network, LSTM*

I. INTRODUCTION

La gestion de la maintenance d'installations industrielles de production est un facteur important de compétitivité. En effet, de tels systèmes sont composés d'une multitude de composants hétérogènes en interactions les uns avec les autres : des composants physiques, des actionneurs, des capteurs, des calculateurs de contrôle/commande. Ajoutons que certains composants embarquent en eux-mêmes de tels éléments logiciels de contrôle/commande, comme les capteurs

dits 'intelligents'. De plus certains de ces systèmes peuvent être distribués en différents endroits physiques, demandant de ce fait des liens de connexions par réseaux (internet par exemple). De tels systèmes combinant des composants physiques, logiciels et en réseaux sont également appelés des 'systèmes cyber-physiques' [10].

Les composants et parties de ces systèmes sont naturellement sujets à des défaillances (qui se nomment également fautes dans la communauté du diagnostic), pouvant mener à des dysfonctionnements ou pannes du système. Certaines de ces défaillances peuvent avoir des conséquences négligeables, même si le système ne remplit plus sa fonction : par exemple l'oxydation d'un câble de haut-parleur, qui occasionne soit un mauvais son, soit pas de son, sortant du haut-parleur, et impactant le confort de l'utilisateur. D'autres défaillances peuvent, au contraire, avoir des conséquences catastrophiques : par exemple l'usure de joints d'étanchéité de durites de freinage, qui amène à un dysfonctionnement, voire même une perte d'un système de freinage. Dans ce cadre et suivant la sévérité des dysfonctionnements et pannes du système considéré, il est nécessaire de mettre en œuvre des solutions de maintien en conditions opérationnelles du système. Même si l'amélioration de la fiabilité des composants, ou les techniques de redondances matérielles, peuvent être des solutions, elles ne sont néanmoins pas suffisantes. En effet, tout composant physique est lié à l'usure matérielle et mènera à des dysfonctionnements ou des pannes. La maintenance joue donc un rôle important pour réduire les risques d'occurrence de pannes, en particulier pour des systèmes dont la panne peut impacter la sécurité des personnes.

Différentes stratégies de maintenances existent, et sont résumées en Figure 1. Les maintenances correctives se réalisent à la suite des occurrences des défaillances. À l'inverse les maintenances préventives anticipent les défaillances en se réalisant avant leurs occurrences. La maintenance préventive est un levier important pour réduire les risques de panne et les coûts de maintenance. Cependant,

réaliser trop d'actions de maintenance préventives pourrait se révéler plus coûteux que nécessaire. Il existe un équilibre entre l'investissement en maintenance préventive et le risque de défaillance. Une analyse de la fiabilité du système, à travers l'étude des données historique de panne et/ou l'éllicitation d'expert de son fonctionnement, permettra de calculer des indicateurs d'aide à la décision permettant de trouver un équilibre optimal.

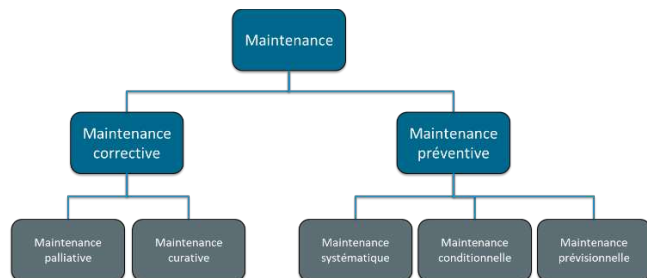


Figure 1 : Les différents types de maintenances

Différentes techniques existent afin d'assurer au mieux les stratégies de maintenance. La surveillance et le diagnostic permettent de détecter et d'identifier un comportement anormal du système (défaillance de l'un des composants) avant que cela ait un impact important. Le pronostic permet d'estimer la durée avant l'occurrence de la défaillance ou la panne (nous reviendrons sur ces notions en section II.C). La première technique est principalement utile dans le cadre de maintenances conditionnelles, et la seconde l'est principalement dans le cadre des maintenances prévisionnelles. Néanmoins quelle que soit la technique, il est nécessaire d'avoir une connaissance du fonctionnement et des dysfonctionnements du système.

Dans le cadre des travaux présentés dans cette publication, nous nous intéressons à l'application de techniques et d'algorithmes de surveillance et diagnostic pour détecter des occurrences de défaillances sur un exemple applicatif. Ces travaux sont réalisés au sein du projet de recherche MPO, pour Maintenance Prévisionnelle et Optimisation, de l'IRT SystemX¹. Ce projet, en partenariat avec plusieurs acteurs industriels et académiques, porte sur l'optimisation des stratégies de maintenance des systèmes de production. L'exemple applicatif considéré est un système virtuel construit durant ce projet : le système 3-Réservoirs présenté dans [2]. Nous avons appliqué un algorithme d'apprentissage automatique sur des données générées par simulation du système 3-Réservoirs.

La suite de cette publication est organisée de la manière suivante. La section II fera un rappel de l'état de l'art sur la surveillance et le diagnostic. Cette section II nous permettra de justifier d'une part la définition de défaillances du système 3-Réservoirs, ainsi que le choix d'un algorithme de diagnostic basé sur les données. La section III fera une présentation succincte du système 3-Réservoirs, issu des travaux présentés dans [2]. Les sections IV et V montreront l'implémentation de l'algorithme de surveillance du système 3-Réservoirs, ainsi que les premières expérimentations réalisées. La section VI discutera des perspectives envisageables sur ces travaux. Enfin la dernière section VII conclura cette publication.

¹ www.irt-systemx.fr/projets/mpo

II. RAPPEL D'ETAT DE L'ART SUR LA SURVEILLANCE ET LE DIAGNOSTIC

Comme indiqué en introduction, les défaillances de composants ou parties d'un système ne peuvent être complètement évitées. Un levier pour limiter le risque d'occurrence de défaillance du système ou de sa conséquence est de mettre en place des techniques permettant de détecter au plus vite une anomalie. Ces techniques sont connues sous les termes de 'surveillance' et 'diagnostic'. Il y a deux principales approches pour la surveillance et le diagnostic [9] : les approches dites 'basées modèles' et les approches dites 'basées données'.

A. Les approches basées modèles

Les approches à base de modèles consistent à comparer le comportement réellement observé du système à un comportement prédit, issu d'un modèle de fonctionnement nominal et avec défaillances du système. Les modèles utilisés par ces méthodes peuvent être de deux types : les modèles quantitatifs et les modèles qualitatifs.

Les approches par modèles quantitatifs sont celles issues de la communauté de l'automatique, et classiquement nommées par l'acronyme FDI pour 'fault detection and isolation'. L'utilisation d'un modèle de fonctionnement nominal du système permet d'engendrer des incompatibilités entre le comportement réel du système et celui prédit par le modèle. Ces incompatibilités, appelées 'résidus', sont générées à partir des mesures effectuées sur le système et de calculs fondés sur le modèle du système. Ces résidus sont des signaux devant refléter la cohérence des données mesurées du système par rapport au modèle de fonctionnement. L'objectif d'un résidu est d'être sensible aux défaillances : c'est-à-dire qu'il doit refléter l'éventuelle présence d'une défaillance. Cela signifie donc qu'un résidu est en général proche d'une valeur de référence si aucune défaillance n'affecte le système, et qu'il est dévié vers une valeur différente dès l'occurrence d'une défaillance.

Les approches basées sur les modèles qualitatifs sont celles issues de la communauté de l'intelligence artificielle (communauté historique, et pas celle actuelle liée à l'apprentissage automatique), et nommées par l'acronyme DX pour 'Data eXtraction'. Les modèles qualitatifs permettent d'abstraire, à un certain degré, le comportement du système à travers des modèles de type symbolique. Ces modèles décrivent d'une manière qualitative l'espace d'état continu du système et ne représentent pas la physique du système, contrairement aux modèles quantitatifs, car ils le décrivent en termes de mode de fonctionnement. Les méthodes à base de modèles qualitatifs peuvent être classifiées soit selon le niveau d'abstraction considéré du système à diagnostiquer (les graphes causaux pour les systèmes continus, les systèmes à événements discrets, ou encore les systèmes hybrides dynamiques) ; soit selon la prise en compte, ou non, des défaillances (les modèles de dysfonctionnement comme dans les techniques de propagation des défaillances ou pour les graphes causaux, ou les modèles de bon fonctionnement dans le cas du diagnostic à partir des principes premiers ou par simulation qualitative).

B. Les approches basées données

Contrairement aux méthodes à base de modèles, celles à base de données reposent sur un nombre important de données

qui sont supposées représenter convenablement le système. Les seules informations disponibles sont les signaux issus des capteurs du système, ce qui implique que ces approches présupposent donc que ce système puisse être complètement décrit par ses observations passées et présentes. L'objectif de ces approches est alors de construire un modèle ajusté sur les données collectées, et la principale difficulté va donc être de définir non seulement la structure appropriée du modèle, mais aussi le calage approprié entre ce modèle et le système.

Les méthodes par reconnaissance de formes ont pour objectif de classer des objets, nommés des '*formes*', qui sont représentées par des données, dans des classes prédéterminées en les comparant à des prototypes. Ces méthodes reposent donc sur une description complète de ces formes et de chacune des différentes classes prototypes. Un problème de diagnostic peut ainsi se définir comme un problème de reconnaissance de formes où les classes sont les modes de fonctionnement du système (nominal ou sous la présence de défaillances) et les formes sont représentées par les observations du système.

Les méthodes par systèmes experts sont utilisées dans des applications où l'expertise humaine y est importante et le développement de modèles y est faible. Ce sont des systèmes à base de règles du type 'si', 'et', 'ou', 'alors' qui utilisent une information heuristique pour lier les symptômes aux défaillances, établissant ainsi des associations empiriques entre effets et causes des défauts. Ces associations sont généralement fondées sur l'expérience de spécialistes, dits '*experts*', plutôt que sur une connaissance de la structure et/ou du comportement du système. Leur fonctionnalité est de trouver la cause de ce qui a été observé en parcourant, par un raisonnement abductif, les règles préalablement établies.

Enfin les méthodes par apprentissage machine (ML pour '*Machine Learning*') appréhendent également la problématique de la surveillance et du diagnostic [8]. De plus récents travaux, [1] et [13] par exemple, motivent d'ailleurs leur démarche par l'apparition de nouveaux challenges pratiques liés à l'arrivée de l'industrie dite '4.0', comme notamment la capacité à gérer des quantités massives de données multi-sources en temps rapide. Ces études présentent les approches de ML comme plus adaptées lorsque les profils de défaillances sont complexes. Les approches utilisent des réseaux de neurones, des outils de traitement du signal (transformées de Fourier et de Laplace), etc.

C. Les notions de défaillances, dysfonctionnements, et pannes

Quelles que soient les approches de surveillance et diagnostic basées modèles ou basées données, nous considérons des défaillances pouvant mener à des dysfonctionnements ou des pannes. Nous présentons donc ces notions, que nous reprenons de [9] :

- Une *défaillance*, également nommée '*faute*' par la communauté du diagnostic, est une dérive non-permise d'au moins une propriété caractéristique du système par rapport aux conditions standard et acceptables de fonctionnement du système. Une défaillance est un état anormal de fonctionnement du système pouvant causer une réduction, voire une perte de la capacité de l'unité fonctionnelle à exécuter sa fonction requise. Une défaillance est indépendante du fait que le système soit opérationnel ou non et peut très bien ne pas affecter le fonctionnement normal du

système. Enfin une défaillance peut initier un dysfonctionnement ou une panne du système.

- Un dysfonctionnement est une irrégularité intermittente dans la réalisation d'une fonction désirée du système. Un dysfonctionnement est donc une interruption temporaire de la fonction du système, et il s'agit d'un événement résultant d'un ou plusieurs défauts.
- Enfin une panne est une interruption permanente de la capacité du système à exécuter une fonction requise sous des conditions opérationnelles spécifiées. Comme pour un dysfonctionnement, une panne est un événement résultant d'un ou plusieurs défauts. Différents types de pannes peuvent être distingués suivant leurs nombres (panne simple ou pannes multiples) et leurs prévisions (panne aléatoire donc non prévisible, panne déterministe donc prévisible sous certaines conditions, panne systématique ou causale dépendant de conditions connues).

Selon [3], une défaillance peut être spécifiée par trois caractéristiques : son comportement, son effet et sa conséquence. Le comportement d'une défaillance qui détermine son instant d'occurrence dans le temps, sa force d'apparition ainsi que sa durée de présence. L'instant d'occurrence peut être aléatoire, systématique ou dépendant d'un événement interne ou externe au système. La force d'apparition peut être brusque ou progressive. La durée de présence d'une défaillance peut être permanente, transitoire ou intermittente. L'effet d'une défaillance détermine sa prise en compte dans le système. Il s'agit de déterminer sa localisation dans le système ainsi que la ou les perturbations induites. Enfin la conséquence engendrée par une défaillance, sur le système lui-même et/ou son environnement, sont à déterminer suivant les pertes potentielles (matérielles et/ou humaines) qu'il peut générer. Ces caractéristiques permettent de bien définir une défaillance afin de la modéliser si nécessaire.

III. LE SYSTÈME 3-RÉSERVOIRS

Le système 3-Réservoirs, présenté dans [2], est un système dynamique hybride, au sens où ils combinent des phénomènes qui seront décrits par des évolutions continues et des phénomènes qui seront décrits par des évolutions discrètes. Comme montré en Figure 2, ce système est constitué de différents composants : deux réservoirs amonts L1 et L2 et un réservoir aval L3, deux pompes P1 et P2, trois vannes V1, V2 et V3, ainsi que trois capteurs CH1 CH2 et CH3 de hauteurs d'eau dans chaque réservoir, et un capteur de température CT3 dans le réservoir L3. Le réservoir aval L3 contient une source de chaleur qui fonctionne en continu et qui doit être refroidie par de l'eau froide venant des deux réservoirs L1 et L2 en amont.

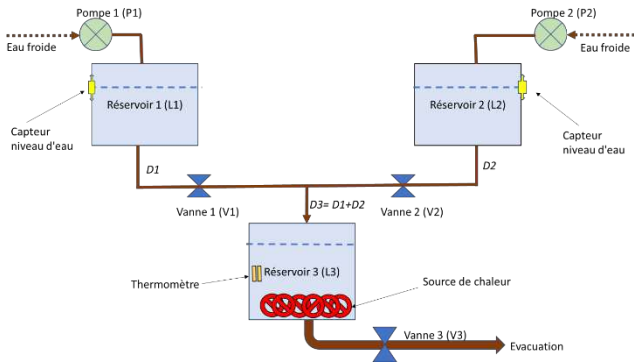


Figure 2 : Représentation schématique du système 3-Réservoirs

A. Fonctionnement du système 3-Réservoirs

L'objectif du système 3-Réservoirs est de refroidir la source de chaleur dans le réservoir L3 avec de l'eau dans les réservoirs L1 et L2. Pour cela il est nécessaire d'assurer un certain niveau de température et de hauteur d'eau dans ce réservoir L3.

L'eau circule de la façon suivante. Les deux réservoirs L1 et L2 sont alimentés par deux sources froides indépendantes grâce aux deux pompes P1 et P2. Ces réservoirs L1 et L2 alimentent en eau le troisième réservoir L3 dans lequel se situe la source de chaleur. L'alimentation de L3 par L1 est gérée par la vanne V1, et l'alimentation de L3 par L2 est gérée par la vanne V2. Enfin l'évacuation de l'eau de L3 est gérée par la vanne V3. Initialement, les deux pompes P1 et P2 fonctionnent et les vannes V1, V2 et V3 sont ouvertes.

Les fonctionnements des ouvertures et fermetures des vannes V1, V2 et V3 dépendent du niveau d'eau dans le réservoir L3. Les vannes V1 et V2 se ferment quand la hauteur d'eau dans le réservoir L3 dépasse une certaine valeur seuil maximum, correspondant à un niveau maximum dans les réservoirs, et elles s'ouvrent quand la hauteur est inférieure à une certaine valeur seuil minimum. La vanne V3 s'ouvre quand la hauteur dépasse la valeur de seuil maximum, et se ferme quand la hauteur est inférieure à la valeur de seuil minimum.

Les fonctionnements des démarrages et arrêts des pompes sont similaire aux fonctionnements des ouvertures et fermeture des vannes. La pompe P1, respectivement P2, démarre quand la hauteur d'eau dans le réservoir L1, respectivement L2, est inférieure à une valeur seuil minimum ; et elle s'arrête quand cette hauteur d'eau est supérieure à une valeur seuil.

B. Dysfonctionnements du système 3-Réservoirs

Les trois évènements redoutés considérés sur ce système 3-Réservoirs sont les suivants :

- Le réservoir L3 a débordé ;
- Le réservoir L3 est vide ;
- La température dans L3 a dépassé un niveau critique.

Les défaillances menant aux dysfonctionnements du système 3-Réservoirs sont multiples. Des défaillances intempestives des vannes : une vanne peut soit se coincer dans l'état dans lequel elle se trouve au moment de la défaillance, soit changer brusquement d'état, c'est-à-dire s'ouvrir si elle est fermée ou se fermer si elle est ouverte. Les défaillances

intempestives des pompes ont les mêmes comportements que celles des vannes. Enfin pour chaque réservoir, une fuite qui apparaît à la suite d'une fissure de la paroi.

C. Modélisation et génération de données du système 3-Réservoirs

Dans [2], nous présentons la modélisation et la génération de données simulées, c'est-à-dire des séries temporelles, pour ce système 3-Réservoirs. Nous avons en effet généré des séries temporelles en fonctionnement normal, et des séries temporelles avec les défaillances.

Le système a été modélisé par un PDMP pour 'Piecewise Deterministic Markov Process' (voir [5] et [6]), et les séries temporelles ont été générées en simulant, par Monte-Carlo, à l'aide de l'outil PyCATSHOO (Pythonic Object Oriented Hybrid Stochastic AuTomata) développé par EDF R&D [4].

IV. IMPLEMENTATION DE L'ALGORITHME DE SURVEILLANCE

Nous avons utilisé les séries temporelles en fonctionnement normal et avec les défaillances des composants afin de produire un outil de surveillance de ce système 3-Réservoirs. La partie surveillance est donc celle qui permet d'établir si le système est en bon fonctionnement ou en fonctionnement dégradé à la suite de l'occurrence d'une défaillance d'un des composants. Nous avons utilisé une approche basée sur les données avec un algorithme d'apprentissage machine.

La construction de l'outil de surveillance s'est réalisée en trois étapes. La première étape a consisté à prétraiter les données. La deuxième étape a consisté à définir et entraîner un modèle d'apprentissage. Enfin la troisième étape a consisté à construire l'outil de surveillance par rapport au modèle d'apprentissage entraîné. Cette implémentation est inspirée de [12].

A. Prétraitement des données

Les séries temporelles, issues de la base de données générées dans [2], ont été prétraitées afin de concaténer les valeurs des capteurs, les valeurs des manœuvres sur les actionneurs (c'est-à-dire les ouvertures et fermetures des vannes, et les démarrages et arrêts de pompes) et les valeurs des défaillances des vannes et des pompes. Certaines modifications ont également été réalisées sur ces séries temporelles. Dans la suite, une défaillance correspond à une défaillance d'un des composants (vannes ou pompes) et pas à la défaillance du système.

D'abord une étiquette (label) a été ajoutée pour réaliser la surveillance. Cette étiquette est à la valeur 0 quand le système n'a pas eu de défaillance à l'instant de temps courant considéré. Cette étiquette est à la valeur 1 à partir de l'instant de temps d'occurrence d'une défaillance (quelconque).

Ensuite les différentes défaillances ont été scindées en deux ensembles. Un ensemble des défaillances visibles des vannes et des pompes : 'blocage en position fermée d'une vanne ouverte' et 'blocage en position ouverte d'une pompe fermée' pour les vannes, et 'blocage en position démarrée d'une pompe arrêtée' et 'blocage en position arrêtée d'une pompe en fonctionnement' pour les pompes. Un ensemble des défaillances invisibles : 'blocage en position ouverte d'une vanne ouverte' et 'blocage en position fermée d'une vanne fermée' pour les vannes, et 'blocage en position arrêtée d'une pompe arrêtée' et 'blocage en position démarrée d'une pompe en fonctionnement' pour les pompes. Seules les défaillances

visibles ont été étiquetées avec la valeur 1. En effet, les défaillances dites invisibles ne sont pas visibles via les mesures des capteurs ; il est ainsi impossible de réaliser la tâche de détection car il n'y a pas d'information sur l'occurrence de cet événement dans les données.

B. Entraînement du modèle de prévision

Le modèle de prévision est un réseau de neurone récurrent de type LSTM, pour 'Long Short-Term Memory' [8], modélisant la dépendance temporelle des capteurs. Ce modèle a été entraîné de façon semi-supervisée sur les séries temporelles saines, c'est-à-dire celles pour lesquelles aucun dysfonctionnement n'a été généré. L'entraînement est réalisé sur une fenêtre glissante de taille L . Au temps t , le modèle estime les valeurs de capteurs des temps $t + 1$ à $t + L$. Ainsi, pour un même pas de temps τ , on peut obtenir L prévisions d'horizons temporels variables (de 1 à L), selon si on se place à $\tau - 1$ ou jusqu'à $\tau - L$. Ces prévisions sont stockées dans un vecteur \hat{x}_τ de taille L , et on peut calculer le vecteur d'erreur e_τ correspondant :

$$e_\tau = \hat{x}_\tau - x_\tau \cdot \mathbf{1}_L$$

en notant $\mathbf{1}_L$ le vecteur de taille L ne contenant que des 1. De là, on peut calculer la moyenne et la variance empiriques de ces vecteurs d'erreur correspondant à un comportement sain du système :

$$\mu = \frac{1}{n} \sum_{i=1}^n e_i, \quad \Sigma = \frac{1}{n} \sum_{i=1}^n (e_i - \mu)(e_i - \mu)^T$$

qui seront utiles pour l'outil de surveillance. L'indice i itère sur l'ensemble des pas de temps de l'ensemble des trajectoires d'entraînement du modèle, pour un total de n points.

Pour l'entraînement du modèle de prévision, les données d'entrée correspondent aux séries temporelles des 4 capteurs (3 de niveaux d'eau et 1 de température), de la première jusqu'à l'avant-dernière mesure (inclusive). Les labels à prédire correspondent à ces mêmes séries temporelles décalées d'un pas de temps dans le futur : de la deuxième mesure jusqu'à la dernière (inclusive) ; le but étant de prédire la prochaine mesure de capteurs à partir des précédentes. Ces valeurs sont enfin standardisées (centrées et réduites). L'outil de surveillance utilisant les prévisions du modèle de prévision LSTM sur un horizon temporel variable (majoré par le paramètre L), il est nécessaire d'entraîner ce LSTM à réaliser des prévisions récursives précises (multi-step). Cet objectif pouvant impliquer un comportement instable lors de l'entraînement (les erreurs de prévision s'accumulent au fil des étapes), on ajoute à la fonction de perte multi-step une fonction de perte one-step, pénalisant l'erreur du modèle sur un seul pas de temps. Ceci est réalisé en suivant une procédure de teacher forcing [14], redonnant la vérité terrain au modèle à chaque pas de temps pour guider son apprentissage. Enfin, pour renforcer cette notion de guidage, on ajoute une dernière fonction de perte, visant à minimiser l'écart entre les états cachés du LSTM, entre la prévision multi-step ou la prévision one-step.

La fonction de perte utilisée est la Mean Squared Error (MSE), et l'outil d'optimisation utilisé pour la descente de gradient est la méthode d'Adam [7].

C. Construction de l'outil de surveillance

L'algorithme de surveillance consiste en la comparaison d'un score d'anomalie à un seuil, permettant de discriminer

entre les comportements normaux et les comportements avec les défaillances.

Pour chaque trajectoire, le score d'anomalie s_τ , correspondant au pas de temps τ , est calculé en fournissant le vecteur d'erreur e_τ à un modèle gaussien multivarié, paramétré par μ et Σ :

$$s_\tau = (e_\tau - \mu)\Sigma^{-1}(e_\tau - \mu)^T$$

Ce score d'anomalie s_τ est ensuite comparé à un seuil ϵ (hyperparamètre optimisé sur un espace de validation) pour obtenir la prédiction du modèle \hat{y}_τ sur la présence de défaillance au temps τ :

$$\hat{y}_\tau = \mathbf{1}_{s_\tau > \epsilon}$$

avec $\mathbf{1}$ la fonction indicatrice.

V. EXPERIMENTATIONS

Les expérimentations sont évaluées via le calcul de plusieurs métriques dépendant de la valeur du seuil ϵ , comme la précision, le rappel, et le score F1. Afin de garantir le meilleur équilibre entre faux-positifs et faux-négatifs, nous avons choisi cette dernière comme métrique de décision pour la valeur du seuil ϵ . Sur la Figure 3, on observe l'évolution de ces métriques sur un ensemble de validation, selon la valeur du seuil choisi. Sur notre jeu de test, l'outil de diagnostic affiche un score F1 de 0.9555

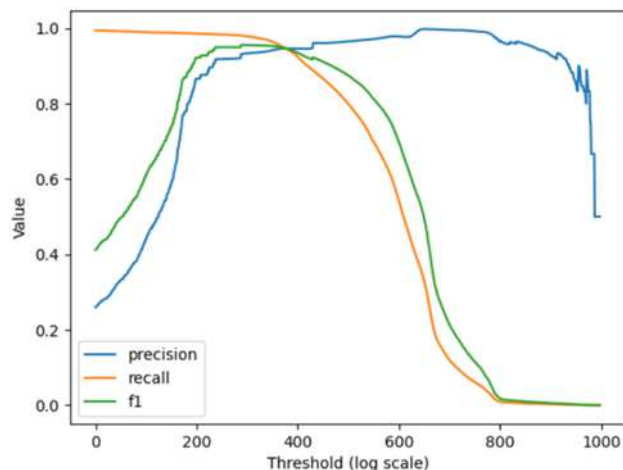


Figure 3 : Evolution des métriques selon la valeur du seuil

VI. PERSPECTIVES

Les travaux présentés dans cette publication concernent la mise en place d'un outil de surveillance, et donc de détection de défaillances du système virtuelle 3-Réservoirs. Ces travaux peuvent être étendus et poursuivis suivant différentes orientations afin d'atteindre un niveau plus élevé de polyvalence, de performance et de généricité.

A. Introduction d'une notion de temporalité

Une première perspective de poursuite serait d'intégrer la notion de temporalité dans la détection d'une défaillance. En effet une exigence communément définie pour un tel outil de surveillance et de diagnostic concerne la temporalité : c'est-à-dire le délai entre l'instant où la défaillance apparaît, et l'instant où elle est détectée, puis isolée et identifiée.

Il y a des cas où ce délai doit être court afin de mettre le système dans un mode sûr. Ce délai doit, bien entendu, être

mis en relation avec la sévérité de la défaillance et la dynamique de ses conséquences. Cette exigence de délai de détection peut de plus impacter le maintien des performances du système. En effet, un tel outil conçu avec une exigence de délai de détection rapide sera très certainement sensible aux bruits ou perturbations furtives (courtes et temporaires), ce qui impliquera une augmentation potentielle des fausses alarmes en fonctionnement normal et impactera ainsi les performances du système.

Dans l'état actuel de l'implémentation de l'algorithme de surveillance, il est nécessaire d'y apporter des modifications complémentaires.

B. Considération de données complémentaires « en l'état »

Une deuxième perspective de poursuite serait de tester l'algorithme sur d'autres données du système 3-Réservoirs, mais sans changer ce système 3-Réservoirs, plus précisément sans changer le modèle de simulation. En effet, les travaux réalisés ont considéré un ensemble de données générées initialement pour une problématique de pronostic (voir [2]) ; ce qui a potentiellement un impact sur la pertinence des données dans un cadre de détection et diagnostic de défaillances, et qui demanderait à être évalué.

Pour le moment, et comme expliqué dans la partie IV, le jeu de données a été divisé en deux parties : une partie servant à l'entraînement du modèle et une autre partie servant de tests, ce qui est d'ailleurs une approche classique. L'ajout d'autres données simulées, suivant bien sûr d'autres consignes de fonctionnement du système 3-Réservoirs, devraient ajouter de la précision dans le modèle de surveillance. Cette deuxième poursuite nécessiterait de réaliser de nouvelles simulations du modèle du système 3-Réservoirs.

C. Prise en compte du diagnostic

Une troisième perspective de poursuite serait d'implémenter la partie diagnostic, plus précisément les étapes d'isolation et d'identification d'une défaillance. En effet, en l'état seule la partie surveillance, c'est-à-dire la détection des occurrences de défaillances, est implémentée. De plus comme le modèle 3-Réservoirs et les simulations générées n'ont pas été initialement construits pour une approche de diagnostic, le passage au diagnostic nécessite des travaux complémentaires à plusieurs niveaux : au niveau du modèle, au niveau des simulations, et au niveau de l'outil de surveillance/diagnostic.

D. Modifications du modèle du système 3-Réservoirs

Au niveau des modifications du modèle du système 3-Réservoirs, nous pouvons envisager différentes perspectives. En premier lieu l'ajout de défaillances ou de pannes. Par exemple un encrassement dans les tuyaux ou les pompes ou encore les vannes mènerait à de mauvais débits qu'il serait possible de modéliser sous la forme d'ajouts d'aléas dans ces calculs de débits dans le modèle. Par exemple encore des fuites des réservoirs qui seraient causées par des fissures sur les parois de ces réservoirs modélisées (les fissures) au moyen d'un processus Markovien pour la taille de la fissure et sa hauteur sur le réservoir.

Il serait également possible de rendre des défaillances qui ne sont pas diagnosticables actuellement en défaillances qui deviendraient diagnosticables par l'ajout de tests dans le modèle. Cela équivaldrait à rajouter une instance virtuelle d'un outil de surveillance dans le modèle afin de fournir les informations de tests.

Ces perspectives nécessitent donc de modifier le modèle de différentes manières :

- Soit en ajoutant de nouveaux observateurs dans le modèle, c'est-à-dire des variables d'intérêt qui n'ont pas d'impact sur les phénomènes physiques représentés ;
- Soit en modifiant les phénomènes physiques représentés au moyen de nouvelles variables et de nouvelles relations liant ces variables, avec potentiellement des impacts sur les variables et relations existantes ;

Ces modifications signifient par la suite de réaliser de nouvelles simulations, comme nous allons l'expliquer dans la sous-partie suivante.

E. Modifications au niveau des simulations du système 3-Réservoirs

Au niveau des simulations, nous pouvons envisager soit la réalisation de nouvelles simulations, soit la modification des simulations existantes.

La réalisation de nouvelles simulations sera nécessaire dans le cas où le modèle a été modifié, comme expliqué dans les perspectives indiquées dans les sous-parties précédentes. Dans le cas où le modèle n'intègre que de nouveaux observateurs, ce pourront être les simulations existantes qui seront rejouées, afin de capturer, dans les données, ces nouvelles observations. Dans le cas où le modèle intègre de nouveaux phénomènes, il faudra d'une part définir les plans de simulation, c'est-à-dire spécifier quelles sont les consignes et trajectoires à simuler, car les simulations existantes seront obsolètes et ne pourront être rejouées, et il faudra d'autre part réaliser ces nouvelles simulations suivant ces nouveaux plans de simulation.

Pour la modification des simulations existantes, il s'agit par exemple de supprimer certaines valeurs ou ensembles de valeurs. Ces suppressions peuvent être soit suivant les observateurs, c'est-à-dire de supprimer les données d'un ou plusieurs observateurs, soit suivant une plage temporelle de fonctionnement. Il peut également s'agir de modifier certaines valeurs, par exemple en ajoutant une valeur aléatoire pour représenter du bruit, ou encore d'ajouter des nouvelles données construites via les données existantes.

Enfin à la suite de la production de nouvelles simulations, ou la modification des simulations existantes, il sera nécessaire d'en faire un prétraitement, c'est-à-dire de les mettre au bon format, afin que l'algorithme de surveillance et de diagnostic puisse les considérer.

F. Implémentation de l'algorithme de diagnostic

Au niveau de l'outil de surveillance/diagnostic, nous pouvons envisager l'implémentation d'algorithmes dédiés pour le diagnostic. Les algorithmes abordant une vision Machine Learning se distinguent selon s'ils traitent de la détection et de l'isolation/identification de manière simultanée, ou de manière séquentielle.

Pour le cas séquentiel, les données d'entrée du module de diagnostic correspondent aux plages temporelles des données ayant conduit à une prévision de défaillance de la part du module de détection. Dans ce cas-là, il serait possible de réutiliser l'approche de détection déjà implémentée comme première brique du modèle de diagnostic global. Dans le cas

d'une détection et isolation/identification simultanées, la majorité des algorithmes se placent dans un cadre supervisé, et conçoivent un modèle de classification en N+1 classes, composée d'une classe correspondant à un fonctionnement normal du système, et de N classes de défaillances différentes.

Les modèles de classification sont en général précédés d'un module d'extraction de 'features' permettant de représenter les données d'entrée sous une forme exploitant leurs caractéristiques pertinentes pour faciliter la tâche de classification. Cela peut être réalisé de façon automatique ou sur la base de compréhension du phénomène physique, et est communément appelé 'feature engineering'.

Dans le cadre du jeu de données 3-Réservoirs, il pourrait être possible d'utiliser des outils de traitement du signal, tels que présentés en sous-partie II.B de l'état de l'art (transformées de Fourier, transformées de Laplace, ou en ondelettes dans le domaine temps-fréquence). Le module de classification pourra ensuite exploiter ces 'features', notamment via l'utilisation de SVM, de réseaux de neurones peu profonds, ou de forêts aléatoires. Des méthodes d'apprentissage profond, intégrant la phase d'apprentissage de représentation de manière automatique, peuvent également s'appliquer à ce jeu de données : des réseaux de neurones convolutifs, des réseaux de neurones récurrents profonds, des transformers, ou des auto-encoders.

VII. CONCLUSION

Dans cette publication, nous avons montré l'implémentation d'un algorithme de surveillance sur un exemple virtuelle du projet MPO de l'IRT SystemX. Cet exemple, nommé système 3-Réservoirs, est constitué d'un ensemble de composants (pompes, vannes, réservoirs, capteurs) sujets à des défaillances. De précédent travaux ont montré la modélisation et la génération de données, plus précisément des séries temporelles, sur cet exemple.

Nous nous sommes donc servis de ces données générées pour définir et implémenter un outil de surveillance de ce système 3-Réservoirs. Nous avons utilisé un modèle d'apprentissage de type réseau de neurone récurrent (plus précisément de type LSTM), qui a été entraîné sur les séries temporelles sans les défaillances. L'algorithme implémenté de surveillance a consisté en un vecteur d'erreur, issu du modèle appris, fourni à un modèle Gaussien afin de produire un score d'anomalie. Ce score est ensuite comparé à un seuil permettant de discriminer entre les comportements normaux et les comportements avec les défaillances.

Nous avons ensuite présenté différentes perspectives permettant de compléter ces travaux dans différentes directions, soit en augmentant l'ensemble des données générées à partir du modèle du système 3-Réservoirs, soit en modifiant le modèle du système 3-Réservoirs, enfin soit en modifiant l'algorithme implémenté. Les objectifs principaux de ces compléments étant d'une part de traiter la partie diagnostic, c'est-à-dire d'identifier la défaillance apparue, et d'autre part d'ajouter des défaillances à diagnostiquer, ou à minima d'en rendre certaines actuelles diagnosticables.

REMERCIEMENTS

Ces travaux ont été réalisés dans le cadre du projet MPO de l'Institut de Recherche Technologique SystemX. Ils ont été soutenus par le gouvernement Français au travers du programme "France 2030".

RÉFÉRENCES

- [1] A. Angelopoulos, E.T. Michailidis, N. Nomikos, P. Trakadas, A. Hatziefremidis, S. Voliotis, T. Zahariadis. "Tackling Faults in the Industry 4.0 Era—A Survey of Machine-Learning Solutions and Key Aspects". *Sensors* 2020, 20, 109. <https://doi.org/10.3390/s20010109>
- [2] M. Batteux, J. Foulliaron, W. Lair, Y. Souami. "Génération de données pour le diagnostic et le pronostic : un exemple applicatif", Actes du congrès Lambda-Mu 22 (actes électroniques). IMdR, Le Havre, France. 2020.
- [3] Michel Batteux, Philippe Dague, Nicolas Rapin & Philippe Fiani . "Caractérisation du comportement observable d'un système pour l'étude de la diagnosticabilité de défauts". QUALITA. Angers, France. mars 2011.
- [4] H. Chraïbi, J. C.Houdebine & A. Sibling. "PyCATSHOO: Toward a new platform dedicated to dynamic reliability assessments of hybrid systems". PSAM, 2016.
- [5] M.H.A. Davis. "Piecewise Deterministic Markov Processes: A general class of non-diffusion stochastic models." In *Journal of the Royal Statistical Society. Series B (Methodological)*, 46(3), pp. 353-388, 1984.
- [6] J. Devooght. "Dynamic Reliability". Springer, 1997.
- [7] Diederik P. Kingma and Jimmy Lei Ba. "Adam: A method for stochastic optimization". ICLR 2015
- [8] S. Hochreiter, J. Schmidhuber. "Long Short-Term Memory". *Neural Comput* 1997; 9 (8): 1735-1780. doi: <https://doi.org/10.1162/neco.1997.9.8.1735>
- [9] Rolf Isermann. "Fault-Diagnosis Systems". Springer Berlin, Heidelberg. 2006. DOI 978-3-540-30368-8.
- [10] S. K. Khaitan and J. D. McCalley, "Design Techniques and Applications of Cyberphysical Systems: A Survey," in *IEEE Systems Journal*, vol. 9, no. 2, pp. 350-365, June 2015, doi: 10.1109/JSYST.2014.2322503.
- [11] V. Palade, L. Jain & C. Bocaniala. "Computational Intelligence in Fault Diagnosis". Springer London. 2006. 10.1007/978-1-84628-631-5.
- [12] J. Park. "RNN based Time-series Anomaly Detector Model Implemented in Pytorch". 2018. <https://github.com/chickenbestlover/RNN-Time-series-Anomaly-Detection>
- [13] M.S. Reis, G. Gins. "Industrial Process Monitoring in the Big Data/Industry 4.0 Era: from Detection, to Diagnosis, to Prognosis". *Processes* 2017, 5, 35. <https://doi.org/10.3390/pr5030035>
- [14] Williams, R. J. and Zipser, D. "A learning algorithm for continually running fully recurrent neural networks". *Neural computation* 1989, 1(2), 270-280

9.3 RÉSUMÉ DE LA THÈSE EN FRANÇAIS

La Quatrième Révolution Industrielle, également appelée Industrie 4.0, marque une transformation profonde du secteur industriel en fusionnant les domaines physiques, numériques et biologiques. Se bâtissant sur une transformation numérique, elle est caractérisée par des avancées telles que l'Internet des Objets, l'Intelligence Artificielle, et les systèmes cyber-physiques. Au cœur de cette révolution se trouve la *smart factory* (usine intelligente), où les machines interagissent en temps réel avec l'homme et d'autres équipements. Un des enjeux majeurs de l'Industrie 4.0 est la maintenance prévisionnelle, visant à prévenir les pannes des systèmes industriels. L'objectif principal de cette thèse de doctorat est d'étudier la maintenance prévisionnelle, en particulier le diagnostic des défauts, à travers le prisme de l'apprentissage profond et des sources de données multimodales et hétérogènes de l'Industrie 4.0. Ces données, qu'elles proviennent de capteurs de vibration, de température, de caméras ou de rapports de maintenances, offrent une perspective multimodale (séries temporelles, images, texte...) riche et détaillée de l'état des systèmes de production. L'analyse intégrée de ces modalités distinctes permet non seulement une détection plus précise des défauts, mais aussi une vérification croisée pour une plus grande fiabilité, révélant parfois des anomalies qui pourraient rester inaperçues si chaque modalité était considérée isolément. L'utilisation de données multimodales offre de nombreux avantages, mais l'importance des données textuelles est également particulièrement remarquable. Ces données, tirées principalement de rapports de maintenance ou de journaux opérationnels (logs), offrent une vue approfondie des opérations des systèmes et des incidents précédents. Elles sont uniques car elles contiennent des détails nuancés provenant de l'expertise humaine, essentiels pour diagnostiquer les défauts. Ces données textuelles relient diverses modalités, ajoutant du contexte et une interprétation aux données numériques et visuelles. Toutefois, leur rareté représente un défi pour leur exploitation optimale dans les analyses.

Bien ancrée dans le contexte très appliqué de l'Industrie 4.0 et du projet "Maintenance Prévisionnelle et Optimisation" de l'IRT SystemX, l'ambition de cette thèse va au-delà du développement de modèles pour des applications spécifiques et vise à répondre méthodologiquement aux défis considérés. Le premier objectif concerne la nature dynamique et en temps réel des systèmes industriels, générant des flux de données continus avec des fréquences d'acquisition hétérogènes. Le second objectif émerge du besoin de gérer la complexité d'intégration de données à structures hétérogènes, en soulignant l'importance des interactions entre les caractéristiques de différentes sources de données. Le troisième objectif se concentre sur l'exploitation de la richesse des données textuelles, en particulier dans les rapports de maintenance. Ces documents encapsulent une information contextuelle riche, mais leur rareté et le vocabulaire spécifique qu'ils contiennent rendent leur traitement difficile.

Conformément aux défis précédemment définis, cette thèse présente deux contributions distinctes, chacune dédiée à un domaine de recherche spécifique : l'Apprentissage Multimodal et l'Apprentissage avec peu de données (Few-Shot Learning) en TAL (Traitement Automatique du Langage). Ceci définit également la structure de la thèse, divisée en deux parties principales.

La première partie commence par un besoin pragmatique clairement défini dans le domaine industriel pour diagnostiquer des pannes dans des systèmes multimodaux complexes. Cette motivation concrète nous a orientés vers le développement d'un cadre théorique basé sur l'apprentissage multimodal, intrinsèquement motivé par la nature multimodale de notre environnement réel. Dans le Chapitre 2, nous fournissons au lecteur les bases nécessaires pour motiver et comprendre la première partie de cette thèse. Nous commençons par présenter les fondamentaux de la théorie du diagnostic de pannes et nous passons en revue les stratégies existantes pour aborder ce problème, en nous concentrant sur les approches basées sur l'apprentissage et en explorant les rares tentatives ayant pris en compte des données issues de modalités hétérogènes. Nous introduisons ensuite le paradigme de l'apprentissage multimodal, avec un accent particulier sur la fusion multimodale. De là, nous proposons un aperçu des méthodologies développées, en commençant par les travaux plus anciens reposant sur des stratégies de fusion simples comme la concaténation, et en se concentrant davantage sur le niveau auquel réaliser la fusion. Nous soulignons ensuite les avantages de construire des représentations de données expressives, qui sont principalement réalisables grâce aux architectures basées sur l'apprentissage profond, et la proximité entre la fusion multimodale et la représentation multimodale. Nous explorons donc les approches d'apprentissage de représentation multimodale, qui sont aujourd'hui principalement basées sur l'architecture Transformer.

Dans le Chapitre 3, nous abordons les nouveaux défis posés par la complexité croissante des systèmes Industrie 4.0 et leur relation avec les tâches de détection et de diagnostic de pannes. Nous explorons ces défis dans un environnement réaliste qui implique des flux de données multi-sources provenant de diverses modalités, incluant des mesures de capteurs en séries temporelles, des images de machines et des rapports de maintenance textuels. Ces flux multimodaux hétérogènes diffèrent également dans leur fréquence d'acquisition, peuvent intégrer des informations temporellement non alignées et peuvent être arbitrairement longs, en fonction du système et de la tâche considérés. S'appuyant sur le chapitre précédent, où nous avons examiné les principales approches de fusion multimodale, nous élargissons notre champ d'application à ce contexte. Nous considérons des flux multimodaux arbitrairement longs conjointement avec des tâches associées, telles que la prédiction dans le temps. Pour relever ce défi, nous proposons StreaMulT, un Transformer multimodal. StreaMulT utilise un mécanisme d'attention cross-modale et une banque de mémoire pour traiter des séquences d'entrée arbitrairement longues pendant l'entraînement et fonctionne au fil de l'eau à l'inférence.

Le Chapitre 4 présente une discussion sur les diverses interactions multimodales. Nous commençons par décomposer le contenu pertinent des données en tant qu'information redondante et complémentaire. Par la suite, nous nous plongeons dans l'exploration des recherches axées sur la maximisation de l'information redondante, principalement dans le cadre multi-vues, et les outils utilisés dans ce domaine. La dernière section tente d'élargir ces approches pour incorporer la caractérisation de l'information complémentaire, et formule des critiques à la fois sur les méthodologies existantes et sur le manque de repères d'évaluation. Cette analyse offre une compréhension exhaustive des défis actuels et des pistes potentielles dans le domaine de l'apprentissage multimodal.

La deuxième partie de la thèse se concentre sur l'exploitation de données textuelles rares et spécifiques dans un contexte réaliste. Elle débute avec le Chapitre 5, qui offre un aperçu des méthodologies de Traitement Automatique du Langage (TAL), jusqu'au développement des récents grands

modèles dits "fondateurs", puis se tourne vers l'apprentissage à partir de peu d'exemples (Few-shot learning), une stratégie pour apprendre à partir de données étiquetées limitées, avant de conclure par une discussion sur l'application du FSL au TAL. La première section de ce chapitre décrit la progression de la recherche en TAL pour comprendre le langage humain. Cela inclut les premières méthodes basées sur des règles établies ou sur de l'ingénierie des caractéristiques (feature engineering), l'utilisation des plongements de mots pour créer des représentations distribuées et significatives, et le développement de diverses architectures pour des modèles de langage efficaces. Nous étudions ensuite l'approche dominante pour traiter les tâches du TAL, qui implique de grands modèles de langage basés sur des transformateurs pré-entraînés et leur évolution ultérieure vers la création de modèles centraux polyvalents capables de gérer une gamme variée de tâches, malgré leurs natures distinctes. Enfin, nous explorons le domaine de l'apprentissage à partir de peu d'exemples, en examinant ses principales techniques et son intersection avec les paradigmes actuels du TAL, tout en mettant en lumière les derniers progrès et défis de ce domaine de recherche.

Dans le Chapitre 6, nous explorons le potentiel des méthodes transductives pour la classification textuelle dans le contexte de l'apprentissage à partir de peu d'exemples, dans le but de pallier les limites des méthodes actuelles de FSL en TAL, notamment les efforts d'ingénierie nécessaires pour des tâches de classification réaliste avec un grand nombre de classes. Nous discutons d'abord des limites des méthodes actuelles de FSL, telles que les stratégies basées sur des prompts ou de l'apprentissage en contexte. Puis, nous explorons l'application des approches transductives - qui ont montré des résultats prometteurs en vision par ordinateur - à la classification en TAL. Enfin, nous évaluons la performance des régularisateurs transductifs traditionnels par rapport aux techniques inductives sur des tâches de classification textuelle avec peu d'exemples et étudions l'impact de différents facteurs, tels que le nombre de paramètres du modèle principal et les stratégies de fine-tuning, sur la performance des méthodes transductives. Les résultats indiquent que les méthodes transductives ont du mal à surpasser le fine-tuning inductif basé sur la cross-entropie lorsqu'il y a une certaine flexibilité dans les paramètres de l'extracteur de caractéristiques pré-entraîné. Cependant, en fixant tous les paramètres de l'extracteur de caractéristiques, l'approche transductive rivalise finalement avec l'approche inductive.

Enfin, dans le Chapitre 7 nous abordons la prévalence croissante des API propriétaires et fermées pour les grands modèles de langage tels que GPT-4 et ChatGPT, qui ont des implications significatives pour les applications pratiques du TAL, y compris la classification avec peu d'exemples. La classification avec peu d'exemples implique de former un modèle pour exécuter une nouvelle tâche de classification avec un minimum de données étiquetées. Notre investigation présente trois contributions clés. Premièrement, nous introduisons une situation dans laquelle un modèle pré-entraîné est accessible via une API protégée, en tenant compte des contraintes de coût de calcul et de confidentialité des données. Deuxièmement, nous approfondissons l'application de l'inférence transductive, un paradigme d'apprentissage qui a été relativement peu exploré au sein de la communauté du TAL. Contrairement à l'apprentissage inductif traditionnel, l'inférence transductive tire parti des statistiques des données non étiquetées. Dans ce contexte, nous introduisons également un nouveau régularisateur transductif sans paramètre basé sur la perte de Fisher-Rao, démontrant son applicabilité et son efficacité dans le cadre de l'incorporation via une API protégée. Cette approche exploite pleinement les données non étiquetées, évite de partager toute information d'étiquette avec les fournisseurs d'API tiers et pourrait servir de référence pour les recherches futures. Enfin, nous proposons un cadre expérimental amélioré et compilons un benchmark de

huit ensembles de données englobant la classification multi-classes dans quatre langues différentes, avec jusqu'à 151 classes. Nous évaluons nos méthodes à l'aide de huit modèles principaux et d'une évaluation épisodique sur 1 000 épisodes, qui démontrent la supériorité de l'inférence transductive par rapport au cadre inductif standard.

NOTATIONS

X	Random variable
\mathbf{X}	Random vector or multivariate random variable
X_α	Random variable modeling the modality α of the random vector
Y_Q	Restriction of random variable Y to set Q
x	Realization of a random variable
\mathbf{x}	Realization of a random vector
x_α^i	Modality α of the i^{th} realization (sample) of random variable \mathbf{X}
$\mathbb{P}(X)$	Probability of X
p_X	Probability distribution of X
$\mathbb{E}_p(X)$	Expectation of $X \sim p$
\mathcal{X}	Input space
\mathcal{X}_α	Associated definition space of the modality α
\mathcal{Z}	Representation space
$Card(\mathcal{A})$	Cardinality of the set \mathcal{A}
\mathcal{Y}	Label space
θ, ϕ, ψ	Model parameters
Θ, Φ, Ψ	Parameters spaces
Ω	Vocabulary
Ω^*	Vocabulary Kleene closure
S, Q	Support and Query sets
N_S, N_Q	Number of support and query shots
K	Number of ways
\wedge	Logical AND
\vee	Logical OR

ACRONYMS

AI	Artificial Intelligence
AM-TRF	Augmented-memory Transformer (C. Wu et al. 2020)
API	Application Programming Interface
ASR	Automatic Speech Recognition
BitFit	BIas-Term FIne-Tuning
BoW	Bag of Words
CBoW	Continuous Bag of Words
CE	Cross Entropy
CLM	Conditional Language Modeling
CNN	Convolutional Neural Network
CRF	Conditional Random Fields
CT	Computerized Tomography
DL	Deep Learning
ERM	Empirical Risk Minimization
ERR	Error Reduction Rate
FFN	Feed Forward Network
FSL	Few-Shot Learning
GAN	Generative Adversarial Network
GRU	Gated Recurrent Unit
HMM	Hidden Markov Model
ICL	In-Context Learning
IoT	Internet of Things
LLM	Large Language Model
LM	Language Modeling
LN	Layer Normalization
LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MI	Mutual Information
ML	Machine Learning
MLE	Maximum Likelihood Estimation
MLM	Masked Language Modelling
MLP	Multi-Layer Perceptron
MSE	Mean Squared Error
MuT	Multimodal Transformer (Tsai et al. 2019)
NER	Named Entity Recognition
NLP	Natural Language Processing
PGM	Probabilistic Graphical Models
PLM	Pretrained Language Models
PMI	Pointwise Mutual Information

POS	Part-Of-Speech
PPMI	Positive Pointwise Mutual Information
RLHF	Reinforcement Learning from Human Feedback
RNN	Recurrent Neural Network
RUL	Remaining Useful Life
SCADA	Supervisory Control And Data Acquisition
SCT	Streaming Crossmodal Transformer
SMT	Simultaneous Machine Translation
SOTA	State-Of-The-Art
SVD	Singular Value Decomposition
TF-IDF	Term Frequency-Inverse Document Frequency
VAE	Variational AutoEncoder
ViT	Visual Transformer

BIBLIOGRAPHY

- Abed, Wathiq (2015). “A Robust Bearing Fault Detection and Diagnosis Technique for Brushless DC Motors Under Non-stationary Operating Conditions”. *Journal of Control, Automation and Electrical Systems*.
- Achille, Alessandro and Stefano Soatto (2018). “Emergence of invariance and disentanglement in deep representations”. *The Journal of Machine Learning Research*.
- Akhbardeh, Farhad, Travis Desell, and Marcos Zampieri (2020). “NLP Tools for Predictive Maintenance Records in MaintNet”. In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: System Demonstrations*.
- Alayrac, Jean-Baptiste, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. (2022). “Flamingo: a visual language model for few-shot learning”. *Advances in Neural Information Processing Systems*.
- Alayrac, Jean-Baptiste, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman (2020). “Self-supervised multimodal versatile networks”. *Advances in Neural Information Processing Systems*.
- Alemi, Alexander A., Ian Fischer, Joshua V. Dillon, and Kevin Murphy (2017). “Deep Variational Information Bottleneck”. In: *5th International Conference on Learning Representations*.
- Androutsopoulos, Ion, Graeme D Ritchie, and Peter Thanisch (1995). “Natural language interfaces to databases—an introduction”. *Natural language engineering*.
- Angelopoulos, Angelos, Emmanouel T Michailidis, Nikolaos Nomikos, Panagiotis Trakadas, Antonis Hatziefremidis, Stamatis Voliotis, and Theodore Zahariadis (2019). “Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects”. *Sensors*.
- Antoniou, Antreas, Harrison Edwards, and Amos J. Storkey (2019). “How to train your MAML”. In: *7th International Conference on Learning Representations*.
- Arivazhagan, Naveen, Colin Cherry, Wolfgang Macherey, Chung Cheng Chiu, Semih Yavuz, Ruoming Pang, Wei Li, and Colin Raffel (2020). “Monotonic infinite lookback attention for simultaneous machine translation”. *57th Annual Meeting of the Association for Computational Linguistics*.
- Atrey, Pradeep K, M Anwar Hossain, Abdulmotaleb El Saddik, and Mohan S Kankanhalli (2010). “Multimodal fusion for multimedia analysis: a survey”. *Multimedia systems*.
- Bachman, Philip, R Devon Hjelm, and William Buchwalter (2019). “Learning representations by maximizing mutual information across views”. *Advances in neural information processing systems*.
- Bagher Zadeh, AmirAli, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency (2018). “Multimodal Language Analysis in the Wild: CMU-MOSEI Dataset and Interpretable

- Dynamic Fusion Graph”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2015). “Neural Machine Translation by Jointly Learning to Align and Translate”. In: *3rd International Conference on Learning Representations*.
- Bai, Yuntao, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. (2022). “Constitutional AI: Harmlessness from AI Feedback”. *arXiv preprint arXiv:2212.08073*.
- Baltrusaitis, Tadas, Chaitanya Ahuja, and Louis Philippe Morency (2019). “Multimodal Machine Learning: A Survey and Taxonomy”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Barbieri, Francesco, Jose Camacho-Collados, Luis Espinosa-Anke, and Leonardo Neves (2020). “TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification”. In: *Proceedings of Findings of EMNLP*.
- Ben Zaken, Elad, Yoav Goldberg, and Shauli Ravfogel (2022). “BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Language-models”. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.
- Bengio, Yoshua, Aaron Courville, and Pascal Vincent (2013). “Representation learning: A review and new perspectives”. *IEEE transactions on pattern analysis and machine intelligence*.
- Bengio, Yoshua, Réjean Ducharme, and Pascal Vincent (2000). “A neural probabilistic language model”. *Advances in neural information processing systems*.
- Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov (2017). “Enriching Word Vectors with Subword Information”. *Transactions of the Association for Computational Linguistics*.
- Bommasani, Rishi, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. (2021). “On the opportunities and risks of foundation models”. *arXiv preprint arXiv:2108.07258*.
- Boudiaf, Malik, Ziko Imtiaz Masud, Jérôme Rony, Jose Dolz, Pablo Piantanida, and Ismail Ben Ayed (2020). “Transductive Information Maximization for Few-Shot Learning”. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*.
- Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei (2020). “Language Models Are Few-Shot Learners”. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*.
- Bruni, Elia, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran (2012). “Distributional Semantics in Technicolor”. In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Bullinaria, John A and Joseph P Levy (2007). “Extracting semantic representations from word co-occurrence statistics: A computational study”. *Behavior research methods*.

- Bunescu, Razvan and Raymond Mooney (2005). “A Shortest Path Dependency Kernel for Relation Extraction”. In: *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.
- Cardoso, J-F (1997). “Infomax and maximum likelihood for blind source separation”. *IEEE Signal processing letters*.
- Casanueva, Iñigo, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić (2020). “Efficient intent detection with dual sentence encoders”. *arXiv preprint arXiv:2003.04807*.
- Castro, Santiago, Devamanyu Hazarika, Verónica Pérez-Rosas, Roger Zimmermann, Rada Mihalcea, and Soujanya Poria (2019). “Towards Multimodal Sarcasm Detection (An _Obviously_ Perfect Paper)”. In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, Volume 1: Long Papers*.
- Chen, Baotong, Jiafu Wan, Lei Shu, Peng Li, Mithun Mukherjee, and Boxing Yin (2017). “Smart factory of industry 4.0: Key technologies, application case, and challenges”. *Ieee Access*.
- Chen, Jiaxin, Xiao-Ming Wu, Yanke Li, Qimai LI, Li-Ming Zhan, and Fu-lai Chung (2020). “A Closer Look at the Training Strategy for Modern Meta-Learning”. In: *Advances in Neural Information Processing Systems*.
- Chen, Junkun, Mingbo Ma, Renjie Zheng, and Liang Huang (2020). “Mam: Masked acoustic modeling for end-to-end speech-to-text translation”. *arXiv preprint arXiv:2010.11445*.
- Chen, Mark, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. (2021). “Evaluating large language models trained on code”. *arXiv preprint arXiv:2107.03374*.
- Chen, Stanley F. and Joshua Goodman (1996). “An Empirical Study of Smoothing Techniques for Language Modeling”. In: Association for Computational Linguistics.
- Chen, Yanda, Ruiqi Zhong, Sheng Zha, George Karypis, and He He (2022). “Meta-learning via Language Model In-context Tuning”. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Cheng, Yiwei, Haiping Zhu, Jun Wu, and Xinyu Shao (2019). “Machine Health Monitoring Using Adaptive Kernel Spectral Clustering and Deep Long Short-Term Memory Recurrent Neural Networks”. *IEEE Transactions on Industrial Informatics*.
- Child, Rewon, Scott Gray, Alec Radford, and Ilya Sutskever (2019). “Generating long sequences with sparse transformers”. *arXiv preprint arXiv:1904.10509*.
- Cho, Kyunghyun, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio (2014). “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.
- Chomsky, N. (1956). “Three models for the description of language”. *IRE Transactions on Information Theory*.
- Chorowski, Jan, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio (2015). “Attention-Based Models for Speech Recognition”. In: *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*.
- Choudhary, Anurag, SL Shimi, and Aparna Akula (2018). “Bearing fault diagnosis of induction motor using thermal imaging”. In: *2018 international conference on computing, power and communication technologies (GUCON)*.

- Chowdhery, Aakanksha, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. (2022). “Palm: Scaling language modeling with pathways”. *arXiv preprint arXiv:2204.02311*.
- Christiano, Paul F, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei (2017). “Deep reinforcement learning from human preferences”. *Advances in neural information processing systems*.
- Chung, Hyung Won, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. (2022). “Scaling instruction-finetuned language models”. *arXiv preprint arXiv:2210.11416*.
- Chung, Joon Son and Andrew Zisserman (2016). “Out of Time: Automated Lip Sync in the Wild”. In: *Computer Vision - ACCV 2016 Workshops - Revised Selected Papers, Part II*. Ed. by Chu-Song Chen, Jiwen Lu, and Kai-Kuang Ma.
- Church, Kenneth, William Gale, Patrick Hanks, and Donald Hindle (2021). “Using statistics in lexical analysis”. In: *Lexical acquisition: exploiting on-line resources to build a lexicon*.
- Church, Kenneth Ward and Patrick Hanks (1990). “Word Association Norms, Mutual Information, and Lexicography”. *Computational Linguistics*.
- Commission, European (2016). *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*.
- (2020). *Proposal for a Regulation of the European Parliament and of the Council on European data governance (Data Governance Act)*.
- Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov (2020). “Unsupervised Cross-lingual Representation Learning at Scale”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.
- Cover, Thomas M (1999). *Elements of information theory*. John Wiley & Sons.
- Dai, Zihang, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov (2019). “Transformer-XL: Attentive Language Models beyond a Fixed-Length Context”. In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, Volume 1: Long Papers*.
- Deerwester, Scott, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman (1990). “Indexing by latent semantic analysis”. *Journal of the American Society for Information Science*.
- Degottex, Gilles, John Kane, Thomas Drugman, Tuomo Raitio, and Stefan Scherer (2014). “COVAREP - A collaborative voice analysis repository for speech technologies”. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Demszky, Dorottya, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi (2020). “GoEmotions: A Dataset of Fine-Grained Emotions”. In: *58th Annual Meeting of the Association for Computational Linguistics*.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

- Dharma, EDDY MUNTINA, F Lumban Gaol, HLHS Warnars, and BENFANO Soewito (2022). “The accuracy comparison among Word2vec, Glove, and Fasttext towards convolution neural network (CNN) text classification”. *Journal of Theoretical and Applied Information Technology*.
- Dhillon, Guneet Singh, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto (2020). “A Baseline for Few-Shot Image Classification”. In: *8th International Conference on Learning Representations*.
- Diaz Roza, Javier et al. (2017). “Machine Learning-based CPS for Clustering High throughput Machining Cycle Conditions”. *Procedia Manufacturing*.
- Ding, Ning, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun (2022). “OpenPrompt: An Open-source Framework for Prompt-learning”. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.
- Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby (2021). “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *9th International Conference on Learning Representations*.
- Elman, Jeffrey L. (1990). “Finding structure in time”. *Cognitive Science*.
- Esteva, Andre, Brett Kuprel, Roberto Novoa, Justin Ko, Susan Swetter, Helen Blau, and Sebastian Thrun (2017). “Dermatologist-level classification of skin cancer with deep neural networks”. In: *Nature*.
- Fan, Angela, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. (2021). “Beyond english-centric multilingual machine translation”. *The Journal of Machine Learning Research*.
- Federici, Marco, Anjan Dutta, Patrick Forré, Nate Kushman, and Zeynep Akata (2020). “Learning Robust Representations via Multi-View Information Bottleneck”. In: *8th International Conference on Learning Representations*.
- Fei, Yu, Zhao Meng, Ping Nie, Roger Wattenhofer, and Mrinmaya Sachan (2022). “Beyond prompting: Making Pre-trained Language Models Better Zero-shot Learners by Clustering Representations”. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.
- Fiérrez-Aguilar, Julian, Javier Ortega-Garcia, Daniel Garcia-Romero, and Joaquin Gonzalez-Rodriguez (2003). “A comparative evaluation of fusion strategies for multimodal biometric verification”. In: *International Conference on Audio-and Video-Based Biometric Person Authentication*.
- Finkel, Jenny Rose, Trond Grenager, and Christopher D Manning (2005). “Incorporating non-local information into information extraction systems by gibbs sampling”. In: *Proceedings of the 43rd annual meeting of the association for computational linguistics*.
- Finn, Chelsea, Pieter Abbeel, and Sergey Levine (2017). “Model-agnostic meta-learning for fast adaptation of deep networks”. In: *International conference on machine learning*.
- Firth, John (1957). “A synopsis of linguistic theory, 1930-1955”. *Studies in linguistic analysis*.
- Ganin, Yaroslav, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky (2016). “Domain-Adversarial Training of Neural Networks”. *Journal of Machine Learning Research*.

- Gao, Tianyu, Adam Fisch, and Danqi Chen (2021). “Making Pre-trained Language Models Better Few-shot Learners”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.
- Gao, Tianyu, Xu Han, Zhiyuan Liu, and Maosong Sun (2019). “Hybrid attention-based prototypical networks for noisy few-shot relation classification”. In: *Proceedings of the AAAI conference on artificial intelligence*.
- Girdhar, Rohit, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra (2023). “ImageBind: One Embedding Space To Bind Them All”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Glaese, Amelia, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Mari-beth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, et al. (2022). “Improving alignment of dialogue agents via targeted human judgements”. *arXiv preprint arXiv:2209.14375*.
- Goldstein, Markus and Seiichi Uchida (2016). “A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data”. *PloS one*.
- Gomes, Eduardo Dadalto Câmara, Florence Alberge, Pierre Duhamel, and Pablo Piantanida (2022). “Igeood: An Information Geometry Approach to Out-of-Distribution Detection”. In: *The Tenth International Conference on Learning Representations*.
- Goodfellow, Ian J., Yoshua Bengio, and Aaron C. Courville (2016). *Deep Learning*. Adaptive computation and machine learning. MIT Press.
- Grandvalet, Yves and Yoshua Bengio (2004). “Semi-supervised learning by entropy minimization”. *Advances in neural information processing systems*.
- Graves, Alex (2013). “Generating sequences with recurrent neural networks”. *arXiv preprint arXiv:1308.0850*.
- Graves, Alex, Greg Wayne, and Ivo Danihelka (2014). “Neural Turing machines”. *arXiv preprint arXiv:1410.5401*.
- Grigorescu, Sorin, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu (2020). “A survey of deep learning techniques for autonomous driving”. *Journal of Field Robotics*.
- Gulshan, Varun, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. (2016). “Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs”. *Jama*.
- Guo, Demi, Alexander Rush, and Yoon Kim (2021). “Parameter-Efficient Transfer Learning with Diff Pruning”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.
- Guo, Liang et al. (2017). “A recurrent neural network based health indicator for remaining useful life prediction of bearings”. *Neurocomputing*.
- Guo, Wenzhong, Jianwen Wang, and Shiping Wang (2019). “Deep Multimodal Representation Learning: A Survey”. *IEEE Access*.
- Guo, Yiluan and Ngai-Man Cheung (2020). “Attentive weights generation for few shot learning via information maximization”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Han, Wei, Hui Chen, and Soujanya Poria (2021). “Improving Multimodal Fusion with Hierarchical Mutual Information Maximization for Multimodal Sentiment Analysis”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

- Harris, Zellig S (1954). “Distributional structure”. *Word*.
- He, Pengcheng, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen (2021). “DeBERTa: decoding-Enhanced Bert with Disentangled Attention”. In: *9th International Conference on Learning Representations*.
- Henaff, Olivier (2020). “Data-efficient image recognition with contrastive predictive coding”. In: *International conference on machine learning*.
- Hinton, Geoffrey, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury (2012). “Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups”. *IEEE Signal Processing Magazine*.
- Hinton, Geoffrey E and Ruslan R Salakhutdinov (2006). “Reducing the dimensionality of data with neural networks”. *Science*.
- Hinton, Geoffrey E., Oriol Vinyals, and Jeffrey Dean (2015). “Distilling the Knowledge in a Neural Network”. *ArXiv*. eprint: [1503.02531](https://arxiv.org/abs/1503.02531).
- Hjelm, R. Devon, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio (2019). “Learning deep representations by mutual information estimation and maximization”. In: *7th International Conference on Learning Representations*.
- Hochreiter, Sepp, Yoshua Bengio, Paolo Frasconi, Jürgen Schmidhuber, et al. (2001). *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*.
- Hochreiter, Sepp and Jürgen Schmidhuber (1997). “Long short-term memory”. *Neural computation*.
- Hoffmann, Jordan, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. (2022). “Training compute-optimal large language models”. *arXiv preprint arXiv:2203.15556*.
- Hospedales, Timothy, Antreas Antoniou, Paul Micaelli, and Amos Storkey (2021). “Meta-learning in neural networks: A survey”. *IEEE transactions on pattern analysis and machine intelligence*.
- Hou, Ruiying, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen (2019). “Cross attention network for few-shot classification”. *Advances in Neural Information Processing Systems*.
- Houlsby, Neil, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly (2019). “Parameter-efficient transfer learning for NLP”. In: *International Conference on Machine Learning*.
- Howard, Jeremy and Sebastian Ruder (2018). “Universal Language Model Fine-tuning for Text Classification”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Hu, Shell Xu, Pablo Garcia Moreno, Yang Xiao, Xi Shen, Guillaume Obozinski, Neil D. Lawrence, and Andreas C. Damianou (2020). “Empirical Bayes Transductive Meta-Learning with Synthetic Gradients”. In: *8th International Conference on Learning Representations*.
- Hu, Weihua, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, and Masashi Sugiyama (2017). “Learning Discrete Representations via Information Maximizing Self-Augmented Training”. In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*.
- Hu, Yuqing, Vincent Gripon, and Stéphane Pateux (2021). “Leveraging the feature distribution in transfer-based few-shot learning”. In: *Artificial Neural Networks and Machine Learning-ICANN 2021: 30th International Conference on Artificial Neural Networks*.

- Huang, Yu, Chenzhuang Du, Zihui Xue, Xuanyao Chen, Hang Zhao, and Longbo Huang (2021). “What makes multi-modal learning better than single (provably)”. *Advances in Neural Information Processing Systems*.
- Huang, Zhiheng, Davis Liang, Peng Xu, and Bing Xiang (2020). “Improve Transformer Models with Better Relative Position Embeddings”. In: *Findings of the Association for Computational Linguistics*.
- Hutchins, W. John (2004). “The Georgetown-IBM Experiment Demonstrated in January 1954”. In: *Machine Translation: From Real Users to Research*.
- Hutchins, William John (1986). *Machine translation: past, present, future*. Citeseer.
- Isermann, Rolf (2005). *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer Science & Business Media.
- Jafar, Raed et al. (2010). “Application of Artificial Neural Networks (ANN) to model the failure of urban water mains”. *Mathematical and Computer Modelling*.
- Janssens, Olivier et al. (2015). “Thermal image based fault diagnosis for rotating machinery”. *Infrared Physics and Technology*.
- Jelinek, Frederick (1991). “Principles of lexical language modeling for speech recognition”. *Advances in speech signal processing*.
- Ji, Xu, Joao F Henriques, and Andrea Vedaldi (2019). “Invariant information clustering for unsupervised image classification and segmentation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Jia, Feng et al. (2015). “Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data”. *Mechanical Systems and Signal Processing*.
- Johnson, Alistair EW, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark (2016). “MIMIC-III, a freely accessible critical care database”. *Scientific data*.
- Jumper, John, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Židek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A A Kohl, Andrew J Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis (2021). “Highly accurate protein structure prediction with AlphaFold”. *Nature*.
- Jurafsky, Dan (2000). *Speech & language processing*. Pearson Education India.
- Kaplan, Jared, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei (2020). “Scaling laws for neural language models”. *arXiv preprint arXiv:2001.08361*.
- Katharopoulos, Angelos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret (2020). “Transformers are rnns: Fast autoregressive transformers with linear attention”. In: *International Conference on Machine Learning*.
- Ke, Guolin, Di He, and Tie-Yan Liu (2021). “Rethinking Positional Encoding in Language Pre-training”. In: *9th International Conference on Learning Representations*.

- Keung, Phillip, Yichao Lu, György Szarvas, and Noah A. Smith (2020). “The Multilingual Amazon Reviews Corpus”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.
- Kingma, Diederik P. and Jimmy Ba (2015). “Adam: A Method for Stochastic Optimization”. In: *3rd International Conference on Learning Representations*.
- Kirkpatrick, James, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. (2017). “Overcoming catastrophic forgetting in neural networks”. *Proceedings of the national academy of sciences*.
- Kitaev, Nikita, Lukasz Kaiser, and Anselm Levskaya (2020). “Reformer: The Efficient Transformer”. In: *8th International Conference on Learning Representations*.
- Kocoń, Jan, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniec, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. (2023). “Chatgpt: Jack of all trades, master of none”. *arXiv preprint arXiv:2302.10724*.
- Koehn, Philipp, Franz J. Och, and Daniel Marcu (2003). “Statistical Phrase-Based Translation”. In: *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.
- Konar, Pratyay et al. (2009). “Bearing Fault Detection of Induction Motor using Wavelet and Neural Networks.” In:
- Lachs, Lorin (2017). “Multi-modal perception”. *Noba textbook series: Psychology. Champaign: DEF Publishers*.
- Lafferty, John D., Andrew McCallum, and Fernando C. N. Pereira (2001). “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data”. In: *Proceedings of the Eighteenth International Conference on Machine Learning*.
- Lake, Brenden M., Ruslan Salakhutdinov, and Joshua B. Tenenbaum (2015). “Human-level concept learning through probabilistic program induction”. *Science*.
- Lan, Zhen-zhong, Lei Bao, Shou-I Yu, Wei Liu, and Alexander G Hauptmann (2014). “Multi-media classification and event detection using double fusion”. *Multimedia tools and applications*.
- Lan, Zhenzhong, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut (2020). “ALBERT: A Lite BERT for Self-supervised Learning of Language Representations”. In: *8th International Conference on Learning Representations*.
- Larson, Stefan, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars (2019). “An Evaluation Dataset for Intent Classification and Out-of-Scope Prediction”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*.
- LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton (2015). “Deep learning”. *nature*.
- Lee, Kwonjoon, Subhansu Maji, Avinash Ravichandran, and Stefano Soatto (2019). “Meta-learning with differentiable convex optimization”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.
- Lee, Mihee and Vladimir Pavlovic (2021). “Private-shared disentangled multimodal vae for learning of latent representations”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

- Lehman, Eric, Evan Hernandez, Diwakar Mahajan, Jonas Wulff, Micah J Smith, Zachary Ziegler, Daniel Nadler, Peter Szolovits, Alistair Johnson, and Emily Alsentzer (2023). “Do We Still Need Clinical Language Models?” *arXiv preprint arXiv:2302.08091*.
- Lewis, Mike, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer (2020). “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.
- Lhoest, Quentin, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Sasko, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander M. Rush, and Thomas Wolf (2021). “Datasets: A Community Library for Natural Language Processing”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.
- Li, Gen, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang (2020). “Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Li, Liunian Harold, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang (2019). “Visualbert: A simple and performant baseline for vision and language”. *arXiv preprint arXiv:1908.03557*.
- Li, Ruilong, Shan Yang, David A Ross, and Angjoo Kanazawa (2021). “Ai choreographer: Music conditioned 3d dance generation with aist++”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Li, Zhe (2018). “Deep learning driven approaches for predictive maintenance: A framework of intelligent fault diagnosis and prognosis in the industry 4.0 era”.
- Liang, Paul Pu, Yun Cheng, Xiang Fan, Chun Kai Ling, Suzanne Nie, Richard Chen, Zihao Deng, Faisal Mahmood, Ruslan Salakhutdinov, and Louis-Philippe Morency (2023). “Quantifying & Modeling Feature Interactions: An Information Decomposition Framework”. *arXiv preprint arXiv:2302.12247*.
- Liang, Paul Pu, Yiwei Lyu, Xiang Fan, Zetian Wu, Yun Cheng, Jason Wu, Leslie Chen, Peter Wu, Michelle A. Lee, Yuke Zhu, Ruslan Salakhutdinov, and Louis-Philippe Morency (2021). “MultiBench: Multiscale Benchmarks for Multimodal Representation Learning”. In: *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1*.
- Liang, Tianchen et al. (2018). “Bearing fault diagnosis based on improved ensemble learning and deep belief network”. *Journal of Physics*.
- Lichtenstein, Moshe, Prasanna Sattigeri, Rogerio Feris, Raja Giryes, and Leonid Karlinsky (2020). “Tafssl: Task-adaptive feature sub-space learning for few-shot classification”. In: *Computer Vision – ECCV 2020: 16th European Conference*.
- Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick (2014). “Microsoft coco: Common objects in context”. In: *European conference on computer vision*.
- Linsker, Ralph (1988). “Self-organization in a perceptual network”. *Computer*.

- Littlewort, Gwen, Jacob Whitehill, Tingfan Wu, Ian Fasel, Mark Frank, Javier Movellan, and Marian Bartlett (2011). “The computer expression recognition toolbox (CERT)”. In: *2011 IEEE International Conference on Automatic Face Gesture Recognition (FG)*.
- Liu, Han et al. (2018). “Unsupervised fault diagnosis of rolling bearings using a deep neural network based on generative adversarial networks”. *Neurocomputing*.
- Liu, Haokun, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel (2022). “Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning”. In: *NeurIPS*.
- Liu, Jinlu, Liang Song, and Yongqiang Qin (2020). “Prototype rectification for few-shot learning”. In: *Computer Vision—ECCV 2020: 16th European Conference*.
- Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig (2023). “Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing”. *ACM Computing Surveys*.
- Liu, Yanbin, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang (2019). “Learning to Propagate Labels: Transductive Propagation Network for Few-Shot Learning”. In: *7th International Conference on Learning Representations*.
- Liu, Yaoyao, Bernt Schiele, and Qianru Sun (2020). “An ensemble of epoch-wise empirical bayes for few-shot learning”. In: *Computer Vision—ECCV 2020: 16th European Conference*.
- Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov (2019). “Roberta: A robustly optimized bert pretraining approach”. *arXiv preprint arXiv:1907.11692*.
- Liu, Yueh-Cheng, Yu-Kai Huang, Hung-Yueh Chiang, Hung-Ting Su, Zhe-Yu Liu, Chin-Tang Chen, Ching-Yu Tseng, and Winston H Hsu (2021). “Learning from 2d: Contrastive pixel-to-point knowledge transfer for 3d pretraining”. *arXiv preprint arXiv:2104.04687*.
- Liu, Yukun et al. (2010). “Application to induction motor faults diagnosis of the amplitude recovery method combined with FFT”. *Mechanical Systems and Signal Processing*.
- Liu, Yunze, Qingnan Fan, Shanghang Zhang, Hao Dong, Thomas Funkhouser, and Li Yi (2021). “Contrastive multimodal fusion with tupleinfonce”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Logan IV, Robert, Ivana Balazevic, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel (2022). “Cutting Down on Prompts and Parameters: Simple Few-Shot Learning with Language Models”. In: *Findings of the Association for Computational Linguistics*.
- Long, Xiang, Chuang Gan, Gerard De Melo, Xiao Liu, Yandong Li, Fu Li, and Shilei Wen (2018). “Multimodal keyless attention fusion for video classification”. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*.
- Lu, Jiasen, Dhruv Batra, Devi Parikh, and Stefan Lee (2019). “ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems*.
- Lu, Yao, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp (2022). “Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity”. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Luhn, Hans Peter (1957). “A statistical approach to mechanized encoding and searching of literary information”. *IBM Journal of research and development*.

- Lund, Kevin and Curt Burgess (1996). “Producing high-dimensional semantic spaces from lexical co-occurrence”. *Behavior research methods, instruments, & computers*.
- Luo, Bo, Haoting Wang, Hongqi Liu, Bin Li, and Fangyu Peng (2018). “Early fault detection of machine tools based on deep learning and dynamic identification”. *IEEE Transactions on Industrial Electronics*.
- Luong, Thang, Hieu Pham, and Christopher D. Manning (2015). “Effective Approaches to Attention-based Neural Machine Translation”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.
- Ma, Xutai, Juan Miguel Pino, James Cross, Liezl Puzon, and Jiatao Gu (2020). “Monotonic Multi-head Attention”. In: *8th International Conference on Learning Representations*.
- Mahabadi, Rabeeh Karimi, Luke Zettlemoyer, James Henderson, Marzieh Saeidi, Lambert Mathias, Veselin Stoyanov, and Majid Yazdani (2022). “PERFECT: Prompt-free and efficient few-shot learning with language models”. *arXiv preprint arXiv:2204.01172*.
- Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze (2008). *Introduction to information retrieval*. Cambridge University Press.
- Manning, Christopher D. and Hinrich Schütze (2001). *Foundations of statistical natural language processing*. MIT Press.
- Maragos, Petros, Alexandros Potamianos, and Patrick Gros (2008). *Multimodal Processing and Interaction, Audio, Video, Text*. Springer.
- Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz (1993). “Building a Large Annotated Corpus of English: The Penn Treebank”. *Computational Linguistics*.
- McCarthy, John, Marvin L Minsky, Nathaniel Rochester, and Claude E Shannon (2006). “A proposal for the dartmouth summer research project on artificial intelligence”. *AI magazine*.
- Mian, Tauheed, Anurag Choudhary, and Shahab Fatima (2022). “A sensor fusion based approach for bearing fault diagnosis of rotating machine”. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*.
- Miech, Antoine, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman (2020). “End-to-End Learning of Visual Representations From Uncurated Instructional Videos”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean (2013). “Distributed Representations of Words and Phrases and their Compositionality”. In: *Advances in Neural Information Processing Systems*.
- Mikolov, Tomás, Kai Chen, Greg Corrado, and Jeffrey Dean (2013). “Efficient Estimation of Word Representations in Vector Space”. In: *1st International Conference on Learning Representations*.
- Mikolov, Tomás, Martin Karafiát, Lukás Burget, Jan Cernocký, and Sanjeev Khudanpur (2010). “Recurrent neural network based language model”. In: *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association*.
- Mikolov, Tomás, Quoc V. Le, and Ilya Sutskever (2013). “Exploiting Similarities among Languages for Machine Translation”. *Computing Research Repository*.
- Min, Sewon, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi (2022). “MetaICL: Learning to Learn In Context”. In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

- Min, Sewon, Xinxu Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer (2022). “Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?” In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.
- Mishra, Nikhil, Mostafa Rohaninejad, Xi Chen, and P. Abbeel (2017). “A Simple Neural Attentive Meta-Learner”. In: *International Conference on Learning Representations*.
- Mobley, R Keith (2002). *An introduction to predictive maintenance*. Elsevier.
- Mohri, Mehryar (1997). “Finite-state transducers in language and speech processing”. *Computational linguistics*.
- Moritz, Niko, Takaaki Hori, and Jonathan Le Roux (2020). “Streaming automatic speech recognition with the transformer model”. *IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*.
- Muennighoff, Niklas, Nouamane Tazi, Loic Magne, and Nils Reimers (2023). “MTEB: Massive Text Embedding Benchmark”. In: *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*.
- Neti, Chalapathy, Benoit Maison, Andrew W Senior, Giridharan Iyengar, P Decuetos, Sankar Basu, and Ashish Verma (2000). “Joint processing of audio and visual information for multimedia indexing and human-computer interaction.” In: *RIAO*.
- Ngiam, Jiquan, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng (2011). “Multimodal Deep Learning”. *Proceedings of the 28th International Conference on Machine Learning*.
- Nichol, Alex, Joshua Achiam, and John Schulman (2018). “On first-order meta-learning algorithms”. *arXiv preprint arXiv:1803.02999*.
- Nor, Norazwan et al. (2019). “A review of data-driven fault detection and diagnosis methods: Applications in chemical process systems”. *Reviews in Chemical Engineering*.
- Oliver, Avital, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow (2018). “Realistic evaluation of deep semi-supervised learning algorithms”. *Advances in neural information processing systems*.
- Oord, Aaron van den, Yazhe Li, and Oriol Vinyals (2018). “Representation learning with contrastive predictive coding”. *arXiv preprint arXiv:1807.03748*.
- OpenAI (2023). *GPT-4 Technical Report*. arXiv: 2303.08774 [cs.CL].
- Ouali, Yassine (2023). “Learning with Limited Labeled Data”. PhD thesis. Université Paris-Saclay.
- Ouyang, Long, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. (2022). “Training language models to follow instructions with human feedback”. *Advances in Neural Information Processing Systems*.
- Palade, Vasile and Cosmin Danut Bocaniala (2006). *Computational intelligence in fault diagnosis*. Springer Science & Business Media.
- Pan, Jun et al. (2017). “LiftingNet: A Novel Deep Learning Network With Layerwise Feature Learning From Noisy Mechanical Data for Fault Classification”. *IEEE TIE*.
- Pan, Sinno Jialin and Qiang Yang (2010). “A survey on transfer learning”. *IEEE Transactions on knowledge and data engineering*.

- Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan (2002). “Thumbs up? Sentiment Classification using Machine Learning Techniques”. In: *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*.
- Pellegrain, Victor, Michel Batteux, William Lair, and Michel Kaczmarek (2022). “Démonstration de surveillance de défaillances sur un exemple applicatif”. In: *Congrès Lambda Mu 23 «Innovations et maîtrise des risques pour un avenir durable»-23e Congrès de Maîtrise des Risques et de Sécurité de Fonctionnement, Institut pour la Maîtrise des Risques*.
- Pellegrain, Victor, Myriam Tami, Michel Batteux, Céline Hudelot, and IRT SystemX (2022). “Apprentissage multimodal pour le diagnostic de fautes sur données séquentielles non alignées et arbitrairement longues”. In: *Conférence Nationale d'Intelligence Artificielle Année 2022*.
- Peng, Ying et al. (2010). “Current status of machine prognostics in condition-based maintenance: A review”. *International Journal of Advanced Manufacturing Technology*.
- Pennington, Jeffrey, Richard Socher, and Christopher D. Manning (2014). In: *Empirical Methods in Natural Language Processing*.
- Perez, Ethan, Douwe Kiela, and Kyunghyun Cho (2021). “True Few-Shot Learning with Language Models”. In: *Advances in Neural Information Processing Systems*.
- Pérez-Rosas, Verónica, Rada Mihalcea, and Louis-Philippe Morency (2013). “Utterance-level multimodal sentiment analysis”. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Peters, Matthew E., Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer (2018). “Deep Contextualized Word Representations”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*.
- Pfeiffer, Jonas, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulic, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych (2020). “AdapterHub: A Framework for Adapting Transformers”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.
- Picot, Marine, Francisco Messina, Malik Boudiaf, Fabrice Labeau, Ismail Ben Ayed, and Pablo Piantanida (2023). “Adversarial Robustness Via Fisher-Rao Regularization”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Poole, Ben, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker (2019). “On variational bounds of mutual information”. In: *International Conference on Machine Learning*.
- Poria, Soujanya, Iti Chaturvedi, Erik Cambria, and Amir Hussain (2016). “Convolutional MKL based multimodal emotion recognition and sentiment analysis”. In: *2016 IEEE 16th international conference on data mining*.
- Porter, Martin F (1980). “An algorithm for suffix stripping”. *Program*.
- Qiao, Limeng, Yemin Shi, Jia Li, Yaowei Wang, Tiejun Huang, and Yonghong Tian (2019). “Transductive episodic-wise adaptive metric for few-shot learning”. In: *Proceedings of the IEEE/CVF international conference on computer vision*.
- Rabiner, Lawrence R (1989). “A tutorial on hidden Markov models and selected applications in speech recognition”. *Proceedings of the IEEE*.
- Radford, Alec, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. (2021). “Learning transferable

- visual models from natural language supervision”. In: *International Conference on Machine Learning*.
- Radford, Alec, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever (2022). “Robust Speech Recognition via Large-Scale Weak Supervision”. *Computing Research Repository* abs/2212.04356.
- Radford, Alec, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. (2018). “Improving language understanding by generative pre-training”.
- Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. (2019). “Language models are unsupervised multitask learners”. *OpenAI blog*.
- Raffel, Colin, Minh Thang Luong, Peter J. Liu, Ron J. Weiss, and Douglas Eck (2017). “Online and linear-time attention by enforcing monotonic alignments”. *34th International Conference on Machine Learning*.
- Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu (2020). “Exploring the limits of transfer learning with a unified text-to-text transformer”. *The Journal of Machine Learning Research*.
- Raghu, Aniruddh, Maithra Raghu, Samy Bengio, and Oriol Vinyals (2020). “Rapid Learning or Feature Reuse? Towards Understanding the Effectiveness of MAML”. In: *8th International Conference on Learning Representations*.
- Raghu, Maithra and Eric Schmidt (2020). “A Survey of Deep Learning for Scientific Discovery”. *Computing Research Repository*.
- Ramesh, Aditya, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen (2022). “Hierarchical text-conditional image generation with clip latents”. *arXiv preprint arXiv:2204.06125*.
- Rasmus, Antti, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko (2015). “Semi-supervised learning with ladder networks”. *Advances in neural information processing systems*.
- Ravi, Sachin and Hugo Larochelle (2017). “Optimization as a model for few-shot learning”. In: *International conference on learning representations*.
- Reid, Alistair et al. (2013). “Fault Location and Diagnosis in a Medium Voltage EPR Power Cable”. *IEEE Transactions on Dielectrics and Electrical Insulation*.
- Reimers, Nils and Iryna Gurevych (2019). “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*.
- Reis, Marco S. and Geert Gins (2017). “Industrial Process Monitoring in the Big Data/Industry 4.0 Era: from Detection, to Diagnosis, to Prognosis”. *Processes*.
- Rogers, A.P. et al. (2019). “A review of fault detection and diagnosis methods for residential air conditioning systems”. *Building and Environment*.
- Rosenfeld, Jonathan S., Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit (2020). “A Constructive Prediction of the Generalization Error Across Scales”. In: *8th International Conference on Learning Representations*.
- Ruder, Sebastian, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf (2019). “Transfer Learning in Natural Language Processing”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*.

- Rush, Alexander M., Sumit Chopra, and Jason Weston (2015). "A Neural Attention Model for Abstractive Sentence Summarization". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.
- Rusu, Andrei, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell (2019). "Meta-Learning with Latent Embedding Optimization". In: *International Conference on Learning Representations*.
- Ryalat, Mutaz, Hisham ElMoaqet, and Marwa AlFaouri (2023). "Design of a smart factory based on cyber-physical systems and internet of things towards industry 4.0". *Applied Sciences*.
- Sain, Stephan R (1996). *The nature of statistical learning theory*.
- Salton, Gerard, Anita Wong, and Chung-Shu Yang (1975). "A vector space model for automatic indexing". *Communications of the ACM*.
- Sanh, Victor, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush (2022). "Multitask Prompted Training Enables Zero-Shot Task Generalization". In: *The Tenth International Conference on Learning Representations*.
- Scao, Teven Le, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. (2022). "Bloom: A 176b-parameter open-access multilingual language model". *arXiv preprint arXiv:2211.05100*.
- Schick, Timo and Hinrich Schütze (2020). "Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference". In: *Conference of the European Chapter of the Association for Computational Linguistics*.
- (2021). "It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.
- (2022). "True Few-Shot Learning with Prompts—A Real-World Perspective". *Transactions of the Association for Computational Linguistics*.
- Schwab, Klaus (2017). *The fourth industrial revolution*. Currency.
- Sha, Fei and Fernando Pereira (2003). "Shallow parsing with conditional random fields". In: *Proceedings of the 2003 human language technology conference of the North American chapter of the association for computational linguistics*.
- Shannon, Claude E (1948). "A mathematical theory of communication". *The Bell system technical journal*.
- Shao, Haidong et al. (2018). "A novel method for intelligent fault diagnosis of rolling bearings using ensemble deep auto-encoders". *Mechanical Systems and Signal Processing*.
- Shaw, Peter, Jakob Uszkoreit, and Ashish Vaswani (2018). "Self-Attention with Relative Position Representations". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*.

- Shi, Yangyang, Yongqiang Wang, Chunyang Wu, Ching-Feng Yeh, Julian Chan, Frank Zhang, Duc Le, and Mike Seltzer (2021). “Emformer: Efficient memory transformer based acoustic model for low latency streaming speech recognition”. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Silberer, Carina and Mirella Lapata (2014). “Learning grounded meaning representations with autoencoders”. *52nd Annual Meeting of the Association for Computational Linguistics*.
- Silver, David, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. (2016). “Mastering the game of Go with deep neural networks and tree search”. *Nature*.
- Sipos, Ruben, Dmitriy Fradkin, Fabian Moerchen, and Zhuang Wang (2014). “Log-based predictive maintenance”. In: *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*.
- Sipple, John (2020). “Interpretable, Multidimensional, Multimodal Anomaly Detection with Negative Sampling for Detection of Device Failure”. In: *Proceedings of the 37th International Conference on Machine Learning*.
- Snell, Jake, Kevin Swersky, and Richard Zemel (2017). “Prototypical networks for few-shot learning”. *Advances in neural information processing systems*.
- Soatto, Stefano and Alessandro Chiuso (2016). “Modeling Visual Representations: Defining Properties and Deep Approximations”. In: *4th International Conference on Learning Representations*.
- Solaiman, Irene (2023). “The Gradient of Generative AI Release: Methods and Considerations”. *arXiv preprint arXiv:2302.04844*.
- Song, Kaitao, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu (2020). “Mpnet: Masked and permuted pre-training for language understanding”. *Advances in Neural Information Processing Systems*.
- Song, Yisheng, Ting Wang, Puyu Cai, Subrota K Mondal, and Jyoti Prakash Sahoo (2022). “A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities”. *ACM Computing Surveys*.
- Souza, Vinicius M.A., Denis M. dos Reis, André G. Maletzke, and Gustavo E.A.P.A. Batista (2020). “Challenges in benchmarking stream learning algorithms with real-world data”. *Data Mining and Knowledge Discovery*.
- Sparck Jones, Karen (1972). “A statistical interpretation of term specificity and its application in retrieval”. *Journal of documentation*.
- Sridharan, Karthik and Sham M. Kakade (2008). “An Information Theoretic Framework for Multi-view Learning”. In: *21st Annual Conference on Learning Theory*.
- Srivastava, Nitish and Ruslan Salakhutdinov (2012). “Learning representations for multimodal data with deep belief nets”. *International Conference on Machine Learning Workshop*.
- Srivastava, Nitish and Russ R Salakhutdinov (2012). “Multimodal learning with deep boltzmann machines”. *Advances in neural information processing systems*.
- Stein, Barry E and M Alex Meredith (1993). *The merging of the senses*. The MIT press.
- Stokes, Jonathan M, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackermann, et al. (2020). “A deep learning approach to antibiotic discovery”. *Cell*.

- Strubell, Emma, Ananya Ganesh, and Andrew McCallum (2019). “Energy and Policy Considerations for Deep Learning in NLP”. In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, Volume 1: Long Papers*.
- Sun, Chen, Fabien Baradel, Kevin Murphy, and Cordelia Schmid (2019). “Contrastive Bidirectional Transformer for Temporal Representation Learning”. *Computing Research Repository* abs/1906.05743.
- Sun, Chen, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid (2019). “Videobert: A joint model for video and language representation learning”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Sun, Jiedi et al. (2017). “Intelligent Bearing Fault Diagnosis Method Combining Compressed Data Acquisition and Deep Learning”. *IEEE Transactions on Instrumentation and Measurement*.
- Sun, Qianru, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele (2019). “Meta-transfer learning for few-shot learning”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Sun, Shengli, Qingfeng Sun, Kevin Zhou, and Tengchao Lv (2019). “Hierarchical attention prototypical networks for few-shot text classification”. In: *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing*.
- Sundermeyer, Martin, Ralf Schlüter, and Hermann Ney (2012). “LSTM neural networks for language modeling”. In: *Thirteenth annual conference of the international speech communication association*.
- Sung, Flood, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales (2018). “Learning to compare: Relation network for few-shot learning”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Sutskever, Ilya, James Martens, George Dahl, and Geoffrey Hinton (2013). “On the importance of initialization and momentum in deep learning”. In: *Proceedings of the 30th International Conference on Machine Learning*.
- Sutskever, Ilya, Oriol Vinyals, and Quoc Le (2014). “Sequence to Sequence Learning with Neural Networks”. *Advances in Neural Information Processing Systems*.
- Taheri-Garavand, Amin et al. (2015). “An intelligent approach for cooling radiator fault diagnosis based on infrared thermal image processing technique”. *Applied Thermal Engineering*.
- Taj, SM, SM Rizwan, BM Alkali, DK Harrison, and GL Taneja (2017). “Reliability analysis of a single machine subsystem of a cable plant with six maintenance categories”. *International Journal of Applied Engineering Research*.
- Talmor, Alon, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannaneh Hajishirzi, and Jonathan Berant (2021). “MultiModalQA: complex question answering over text, tables and images”. In: *9th International Conference on Learning Representations*.
- Tam, Derek, Rakesh R. Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel (2021). “Improving and Simplifying Pattern Exploiting Training”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.
- Tay, Yi, Mostafa Dehghani, Dara Bahri, and Donald Metzler (2022). “Efficient transformers: A survey”. *ACM Computing Surveys*.

- Taylor, Ross, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic (2022). “Galactica: A large language model for science”. *arXiv preprint arXiv:2211.09085*.
- Taylor, Wilson L (1953). ““Cloze procedure”: A new tool for measuring readability”. *Journalism quarterly*.
- Teske, JJ, JC Liljegren, and DL Sisterson (2001). *Long-term analysis of the corrective maintenance records of the ARM SGP CART*. Technical report. Argonne National Lab., IL (US).
- Thoppilan, Romal, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. (2022). “Lamda: Language models for dialog applications”. *arXiv preprint arXiv:2201.08239*.
- Thrun, Sebastian and Lorien Y. Pratt, eds. (1998). *Learning to Learn*. Springer.
- Tian, Yonglong, Dilip Krishnan, and Phillip Isola (2020). “Contrastive multiview coding”. In: *Computer Vision—ECCV 2020: 16th European Conference*.
- Tian, Yonglong, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola (2020). “What makes for good views for contrastive learning?” *Advances in neural information processing systems*.
- Tian, Yonglong, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola (2020). “Rethinking few-shot image classification: a good embedding is all you need?” In: *Computer Vision—ECCV 2020: 16th European Conference*.
- Tian, Zhengkun, Jiangyan Yi, Ye Bai, Jianhua Tao, Shuai Zhang, and Zhengqi Wen (2020). “Synchronous Transformers for end-to-end Speech Recognition”. *IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*.
- Tishby, Naftali, Fernando C Pereira, and William Bialek (2000). “The information bottleneck method”. *arXiv preprint physics/0004057*.
- Tosh, Christopher, Akshay Krishnamurthy, and Daniel Hsu (2021). “Contrastive learning, multi-view redundancy, and linear models”. In: *Algorithmic Learning Theory*.
- Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. (2023). “Llama: Open and efficient foundation language models”. *arXiv preprint arXiv:2302.13971*.
- Tripathi, Anshuman, Jaeyoung Kim, Qian Zhang, Han Lu, and Hasim Sak (2020). “Transformer transducer: One model unifying streaming and non-streaming speech recognition”. *arXiv preprint arXiv:2010.03192*.
- Tsai, Yao-Hung Hubert, Shaojie Bai, Paul Pu Liang, J Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov (2019). “Multimodal transformer for unaligned multimodal language sequences”. In: *Proceedings of the conference. Association for Computational Linguistics. Meeting*.
- Turing, Alan M (1950). *Computing machinery and intelligence*.
- Vapnik, Vladimir (2000). *The Nature of Statistical Learning Theory*. Statistics for Engineering and Information Science. Springer.
- Vapnik, Vladimir N (1999). “An overview of statistical learning theory”. *IEEE transactions on neural networks*.
- Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin (2017). “Attention is all you need”. *Advances in neural information processing systems*.

- Veilleux, Olivier, Malik Boudiaf, Pablo Piantanida, and Ismail Ben Ayed (2021). “Realistic evaluation of transductive few-shot learning”. *Advances in Neural Information Processing Systems*.
- Venkatasubramanian, Venkat et al. (2003). “A review of process fault detection and diagnosis. Part I: Quantitative model-based methods 27(3), 293–311. Part II: Qualitative models and search strategies 27(3), 313–32. Part III: Process history based methods 27(3), 327–346”. *Computers & Chemical Engineering*.
- Vinyals, Oriol, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojciech M Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, et al. (2019). “AlphaStar: Mastering the real-time strategy game starcraft ii”. *DeepMind blog*.
- Vinyals, Oriol, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu koray, and Daan Wierstra (2016). “Matching Networks for One Shot Learning”. In: *Advances in Neural Information Processing Systems*.
- Wan, Zhibin, Changqing Zhang, Pengfei Zhu, and Qinghua Hu (2021). “Multi-view information-bottleneck representation learning”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Wang, Chengyi, Yu Wu, Liang Lu, Shujie Liu, Jinyu Li, Guoli Ye, and Ming Zhou (2020). “Low Latency End-to-End Streaming Speech Recognition with a Scout Network”. In: *21st Annual Conference of the International Speech Communication Association*.
- Wang, Feng et al. (2016). “Bilevel feature extraction-based text mining for fault diagnosis of railway systems”. *IEEE TITS*.
- Wang, Jinjiang et al. (2019). “Machine vision intelligence for product defect inspection based on deep learning and Hough transform”. *Journal of Manufacturing Systems*.
- Wang, Qi, Claire Boudreau, Qixing Luo, Pang-Ning Tan, and Jiayu Zhou (2019). “Deep multi-view information bottleneck”. In: *Proceedings of the 2019 SIAM International Conference on Data Mining*.
- Wang, Sen, Xiaoqin Liu, Tangfeng Yang, and Xing Wu (2018). “Panoramic crack detection for steel beam based on structured random forests”. *IEEE Access*.
- Wang, Sinong, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma (2020). “Linformer: Self-attention with linear complexity”. *arXiv preprint arXiv:2006.04768*.
- Wang, Wenhui, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou (2020). “Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers”. *Advances in Neural Information Processing Systems*.
- Wang, Yikai, Chengming Xu, Chen Liu, Li Zhang, and Yanwei Fu (2020). “Instance credibility inference for few-shot learning”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.
- Wei, Jason, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le (2022). “Finetuned Language Models are Zero-Shot Learners”. In: *The Tenth International Conference on Learning Representations*.
- Wei, Jason, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus (2022). “Emergent Abilities of Large Language Models”. *Transactions on Machine Learning Research*.

- Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou (2022). "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models". In: *NeurIPS*.
- Wen, Long et al. (2017). "A New Convolutional Neural Network Based Data-Driven Fault Diagnosis Method". *IEEE Transactions on Industrial Electronics*.
- (2019). "A New Snapshot Ensemble Convolutional Neural Network for Fault Diagnosis". *IEEE Access*.
- Woods, William A (1977). "Lunar rocks in natural English: Explorations in natural language question answering."
- Wu, Bingjie et al. (2021). "Simultaneous-fault diagnosis considering time series with a deep learning transformer architecture for air handling units". *Energy and Buildings*.
- Wu, Chunyang, Yongqiang Wang, Yangyang Shi, Ching-Feng Yeh, and Frank Zhang (2020). "Streaming Transformer-Based Acoustic Models Using Self-Attention with Augmented Memory". In: *21st Annual Conference of the International Speech Communication Association*.
- Wu, Mike, Chengxu Zhuang, Milan Mosse, Daniel Yamins, and Noah D. Goodman (2020). "On Mutual Information in Contrastive Learning for Visual Representations". *Computing Research Repository* abs/2005.13149.
- Xia, Min et al. (2017). "Fault Diagnosis for Rotating Machinery Using Multiple Sensors and Convolutional Neural Networks". *IEEE/ASME Transactions on Mechatronics*.
- Xie, Junyuan, Ross Girshick, and Ali Farhadi (2016). "Unsupervised deep embedding for clustering analysis". In: *International conference on machine learning*.
- Xie, Yuan and Tao Zhang (2018). "Imbalanced learning for fault diagnosis problem of rotating machinery based on generative adversarial networks". In: *37th Chinese Control Conference*.
- Xu, Peng, Xiatian Zhu, and David A Clifton (2022). "Multimodal learning with transformers: a survey". *arXiv preprint arXiv:2206.06488*.
- Yam, R. et al. (2001). "Intelligent Predictive Decision Support System for Condition-Based Maintenance". *International Journal of Advanced Manufacturing Technology*.
- Yamaguchi, Atsuki, George Chrysostomou, Katerina Margatina, and Nikolaos Aletras (2021). "Frustratingly Simple Pretraining Alternatives to Masked Language Modeling". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.
- Yang, Ling, Liangliang Li, Zilun Zhang, Xinyu Zhou, Erjin Zhou, and Yu Liu (2020). "Dpgn: Distribution propagation graph network for few-shot learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.
- Yang, Bo-Suk et al. (2008). "Random forests classifier for machine fault diagnosis". *Journal of Mechanical Science and Technology*.
- Yang, Zhe et al. (2021). "A multi-branch deep neural network model for failure prognostics based on multimodal data". *Journal of Manufacturing Systems*.
- Yang, Zhilin, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le (2019). "XLnet: Generalized autoregressive pretraining for language understanding". *Advances in neural information processing systems*.
- Yang, Zichao, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola (2016). "Stacked attention networks for image question answering". *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

- Ye, Qinyuan, Bill Yuchen Lin, and Xiang Ren (2021). “CrossFit: A Few-shot Learning Challenge for Cross-task Generalization in NLP”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.
- Yeh, Ching-Feng, Yongqiang Wang, Yangyang Shi, Chunyang Wu, Frank Zhang, Julian Chan, and Michael L Seltzer (2021). “Streaming attention-based models with augmented memory for end-to-end speech recognition”. In: *2021 IEEE Spoken Language Technology Workshop*.
- Yu, Kun et al. (2019). “A bearing fault and severity diagnostic technique using adaptive deep belief networks and Dempster–Shafer theory”. *Structural Health Monitoring*.
- Yu, Wenmeng, Hua Xu, Ziqi Yuan, and Jiele Wu (2021). “Learning modality-specific representations with self-supervised multi-task learning for multimodal sentiment analysis”. In: *Proceedings of the AAAI conference on artificial intelligence*.
- Yuan, Jiahong and Mark Y. Liberman (2008). “Speaker identification on the SCOTUS corpus”. *Journal of the Acoustical Society of America*.
- Zadeh, Amir, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency (2017). “Tensor Fusion Network for Multimodal Sentiment Analysis”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.
- Zadeh, Amir, Paul Pu Liang, Jonathan Vanbriesen, Soujanya Poria, Edmund Tong, Erik Cambria, Minghai Chen, and Louis Philippe Morency (2018). “Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph”. *56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*.
- Zadeh, Amir, Soujanya Poria, Paul Pu Liang, Erik Cambria, Navonil Mazumder, and Louis Philippe Morency (2018). “Memory fusion network for multi-view sequential learning”. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*.
- Zaheer, Manzil, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. (2020). “Big bird: Transformers for longer sequences”. *Advances in neural information processing systems*.
- Zarei, Jafar et al. (2014). “Vibration analysis for bearing fault detection and classification using an intelligent filter”. *Mechatronics*.
- Zhang, Qian, Han Lu, Hasim Sak, Anshuman Tripathi, Erik McDermott, Stephen Koo, and Shankar Kumar (2020). “Transformer Transducer: A Streamable Speech Recognition Model with Transformer Encoders and RNN-T Loss”. *IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*.
- Zhang, Shen et al. (2020). “Deep Learning Algorithms for Bearing Fault Diagnostics—A Comprehensive Review”. *IEEE Access*.
- Zhang, Yue and Joakim Nivre (2011). “Transition-based Dependency Parsing with Rich Non-local Features”. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.
- Zhang, Zhenyou et al. (2013). “Fault diagnosis and prognosis using wavelet packet decomposition, Fourier transform and artificial neural network”. *Journal of Intelligent Manufacturing*.
- Zhao, Wayne Xin, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. (2023). “A survey of large language models”. *arXiv preprint arXiv:2303.18223*.

- Zhao, Zihao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh (2021). “Calibrate Before Use: Improving Few-shot Performance of Language Models”. In: *Proceedings of the 38th International Conference on Machine Learning*.
- Zhou, Funa et al. (2018). “A Multimodal Feature Fusion-Based Deep Learning Method for On-line Fault Diagnosis of Rotating Machinery”. *Sensors*.
- Zhu, Xiaojin Jerry (2005). “Semi-supervised learning literature survey”.
- Ziegler, Daniel M, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving (2019). “Fine-tuning language models from human preferences”. *arXiv preprint arXiv:1909.08593*.
- Ziko, Imtiaz, Jose Dolz, Eric Granger, and Ismail Ben Ayed (2020). “Laplacian regularized few-shot learning”. In: *International conference on machine learning*.