Multidimensional Adaptation of Language Models: Domains, Languages, and Social Dimensions

Inauguraldissertation
zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
der Universität Mannheim

vorgelegt von

Chia-Chien Hung
aus Tainan

Mannheim, 2024

Dekan:	Professor Dr. Claus Hertling, Universität Mannheim
Referent:	Professor Dr. Simone Paolo Ponzetto, Universität Mannheim
Korreferent:	Professor Dr. Goran Glavaš, University of Würzburg
Korreferent:	Professor Dr. Anne Lauscher, University of Hamburg
Korreferent:	Professor Dr. Lucie Flek, University of Bonn

Tag der mündlichen Prüfung: 06. December 2024

Acknowledgements

I would like to express my sincere gratitude to everyone who supported me throughout my Ph.D. journey. You thoroughly considered the obstacles that I had faced and cheered with me for the achievement I had made.

I would like to thank my supervisor, Prof. Simone Paolo Ponzetto. You supported me with both academic and industrial pursuits and gave me the freedom to explore my research passions. The same honors go to my supervisor, Prof. Goran Glavaš. Your closer guidance, mental support, and encouragement to explore diverse research angles have been invaluable. Thanks for your exceptional dedication as both a great mentor and supervisor. I also want to thank my third supervisor, Prof. Anne Lauscher, for guiding me in both my research writing and presentation, and inspiring me with your enthusiasm for academic research. In addition, I am grateful to my Ph.D. committee member, Prof. Lucie Flek – without you, it would not have been possible to complete the journey. Thank you for your valuable feedback.

I would like to thank all the collaborators throughout my Ph.D. journey, especially Dr. Ivan Vulić and Prof. Dirk Hovy, for their valuable guidance and insights. Working with you has been a great learning experience. I also want to thank my mentors at Bosch during my Ph.D. sabbatical: Prof. Jannik Strötgen and Dr. Lukas Lange. You've made research discussions exciting and fulfilling, and I am so grateful for the opportunity to work and learn from you. Special thanks to my support at NEC, especially Dr. Carolin Lawrence, for guiding and caring for my feelings along with my achievements.

Further, I would like to extend my gratitude to the amazing people in DWS Group, especially Sotaro, Tommaso, Robert, Tornike, Daniel, Anna, Pedro, Daryna, Ines, Ralph, Alex, and Keti. Our collaborations, discussions, and coffee/tea-break chats have added so much value to my Ph.D. journey. You've all been such an integral part of this experience.

I would like to thank my friends: Qi, Yaqi, and Xinmei, who stood by me during tough pandemic times. Your constant support was my anchor. A huge shout-out to Jun-Ting, Yen-Ting, Chia-Hsin, Fan-Ni, Han-Ni, Ming, Fang, Jade, Ta-Wei, Meng-Chin, Pei, Julia, Puli, Juliena, Jeff, Yen-Ju, Nipuni, Jeanie, Ciao-Hsin, Chieh-Han, and Simona. Your belief in me never wavered – knowing you were cheering me on made all the difference.

And finally, to my family – my parents and my sister. Your love and support have been my foundation. This journey wasn't just mine; it was ours. You were there for every step, every stumble, every success. This achievement is as much yours as it is mine.

Sincerely, with all my appreciation and love.

Abstract

The emergence of Pre-trained Language Models (PLMs) has revolutionized the field of natural language processing, yielding remarkable improvements for a wide range of tasks. While PLMs excel at discerning language patterns and crafting coherent narratives, they exhibit limitations when adapting to specialized topics, such as medical concepts, cross-lingual conversational systems, and contents derived from diverse sociodemographic backgrounds – necessitating specific terminologies, deep contextual understanding, and sensitivity to socially-aware language use. These limitations, collectively referred to as *adaptation barrier*, underscore the difficulty of adapting PLMs to fields demanding expert knowledge, e.g., medical terminology, multilingual applications, or social-contextual interactions for understanding cultural idioms. In such contexts, the deficiency of topic-relevant information within the model's training data can significantly impair the performance of language models in practical applications.

To address the *adaptation barrier* of PLMs, recent studies focused on techniques, such as domain adaptation, cross-lingual transfer learning, or incorporating external knowledge sources from diverse sociodemographic contexts to bridge the gap between generalizability and the specific demands of specialized topics. These studies highlight the considerable potential for enhancing performance by integrating knowledge from domain-specific, language-specific, or social-related resources into general-purpose PLMs. While research on (i) adapting PLMs to specific topics is an active and dynamic field, and (ii) investigating the pressing need for effective adaptation methods is conducted in prior studies, these research directions have been confined to a singular perspective: either domain-specific, language-specific, or solely centered on the social dimension. This limited scope has constrained the in-depth exploration of multiple facets and viewpoints concerning the efficacy of the proposed adaptation approaches. Furthermore, several promising research directions are under-explored and warrant attention. These include the development of task-agnostic adaptation techniques applicable across tasks, as well as adaptation strategies suitable for deployment in multi-domain and multilingual contexts. Moreover, there is a pressing need to enhance the efficiency of adapting PLMs, striking a balance between parameter utilization and data requirements – a set of vital factors essential for the viability of these models in applied scenarios.

In this thesis, we systematically conduct experiments across multiple dimensions and perspectives, aimed at closing the research gap concerning the *adaptation barrier* of PLMs. Concretely, we center on three key challenges (**C1**, **C2**, **C3**) – *effectiveness*, *efficiency*, *interpretability*, inherent in adapting current state-of-the-art PLMs to diverse *domains*, *languages*, and *social dimensions*:

(C1) Effectiveness: PLMs are typically trained on heterogeneous data with an emphasis on achieving generalizability – *not* tailored for specific domains, languages, or social contexts. We introduce *task-agnostic* adaptive pre-training methods with multiple transfer learning objectives. We conduct experiments on self-supervised domain adaptation and cross-lingual transfer for task-oriented dialogs, along with hybrid setups combining

multiple transfer learning methods for demographic adaptation. We demonstrate the effectiveness of our proposed methods, enabling versatile use across multi-domain and multilingual use cases, and contrasting them with task-specific approaches tailored for singular task.

(C2) Efficiency: given the constraints posed by computational resources and the challenges of acquiring extensive labeled training data in practical scenarios, we delve into methodologies that are considered data-efficient and parameter-efficient. We introduce a term-matching technique to gather domain-specific data efficiently and propose leveraging cross-lingual corpora for multilingual dialog specialization, facilitating both domain and language adaptation. We further present a novel task-agnostic domain adaptation approach, enhancing its efficacy in scenarios involving multiple domains and with limited data availability. Our findings demonstrate the feasibility and scalability of language model adaptation in resource-limited settings.

(C3) Interpretability: to enhance model transparency and unveil both strengths and limitations, we conduct multi-faceted controlled experiments to evaluate the implications of domain, language, and demographic adaptation techniques. We study the effects of cross-domain transfer and token-level segmentation use cases, providing insights into how cross-domain knowledge influences model behavior and how domain-specific information is captured with token-level control. We further conduct controlled experiments to examine the impact of demographic adaptation, highlighting the negligible effect and suggesting future research attention. Through systematic analyses, we provide a comprehensive understanding of how adaptation techniques influence model interpretability, while shedding light on further challenges and adaptability.

Through our findings, we hope to contribute to the advancement of more effective and adaptable language models across multiple dimensions. Further, we hope that our research outcomes will pave the way for the applicability of these models in addressing a broad spectrum of real-world challenges, thereby mitigating the *adaptation barrier* associated with PLMs.

Zusammenfassung

Die Entstehung von *vortrainierten Sprachmodellen* (Pre-trained Language Models, PLMs) hat den Bereich der Verarbeitung natürlicher Sprache revolutioniert und bemerkenswerte Verbesserungen für ein breites Spektrum an Aufgaben gebracht. PLMs sind zwar hervorragend in der Lage, Sprachmuster zu erkennen und kohärente Erzählungen zu erstellen, weisen aber Einschränkungen bei der Anpassung an spezielle Themen auf, darunter medizinische Konzepte, sprachübergreifende Konversationssysteme und Inhalte, die aus unterschiedlichen sozio-demografischen Hintergründen stammen, da diese spezifischen Terminologien ein tiefes kontextuelles Verständnis und Sensibilität für soziale Nuancen erfordern. Diese Einschränkungen, die zusammenfassend als *Anpassungsbarriere* bezeichnet werden, unterstreichen die Schwierigkeit der Anpassung von PLMs an Bereiche, die Expertenwissen erfordern, z. B. medizinische Terminologie, mehrsprachige Anwendungen oder sozial-kontextuelle Interaktionen zum Verständnis kultureller Idiome. In solchen Kontexten kann der Mangel an themenrelevanten Informationen in den Trainingsdaten des Modells die Leistung von Sprachmodellen in praktischen Anwendungen erheblich beeinträchtigen.

Um die *Anpassungsbarriere* von PLMs zu überwinden, haben sich neuere Studien auf Techniken wie Domänenanpassung, sprachübergreifendes Transferlernen oder die Einbeziehung externer Wissensquellen aus verschiedenen soziodemografischen Kontexten konzentriert, um die Kluft zwischen Verallgemeinerbarkeit und den spezifischen Anforderungen spezialisierter Themen zu überbrücken. Diese Studien verdeutlichen das beträchtliche Potenzial zur Leistungssteigerung durch die Integration von Wissen aus domänenspezifischen, sprachspezifischen oder sozialbezogenen Ressourcen in allgemein einsetzbare PLMs. Während die Forschung zur (i) Anpassung von PLMs an spezifische Themen ein aktives und dynamisches Feld ist und (ii) der dringende Bedarf an effektiven Anpassungsmethoden in früheren Studien untersucht wurde, waren diese Forschungsrichtungen auf eine singuläre Perspektive beschränkt: entweder domänenspezifisch, sprachspezifisch oder ausschließlich auf die soziale Dimension konzentriert. Dieser begrenzte Umfang hat die eingehende Erforschung der verschiedenen Facetten und Standpunkte in Bezug auf die Wirksamkeit der vorgeschlagenen Anpassungsansätze eingeschränkt. Darüber hinaus sind mehrere vielversprechende Forschungsrichtungen noch nicht ausreichend erforscht und verdienen Aufmerksamkeit. Dazu gehören die Entwicklung von aufgabenanagnotischen Adaptionstechniken, die aufgabenübergreifend anwendbar sind, sowie Adaptionsstrategien, die sich für den Einsatz in multidisziplinären und mehrsprachigen Kontexten eignen. Darüber hinaus besteht die dringende Notwendigkeit, die Effizienz der Anpassung von PLMs zu verbessern und ein Gleichgewicht zwischen Parameternutzung und Datenanforderungen herzustellen – eine Reihe von entscheidenden Faktoren, die für die Realisierbarkeit dieser Modelle in Anwendungsszenarien unerlässlich sind.

In dieser Arbeit führen wir systematisch Experimente durch, um die Forschungslücke zur *Anpassungsbarriere* von PLMs zu schließen. Dabei konzentrieren wir uns auf drei

zentrale Herausforderungen (**C1**, **C2**, **C3**) – *Effektivität*, *Effizienz*, *Interpretierbarkeit*, die mit der Anpassung moderner PLMs an unterschiedliche *Domänen*, *Sprachen* und *soziale Dimensionen* verbunden sind:

(C1) Effektivität: PLMs werden meist auf heterogenen Daten trainiert, um Verallgemeinerbarkeit zu erreichen, sind aber *nicht* auf spezifische Domänen, Sprachen oder soziale Kontexte abgestimmt. Wir stellen *aufgaben-agnostische* adaptive Pre-Training-Methoden mit mehreren Transfer-Lernzielen vor. Unsere Experimente umfassen selbstüberwachte Domänenanpassung, sprachübergreifenden Transfer für aufgabenorientierte Dialoge und hybride Ansätze zur demographischen Anpassung. Wir zeigen die Effektivität unserer Methoden, die vielseitig in verschiedenen Domänen und Sprachen einsetzbar sind, und vergleichen sie mit aufgabenspezifischen Ansätzen.

(C2) Effizienz: In Anbetracht der Beschränkungen, die sich aus den Rechenressourcen ergeben, und der Herausforderungen, die sich aus der Beschaffung umfangreicher gelabelter Trainingsdaten in praktischen Szenarien ergeben, befassen wir uns mit Methoden, die als daten- und parametereffizient gelten. Wir stellen ein Term-Matching-Verfahren vor, um domänenspezifische Daten effizient zu erfassen, und schlagen vor, sprachübergreifende Korpora für die mehrsprachige Dialogspezialisierung zu nutzen, was sowohl die Domänen- als auch die Sprachanpassung erleichtert. Darüber hinaus stellen wir einen neuartigen, aufgabenagnostischen Ansatz zur Domänenanpassung vor, der die Wirksamkeit des Term-Matching-Verfahrens in Szenarien mit mehreren Domänen und begrenzter Datenverfügbarkeit erhöht. Unsere Ergebnisse zeigen die Machbarkeit und Skalierbarkeit der Sprachmodellanpassung in ressourcenbeschränkten Umgebungen.

(C3) Interpretierbarkeit: Um die Transparenz des Modells zu erhöhen und sowohl Stärken als auch Grenzen aufzuzeigen, führen wir vielschichtige kontrollierte Experimente durch, um die Auswirkungen von Techniken zur Anpassung an Domänen, Sprache und Demografie zu bewerten. Wir untersuchen die Auswirkungen des domänenübergreifenden Transfers und der Segmentierung auf Token-Ebene in Anwendungsfällen, die Aufschluss darüber geben, wie domänenübergreifendes Wissen das Modellverhalten beeinflusst und wie domänenspezifische Informationen mit der Kontrolle auf Token-Ebene erfasst werden. Darüber hinaus führen wir kontrollierte Experimente durch, um die Auswirkungen der demografischen Anpassung zu untersuchen, wobei wir deren vernachlässigbaren Effekt hervorheben und künftige Forschungsschwerpunkte vorschlagen. Durch systematische Analysen vermitteln wir ein umfassendes Verständnis dafür, wie Anpassungstechniken die Interpretierbarkeit von Modellen beeinflussen, und beleuchten gleichzeitig weitere Herausforderungen und Anpassungsmöglichkeiten.

Wir hoffen, mit unseren Ergebnissen einen Beitrag zur Entwicklung effektiverer und anpassungsfähigerer Sprachmodelle über mehrere Dimensionen hinweg leisten zu können. Darüber hinaus hoffen wir, dass unsere Forschungsergebnisse den Weg für die Anwendbarkeit dieser Modelle bei der Bewältigung eines breiten Spektrums von Herausforderungen in der realen Welt ebnen und damit die mit PLMs verbundene *Anpassungsbarriere* abmildern.

Contents

List of Publications

The work presented in this thesis has been previously published in proceedings of top-tier international conferences, encompassing textual materials, tables, and figures. The publications are referenced in the subsequent Chapters and are listed here in inverse chronological order:

Chia-Chien Hung, Lukas Lange, and Jannik Strötgen. 2023. TADA: Efficient Task-Agnostic Domain Adaptation for Transformers. In *Findings of the Association for Computational Linguistics (ACL 2023)*, pages 487–503, Toronto, Canada, July 2023. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Dirk Hovy, Simone Paolo Ponzetto, and Goran Glavaš. 2023. Can Demographic Factors Improve Text Classification? Revisiting Demographic Adaptation in the Age of Transformers. In *Findings of the Association for Computational Linguistics (EACL 2023)*, pages 1565–1580, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. 2022. Multi^{2}WOZ: A Robust Multilingual Dataset and Conversational Pretraining for Task-Oriented Dialog. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2022)*, pages 3687–3703, Seattle, United States, July 2022. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Simone Paolo Ponzetto, and Goran Glavaš. 2022. DS-TOD: Efficient Domain Specialization for Task-Oriented Dialog. In *Findings of the Association for Computational Linguistics (ACL 2022)*, pages 891–904, Dublin, Ireland, May 2022. Association for Computational Linguistics.

The publications listed before are specifically pertinent to the content of this thesis, whereas the other published research work that the author contributed during the course of her doctoral studies is listed here in inverse chronological order:

Janek Herrlein, **Chia-Chien Hung**, and Goran Glavaš. 2024. anHalten: Cross-Lingual Transfer for German Token-Level Reference-Free Hallucination Detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop) (ACL 2024)*, pages 186–194, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

Chia-Chien Hung, Wiem Ben Rim, Lindsay Frost, Lars Bruckner, and Carolin Lawrence. 2023. Walking a Tightrope – Evaluating Large Language Models in High-Risk Domains. In *Proceedings of the 1st GenBench Workshop on (Benchmarking) Generalisation in NLP (GenBench@EMNLP 2023)*, pages 99–111, Singapore, December 2023. Association for Computational Linguistics.

Gorjan Radevski, Kiril Gashteovski, **Chia-Chien Hung**, Carolin Lawrence, and Goran Glavaš. 2023. Linking Surface Facts to Large-Scale Knowledge Graphs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*, pages 7189–7207, Singapore, December 2023. Association for Computational Linguistics.

Ashish Rana, Pujit Golchha, Roni Juntunen, Andreea Coaja, Ahmed Elzamarany, **Chia-Chien Hung**, and Simone Paolo Ponzetto. 2022. LeviRANK: Limited Query Expansion with Voting Integration for Document Retrieval and Ranking. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum (CLEF 2022)*, pages 3074–3089, Bologna, Italy, September 2022. CEUR Workshop Proceedings.

Pavani Rajula, **Chia-Chien Hung**, and Simone Paolo Ponzetto. 2022. Stacked Model based Argument Extraction and Stance Detection using Embedded LSTM model. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum (CLEF 2022)*, pages 3064–3073, Bologna, Italy, September 2022. CEUR Workshop Proceedings.

Chia-Chien Hung, Tommaso Green, Robert Litschko, Tornike Tsereteli, Sotaro Takeshita, Marco Bombieri, Goran Glavaš, and Simone Paolo Ponzetto. 2022. ZusammenQA: Data Augmentation with Specialized Models for Cross-lingual Open-retrieval Question Answering System. In *Proceedings of the Workshop on Multilingual Information Access (MIA@NAACL 2022)*, pages 77–90, Seattle, USA, July 2022. Association for Computational Linguistics.

List of Figures

List of Tables

List of Acronyms

BPE Byte-Pair Encoding. 18, 19, 27

CBOW Continuous Bag-of-Words. 16

DST Dialog State Tracking. 6, 45, 56–62, 70, 72–75, 77, 78, 83, 84, 88, 90–93, 95, 155, 157

MLM Masked Language Modeling. 7, 24, 26, 27, 45, 48, 51, 53, 57, 58, 65–67, 71–74, 83, 88, 89, 91–93, 95, 102, 103, 106–113, 122

MT machine translation. 22, 96

MTL Multi-Task Learning. 103, 106, 107

NER Named Entity Recognition. 6, 44, 70–75, 77, 78, 122, 155, 157

NLI Natural Language Inference. 6, 44, 70–78, 155–157

NLP Natural Language Processing. 3, 4, 6, 9, 11–13, 15, 16, 19, 23, 24, 27, 29, 30, 32, 33, 39, 43–45, 63, 81, 82, 96, 99–103, 115, 116, 123

NLU Natural Language Understanding. 6, 23

NSP Next Sentence Prediction. 24–27

OOV Out-of-Vocabulary. 12, 17, 18, 21

OS OpenSubtitles. 89, 90, 93–95

PLM Pre-trained Language Model. 24, 36, 39, 45, 46, 53, 54, 57, 63, 64, 66–68, 72, 83, 84, 88, 91, 96, 100–103, 105–107, 109, 110, 112–115, 120, 121

PLMs Pre-trained Language Models. 3–9, 18, 23, 24, 27, 29, 32–34, 36–38, 43–45, 47–49, 51, 53, 57–59, 62–64, 66–68, 72, 73, 77–79, 82–84, 90, 96, 97, 99–102, 104–107, 109, 110, 113–116, 119, 121–123

RR Response Retrieval. 6, 45, 56–58, 62, 70–75, 77, 78, 83, 88, 90–93, 95, 155, 157

RS Response Selection. 7, 45, 48, 53, 54, 57, 58, 83, 88, 89, 91–93, 95, 102

SG Skip-Gram. 16–18

SPA Sentence Piece Algorithm. 18, 25, 27

TLM Translation Language Modeling. 83, 89–95

TOD Task-oriented Dialog. 6–8, 28, 37, 44–46, 48–51, 53, 56, 58–60, 62, 63, 73, 81–85, 87–91, 93, 96, 97, 120

WPA Word-Piece Algorithm. 18, 25, 27, 68, 74

XLM-R XLM-RoBERTa. 27, 28, 34, 84, 88, 91, 109

Referring to specific languages, we use ISO 639-1 codes, e.g., EN for English.

Part I

Introduction and Theoretical Background

Chapter 1

Introduction

"Adapt or perish, now as ever, is Nature's inexorable imperative."

Herbert George Wells
«Mind at the End of Its Tether»

1.1 Motivation and Problem Statement

Language is a fundamental tool for human communication and understanding, enabling us to convey our thoughts, ideas, and emotions. Language as a tool for communication implies that there are underlying patterns and structures to language that could be analyzed and modeled. Language models are a technological innovation that relies on these patterns and structures to understand and generate natural language text (i.e., coherent and meaningful text) (Jurafsky and Martin, 2000), and are designed to mimic the way humans use language to communicate.

To illustrate the power of language models, let's consider the story of a chef who is just starting out her culinary career. At first, the chef is overwhelmed by the sheer number of ingredients and recipes available and struggles to create dishes that are both delicious and extraordinary. However, as she gains more experience with heterogeneous recipes, she begins to understand the patterns and rules that govern cooking and is capable of experimenting with new ingredients and techniques.

Similarly, language models learn from vast amounts of text data, identifying patterns and structures that could be used to understand and generate new texts. By analyzing large amounts of text data, a language model can learn the patterns of language use and identify common phrases and idioms. Pre-trained Language Models (PLMs) (Radford et al., 2018; Devlin et al., 2019) utilizing the *Transformer* (Vaswani et al., 2017) architecture, in particular, have gained widespread attention in recent years for their ability to perform a variety of Natural Language Processing (NLP) tasks without the necessity to be trained from scratch (Min et al., 2023). These PLMs, such as GPT (Radford et al., 2018) and

BERT (Devlin et al., 2019), are trained on massive amounts of general and heterogeneous corpora, enabling them to understand a wide range of texts, achieve outstanding performance across multiple tasks (e.g., Wang et al., 2022; Zhang et al., 2023), and be utilized for applications (Zhou et al., 2020; Valizadeh and Parde, 2022, *inter alia*).

While PLMs offer a powerful tool for NLP, they face several barriers to effectively deploy in real-world applications. One of the most significant barriers is the need to adapt language models to specific topics.[1] Just as a chef must adjust their recipes to fit different cuisines and dietary restrictions, language models must be tailored to the specific characteristics of different domains, languages, and social contexts in order to achieve optimal performance on selected tasks in practical use cases. For example, a language model trained on English data may not be as effective when applied to the Arabic text (Antoun et al., 2020; Lan et al., 2020), and a language model trained on formal language in books may struggle with informal language used by specific communities (e.g., slang and colloquialisms in social media) (Sun et al., 2024).

Here, we introduce the concept of *adaptation barrier*, indicating one of the key challenges of PLMs – these models are trained on massive and heterogeneous corpora with a focus on generalizability without addressing particular topic concerns. This means that while these models can recognize patterns in language and generate coherent text, they may not be well-suited for specific topics of interest (e.g., medical concepts, cross-lingual conversational systems, sociocultural contents), which require specialized terminologies, deep contextual understanding, and awareness of social nuances. In practice, the absence of such topic-relevant information can severely hurt performance in downstream applications, as shown in numerous studies (Ruder and Plank, 2018; Friedrich et al., 2020; Ben-David et al., 2020).

To address the challenge of *adaptation barrier* in PLMs, recent work has actively explored methods to adapt PLMs to specific topics of interest (Lee et al., 2019; Beltagy et al., 2019). These studies highlight the potential for significant performance improvements by incorporating knowledge from domain-specific, language-specific, or social-related resources into general-purpose PLMs (Gururangan et al., 2020; Xue et al., 2021; Lauscher et al., 2022a). While (i) adapting PLMs toward specific topics of interest is an active research topic, and (ii) the specific need for effective adaptation methods has been conducted in previous research, these studies, however, focused mostly on a narrow perspective: either on domain-only, language-only, or solely centered on the social dimension. The limited scope has restricted the exploration of multidimensional prospects and viewpoints on the efficacy of proposed adaptation approaches. Moreover, several underexplored research directions merit attention. These include the development of task-agnostic adaptation techniques for versatile task accommodation, adaptation methods for deployment in multi-domain and multilingual scenarios, and the optimization of efficiency concerning parameter and data usage – a set of crucial considerations for real-world applications.

[1] Here, the *topic* indicates all perspectives of applied scenarios, which focus on specific interests.

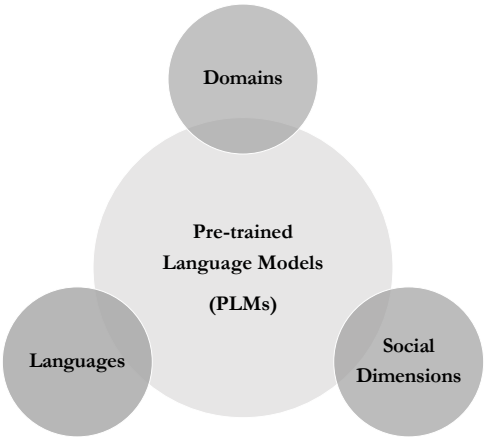


Figure 1.1: Illustration of multidimensional aspects of adapting PLMs investigated in this thesis: domains, languages, and social dimensions.

In this thesis, we aim towards closing the research gap concerning the existing *adaptation barrier* of PLMs by considering multidimensional perspectives as illustrated in Figure 1.1: domains, languages, and social dimensions. Domain adaptation involves specializing PLMs to handle specific fields of knowledge, such as medicine or finance, ensuring the understanding of specialized terminologies in domain-specific contents. Language adaptation focuses on enhancing the model's ability to process texts in the selected language(s), allowing it to handle linguistic diversity across those languages. Social dimension adaptation ensures PLMs are sensitive to cultural and societal factors, enabling them to align with social aspects (e.g., demographics), adhere to norms, and promote inclusivity. In particular, we systematically study the following challenges:

(C1) Effectiveness: Pre-trained language models are typically trained on large and heterogeneous corpora, which encompass generalizability but may not effectively capture the characteristics of specific fields in need. *How could we effectively adapt language representations to encode knowledge relevant to specific domains, languages, and social dimensions, leveraging task-agnostic methods?*

(C2) Efficiency: Computational resources are typically limited in real-world applications, and training data is often scarce in specific fields and languages. *How could we improve adaptation methods to optimize parameter efficiency and handle limited training data, ensuring practical deployment without substantial computational or resource overhead?*

(C3) Interpretability: Adaptation methods, despite their focus on improving model performance, often lack interpretable analyses, hindering the reliability and robustness of the models. *How could we ensure that the adapted PLMs truly encode the knowledge? How could we foster transparency with interpretable analyses that are understandable and meaningful to humans?*

For the key challenges listed above, we conduct a series of experiments and instantiate analyses, address the strengths and limitations of PLMs, and provide practical guidance on how to adapt these models to a specific subject area of interest. Through our findings, we hope to contribute to the development of more effective and adaptable language models that could be tailored to diverse NLP tasks, mitigating the *adaptation barrier* of PLMs.

1.2 Contributions

This thesis constitutes a substantial contribution to the field of NLP by addressing the challenges associated with adapting language models to multidimensional perspectives: domains, languages, and social dimensions. Our foremost aim is to develop robust and efficient techniques capable of surmounting *adaptation barrier* that may arise from these varied perspectives. Besides, we conduct thorough analyses to improve the interpretability of our findings. The efficacy of newly proposed adaptation techniques is assessed through the evaluation on Natural Language Understanding (NLU) tasks:[2] (a) Task-oriented Dialog (TOD) tasks, including Dialog State Tracking (DST) (Wu et al., 2020) and Response Retrieval (RR) (Henderson et al., 2020); (b) sequence tagging tasks (i.e., Named Entity Recognition (NER)) (Tjong Kim Sang and De Meulder, 2003; Uzuner et al., 2011; Salinas Alvarado et al., 2015; Bamman et al., 2019; Friedrich et al., 2020); (c) Natural Language Inference (NLI) task (Williams et al., 2018); and (d) text classification tasks (Hovy et al., 2015). NLU tasks are essential for comprehending and evaluating the adaptation of language models, ensuring their robustness and reliability across diverse contexts. We cover 14 domains, 7 languages, and 2 social dimensions,[3] encompassing multifaceted contexts.[4] To address the challenges identified in § 1.1, we introduce novel methods and resources as well as analytical insights. Concretely, our contributions are as follows:

Corpora. We construct textual resources for training and evaluating the adaptation approaches. We focus primarily on data-limited scenarios: how to efficiently collect sufficient data to facilitate effective knowledge transfer (**C2**). To tackle data limitations, we streamline the data collection process, ensuring efficient acquisition and utilization of limited resources across various domains and languages.

[2] NLU focuses on machine reading comprehension, interpretation, and extracting meanings from human language to simulate the human understanding process, enabling machines to understand the context, semantics, and intent behind the words and phrases used in human communication. NLU has various applications in fields such as sentiment analysis and information retrieval.

[3] Here, the social dimensions are explored through the lens of demographic adaptation, focusing on two factors: age and gender.

[4] Domains: taxi, restaurant, hotel, train, attraction, government, telephone, fiction, slate, travel, finance, news, clinical, science. Languages: English, German, Russian, Chinese, Arabic, Danish, French. Social Dimensions: age, gender.

1. DomainCC and DomainReddit: in order to advance research on domain adaptation with in-domain corpus training, we leverage a simple terminology extraction method to construct DomainCC and DomainReddit corpora, which are extended from CCNet (Wenzek et al., 2020) and Pushshift API (Baumgartner et al., 2020) respectively. The in-domain corpora are then utilized for conducting dialogic pre-training to PLMs for domain adaptation (see Chapter 3).
2. LangCC and LangOpenSubtitles: we compile target-language-specific as well as cross-lingual corpora for language adaptation and construct LangCC and LangOpenSubtitles. They are acquired from CCNet (Wenzek et al., 2020) and OpenSubtitles (Lison and Tiedemann, 2016) respectively. The collected corpora are then utilized for conducting dialogic pre-training to PLMs for language adaptation (see Chapter 4).
3. Multi^{2}WOZ: we introduce a novel multilingual multi-domain TOD dataset, derived from the well-established English dataset MultiWOZ (Budzianowski et al., 2018; Eric et al., 2020). Multi^{2}WOZ spans four typologically diverse languages (Chinese, German, Arabic, and Russian), containing gold-standard dialogs in target languages that are directly comparable with development and test portions of the English dataset, enabling reliable and comparative estimates of cross-lingual transfer performance for TOD. We carry out a two-phase translation of the English data: (a) *automatic translation*, and (b) *manual post-editing* of the translations. To measure the translation quality, an additional quality assurance step is further controlled to highlight the robustness and reliability of the newly introduced multilingual multi-domain TOD dataset (see Chapter 4).

Methods. To effectively and efficiently adapt language models to domains, languages, and social dimensions (**C1**, **C2**), we present the following four contributions that are aligned with the proposed task-agnostic adaptation methodology.

1. Dialogic Domain-Adaptive Pre-training: to address the challenge of adapting PLMs to conversational nature of dialogic tasks (Wu et al., 2020), we propose novel task-agnostic Response Selection (RS) objectives (Oord et al., 2018; Henderson et al., 2019c) applied on dialogic DomainReddit corpus – compared against Masked Language Modeling (MLM) objective (Devlin et al., 2019; Gururangan et al., 2020) on DomainCC corpus (see § 3.1). The comparison highlights how our proposed methods more effectively integrate and leverage domain-specific knowledge in dialogic contexts.

2. Efficient Domain-Adaptive Pre-training: the risk of catastrophic forgetting of the previously acquired knowledge (French, 1999) alleviates the effectiveness of the proposed domain adaptation approach (Pfeiffer et al., 2023). To address the issue, we propose a novel modular and parameter-efficient domain specialization method: TADA, and compare it against the domain-aware adapters (Houlsby et al., 2019) (see § 3.2).
3. Dialogic Language-Adaptive Pre-training: we introduce TOD-XLMR – a multilingual pre-trained language model, specialized in the English conversational corpus. LangCC and LangOpenSubtitles are further leveraged to facilitate language adaptation for a specific target language with the variants of conversational training objectives (see Chapter 4).
4. Multi-task Learning for Demographic-Adaptive Pre-training: we propose a novel task-agnostic method to adapt the language representations for the demographic factors of gender and age, using dynamic multi-task learning for adaptation. The approach couples language modeling objectives with the prediction of demographic classes to jointly learn the contextualized text representations while being sensitive to demographic variations (see Chapter 5).

Analyses. Based on the newly introduced resources and methods outlined above, we perform a series of analyses to gain deeper insights into adapting language models for domains, languages, and social dimensions (**C1**, **C2**), and further enhance model transparency with the concerns of human interpretability (**C3**).

1. To increase efficiency and alleviate the issue of catastrophic forgetting as compared to *full* domain-adaptive pre-training (Gururangan et al., 2020), we demonstrate consistent performance improvements of proposed *modular* and *parameter-efficient* domain specialization approach in both single-domain and multi-domain scenarios. Furthermore, we conduct systematic analyses to assess (a) the effect of domain-aware token representation, and (b) the few-shot transfer capability, ensuring the robustness of our proposed method (see § 3.2).
2. To handle the challenge of language ambiguity beyond English and leverage the multilingual PLMs, we present a new framework for *multilingual conversational specialization* of PLMs that aims to facilitate cross-lingual transfer for arbitrary downstream TOD tasks. We systematically benchmark a number of zero-shot and few-shot cross-lingual transfer approaches on Multi^{2}WOZ dataset. Our experimental results demonstrate that our proposed language adaptation approach enables an exceptionally *sample-efficient few-shot transfer* for downstream TOD tasks (see Chapter 4).

3. To evaluate the effectiveness of demographic specialization in multilingual PLMs, we evaluate across four languages (English, German, Danish, French) using a multilingual corpus of reviews annotated with demographic information. The initial experiments yield gains in most tasks and settings, consistent with earlier findings. With further analysis, the results reveal that most gains can be attributed to confounding effects of language and/or domain adaptation. The findings suggest that adaptation approaches in social dimensions – specifically in demographic factors, fail to instill demographic knowledge into PLMs, making it an open problem in the age of PLMs (see Chapter 5).

We hope that our work catalyzes future research into the challenges of the *adaptation barrier* of PLMs and draws further attention to the development of more effective and efficient approaches of adapting language models to applied perspectives. By incorporating interpretable analyses, we aim to provide deeper insights into the strengths and limitations of adaptation methods for multidimensional aspects, ultimately leading to more robust and reliable NLP applications.

1.3 Outline

This thesis consists of three parts, each of which covers multiple Chapters:

Part I. Contains the introduction (Chapter 1), which states the overview of the thesis. The subsequent theoretical background (Chapter 2) presents the fundamental knowledge related to language modeling and pre-trained language models. Building on the inherent characteristics of pre-trained language models and insights from previous research, we identify the challenges associated with adapting language models to multidimensional aspects.

Part II. Comprises a detailed discussion, spanning from domain adaptation (Chapter 3), language adaptation (Chapter 4) to demographic adaptation (Chapter 5). The discussion includes the curation of datasets, the proposed task-agnostic approaches of adapting language models to different objectives for diverse downstream tasks, and further ablation studies to demonstrate the effectiveness (**C1**), efficiency (**C2**), and interpretability (**C3**) of our proposed methods.

Part III. Concludes the thesis by summarizing the key findings and contributions. We discuss the limitations and provide directions for future research (Chapter 6).

Chapter 2

Theoretical Background

"All men have the stars, but they are not the same things for different people.
For some, who are travelers, the stars are guides.
For others they are no more than little lights in the sky.
For others, who are scholars, they are problems."

Antoine de Saint-Exupéry
«Le Petit Prince»

In this Chapter, we first give an overview of the advancements in language modeling techniques that have emerged in recent years (§ 2.1), and the extension of language models to domain-specific and multilingual use cases (§ 2.2). Further, we delve into the concepts of transfer learning and its effectiveness in adapting models for a wide range of NLP tasks, discussing promising methods and recent research directions (§ 2.3). We then highlight the challenges associated with language modeling and establish their connections to the subsequent Chapters, where we propose adaptation approaches concerning multi-dimensional perspectives: domain adaptation, language adaptation, and demographic adaptation (§ 2.4).

2.1 Language Modeling

Language is like a vast and intricate puzzle, with countless pieces that fit together to form meaning. Language models serve as tools to analyze and mimic this complex puzzle, helping us process and interpret language patterns. Over the years, language modeling has undergone a significant evolution, transitioning from early statistical methods to modern neural language models. Fundamentally, language modeling involves the development of machine-based systems that can comprehend and mimic human languages. It encompasses the task of estimating the probability distribution of word sequences within a language, thereby enabling the prediction of the most probable word or sequence of words to follow based on the words that have previously been observed. In the early development stage of language models, statistical methods, such as n-gram models, gained widespread popularity. These models relied on the frequency count of

n-grams (i.e., sequence of n words) in a text corpus to predict the subsequent word in a sequence. Despite their effectiveness, these models encounter difficulties in managing long-range dependencies and overcoming data sparsity issues. With the advent of neural networks, Bengio et al. (2000) addressed the limitation of data sparsity by introducing *neural language model*, utilizing a shallow feed-forward neural network to learn distributed representations of words and the language model jointly.[1] This approach tackled the "curse of dimensionality" by representing words as dense, low-dimensional vectors, known as *word embeddings* or *embeddings*, which encode semantic information based on their contexts in the training corpus. The introduction of neural language model, where word embeddings are learned parameters within the neural network, emerges as a solution to represent words numerically and capture their semantic meanings based on the distributional properties of words.

Subsequent work demonstrated the effectiveness of word embeddings in various NLP downstream tasks (Collobert and Weston, 2008). Mikolov et al. (2013) further advanced the quality of word embeddings with significant reduction of computational costs by introducing Word2Vec. However, these methods were tied to fixed window size and context-independent representation of words, struggling with handling long-range dependencies, word sense variations (i.e., polysemy), and Out-of-Vocabulary (OOV) issues. This led to the exploration of *contextualized embeddings* (Peters et al., 2018), which aimed to capture not only the semantic meaning of individual words but also their meaning within the context of a longer sentence or document.

More recently, attention-based models such as the *Transformer* architecture (Vaswani et al., 2017) with deep neural networks have gained widespread recognition and adoption in language modeling (Devlin et al., 2019). These models employ subword unit tokenization (Gage, 1994; Sennrich et al., 2016) to better handle OOV issue. Moreover, self-attention mechanisms enable them to assess and prioritize the relative significance of different words in a sequence. By doing so, they are able to more effectively capture long-term dependencies and handle polysemy with contextual representations, leading to significant enhancements in performance across a range of NLP tasks.

In the following Sections, we will delve deeper into the history of language models (§ 2.1.1), tracing their evolution and the key milestones. We explore the development of language models from early approaches like n-grams to more sophisticated neural language models with the discussion of word embeddings (§ 2.1.2) and Transformer (§ 2.1.3). We hope to provide a comprehensive understanding of language modeling and its role in advancing NLP techniques.

[1] The distributional hypothesis, proposed by Harris (1954), suggests that words with similar meanings tend to occur in similar contexts, and was captured by Firth (1957) by the well-known quote: "You shall know a word by the company it keeps." Embeddings allow the quantification of these distributional properties and enable capturing word relationships and similarities.

2.1.1 From N-Gram Models to Neural Language Models

In the realm of natural language processing, *language modeling* stands as a fundamental task, aiming to *estimate the probability of a given sequence*. The stream of language models can be traced back to the N-gram model, which has paved the way for more advanced neural language modeling techniques. To comprehend the connection between N-gram models and neural language models, let us embark on the journey.

Language modeling is a probability distribution over word sequences. Given a sequence with a number of s words ($w_1, w_2, \ldots, w_s$; or $w_{1:s}$), the probability of the entire sequence $P(w_{1:s})$ can be decomposed as the chain rule of probability (Jurafsky and Martin, 2000):

$$\begin{aligned} P(w_{1:s}) &= P(w_1)P(w_2|w_1)P(w_3|w_{1:2})\ldots P(w_s|w_{1:(s-1)}) \\ &= \prod_{k=1}^{s} P(w_k|w_{1:(k-1)}) \end{aligned} \tag{2.1}$$

The chain rule shows that the joint probability of the entire word sequences could be estimated by multiplying the number of conditional probabilities of a word given previous words. To reduce the difficulty of the modeling problem by taking advantage of the word order, and the fact that temporally closer words in the word sequence are statistically more dependent (Bengio et al., 2000), N-gram language models instead approximate the probability of a word by just the last few words. N-gram language model (or N-gram model in short) is a statistical approach to language modeling rooted in Markov assumptions: the probability of a word depends only on the previous word(s), which indicates that we can predict the probability of some future unit without looking too far into the past (Jurafsky and Martin, 2000). N-gram models estimate the probability of the next word in a sequence given the preceding $N-1$ words. The probability of the entire sequence $P(w_{1:s})$ can be approximated by:

$$P(w_{1:s}) \approx \prod_{k=1}^{s} P(w_k|w_{(k-1):(k-N+1)})\,, \tag{2.2}$$

where N is predefined and the probability is estimated by frequency counts of words within the corpus.[2]

N-gram models offer a probabilistic framework for language modeling and capture local dependencies within N words. Moreover, the simplicity and efficiency of the N-gram models make it a valuable tool in various practical applications in NLP. For text generation, N-gram models can be used to generate sentences by selecting the most probable next word based on preceding words. In sentiment analysis, N-gram models can

[2]The N in N-gram models represent the number of words considered together as a unit. For example, in a bigram (N=2; 2-Gram) model, two adjacent words are treated as a unit, while a trigram (N=3; 3-Gram) model considers three adjacent words.

be employed to determine the sentiment of a given text by analyzing the frequency and distribution of words or phrases that carry sentiments. By calculating the probabilities of specific N-grams associated with positive or negative sentiments, N-gram models can perform sentiment classification.

Despite their usefulness, N-gram models have notable limitations in three main perspectives: (1) Limited comprehension of long-range dependencies: N-gram models only consider a fixed number of preceding words (up to N-1), which restricts their ability to capture broader context, often resulting in limited capacity and repetitive patterns. (2) Lack of semantic understanding: no semantic information is conveyed by frequency counts, which may struggle with understanding negations, sarcasm, or more complex linguistic expressions of sentiment. (3) Data sparsity issue to handle out-of-vocabulary words.[3] The realization of the above issues has led to the development of more advanced approaches: neural language models.

Neural Language Models To overcome the limitations of N-gram models, neural language models emerged. Neural language models leverage the power of Artificial Neural Networks (ANN) (or NN in short) to learn complex patterns and representations from large-scale textual data. NN are a class of machine learning models inspired by the structure and function of biological neurons in the human brain. They consist of interconnected artificial neurons, called *nodes*, organized in multiple *layers*. Each node receives input signals, processes them using an *activation function*, and produces an output signal which further passes to other nodes. The connections between nodes are represented by *parameters*, which determine the strength of the influence between nodes and are learned during the machine learning process.

A neural network $f_\theta(x)$ can be viewed as a function parameterized by θ that takes an input x and maps it to a probability distribution over possible output(s) y. Any (deep) neural networks with K layers and without residual connections (He et al., 2016), could be considered as a composition of functions $f_\theta^k(\cdot)$, corresponding to each layer k (Collobert et al., 2011):

$$f_\theta(x) = f_\theta^K(f_\theta^{K-1}(\cdots f_\theta^1(x)\cdots)) \tag{2.3}$$

Compared to the neural network, a neural language model is identical to a (deep) neural network: a function $f_\theta(\cdot)$ parameterized by a large number of parameters θ (nowadays millions to trillions) that takes input texts T, containing sequences of words (each word is represented by a dense feature vector), and maps it to outputs y (if considering classification tasks, then y would be a finite set of candidates):

$$f_\theta(T) = f_\theta^K(f_\theta^{K-1}(\cdots f_\theta^1(T)\cdots))\,, \tag{2.4}$$

[3]The number of unique N-grams grows exponentially with the size of vocabulary, leading to sparse data and making it difficult to estimate probability and results in poor generalization and limited coverage of rare or unseen n-grams.

where the objective of learning a good neural language model (i.e., good function) is to learn the underlying distribution of words or sequences of words in a given language corpus. This enables the model to learn meaningful representations of words that can be effectively utilized for various NLP tasks.

Neural language models have transformed language processing through the utilization of non-linear activation functions and layered architectures. Activation functions play a significant role in introducing non-linearity within these models, enabling the approximation of complex functions and facilitating the handling of language's inherent non-linear nature, including syntax, semantics, and contextual dependencies. Additionally, the layered architecture of neural language models contributes to their non-linear behavior as each layer applies non-linear transformations to input data, enabling the acquisition of hierarchical representations and the capturing of increasingly intricate patterns as information progresses through the layers. Neural networks have been widely used for language modeling due to their ability to capture complex patterns and dependencies in the data, which alleviates the major limitations of N-gram models:

(a) Capture semantic information: neural language models use distributed representations for words (i.e., word embeddings). These representations capture semantic relationships between words, meaning that similar words are closer together in the embedding space. As a result, neural language models can encode higher similarity scores for relevant words and better capture semantic information.

(b) Avoid data scarcity problem: the number of unique N-grams grows exponentially with the size of the vocabularies. This leads to the data scarcity problem, as many N-grams may have low or zero occurrences in the training data, making it challenging to estimate their probabilities accurately (Bengio et al., 2000).[4] Neural language models address the data scarcity problem by learning dense representations of words that generalize well across different contexts. Instead of relying on exact matches of N-grams, neural language models can learn from similar contexts, even if specific combinations of words are rare in the training data. This allows neural language models to better estimate probabilities for unseen word combinations.

(c) Mitigate local dependency issue: N-gram models have a fixed context window size, which restricts the model's ability to capture long-range dependencies between words. Neural language models can capture long-range dependencies through advanced mechanisms (e.g., gating mechanism (Hochreiter and Schmidhuber, 1997), self-attention mechanism (Vaswani et al., 2017)), taking into account information from all context words in the sequence, enabling the model to capture long-range dependencies effectively.

[4]While N-gram models demonstrate simplicity and effectiveness, incorporating longer contexts often results in significant data scarcity challenges. Consequently, context length is typically restricted to 3 (i.e., trigram) or 4, leading to the neglect of any valuable information beyond this limited scope.

In this work, we focus on two major types of neural networks architectures applied to neural language models: *feed-forward*, and *Transformer-based* (i.e., combine feed-forward and self-attention mechanism). Feed-forward neural networks are foundational models that process data in a forward direction, while Transformer-based models combine feed-forward architecture with a self-attention mechanism to effectively capture dependencies in sequential data. Neural language models utilizing feed-forward neural networks are discussed in § 2.1.2 and the Transformer using feed-forward and self-attention mechanism to capture word relationships in arbitrary longer contexts will follow in § 2.1.3.

2.1.2 Emergence of Word Embeddings

Neural language models have been introduced to address the limitations posed by N-gram models, such as their inability to capture semantic information, handle data scarcity issues, and manage local dependencies. One significant advancement is the introduction of learning dense word representations with a feed-forward neural network.

Static word embeddings depict words as low-dimensional, real-valued vectors. This method operates based on the distributional hypothesis (Harris, 1954), which asserts that word meaning can be inferred from the distribution of surrounding context. This hypothesis posits that words appearing in similar contexts share semantic meanings, thus enabling the capture of semantic similarity between word pairs (i.e., similar words will exhibit comparable contexts and possess similar dense vector representations).

The notion of static word embeddings as dense vectors (or distributed representations) was originally coined by Bengio et al. (2000), who trained the word embeddings as feature vectors alongside the parameters of a neural language model (see Figure 2.1). Building on this work, Collobert and Weston (2008) demonstrated the effectiveness of pre-trained word embeddings as a powerful tool applied in various NLP downstream tasks.[5] Besides, a neural network architecture introduced in their paper has become the prototype of the current research paper of training the neural-based word embeddings.

However, it was Mikolov et al. (2013) who brought the static word embeddings to great attention with the development of the Word2Vec model. The architecture presented two neural-based approaches with a shallow feed-forward neural network: *Skip-Gram (SG)* and *Continuous Bag-of-Words (CBOW)*. Algorithmically, both methods are analogous, with the distinction that *CBOW* predicts target words based on source context words, while *SG* performs the inverse by predicting source context words from the target words. Although this inversion seems like an arbitrary choice, statistically, *CBOW* has the effect of smoothing over a significant amount of distributional information by treating an

[5] Although the number of parameters in Bengio's model scales linearly with the input window and vocabulary size, computing output probabilities is much more computationally intensive than in n-gram models, due to the need to compute activations for all vocabulary words. Collobert and Weston mitigated this by using a pairwise ranking criterion instead of softmax, reducing computational demands and enabling training on larger datasets.

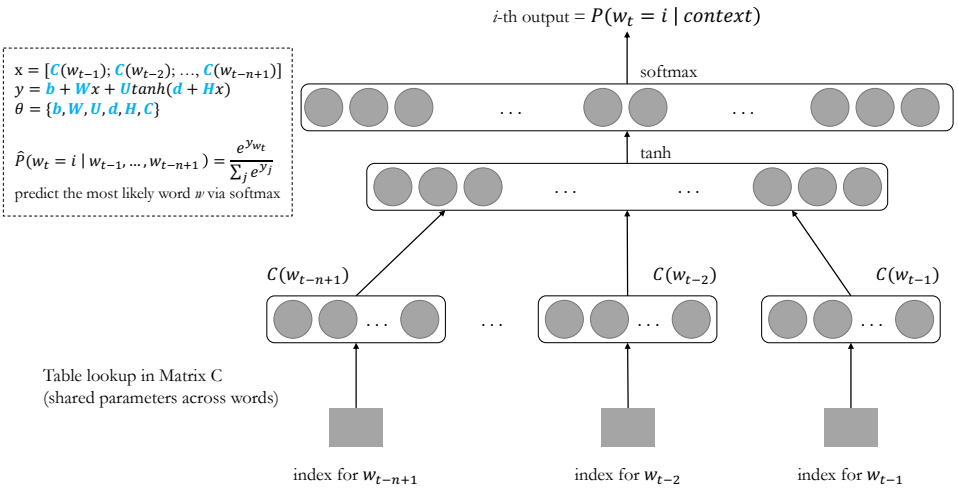


Figure 2.1: Overview of the neural probabilistic language model architecture introduced by Bengio et al. (2000). The neural network model is developed to jointly (i) learn the dense representations of each word, and (ii) estimate the probabilities for word sequences. The model comprises an input layer with dense word representations, a hidden layer with *tanh* activation, and an output softmax layer to predict the probabilities of words in the vocabulary given the context of previous $(n-1)$ words. With vocabulary size $|V|$, the parameters of the model are the word feature matrix $C \in \mathbb{R}^{|V| \times m}$, the word features to output weights $W \in \mathbb{R}^{|V| \times (n-1)m}$, the output biases b (with $|V|$ elements), the hidden layer biases d (with h elements), the hidden-to-output weights $U \in \mathbb{R}^{|V| \times h}$, and the hidden layer weight matrix $H \in \mathbb{R}^{h \times (n-1)m}$, where h is the number of hidden units, and m is the dimension of dense representation of each word.

entire context as a single observation. This smoothing turns out useful for learning better representation of frequent words. On the other hand, *SG* treats each context-target pair as a distinct observation, resulting in better representations for infrequent words. To accelerate the training process, Mikolov et al. (2013) proposed an approach called *negative sampling*, implemented specifically for *SG* method. Rather than predicting the context word from the entire vocabulary in the training corpus, a small number of suitable training instances are considered. In these selected instances, one word is the actual context word, while the others are randomly chosen from the vocabulary. Despite the efficiency, Word2Vec is tied to a fixed vocabulary size and cannot handle OOV words effectively.

Subword-based embeddings (e.g., *FastText*, *BPE*, *WPA*, *SPA*) tackle the OOV problem by breaking words into multiple subword tokens, which can be combined to represent the original word. *FastText* (Bojanowski et al., 2017) enriches the word representations by treating each word as a collection of fixed-length N-grams (subword units). The word representation is obtained by summing the representations of its N-grams representations trained with a *SG* model, which particularly benefits from handling OOV words effectively. Alternatives such as Byte-Pair Encoding (BPE) (Gage, 1994; Sennrich et al., 2016), Word-Piece Algorithm (WPA) (Schuster and Nakajima, 2012), and Sentence Piece Algorithm (SPA) (Kudo and Richardson, 2018), advance subword-based embeddings by introducing variable-length subwords (compared to fixed character N-grams). BPE starts with the base vocabulary of individual characters and iteratively merges the most frequent pair of tokens to form subwords, which is widely used in generative PLMs such as GPT-2 (Radford et al., 2019). Similarly, WPA starts with individual characters and iteratively merges them into subwords that maximize the likelihood of the entire training data, differing from BPE by ensuring the merging process optimally improves the likelihood of the training data, which is used in models like BERT (Devlin et al., 2019). Both BPE and WPA focus on word-level decomposition and are well-suited for languages with clear word boundaries. SPA instead offers a more universal solution by tokenizing raw and unsegmented text, which is effective for languages without clear word boundaries (e.g., Chinese). SPA is highly versatile and language agnostic, making it especially suitable for multilingual or script-diverse contexts. Subword-based embeddings offer: (i) Enhanced word representation at different levels of granularity, capturing morphological structure (e.g., prefixes, suffixes) and handling OOV words effectively. (ii) Improved efficiency through adaptable subword units, resulting in a more compact and expressive vocabulary that captures the most frequent and informative subword patterns.

Despite the success, static word embeddings encounter three principal issues and are further addressed by the Transformer (Vaswani et al., 2017) utilizing contextualized embeddings in § 2.1.3:

(a) Lack of contextual information: consider words in isolation and do not take into account the surrounding context. As a result, they fail to capture the full range of contextual dependencies and nuances present in natural language.

(b) Difficulty in handling polysemy and homonymy: words with multiple meanings (polysemy) or words that share the same form but have different meanings (homonymy) pose challenges for static word embeddings. These models assign a single embedding to each word, regardless of its context or meaning, resulting in ambiguous representations.

(c) Constraint of dynamic adaptation: Static word embeddings are not designed to adapt to new information or changes in language usage over time. They are fixed and unable to learn from new data without retraining the entire embedding model, hindering their ability to capture evolving language patterns and trends.

We further discuss their successors of utilizing *contextualized embeddings*: the Transformer (Vaswani et al., 2017).

2.1.3 Evolution of Transformer

The Transformer, introduced by Vaswani et al. (2017), revolutionized NLP by addressing (i) the limitations of static word embeddings (see § 2.1.2), and (ii) the long-range dependency issue from recurrent neural networks (RNNs; Hopfield (1982); Elman (1990)).[6] An overview of the Transformer architecture is illustrated in Figure 2.2. The Transformer architecture introduces *self-attention mechanisms* to simultaneously capture dependencies across the entire input sequence. This parallel processing approach enables the Transformer to overcome the limitations of RNNs and their variants, significantly reducing training time while effectively capturing long-range contextual information and outperform in various NLP tasks.

The Transformer architecture consists of two main components: *encoder* and *decoder*. The encoder in the Transformer processes the input sequence, applying self-attention mechanisms and feed-forward networks to capture dependencies and encode the information effectively. The encoded representations capture the contextual information of the input sequence and serve as the input for the decoder. The decoder then utilizes the encoded information to generate the output sequence. In this Section, we would introduce the Transformer architecture focusing on the overview of embeddings, multi-head self-attention mechanism and encoder-decoder architecture.

Embeddings. In the Transformer architecture, two primary types of embeddings are used: *token embeddings* and *positional encodings*. *Token embeddings* represent individual tokens or words in the input sequence. Each token is initially represented as a dense vector, capturing its semantic and syntactic properties (here: input and output embedding). These token embeddings are learned during the training process and can be fine-tuned to improve the model's performance on specific tasks. Vaswani et al. (2017) utilized subword-based token embeddings by employing BPE (Gage, 1994; Sennrich et al., 2016). This allows

[6] Recurrent Neural Networks (RNNs) are tailored for sequential data processing, maintaining a hidden state to retain information from prior time steps. They iterate over input sequences, updating their hidden state based on the current input and previous states. However, RNNs are hindered by the vanishing gradient problem, where gradients become increasingly small as they propagate back through time during training. This problem limits their ability to capture long-range dependencies in sequences. Long Short-Term Memory Networks (LSTMs) (Hochreiter and Schmidhuber, 1997) is a variant of RNNs developed to mitigate the gradient vanishing problem with memory cells and gating mechanisms to manage information flow. Despite their advancements, LSTMs still struggle with sequential processing and can be computationally intensive.

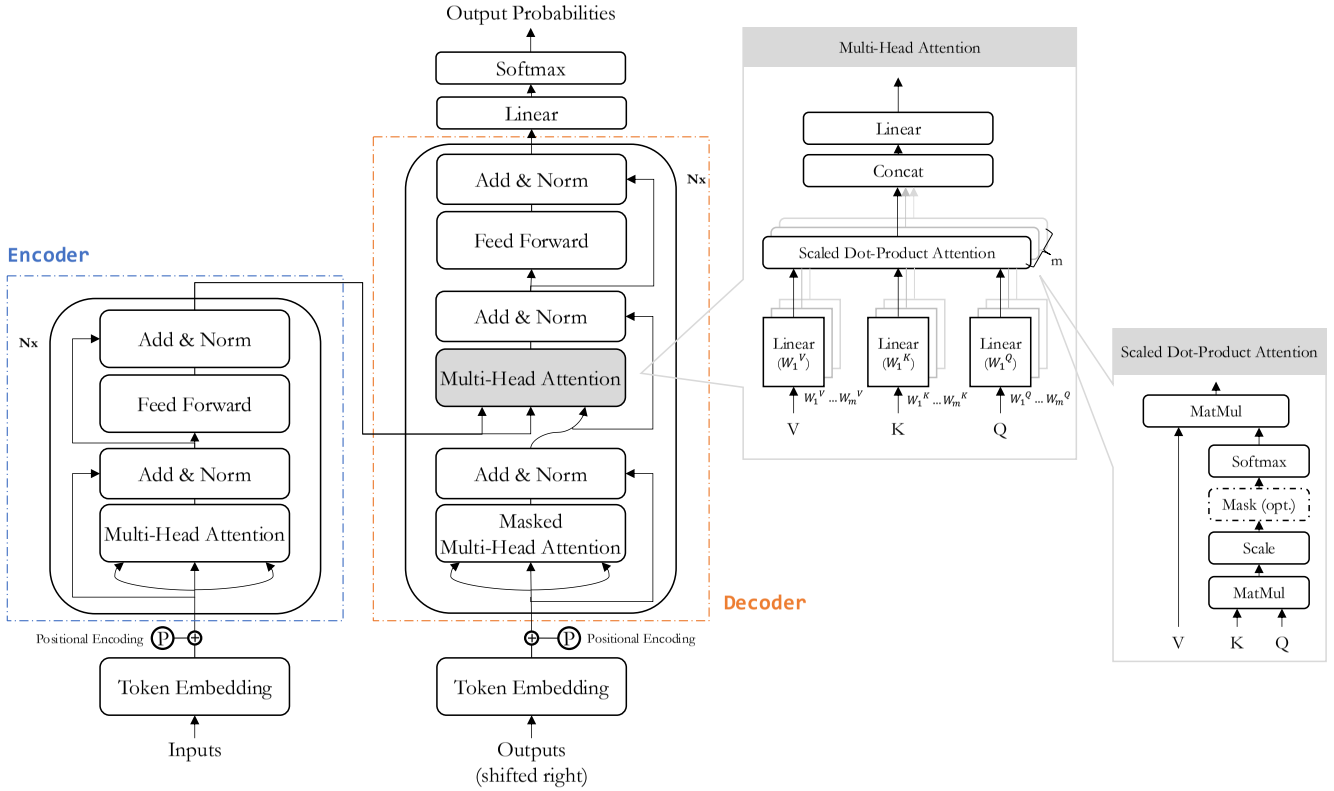


Figure 2.2: Overview of the Transformer architecture introduced by Vaswani et al. (2017). Encoder (*left*) and Decoder (*middle*) are illustrated where each consists of $N = 6$ layers. Multi-Head Attention (*right*) consists of several attention layers running in parallel, enabling the model to capture various aspects of the input by jointly attending information from different representation subspaces at different positions.

the model to effectively handle OOV words by tokenizing them into subword units. Since the Transformer lacks explicit sequential information like recurrent models (Hopfield, 1982; Elman, 1990), positional encoding is added to the token embedding, enabling the model to incorporate positional information of each token in the sequence. By combining the token embedding with the positional encoding, the Transformer model can effectively capture both the semantic properties of individual tokens and their relative positions in the sequence.

Multi-Head Self-Attention Mechanism. To enable the model to determine the importance of tokens within an input sequence based on their contextual relevance, the self-attention mechanism is incorporated into the Transformer architecture. The term "self-attention" refers to the fact that attention scores are computed within the same sequence. This mechanism allows the model to attend to different positions or tokens within the sequence and learn their dependencies. The self-attention mechanism utilizes scaled dot-product attention, and involves the creation of three attention matrices: query matrix Q, key matrix K, and value matrix V. These matrices are formed by an input matrix X, where each row represents an input token representation x_i (i.e., the combination of its token embedding and positional encoding). By multiplying the input matrix with three trainable weight matrices (W^Q, W^K, W^V), three attention matrices are formed: query matrix Q, key matrix K, and value matrix V:

$$Q = XW^Q; \quad K = XW^K; \quad V = XW^V\,, \tag{2.5}$$

where each row of each weight matrix corresponds to the query vector q_i, key vector k_i, and value vector v_i for an input token representation x_i.

The self-attention mechanism utilizes the query, key, and value matrix (Q, K, and V) to compute attention scores. The attention scores determine the importance or relevance of each token in the input sequence in relation to other tokens. To calculate the attention scores, the dot product is computed between the query matrix Q and key matrix K, scaled by a factor $\sqrt{d_k}$ (d_k is the dimension of query and key vector) to ensure stable gradients, and passed through a *softmax* operation to normalize the scores (i.e., making all values positive and summing up to 1). To weight the value vectors, the value matrix V is then multiplied by the attention scores. This approach allows the model to focus on preserving the values of the relevant words while minimizing the influence of irrelevant words. The resulting weighted value matrix serves as the final attention output (Equation 2.6), further processed by subsequent layers in the Transformer architecture:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{2.6}$$

The self-attention mechanism allows the model to attend to different parts of the input sequence and capture the long-range dependencies and relationships between tokens,

contributing to its ability to learn contextual representations effectively. Multi-head self-attention is an extension of the self-attention mechanism: instead of relying on a single attention head (i.e., a single set of W^Q, W^K, W^V), it utilizes multiple attention heads to capture different types of dependencies and attend to multiple aspects of the input sequence simultaneously. For example, one attention head might learn information between subject-verb relationships, while the other might focus on capturing long-range dependencies between tokens. In multi-head self-attention, the input token representation is multiplied by the corresponding weight matrices to obtain the query, key, and value vectors specific to each attention head. The number of attention heads m is a hyperparameter that determines the capacity of the model to capture different patterns and dependencies within the sequence.[7]

$$\begin{aligned} \text{MultiHeadAttention}(Q, K, V) &= \text{Concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_m)W^O \\ \text{where head}_i &= \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \tag{2.7}$$

To enhance training efficiency, the attention computations for each head are performed in parallel, and the results are concatenated and projected using an additional weight matrix W^O to form the final output. Specifically, for each set of attention matrices of each head are: $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$. After concatenation, the projected weight matrix is: $W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$. According to Vaswani et al. (2017), the Transformer model employed $h = 8$ heads and $d_{\text{model}} = 512$ embedding dimensions, which gives $d_k = d_v = d_{\text{model}}/h = 64$. The reduced dimension of each head allows for efficient parallelization while maintaining the computational cost similar to that of single-head self-attention with full dimensionality.

Encoder-Decoder Architecture. The Transformer architecture was originally introduced as a sequence-to-sequence model specifically for machine translation (MT) tasks. This architecture utilizes a multi-layer encoder-decoder framework as depicted in Figure 2.2. *Encoder* consists of N identical layers, each including a *multi-head self-attention* and *feed-forward network* sub-layer. A residual connection (He et al., 2016) is employed around each of the two sub-layers to allow the model to retain and propagate useful information from the earlier layer to the subsequent layer, avoiding the gradient vanishing problem.[8] The residual connection is then followed by layer normalization (Ba et al.,

[7] Increasing the number of attention heads allows for more diverse and fine-grained attention patterns to be learned. However, it also increases the computational complexity of the model.

[8] Deep neural networks often suffer from the vanishing gradient problem, where gradients become extremely small as they propagate through many layers, eventually vanishing to zero. When the gradients vanish, it becomes challenging for the network to update the weights of earlier layers effectively. This leads to slower convergence and can prevent the network from learning complex patterns and dependencies in the data. Residual connections allow the gradients to flow more easily through the network by providing shortcuts from earlier layers to later layers. This helps to mitigate the vanishing gradient problem and facilitates better optimization during training.

2016) to mitigate the internal covariate shift problem.[9] *Decoder* has similar architecture as *encoder* consisting of N identical layers, except it incorporates the encoder output with a second multi-head self-attention module. Besides, the decoder utilizes *masked* multi-head self-attention while preventing the inputs from attending to future positions. This is crucial to ensure that the model only attends to information that is available at the current decoding step to prevent reverse information flow.

The last layer of the Transformer is typically one or more linear layers followed by a softmax function. The number of linear layers depends on the target tasks, which learn representations into the desired output format for the task at hand. By employing this final layer, the Transformer can make predictions or generate text sequences by mapping its learned representations to class labels or specific tokens from a fixed vocabulary, facilitating various NLP tasks such as sentiment analysis and machine translation.

The advancement of the Transformer architecture enables the model to effectively encode the contextual information in the input sequence, capture long-range dependencies, and process sequences through efficient parallel training. The modularity and adaptability of Transformer architecture have inspired the development of language models tailored to specific tasks, where subsequent research has shown that using decoder-only layers for auto-regressive models (Radford et al., 2018) or encoder-only layers for bidirectional auto-encoding models (Devlin et al., 2019; Liu et al., 2019c) can be sufficient. Decoder-only model excels in text generation tasks due to its autoregressive nature, while encoder-only model is highly effective for NLU tasks because of its bidirectional contextual representation (Wu et al., 2020).[10] These Pre-trained Language Models (PLMs) utilizing the Transformer architecture have led to significant improvements in various NLP tasks, making the Transformer a highly influential model architecture in the NLP field. In the following Section, we will delve exclusively into the advancements of encoder-based pre-trained Transformer-based language models. Due to the focus on NLU tasks, we would rely on the strengths of encoder-only models (Devlin et al., 2019; Liu et al., 2019c), which leverage bidirectional learning for better capturing and handling NLU tasks by understanding the nuances and intricacies within text, and the effective use of token representations (i.e., Embeddings layer) (Reimers and Gurevych, 2019; Su et al., 2023). Specifically, the exploration would cater the PLMs to various domains and languages usage (Wu et al., 2020; Conneau et al., 2020a) (§ 2.2). We hope to provide a comprehensive understanding of how language models can continue to evolve to meet the growing demands of domain- and language-related applications.

[9] In deep neural networks, the distribution of inputs to each layer can change during training, making it difficult for subsequent layers to learn effectively. Layer normalization normalizes the inputs to each layer, making the network more robust to these distribution shifts and ensuring that the inputs of each layer are centered and have unit variance. This stabilizes the training process and allows for faster convergence.

[10] NLU tasks have also been cast as text generation setup (Brown et al., 2020), highlighting the versatility of decoder-only models. However, encoder-only models remain highly effective for NLU tasks because of their bidirectional contextual representation (Wu et al., 2020).

2.2 Pre-trained Language Models

Building upon the success of the Transformer, Pre-trained Language Models (PLMs) utilizing Transformer-based architecture (Radford et al., 2018; Devlin et al., 2019; Liu et al., 2019c) has emerged as a transformative breakthrough in the field of NLP.[11] These PLMs typically undergo a two-stage training process. In the first stage, known as *pre-training*, the model is trained on a massive (un)labeled text corpora, which helps to grasp the statistical patterns and contextual representation of words presented in the data. The second stage, known as *fine-tuning*, involves training the pre-trained model further on specific downstream tasks. This stage typically requires labeled data and task-specific annotations. By fine-tuning the pre-trained model on specific downstream tasks (e.g., sentiment analysis, named entity recognition), the model's parameters are adjusted to better align with the specific task's objective, enabling it to make accurate predictions or generate desired outputs. PLMs achieve state-of-the-art performance, making them invaluable assets for a wide range of NLP applications. We would introduce two prominent encoder-based PLMs: BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019c), utilized in our following Chapters and explore their advanced domain and multilingual pre-training variations. The overall comparisons are shown in Table 2.1.

2.2.1 General-Purpose Pre-training

General-purpose pre-training refers to the training of a language model on a large corpus of unlabeled text data from heterogeneous sources. The objective is to develop a deep understanding of language and capture general language representations. These pre-trained models are designed to be versatile and perform well across a wide range of NLP tasks. Here we introduce two widely-used variations: BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019c).

Bidirectional Encoder Representations from Transformers (BERT). BERT proposed by Devlin et al. (2019) is a Pre-trained Language Model (PLM) that builds upon the Transformer architecture. While the Transformer architecture consists of both encoder and decoder components, BERT focuses primarily on the encoder part.[12] For the *pre-training* stage, BERT employs two objectives for self-supervised training as shown in Figure 2.3: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM, referred to as *Cloze* task (Taylor, 1953), is a token-level prediction task that involves masking certain tokens in each training example and training the model to predict the masked tokens based on the surrounding context. MLM enables the model to grasp bidi-

[11] For brevity, in the following Sections and Chapters, we will use pre-trained language models (PLMs) to refer to all notions of pre-trained Transformer-based language models.

[12] BERT is originally designed for natural language understanding (NLU) tasks, aiming to learn contextualized word representations.

Model	**BERT** (Devlin et al., 2019)	**mBERT** (Devlin et al., 2019)	**TOD-BERT** (Wu et al., 2020)
Purpose	General	Multilingual	Domain
Tokenization	WPA	WPA	WPA
# Params	109 M (CASED) / 110 M (UNCASED)	179 M	110 M
# Vocabs ‡	28996 (CASED) / 30522 (UNCASED)	119547	30524 (UNCASED)
Objective	MLM^S + NSP	MLM^S + NSP	MLM^D + RCL
Training Data	16 GB	-	1 GB
# Languages	1 (EN)	104	1 (EN)

Model	**RoBERTa** (Liu et al., 2019c)	**XLM-R** (Conneau et al., 2020a)	**TOD-XLMR** (Hung et al., 2022b)
Purpose	General	Multilingual	Domain & Multilingual
Tokenization	bBPE	SPA	SPA
# Params	125 M	279 M	279 M
# Vocabs ‡	50265	250002	250004
Objective	MLM^D	MLM^D	MLM^D + RCL
Training Data	160 GB	2.5 TB	1 GB
# Languages	1 (EN)	100	100

Table 2.1: Overview of encoder-based Transformer models. The listed models for comparison are utilized in the following Chapters. We consider only the BASE architecture with 12 layers for each model. It is noted that there might be case-sensitive (CASED) or case-insensitive (UNCASED) variations. MLM^S refers to static masked tokens and MLM^D refers to dynamic masking of MLM. Tokenization methods are discussed in § 2.1.2. ‡Unless explicitly specified, the vocabulary size corresponds to the CASED model.

rectional relationships optimized on the token level. NSP aims to predict whether two sentences are consecutive in the original text or randomly sampled from different sources. This teaches them to comprehend the sentence relationship and grasp the coherence and context transitions within the sentence level. To represent the input, BERT utilizes Word Piece Algorithm (WPA) (Schuster and Nakajima, 2012) for breaking the words into subword tokens, allowing effective handling of known and unknown words. BERT incorporates special tokens like the separator token ([SEP]), classification token ([CLS]), and padding token ([PAD]). The [SEP] token indicates different parts of the input sequence, while the [CLS] token represents the entire input sequence for classification tasks, and [PAD] token is used to fill the remaining positions in the input sequence of varying lengths. These special tokens, along with tokens from vocabulary, are embedded as token embeddings and combined with position and segment embeddings to form the first layer of the model. The original BERT model is pre-trained on BookCorpus (Zhu et al., 2015) and English Wikipedia. The pre-trained BERT model can be used to generate contextualized input representations as features (Devlin et al., 2019; Reimers and Gurevych, 2019) or can be directly fine-tuned for downstream tasks (Devlin et al., 2019).

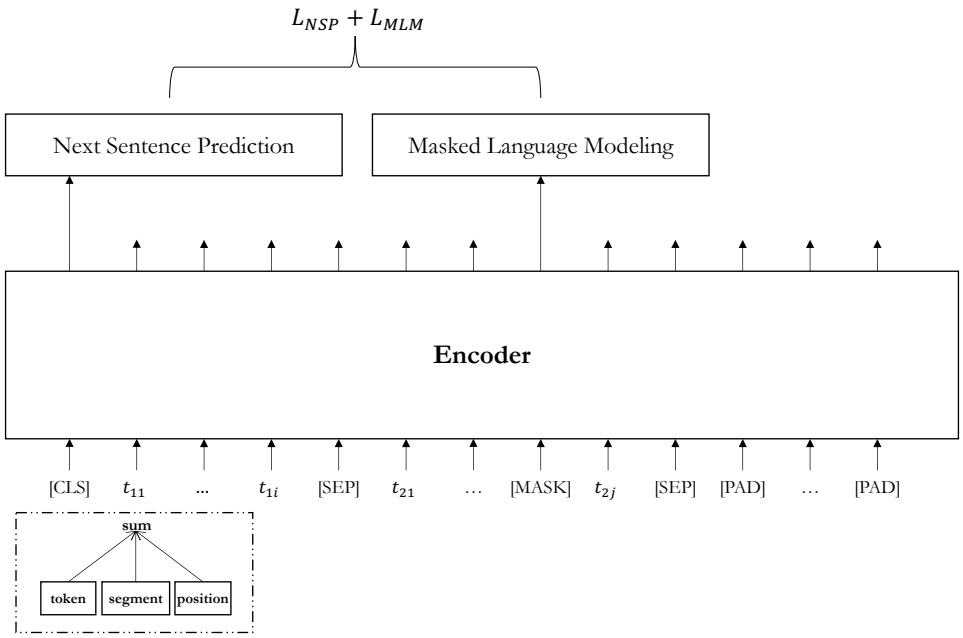


Figure 2.3: Overview of BERT pre-training architecture. BERT employs two objectives for self-supervised training: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP).

Robustly Optimized BERT Pre-Training Approach (RoBERTa). RoBERTa is a variant of the BERT model that was introduced by Liu et al. (2019c). While RoBERTa shares many similarities with BERT, there are a few key differences in their training procedures. One major difference is in the training data. RoBERTa is trained on a significantly larger corpus that includes original BERT training data and additional publicly available text from the internet (BERT: 16 GB vs. RoBERTa: 160 GB), which allows the model to capture more diverse and comprehensive language patterns. RoBERTa eliminates the NSP task, while introducing dynamic masking (MLM^{D}) so that a new masking pattern is generated each time a sentence is fed into training, whereas BERT uses static masked tokens (MLM^{S}) during training. This dynamic masking approach allows RoBERTa to benefit from more training steps and effectively utilize more data during pre-training. It also applies various optimization techniques, such as training with larger batches, and using byte-level Byte-Pair Encoding (bBPE) (Radford et al., 2019) to efficiently handle different character encodings. Overall, these modifications in the training process of RoBERTa lead to improved performance compared to BERT model, demonstrating the effectiveness of the optimized pre-training approach with larger training data.

2.2.2 Multilingual Pre-training

Multilingual pre-training focuses on training a model on a diverse multilingual corpus to efficiently comprehend and process multiple languages. This approach acknowledges the importance of language representations capable of accommodating the nuances and complexities of different languages while utilizing a shared set of subword units during pre-training. The models vary from bilingual (Kim et al., 2019) to multilingual (Conneau and Lample, 2019; Conneau et al., 2020a), aiming to develop language representations that can handle a multitude of languages effectively. This enables the pre-trained language model to perform effectively on various multilingual tasks, showcasing its versatility and adaptability across different languages (Pires et al., 2019). Here we introduce two prominent multilingual encoder-based PLMs: mBERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020a), utilized in the subsequent Chapters.

Multilingual BERT (mBERT). mBERT is a multilingual variant of BERT, trained on a multilingual corpus of 100+ languages sourced from Wikipedia.[13] It has the ability to handle multiple languages effectively and has been extensively used for various multilingual NLP tasks (Wu and Dredze, 2019; Kassner et al., 2021). Identical to BERT, mBERT pre-training involves two training objectives (MLM, NSP) and the Word Piece Algorithm (WPA) (Schuster and Nakajima, 2012) which utilizes the shared subword units across different languages, enabling consistent tokenization and facilitating cross-lingual transfer of learned representations (Karthikeyan et al., 2020).

Unsupervised Cross-Lingual Representation Learning at Scale (XLM-R). XLM-RoBERTa (XLM-R) (Conneau et al., 2020a) is an enhanced version of XLM (Cross-lingual Language Model Pre-training; Conneau and Lample (2019)) and RoBERTa (Liu et al., 2019c). Conneau and Lample (2019) initially proposed XLM, utilizing Translation Language Modeling (TLM) to leverage parallel data from Wikipedia for cross-lingual pre-training. XLM-R follows similar training setups as XLM, while incorporating the following distinct elements: (i) Expand training data: XLM-R is instead trained on a vast multilingual unlabeled corpus (2.5 TB) of 100 languages sourced from CCNet (Wenzek et al., 2020). Besides, the amount of data for low-resource languages is increased by two orders of magnitude on average. (ii) Avoid parallel training data: XLM leverages parallel data for cross-lingual supervision, while XLM-R is trained with dynamic masking (MLM^{D}) training objective (i.e., similar to RoBERTa, without NSP) on a large scale multilingual dataset without utilizing parallel data for cross-lingual supervision. (iii) Leverage Sentence Piece Algorithm (SPA) for tokenization: instead of BPE (Gage, 1994; Sennrich et al., 2016), XLM-R employs SPA (Kudo and Richardson, 2018), which allows for more fine-grained tokenization, making it particularly advantageous for languages

[13] https://github.com/google-research/bert/blob/master/multilingual.md

lacking clear word boundaries. Compared to mBERT, XLM-R encodes a larger vocabulary size (mBERT: 120 K vs. XLM-R: 250K), resulting in capturing a more extensive range of subword units and linguistic patterns across languages.

2.2.3 Domain-Specific Pre-training

Domain-specific pre-training involves training a model on a diverse range of domain-specific corpus. The objective is to develop domain-specialized language representations. This process allows the language model to adapt and learn domain-specific features, vocabularies, and patterns that are relevant to the target domain(s). Domain-specialized language models offer more effective and efficient solutions for domain-specific tasks and applications. They can provide more accurate predictions, better insights, and improved performance compared to general-purpose pre-training models that lack domain-specific knowledge. For the purpose of conversational knowledge, two domain-purpose pre-training models are utilized in the following Chapters: TOD-BERT (Wu et al., 2020) and TOD-XLMR (Hung et al., 2022b).

Task-Oriented Dialog BERT (TOD-BERT). TOD-BERT (Wu et al., 2020) is a domain-specialized variant of BERT model. It is designed for English task-oriented dialog (TOD) systems, aiming to assist users in completing specific tasks (e.g., booking a flight or making a restaurant reservation), through conversational interactions. To better model dialog behavior during pre-training, TOD-BERT continually trains on general-purpose BERT model on nine English TOD datasets across 60+ subdomains to capture conversational TOD structure. During pre-training, two objectives are jointly trained: (i) MLM: similar to RoBERTa with dynamic masking, and (ii) RCL (Response Contrastive Loss): a newly proposed method utilizing in-batch contrastive learning to capture sequential order, structural information, and response similarity within dialogs. TOD-BERT outperforms BERT on four TOD downstream tasks, and shows a stronger few-shot ability to mitigate the data scarcity problem for TOD.

Task-Oriented Dialog XLM-R (TOD-XLMR). TOD-XLMR, proposed by Hung et al. (2022b), is a domain-specialized variant of XLM-R model, which will be demonstrated in Chapter 4. Unlike TOD-BERT, tailored for English-only TOD systems, TOD-XLMR is specifically designed to cater needs of multilingual TOD systems. To encode conversational structure to a multilingual pre-trained language model, TOD-XLMR is trained in the same manner as TOD-BERT: employing two training objectives (MLM, RCL) on nine English TOD datasets to cater conversational structure in XLM-R. Since TOD-XLMR has been conversationally specialized only in English data, it is shown to be beneficial to further language-specific training through transfer learning approaches.

2.3 Enhancing Transfer Learning through Adaptive Pre-training

The field of NLP has experienced remarkable progress with the advent of PLMs (see § 2.1.3). These models typically employ a *sequential transfer learning* paradigm, involving a two-stage training process: *pre-training* then *fine-tuning*. In the *pre-training* phase, the models are pre-trained on massive corpora to capture linguistic patterns and semantic representations. This initial training phase equips the models with a broad understanding of language(s). Subsequently, they are *fine-tuned* on downstream tasks, typically smaller task-specific datasets compared to the pre-trained data. The sequential transfer learning approach allows the general language knowledge gained during pre-training to be effectively transferred and adapted, refining the model's capabilities to meet the specific requirements of the target tasks through further fine-tuning. However, *sequential transfer learning* has certain limitations. While PLMs capture a broad range of language features, they may not possess domain-specific, language-specific, or demographic-aware knowledge necessary for optimal performance on downstream tasks. Additionally, *catastrophic forgetting*, where fine-tuning disrupts previously learned representations, can hinder the transfer of knowledge across tasks (McCloskey and Cohen, 1989; French, 1999).

In the following Sections, we will delve into the fundamentals of transfer learning, specifically focusing on the *sequential transfer learning* paradigm. We will then discuss the limitations associated with sequential transfer learning and introduce a more recent approach known as *adaptive pre-training* (Gururangan et al., 2020). The adaptive pre-training approach aims to enhance knowledge transfer through a three-stage training paradigm: *pre-training*, *adaptation*, then *fine-tuning*, utilizing pre-trained language models. This approach effectively overcomes the limitations of sequential transfer learning by incorporating an *adaptation* stage between pre-training and fine-tuning. It offers a more efficient mechanism for acquiring specialized knowledge while retaining previously acquired information during the pre-training phase.

2.3.1 Transfer Learning and Adaptive Pre-training

Transfer learning in NLP encompasses the notions of a *source* and a *target*, with the objective of extracting knowledge from a source setting and applying it to a distinct target setting (Pan and Yang, 2010). The *source* refers to the dataset that serves as the knowledge source for transferring information. It can be a large-scale general dataset, a dataset specific to a particular subdomain(s), or a collection of language(s) corpora. On the other hand, the *target* refers to any NLP tasks (e.g., sentiment analysis, named entity recognition). The fundamental principle of transfer learning in NLP entails utilizing pre-existing knowledge or representations from the source to improve the model's performance on the target task. By effectively transferring knowledge from the source to the target, models can leverage prior learning to achieve better generalization on unseen tasks related to

effectively tackling a wide range of NLP tasks.

Sequential transfer learning is a specific approach within transfer learning that involves utilizing a sequence of related tasks to enhance model performance. It aims to capture and transfer knowledge from earlier tasks to improve performance on subsequent tasks. Pre-trained Transformer-based language models (see § 2.2), like BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019c) follow the two-stage training paradigms sequentially: *pre-training*, then *fine-tuning*. In the *pre-training* stage, the language model is trained on a large corpus of text data to learn general language representations. These representations capture the relationships and patterns within the language. In the *fine-tuning* stage, the word or wordpiece (Schuster and Nakajima, 2012; Basmatkar et al., 2019) representations learned during pre-training are reused in supervised training for a downstream task, with fine-tuning (i.e., optionally updates) to adapt the model to the task-specific knowledge required for the downstream task.

Following the two-stage sequential transfer learning paradigm enables the model to benefit from the general language understanding acquired during pre-training and later adapt to the specific target task. Approaches based on sequential transfer learning have achieved state-of-the-art results on a wide range of NLP tasks. However, the two-stage sequential transfer learning paradigms face the following major challenges:

(1) *Mismatch of Domain(s) and Language(s)*: The representations learned from the pre-trained model might not be perfectly aligned with the domain, or language-specific tasks, which can limit its capacity on downstream tasks performance. This is commonly addressed as a problem of *domain shift* (Lekhtman et al., 2021; Guo and Yu, 2022), where the data distribution for pre-training is different from fine-tuning data, which can incur negative transfer in performance during fine-tuning, as the model's knowledge might not be directly applicable to the new domain(s) or language(s).

(2) *Scarcity of Task-Annotated Data*: Fine-tuning typically requires a task-specific annotated dataset. However, in many real-world scenarios, obtaining a large labeled dataset for the target task might be challenging or expensive, leading to suboptimal results due to the scarcity of task-annotated data.

To overcome these challenges and enhance the model adaptability of sequential transfer learning, an *adaptive pre-training* approach with a three-stage process is addressed (Gururangan et al., 2020; Jiang et al., 2022): *pre-training*, *adaptation*, then *fine-tuning*. The overview of the adaptive pre-training utilizing pre-trained language models is depicted in Figure 2.4. The general workflow of adaptive pre-training (i.e., a three-stage sequential transfer learning) involves the following steps:

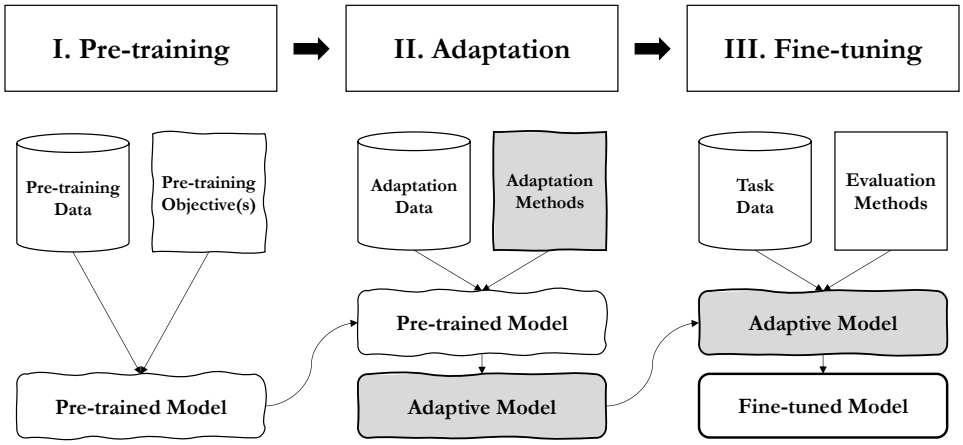


Figure 2.4: Overview of the *adaptive pre-training* framework. The framework involves a three-stage process: *Pre-training*, *Adaptation*, *Fine-tuning*.

- *Pre-training*: This stage involves training a language model on a vast amount of heterogeneous datasets (typically sourced from general web resources (Liu et al., 2019c), domains (Wu et al., 2020), languages (Conneau et al., 2020a)). Self-supervised learning objectives, such as next sentence prediction (Devlin et al., 2019) and masked language modeling (Devlin et al., 2019; Liu et al., 2019c), are commonly conducted during pre-training. The process serves as a foundation for the natural language understanding of the model, capturing the contextual representation of tokens, which are essential for subsequent tasks.
- *Adaptation*: Following pre-training, the model is further trained on a smaller corpus of narrower domain- and/or language-specific (un)labeled data. The continual learning phase enables the model to acquire specialized knowledge, tailoring it to become more relevant for the target tasks during fine-tuning. The primary focus in this stage shifts towards adapting the model's representations to align with the domains and languages associated with the downstream tasks, which mitigates the scarcity of task-annotated data, as mentioned above.
- *Fine-tuning*: Once the model has undergone adaptation with acquired knowledge, it can be fine-tuned on a labeled dataset tailored to a specific downstream task in a particular domain and/or language. The fine-tuning process involves updating the model's parameters to make it more suitable and accurate for the task at hand. This stage ensures that the model becomes specialized for the target task while leveraging the knowledge captured during pre-training and adaptation.

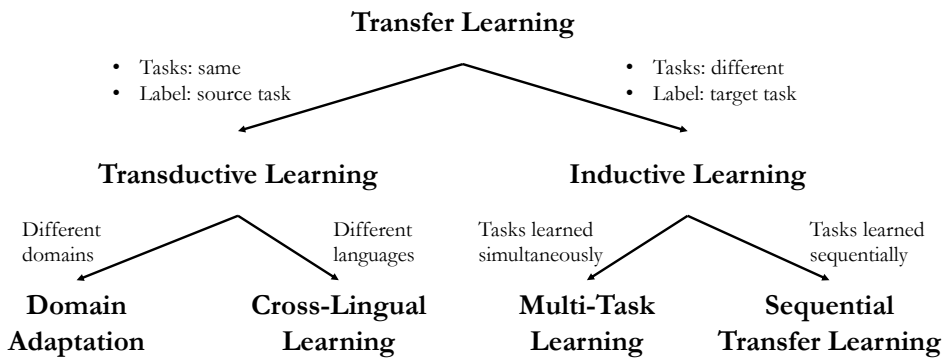


Figure 2.5: Overview of the transfer learning taxonomy proposed by Ruder (2019). The taxonomy is structured according to: (a) whether the source and target settings involve the same task; (b) the characteristics of the source and target domains; and (c) the sequence in which the tasks are acquired.

By incorporating an *adaptation* phase between pre-training and fine-tuning, the intermediary step allows the model to better align its representations to the task at hand. This adaptive pre-training approach offers a more flexible and effective way to utilize PLMs in NLP tasks.

2.3.2 Transferability of Pre-trained Language Models

Building upon the notion of transferability in language models, our work draws inspiration from the research conducted by Gururangan et al. (2020). We adopt and refine the transfer learning techniques proposed in their study, which proved to be successful in harnessing the capabilities of PLMs. This enables us to leverage the knowledge and representations learned during pre-training, conduct an adaptation step for acquiring specialized knowledge, and further fine-tune on various downstream tasks. The *adaptation* phase in the adaptive pre-training framework plays a vital role in harnessing the power of transferability for PLMs, which is also the core of adapting different transfer learning methods, offering flexible and highly effective solutions for leveraging PLMs regarding transferability.

To classify the field of transfer learning and its methods, a taxonomy was introduced by Pan and Yang (2010) and later adapted to the field of NLP by Ruder (2019). The taxonomy as illustrated in Figure 2.5 comprises the following categories: transductive learning (domain adaptation, cross-lingual transfer) and inductive learning (multi-task learning, sequential transfer learning). These categories provide a rough classification based on the similarity between source and target tasks, the nature of the domains involved, and the order in which tasks are learned. Transductive learning focuses on adapting models across domains or languages, while inductive learning explores knowledge transfer from multiple tasks or sequentially learned representations.

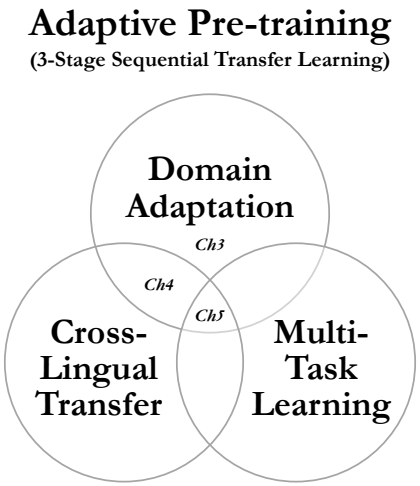


Figure 2.6: Overview of the transfer learning methods utilized in the *adaptation* phase of the adaptive pre-training framework proposed in this work. Each Chapter (Ch3, Ch4, Ch5) employs a combination of transfer learning methods. This integrative approach allows for a comprehensive adaptation strategy that enhances the model's capability to handle multidimensional adaptation and tasks effectively.

However, the proposed taxonomy has certain limitations, as it fails to consider the practical aspects of combining multiple transfer learning methods, such as domain adaptation with sequential transfer learning. Moreover, determining the degree of similarity or difference between source and target tasks can be challenging, as the categories in the taxonomy are not mutually exclusive. To address these limitations and emphasize the flexibility and adaptability of transfer learning methods in the *adaptation* phase for harnessing the power of PLMs in adaptive pre-training discussed in the subsequent Chapters, it is essential to acknowledge the interconnectedness and overlaps between different approaches. By exploring the interconnections among various transfer learning techniques, we can learn more robust and versatile representations tailored for multidimensional adaptation of NLP tasks – for domains, languages, and social dimensions in this work. In the upcoming Chapters (Chapter 3, Chapter 4, Chapter 5), we delve into the core of the adaptive pre-training framework utilizing transferability of PLMs. The transfer learning methods conducted in the adaptation phase of the adaptive pre-training framework are depicted in Figure 2.6. The Figure presents a comprehensive integration of three transfer learning methods for adaptation: domain adaptation, cross-lingual transfer, and multi-task learning. These approaches are strategically employed to augment the adaptive pre-training process and enhance its effectiveness. By incorporating these transfer learning methods, we aim to showcase the remarkable flexibility and adaptability of the adaptive pre-training framework leveraging PLMs. We would introduce the concepts of *domain adaptation*, *cross-lingual transfer*, and *multi-task learning* in the following.

Domain Adaptation. The method specifically focuses on transferring knowledge or representations from a source domain to a target domain where the distributions of data may differ. The goal is to bridge the domain gap and improve the performance of the target task in the new domain, especially when the domain-specific target tasks suffer from *data scarcity* issues. Domain adaptation techniques often involve aligning the source and target domains, either by feature transformations or by learning domain-invariant representations. Recent advancements in this field have focused on unsupervised domain adaptation, where labeled target domain data is not required for domain adaptation. Ramponi and Plank (2020) provided an overview of unsupervised domain adaptation methods categorized into: *model-centric*, *data-centric*, and *hybrid* approaches. In this work, we focus on *data-centric* methods, leveraging in-domain data for adaptive pre-training utilizing PLMs for task-oriented dialogs (Chapter 3), multilingual dialogs (Chapter 4), and demographic factors (Chapter 5).[14]

Cross-Lingual Transfer. This is a technique that facilitates the transfer of knowledge across different languages. The main objective is to align the representation spaces of text between two or more languages. This approach is especially valuable when dealing with language-specific target tasks that suffer from *data scarcity* issues (e.g., resource-lean languages). To achieve the alignment of representation spaces, recent work proposed unsupervised methods or methods that incorporate cross-lingual supervision signals, either relying on cross-lingual word embedding spaces (Ruder et al., 2019) or more recent multilingual language models (Pires et al., 2019; Lauscher et al., 2020; Conneau et al., 2020b). These approaches aim to bridge the linguistic and semantic gaps across languages, enabling effective knowledge transfer and improving performance on tasks in resource-lean languages. In the subsequent Chapters, we analyze how cross-lingual transfer helps to improve downstream task performance in both zero-shot and few-shot scenarios by utilizing multilingual pre-trained language models (i.e., mBERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020a)) in task-oriented dialogs (Chapter 4) and whether language-adaptive pre-training helps in multilingual demographic adaptation (Chapter 5).

Multi-Task Learning. In addition to domain adaptation and cross-lingual transfer, the incorporation of multi-task learning enriches the adaptive pre-training process by enabling the model to learn multiple tasks simultaneously (Caruana, 1997). Through joint training on various tasks alongside in-domain or language-specific texts, the model can effectively extract and leverage shared information across tasks. This multi-task learning

[14] It is noted that the term domain adaptation is widely used in various concepts. In the context of this thesis, to make a clear distinction, *domain-specific pre-training* (see § 2.2.3) refers to the initial training phase, where a model is trained on a domain-specific corpus encompassing various subdomains. And *domain adaptation* (or *domain-adaptive pre-training* introduced in Gururangan et al. (2020)) refers to the process of further training a general- or domain-specialized pre-trained model on specific subdomains or narrower domains.

approach facilitates the transfer of knowledge acquired from one task to improve its performance on other tasks, encouraging generalization across multiple tasks due to the simultaneous training process. Recent research has focused on conducting multi-task learning during the task fine-tuning stage to jointly learn multiple tasks at once (Guda et al., 2021), while we instead focus on the adaptation stage (Liu et al., 2019b; Aghajanyan et al., 2021), which has greater capability to incorporate (un)labeled data for adaptive pre-training while mitigating the scarcity of task annotated data. We analyze how multi-task learning impacts the model's performance concerning demographic factors (Chapter 5).

2.4 Challenges

Adaptive pre-training methods have demonstrated significant success in a wide range of downstream tasks, spanning various aspects from (i) domain-adaptive pre-training for applications such as dialog understanding (Wu et al., 2021) and geographic adaptation (Hofmann et al., 2024), (ii) language-adaptive pre-training for tasks like causal commonsense reasoning (Ponti et al., 2020) and syntactic parsing (Glavaš and Vulić, 2021), and (iii) a hybrid approach that integrates both domain- and language-adaptive pre-training, has shown promise in hate speech detection (Glavaš et al., 2020). Despite the success, there remain challenges to be addressed: *effectiveness* (**C1**), *efficiency* (**C2**), and *interpretability* (**C3**).

2.4.1 Towards Effective Adaptive Pre-training

To enhance the effectiveness of adaptive pre-training methods, three critical perspectives come to the forefront (**C1**): (1) *Task-specific versus task-agnostic approaches*, where the former requires labeled task data and is limited to a single task, while the latter aims to develop versatile models adaptable to multiple tasks without the specific task data constraint. (2) *Handling multi-domain and multilingual use cases*, becomes imperative in real-world applications, where a single model must seamlessly accommodate various domains and languages to avoid the impracticality of deploying multiple models. (3) *Addressing low-resource scenarios for task fine-tuning*, is paramount as many languages and domains lack sufficient annotated data for effective task fine-tuning. Exploring these perspectives opens the door to advancing adaptive pre-training methods and empowering language models to perform optimally across diverse settings.

Task-Specific vs. Task-Agnostic. Compared to *task-specific pre-training* methods that are limited to only single-task usage (Zeng and Nie, 2020; Liu et al., 2021c), *task-agnostic adaptive pre-training* techniques instead aim to develop a versatile model that is capable of effectively adapting to various tasks (Bhattacharjee et al., 2020). In this work, we thus focus on *task-agnostic adaptive pre-training* approaches, which can handle multidimensional use cases across multiple tasks effectively. For instance, a model specialized in the financial domain can be applied to any financial-related tasks, including named entity recognition

and sentiment analysis. To tackle the challenge, we propose task-agnostic methods for self-supervised domain adaptation (Chapter 3) and cross-lingual transfer (Chapter 4) for task-oriented dialogs, along with hybrid setups combining multiple transfer learning methods (see Figure 2.6) for demographic adaptation (Chapter 5).

Multi-Domain and Multilingual Use Cases. The necessity of handling multi-domain and multilingual scenarios with a single model arises from the increasing complexity and diversity of real-world data. In practical applications, data often comes from various sources with diverse domains and languages, each with its own unique characteristics and patterns. For instance, a language model may need to process text data from fields as diverse as healthcare, finance, and technology, and cater to a wide range of language-speaking users. Using separate models for each domain and language would not only be resource-intensive but also impractical to manage, update, and maintain. Therefore, a single model capable of handling multi-domain and multilingual use cases becomes essential. For multi-domain scenarios, we propose two ways of utilizing the single-domain adaptive methods and aggregating in the task fine-tuning stage by (i) employing *adapters* (Houlsby et al., 2019; Pfeiffer et al., 2020, 2021; Parović et al., 2022) for multi-domain task-oriented dialog systems (see § 3.1), and (ii) introducing a novel method considering meta-embeddings and meta-tokenizers (see § 3.2). For multilingual scenarios, we utilize augmented in-domain and language-specific texts with the proposed multi-task learning objectives, to encode the demographic knowledge into a multilingual PLM (see Chapter 5). By incorporating these approaches, we aim to enhance the adaptability of PLMs for multi-domain and multilingual use cases.

Low-Resource Scenarios for Task Fine-Tuning. Additionally, adaptive pre-training methods should be able to handle low-resource scenarios for task fine-tuning. In situations where labeled data for task fine-tuning is scarce, such as for low-resource languages or niche domains, models are required to adapt effectively to limited task training data while preserving their performance. Gururangan et al. (2020) showed that adaptive pre-training with unlabeled data leads to performance gains in low-resource settings for diverse domains and tasks in English. Aharoni and Goldberg (2020) proposed to find domain-specific clusters in PLMs to aid domain data selection, requiring only a small set of in-domain data for unsupervised domain adaptation training. Apart from data-centric approaches, another line of research focuses on aggregating representations of high-resource embeddings from the source domain and low-resource embeddings from the target domain with attention-based meta-embeddings (Kiela et al., 2018) or adversarial training approach (Lange et al., 2021a). For resource-lean languages, Lauscher et al. (2020) introduced continual few-shot transfer learning after task fine-tuning for resource-lean languages, and has proved to significantly reduce the performance gap observed for zero-shot transfer scenarios. Another area of research concentrates on the representation

space alignment in either cross-lingual (Liu et al., 2019a) or multilingual settings (Cao et al., 2019). In this work, we investigate the sample efficient few-shot transfer scenarios (Chapter 3) and the effect of continual few-shot transfer for multilingual task-oriented dialogs (Chapter 4). These efforts aim to improve the adaptability of PLMs, even with limited available task data.

2.4.2 Towards Efficient Adaptive Pre-training

As the size and complexity of PLMs continue to expand, the computational resources required for adaptive pre-training also increase substantially. This poses challenges for practical deployment and scalability. Enhancing the efficiency of adaptive pre-training methods revolves around two primary concerns (**C2**): (1) *Data-efficient* methods, which aim to maximize the use of limited data by optimizing data collection and utilization strategies; and (2) *Parameter-efficient* methods, that aim to reduce training time and optimize resource utilization while maintaining adaptability across multiple dimensions. These areas of investigation hold the key to overcoming computational obstacles and making adaptive pre-training more efficient and accessible in real-world applications.

Data-Efficient. Efficiently collecting sufficient and representative data for domains, languages, and social contexts for the knowledge transfer of model adaptation poses a significant challenge. Acquiring a substantial and diverse dataset that accurately represents the target domain and/or language tasks can be time-consuming, expensive, or even unfeasible in some cases. Traditional supervised learning approaches heavily rely on labeled data, which may be limited or unavailable for specific domains or niche areas (Ramponi and Plank, 2020; Ivison et al., 2023). As a result, data-efficient methods become essential in adaptive pre-training to make the most out of the available data, leverage transfer learning, and adapt language models effectively to domains and languages without an overwhelming reliance on extensive labeled datasets. In this work, we focus on utilizing a small amount of unlabeled domain- and language-specific datasets (around 50K to 200K). We propose a simple term-matching method to efficiently acquire in-domain data for task-oriented dialog (DomainCC and DomainReddit; see § 3.1). In Chapter 4, we introduce a novel multilingual multi-domain TOD datasets: Multi^{2}WOZ, enabling a reliable and robust resource to facilitate cross-lingual transfer studies on TOD. Further, we present target-language-specific (LangCC) and cross-lingual (LangOpenSubtitles) corpora, for conducting efficient dialogic pre-training for language adaptation. In Chapter 5, we adapt the language representation from demographic corpora of gender and age for efficient demographic adaptation.

Parameter-Efficient. Parameter-efficient adaptive pre-training seeks to optimize the utilization of model parameters to achieve effective transfer learning while minimizing the excessive computational overhead. Traditional approaches, like *full fine-tuning* (Gururangan et al., 2020), involve updating all parameters of PLMs, which can be computationally expensive and memory-intensive. To mitigate the computational complexity and memory requirements of adaptive pre-training, making it more feasible for resource-constrained environments, the goal is to design parameter-efficient methods that can efficiently adapt to multiple domains and languages, while retaining the pre-trained knowledge intact. Recent work proposed *modular-based* approaches (Pfeiffer et al., 2023) to adapt the model while only updating or adding a relatively small number of parameters.[15] *Adapters* (Rebuffi et al., 2017; Houlsby et al., 2019) introduce new trainable dense layers into PLMs while keeping the original model parameters fixed. Adapters have been proven effective in cross-lingual transfer (Pfeiffer et al., 2020) and are further extended to multi-domain and multilingual scenarios, including advanced stacking (Pfeiffer et al., 2020) and fusion (Pfeiffer et al., 2021). Though adapters are capable of modular portability to various downstream tasks and reducing computational cost by avoiding full fine-tuning, two main shortcomings are indicated: (1) training time significantly increases while preserving more parameters for training (Rücklé et al., 2021); (2) lack of expressiveness (Ansell et al., 2022). To address the issues, we propose a novel task-agnostic domain adaptation method, leveraging domain-specialized embeddings and tokenizers (see § 3.2; Hung et al. (2023b)). The proposed approach allows for parameter-efficient adaptation, and proves to be comparably better than adapters in few-shot scenarios.

2.4.3 Towards Interpretability of Adaptive Pre-training

Attaining a comprehensive understanding of the behavior and decision-making mechanisms exhibited by PLMs, holds utmost importance in maximizing their capabilities and addressing potential limitations. Interpretability, defined as "*the ability [of a model] to explain or to present [its predictions] in understandable terms to a human*" (Doshi-Velez and Kim, 2018; Luo et al., 2024), holds key significance in this pursuit. To enhance interpretability with our proposed adaptive pre-training methods for multidimensional adaptation of PLMs, we examine how models adapt not just in terms of performance gains but also in their responses across domains, languages, and demographic factors under different controlled conditions (**C3**). We aim to improve model transparency, uncover both strengths and limitations, and ultimately advance the field of adaptive pre-training toward more robust and comprehensive approaches.

In this work, we explore a set of analyses to enhance interpretability. We study the cross-domain transfer (§ 3.1.6) and token-level segmentation analysis (§ 3.2.6), showing the strength of the adaptive pre-training approach beneficial in closely related domains and

[15] Parameter-efficient fine-tuning methods typically have up to four orders of magnitude fewer parameters than full fine-tuning (Treviso et al., 2023).

how tokenization enhances in-domain terminology for better adaptation. Additionally, we conduct control experiments to examine the impact of demographic adaptation utilizing a multilingual PLM (§ 5.5.2). Through the analyses, we hope to shed light on how the multilingual PLM respond to various demographic factors and discern potential variations in their performance across different demographic subgroups. We expect the analyses will shift the focus of the research community beyond merely evaluating the performance gains achieved by adaptive pre-training methods. Instead, it is encouraged to further investigate how and why the proposed techniques operate, and foster diverse perspectives and insights in relation to the control experiments. Equipped with the awareness of interpretability concerns, we can take significant strides toward enhancing the diversity and inclusivity of proposed methods and trained models. This, in turn, will encourage innovation and progress in the field of NLP and promote more robust and effective methods for a broader spectrum of users and applications.

While adaptive pre-training methods have achieved performance gains in various downstream applications, continuous research and development are required to overcome the challenges of *effectiveness* (**C1**), *efficiency* (**C2**), and *interpretability* (**C3**). In the forthcoming Chapters, we delve into each of the outlined challenges, presenting and discussing the proposed solutions. By addressing these aspects, we hope to propel adaptive pre-training methods across multiple dimensions, and pave the way for more versatile and reliable language models in real-world use cases.

Part II

Adaptation

Chapter 3

Domain Adaptation

> *"The only way to make sense out of change is to plunge into it, move with it, and join the dance."*
>
> Alan Wilson Watts
> «Wisdom Of Insecurity: A Message for an Age of Anxiety»

The advent of Pre-trained Language Models (PLMs) based on Transformer (Vaswani et al., 2017) has ushered in remarkable advancements across a spectrum of NLP applications (see § 2.2). However, seamlessly adapting those PLMs to new domains persists as a central concern. The major two challenges arise (see § 2.3.1): (1) *Mismatch of Domain(s)*: in the realm of many machine learning algorithms, a default assumption arises that the training and test datasets follow the same underlying distribution. However, when these distributions do not match, a *domain shift* problem emerges (Lekhtman et al., 2021; Farahani et al., 2021). In this scenario, the source training data and the target task domain differ (i.e., they are not sampled from the same underlying distribution). Consequently, performance drops on the target task, which undermines the ability of models to truly generalize across varied contexts. (2) *Scarcity of task-annotated data*: the scarcity of annotated task data presents a substantial challenge (i.e., "Y scarcity"), while the availability of domain-specific data for domain specialization might be constrained (i.e., "X scarcity") (Ramponi and Plank, 2020).

The above challenges underscore the importance of domain adaptation, a crucial transfer learning technique aimed at mitigating the *domain shift* problem and enhancing the robustness and efficacy of language models in real-world settings.[1] Recent work proposed domain-adaptive pre-training utilizing PLMs (see § 2.3.1) as a three-stage training framework to mitigate the above challenges. However, two perspectives are addressed

[1]The extensive concern arises for domain adaptation: generalization beyond the training distribution. Ultimately, the models should possess the ability to adapt and robustly handle any test distribution without prior exposure to corresponding data. This broader imperative encompasses the necessity of effectively managing *out-of-distribution* scenarios, particularly when dealing with *unknown* domains (Volpi et al., 2018; Zhou et al., 2023).

to further enhance the adaptive pre-training methods via the *effectiveness* and *efficiency* perspectives for domain adaptation (**C1** and **C2**; see § 2.4).

In this Chapter, we first investigate domain-adaptive pre-training methods for task-oriented dialog systems. The capability of a task-oriented dialog system to seamlessly adapt to different domains holds paramount importance for its practical deployment. To address this crucial aspect, we (i) present data collection methods designed to tackle the challenges stemming from the scarcity of annotated in-domain data; (ii) introduce the integration of dialogic objectives to enhance the adaptability of PLMs to specific subdomains; and (iii) further substantiate the effectiveness of employing *adapters* (Houlsby et al., 2019) to foster more efficient domain adaptation in task-oriented dialog systems (see § 3.1; Hung et al. (2022a)). Despite the advantages that adapters bring, such as diminished training time due to a reduced number of parameters and decreased deployment costs through memory storage reduction, two primary shortcomings are indicated: (a) training time significantly increases due to the inclusion of additional parameters (Rücklé et al., 2021); (b) adapters are perceived to lack expressiveness (Ansell et al., 2022). To address these concerns, we propose a novel task-agnostic domain adaptation method, leveraging domain-specialized embeddings and tokenizers (see § 3.2; Hung et al. (2023b)). The efficacy of the proposed approach is demonstrated across 4 downstream tasks (including task-oriented dialog tasks, NER, and NLI) in both (i) single- and multi-domain; and (ii) high- and low-resource scenarios. Notably, our findings underscore that the proposed approach has shown comparably better performance in few-shot scenarios compared to *adapters*.

In the context of task-agnostic domain adaptation methods proposed in this Chapter, our aim is to shed light on the advantages inherent in various approaches across a spectrum of downstream tasks. We hope that our contributions will pave the way for more effective and efficient domain adaptation techniques, with implications that extend across various domains for NLP applications.

3.1 Domain Specialization for Task-Oriented Dialog

[*]Recent work has shown that self-supervised dialog-specific pre-training on large conversational datasets yields substantial gains over traditional language modeling pre-training in downstream Task-oriented Dialog (TOD) tasks (Henderson et al., 2019c, 2020). These approaches, however, exploit general dialogic corpora (e.g., Reddit) and thus presumably fail to reliably embed domain-specific knowledge useful for concrete downstream TOD domains. In this Section, we investigate the effects of domain specialization of PLMs for TOD. Within our proposed **D**omain **S**pecialization framework for **TOD** (**DS-TOD**),

[*]This Section is adapted from: **Chia-Chien Hung**, Anne Lauscher, Simone Paolo Ponzetto, and Goran Glavaš. 2022. DS-TOD: Efficient Domain Specialization for Task-Oriented Dialog. In *Findings of the Association for Computational Linguistics (ACL 2022)*, pages 891–904, Dublin, Ireland, May 2022. Association for Computational Linguistics.

we first automatically extract salient domain-specific terms, and then utilize them to construct DomainCC and DomainReddit – resources that we leverage for domain-specific pre-training (**C2**; § 3.1.3), based on (i) Masked Language Modeling (MLM) and (ii) Response Selection (RS) objectives, respectively. We further propose a resource-efficient and modular domain specialization by means of *domain adapters* – additional parameter-light layers in which we encode the domain knowledge (**C2**; § 3.1.4). Our experiments with prominent TOD tasks – dialog state tracking (DST) and response retrieval (RR) – encompassing five domains from the MultiWOZ benchmark (Budzianowski et al., 2018; Eric et al., 2020) demonstrate the effectiveness of DS-TOD (§ 3.1.6). Moreover, we show that the light-weight adapter-based specialization (1) performs comparably to full fine-tuning[2] in single-domain setups, and (2) is particularly suitable for multi-domain specialization, where besides advantageous computational footprint, it can offer better TOD performance (**C1**).

3.1.1 Introduction

Task-oriented dialog (TOD) as shown in Figure 3.1, where conversational agents help users complete concrete tasks (e.g., book flights or order food), has arguably been one of the most prominent NLP applications in recent years, both in academia (Budzianowski et al., 2018; Henderson et al., 2019c; Liu et al., 2021a, *inter alia*) and industrial applications (e.g., Yan et al., 2017; Henderson et al., 2019b; Gupta et al., 2022; Valizadeh and Parde, 2022). Like for most other NLP tasks, fine-tuning of PLMs (e.g., BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019); see § 2.2) pushed the state-of-the-art in TOD tasks (Budzianowski and Vulić, 2019; Hosseini-Asl et al., 2020), with language model pre-training at the same time alleviating the need for large labeled datasets (Ramadan et al., 2018).

More recent TOD work recognized the idiosyncrasy of dialog – i.e., dialogs represent interleaved exchanges of utterances between two (or more) participants – and proposed pre-training objectives specifically tailored for dialogic corpora (Henderson et al., 2019c; Wu et al., 2020; Bao et al., 2020, *inter alia*). For instance, Wu et al. (2020) pre-train TOD-BERT model on the concatenation of nine human-to-human multi-turn dialog datasets (see § 2.2.3). Similarly, Henderson et al. (2019c, 2020) pre-train a general-purpose dialog encoder on a large corpus from Reddit by means of response selection objectives. Encoding dialogic linguistic knowledge in this way led to significant performance improvements in downstream TOD tasks.

While these approaches impart useful dialogic linguistic knowledge, they fail to exploit the fact that individual task-oriented dialogs typically belong to one narrow domain (e.g., *food* ordering) or few closely related domains (e.g., booking a *train* and *hotel*; Budzianowski et al., 2018; Ramadan et al., 2018). Given the multitude of different

[2]To clarify, *full fine-tuning* (Houlsby et al., 2019) is considered in the adaptation phase of adaptive pre-training framework to update all PLM parameters (detailed in § 2.3.1) in this Chapter. While fine-tuning indicates the downstream task fine-tuning, as the third stage illustrated in Figure 2.4.

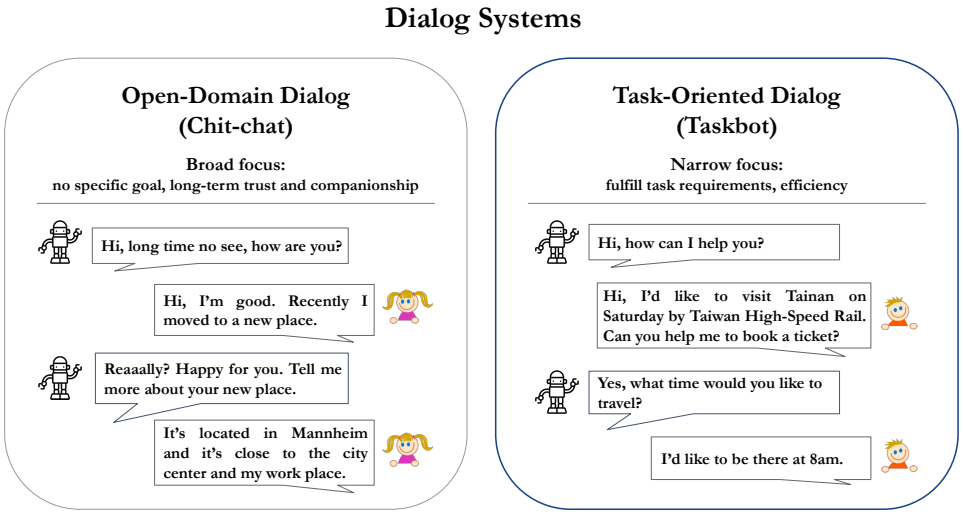


Figure 3.1: Examples of dialog systems, including an open-domain dialog (chit-chat), and a task-oriented dialog (taskbot, or TOD) system. Open-domain dialog involves engaging in conversations without a particular subject or user objective, with the primary aim of establishing the long-term user engagement. In contrast, task-oriented dialog systems are designed to assist users in achieving their goals in specific domain(s) (Huang et al., 2020).

downstream TOD domains (e.g., ordering *food* is quite different from booking a *flight*), it is, intuitively, unlikely that general dialogic pre-training reliably encodes domain-specific knowledge for all of them.

In this Section, we propose **D**omain **S**pecialization for **T**ask **O**riented **D**ialog (**DS-TOD**), a novel task-*agnostic* domain-adaptive pre-training framework for task-oriented dialog. DS-TOD, depicted in Figure 3.2, encloses three steps: (1) we extract domain-specific terms (e.g., terms related to ordering *taxi* or terms related to buying a *train* ticket) from the training portions of a task-specific TOD corpus; (2) we next use the extracted terms to obtain domain-specific data from large unlabeled corpora (e.g., Reddit); (3) finally, we conduct intermediate training of a PLM (e.g., BERT) on the domain-specific data, in order to inject the domain-specific knowledge into the encoder. This intermediate training step ensures domain specificity and is designed to be easily adaptable to any downstream TOD tasks in a task-*agnostic* manner. As a result, we obtain a domain-specialized PLM, which can then be fine-tuned for *any* downstream TOD tasks, e.g., dialog state tracking. An overview of domain-adaptive pre-training framework for TOD is illustrated in Figure 3.3.

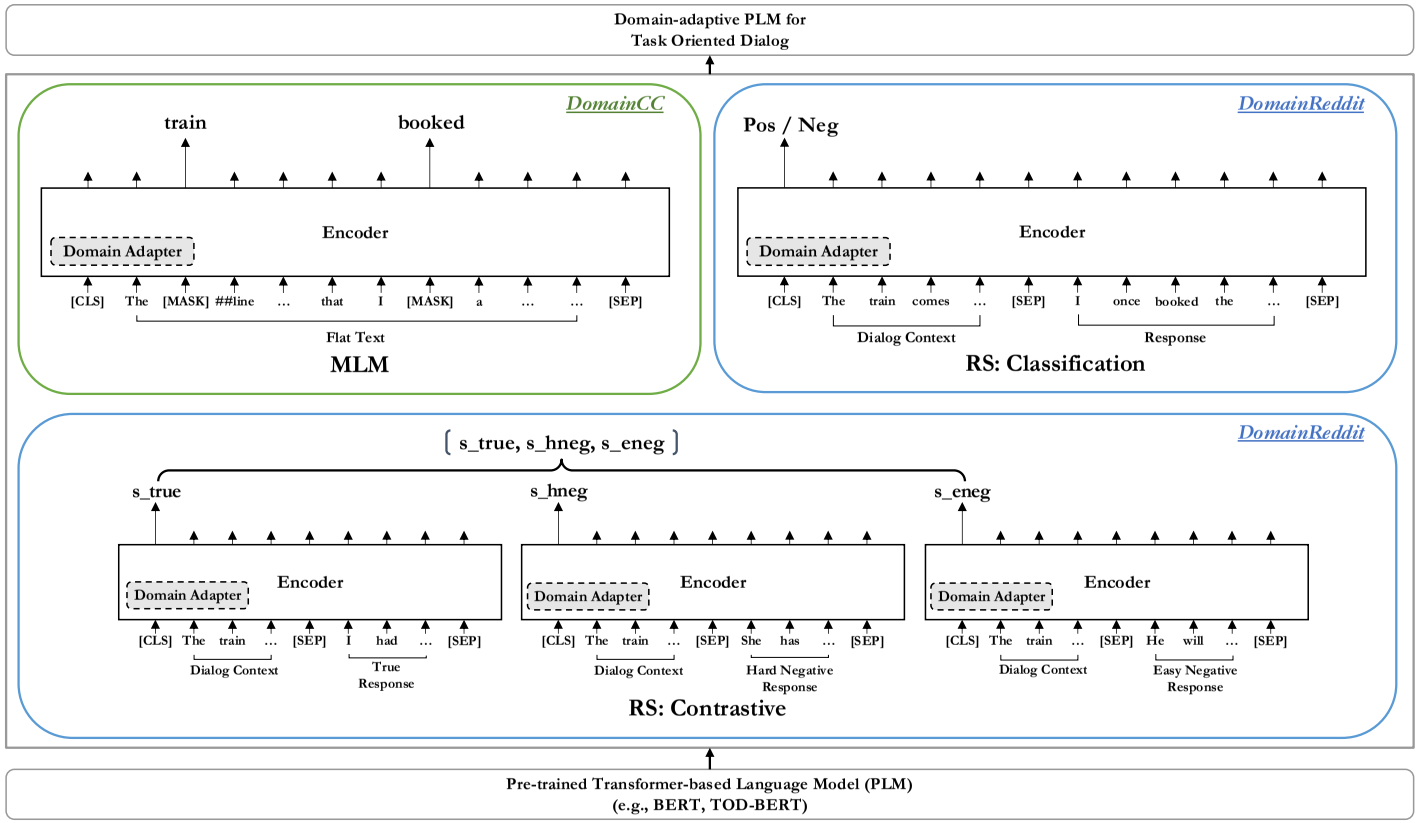


Figure 3.2: Overview of DS-TOD. Three different specialization objectives for injecting domain-specific knowledge into PLMs (§ 3.1.4): (1) Masked Language Modeling (MLM) on the "flat" domain corpus DomainCC, (2) Response Selection (RS) via Classification, and (3) Response Selection via Contrastive Learning operating on the dialogic DomainReddit. Domain adaptation is performed either via (a) *full fine-tuning* or (b) *adapters*.

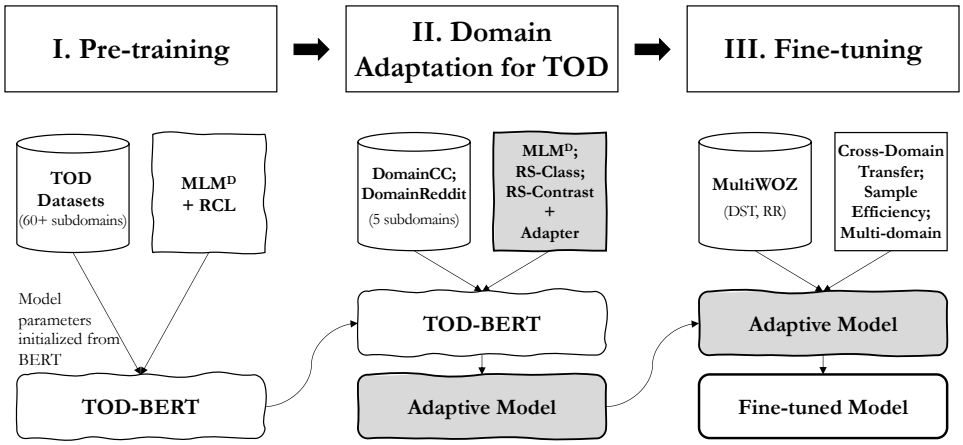


Figure 3.3: Overview of the domain-adaptive pre-training framework for TOD. The framework involves a three-stage process: *Pre-training*, *Domain Adaptation for TOD*, *Fine-tuning*. In this Chapter, intermediate training indicates the second (II) stage: domain adaptation phase.

Contributions. We advance the state-of-the-art in TOD with the following contributions: **(i)** Departing from general-purpose dialogic pre-training (e.g., Henderson et al., 2019a), we leverage a simple terminology extraction method to construct DomainCC and DomainReddit corpora which we then use for domain-specific LM and dialogic pre-training, respectively. **(ii)** We examine different objectives for injecting domain-specific knowledge into PLMs: we empirically compare Masked Language Modeling (MLM) applied on the "flat" domain dataset DomainCC against two different Response Selection (RS) objectives (Oord et al., 2018; Henderson et al., 2019c) applied on the dialogic DomainReddit corpus. We demonstrate the effectiveness of our specialization on two TOD tasks – dialog state tracking (DST) and response retrieval (RR) – for five domains from the MultiWOZ dataset (Budzianowski et al., 2018; Eric et al., 2020). **(iii)** We propose modular domain specialization for TOD via *adapter* modules (Houlsby et al., 2019; Pfeiffer et al., 2020). Additional experiments reveal the advantages of adapter-based specialization in *multi-domain* TOD: combining domain-specific adapters via stacking (Pfeiffer et al., 2020) or fusion (Pfeiffer et al., 2021) (a) performs *en par* with or outperforms expensive multi-domain pre-training, while (b) having a much smaller computational footprint.[3] **(iv)** The code and resources developed for **DS-TOD** are publicly available.[4]

[3]Assume N mutually close domains and a bi-domain downstream setup (any two domains). With an adapter-based approach, we pre-train one adapter for each domain (complexity: N) and then combine the adapters of the two domains intertwined in the concrete downstream setup. In contrast, multi-domain specialization would require one bi-domain pre-training for each two-domain combination (complexity: N^2).

[4]https://github.com/umanlp/DS-TOD

3.1.2 Related Work

TOD Datasets. Datasets for TOD can be divided into single-domain (Wen et al., 2017; Mrkšić et al., 2017a) and multi-domain ones (Budzianowski et al., 2018; Rastogi et al., 2020; Moghe et al., 2023a). The latter are generally seen as closer to real-world situations and intended usages of personal assistants, where strict adherence to a single domain is unlikely. While downstream TOD datasets exist for specific domains, corresponding large(er)-scale datasets that would enable domain-adaptive pre-training have been limited to the general domain (Henderson et al., 2019a).

Pre-trained Language Models in Dialog. The advantages of large-scale pre-training of language models on massive amounts of text (Devlin et al., 2019; Radford et al., 2019; Lewis et al., 2020a), ubiquitous in natural language tasks, have also spilled over to task-oriented dialog. Recent research focused on either (1) leveraging general-domain dialogic resources (e.g., Reddit, Twitter) in order to improve downstream TOD tasks (Henderson et al., 2019c, 2020; Zhang et al., 2020; Bao et al., 2020; Liu et al., 2021b) or (2) using TOD datasets to inject dialogic structure into PLMs (Wu et al., 2020; Peng et al., 2021; Su et al., 2022). Neither of the two, however, considers task-agnostic domain-adaptive pre-training methods, that could support any downstream TOD tasks.

Domain Adaptation and Knowledge Reuse. Common unsupervised approaches for extracting domain-specific portions of large general domain corpora, rely on term and document frequencies (Kim et al., 2009), learn a candidate retrieval-based classifier (Glavaš et al., 2020) or perform unsupervised domain clustering with PLMs (Aharoni and Goldberg, 2020). In this Section, we address the lack of in-domain resources by creating large-scale domain-specific corpora – flat as well as dialogic – for the five domains of the MultiWOZ dataset using a simple TF-IDF based term filtering approach.

To inject domain knowledge into PLMs, recent work proposed the approach either as a step before the downstream task-specific fine-tuning (Glavaš et al., 2020) or in parallel with it (i.e., in a multi-task training setup) (Gururangan et al., 2020). In the narrower context of TOD, Whang et al. (2020) present the lone effort on domain specialization for TOD: they focus on easier, single-domain TOD and investigate the specialization effect with a single task, response retrieval. In this work, in contrast, we focus on dialogic domain-adaptive pre-training, as depicted in Figure 3.3, and further demonstrate its effectiveness in multi-domain use-cases for TOD (**C1**). For efficiency and to avoid catastrophic forgetting, adapter modules have been widely used for parameter-efficient fine-tuning of PLMs for new tasks (Houlsby et al., 2019) and languages (Pfeiffer et al., 2020). Non-destructive adapter compositions (e.g., stacking or fusion) can be beneficial if multiple knowledge facets, stored in separate adapters, need to be leveraged (Pfeiffer et al., 2020, 2021) (**C2**).

	Taxi	Restaurant	Hotel	Train	Attraction
Slot names	*destination*, *departure*, *arriveBy*, *leaveAt*	*pricerange*, *area*, *day*, *people*, *food*, *name*, *time*	*pricerange*, *area*, *day*, *people*, *type*, *parking*, *stars*, *internet*, *stay*, *name*	*destination*, *departure*, *day*, *people*, *arriveBy*, *leaveAt*	*area*, *type*, *name*
# Total‡	1654, 207, 195	3813, 438, 437	3381, 416, 394	3103, 484, 494	2717, 401, 395
# Multi-domain‡	1329, 150, 143	2616, 388, 375	2868, 360, 327	2828, 454, 461	2590, 390, 383
# Single domain‡	325, 57, 52	1197, 50, 62	513, 56, 67	275, 30, 33	127, 11, 12
% Single domain	24.62%	19.00%	15.21%	7.25%	3.49%

Table 3.1: Statistics for MULTIWOZ 2.1 dataset. For each domain, we report slot names, the total number of dialogs as well as the number of single-domain and multi-domain dialogs. ‡The number of dialogs for training, development, and test set respectively.

3.1.3 Domain-Specialized Corpora

We create large-scale domain-specific corpora in two steps: given a collection of in-domain dialogs, we first extract salient domain terms (§ 3.1.3.1); we then use these domain terms to filter content from CCNET (Wenzek et al., 2020) as a large general corpus and Reddit (Baumgartner et al., 2020) as a source of dialogic data (§ 3.1.3.2).

3.1.3.1 Domain-Specific Ngrams

We start from Wizard-of-Oz, a widely used multi-domain TOD dataset (MULTIWOZ; Budzianowski et al., 2018): we resort to the revised version 2.1 (Eric et al., 2020) and work with the five domains that have test dialogs: *Taxi*, *Attraction*, *Train*, *Hotel*, and *Restaurant*. Table 3.1 shows the statistics of domain-specific MultiWOZ subsets.

To obtain large domain-specific corpora for domain adaptation phase as illustrated in Figure 3.3, we first construct sets of domain-specific ngrams for each domain. To this end, we first compute TF-IDF scores for all {1,2,3}-grams found in single-domain dialogs from MULTIWOZ training sets[5]: our term frequency (TF) is the total ngram frequency in all domain dialogs; the inverse document frequency (IDF) is here the inverse of the proportion of dialogs that contain the ngram. We then select N ngrams with the largest TF-IDF scores (in all our experiments, we set $N = 80$) and manually eliminate from the list ngrams that are not intrinsic to the domain (e.g., weekdays, named locations). Finally, since MULTIWOZ terms follow the British English spelling (e.g., *centre*, *theatre*), we add the corresponding American English word forms (e.g., *center*, *theater*). The complete resulting ngram sets for all domains are given in Table 3.2.

[5]E.g., for the *Taxi* domain, we collect all training dialogs that span only that domain (i.e., only taxi ordering) and omit dialogs that besides *Taxi* involve one or more other domains (e.g., taxi ordering and hotel booking in the same dialog).

Domain	Ngrams
Taxi	*taxi, contact number, book a taxi, booked, time schedule, pickup, leaving, booked type, booking completed, departing, destination, cab, completed booked, honda, ford, audi, lexus, toyota, departure, skoda, lexus contact, toyota contact, ford contact, volvo, train station, departure site, tesla, audi contact, honda contact, skoda contact, picking, departing, volkswagen*
Attraction	*museum, college, entrance, attraction, information, centre town, center town, entertainment, swimming pool, gallery, sports, nightclub, pounds, park, postcode, architecture, centre area, center area, cinema, church, trinity college, entrance free, jello gallery, post code, town centre, town center, downing college*
Train	*train station, travel time, leaving, pounds, train ticket, departing, payable, train leaving, cambridge, london, reference id, arrive, destination, kings cross, total fee, departure, arriving, book a train, booked, stansted, stansted airport, peterborough, traveling, trip, airport, booking successful, norwich*
Hotel	*hotel, nights, parking, free parking, wifi, star hotel, price range, free wifi, guesthouse, guest house, internet, guest, hotel room, star rating, expensive room, priced, rating, book room, moderately priced, moderate price, stay for, reservation, breakfast available, book people, fully booked, booking, reference*
Restaurant	*restaurant, food, price range, expensive, cheap, priced, chinese food, italian food, moderately priced, south town, book table, city, north town, serving, city centre, city center, european food, reservation, food type, phone address, centre town, center town, expensive restaurant, moderate price, cuisine, restaurant center, restaurant centre, south town, expensive price, east town, cheap restaurant, indian food, asian food, british food, book people*

Table 3.2: Salient domain ngrams extracted from the single-domain training portions of MULTIWOZ 2.1.

3.1.3.2 Domain-Specific Corpora

We next use the extracted domain ngrams to retrieve two types of in-domain data for domain specialization: (i) flat text and (ii) dialogic data.

DOMAINCC. For each of the five MULTIWOZ domains, we create the corresponding flat text corpus for MLM training by filtering out 200K sentences from the English portion of CCNet (Wenzek et al., 2020) – a high-quality collection of monolingual corpora extracted from CommonCrawl[6] that has been used for pre-training multilingual PLMs (Conneau et al., 2020a; Liu et al., 2020) – that contain one or more of the previously extracted domain terms. We additionally clean all DOMAINCC portions by removing email addresses and URLs, and lower-casing all terms. We provide example excerpts for each domain in Appendix B.1.

DOMAINREDDIT. Being constructed from CommonCrawl, DOMAINCC portions do not exhibit any natural conversational structure, encoding of which has been shown beneficial for downstream TOD tasks (Henderson et al., 2019c; Wu et al., 2020). We thus additionally create a dialogic corpus for each domain: we employ the Pushshift API (Baumgartner et al., 2020) to extract dialogic data from Reddit (period 2015–2019). To this end, we select subreddits related to *traveling* (listed in Table 3.3) which we believe align well with the content of MULTIWOZ, which was created by simulating conversations between tourists and clerks in a tourist information center. Each of the subreddits contains threads composed of a series of comments, each of which can serve as a *context*

[6] https://commoncrawl.org/

Subreddit	# Members‡	Domains
travel	5.8M	Taxi, Attraction, Train, Hotel, Restaurant
backpacking	2.5M	Taxi, Attraction, Train, Hotel, Restaurant
solotravel	1.7M	Taxi, Attraction, Train, Hotel, Restaurant
CasualUK	797K	Taxi, Attraction, Train, Hotel, Restaurant
unitedkingdom	553K	Taxi, Attraction, Train, Hotel, Restaurant
restaurant	81.6K	Restaurant
trains	64.8K	Train, Attraction
hotel	1.8K	Hotel
hotels	4.9K	Hotel
tourism	3.9K	Taxi, Attraction, Train, Hotel, Restaurant
uktravel	1.5K	Taxi, Attraction, Train, Hotel, Restaurant
taxi	0.6K	Taxi

Table 3.3: Subreddits and associated domains selected for creating DomainReddit. ‡The recorded number of members for each subreddit is based on the crawling date (24/05/2021).

Field	Example
Subreddit	restaurant
Context	*Hosts don't get tips? That's news to me. Most host positions in my area get at least 1% of sales; they make anywhere between 60-100 per night in tips!*
True Response	*We get tips but definitely not that much (in my experience). The tip out in my restaurant is 1% split between shift leaders, food runners, and any other FOH other than servers/bartenders. Full time hosts get about 50-75 every other week*
False Response	*Wow that's terrible. Then again, my restaurant is in CA, so wages and guest check averages are usually higher.*

Table 3.4: Example from DomainReddit dataset.

followed by a series of *responses*. For DomainReddit we select context-response pairs where either the context utterance or the response contains at least one of the domain-specific terms. To construct examples for injecting conversational knowledge, we follow Henderson et al. (2019a) and couple each *true* context-response pair (i.e., a comment and its immediate response) with a *false response* – a non-immediate response from the same thread. Table 3.4 provides an example context with its true and one false response; further examples, for all domains, are available in Appendix B.1. Finally, we also clean DomainReddit by removing email addresses and URLs as well as comments having fewer than 10 characters. The total number of Reddit triples *(context, true response, false response)* that we extract this way for the MultiWOZ domains is as follows: *Taxi* – 120K; *Attraction* – 157K; *Hotel* – 229K; *Train* – 229K; and *Restaurant* – 243K.

3.1.4 Domain-Adaptive Pre-training for TOD

The next step in DS-TOD is the injection of domain-specific knowledge through intermediate training on DomainCC and DomainReddit. To this end, we train a PLM (1) via Masked Language Modeling on DomainCC, and (2) using two different Response Selection objectives on DomainReddit. Finally, for all objectives, we compare full fine-tuning (i.e., update *all* PLM parameters in domain adaptation phase) against adapter-based specialization, where we freeze the PLM parameters and inject domain knowledge into new adapter layers.

3.1.4.1 Training Objectives

Masked Language Modeling (MLM). Following successful work on *adaptive pre-training* leveraging language modeling for domain-adaptation (Gururangan et al., 2020; Aharoni and Goldberg, 2020; Glavaš et al., 2020), we delve into the advantages of further pre-training PLMs via Masked Language Modeling (MLM) with a small subset of in-domain data (typically around 100 K). Our research focuses on investigating the effect of applying standard MLM on small domain-specific portions of DomainCC to inject domain-specific knowledge for TOD (**C2**; see § 2.4.2). In this context, the MLM loss L_{mlm} is computed as the negative log-likelihood of the predicted probability of the true token (Devlin et al., 2019; Liu et al., 2019c).

$$L_{mlm} = -\sum_{m=1}^{M} \log P(t_m)\,, \tag{3.1}$$

where M is the total number of masked tokens in a given text and $P(t_m)$ is the predicted probability of the token t_m over the vocabulary size.

Response Selection (RS). RS objectives force the model to recognize the correct response utterance given the context – pre-training with such objectives is particularly useful for conversational settings, including TOD tasks (Henderson et al., 2019c, 2020). We consider two RS objectives. The first is a simple pairwise binary classification formulation (**RS-Class**): given a context-response pair, predict whether the response is a true (i.e., immediate) response to the context. The loss function for RS-Class loss is formulated as:

$$L_{RS-class} = -\left(y \log(f(c,r)) + (1-y)\log(1-f(c,r))\right)\,, \tag{3.2}$$

where c represents the context, r denotes the response, y is the actual label indicating whether the response is true (1) or false (0), and f is the function calculating the predicted probability of the response to the given context.

We straightforwardly use pairs of contexts and their true responses from DomainReddit as positive training instances. Next, we create negative samples for each positive

instance as follows: (a) we use the crawled *false response* from DomainReddit, which represents a relevant but non-consecutive response from the same thread; such non-immediate responses from the same thread represent the so-called *hard negatives* introduced to prevent the model from learning simple lexical cues and similar heuristics that poorly generalize; (b) we additionally randomly sample k utterances from the same domain but different threads (these represent the so-called *easy negatives*).[7]

The second response selection objective (**RS-Contrast**) that we adopt is a type of loss function used for contrastive model training based on the representational similarities between sampled positive and negative pairs (Oord et al., 2018). It has been used for pre-training cross-lingual language models (Chi et al., 2021) and shown to be useful in information retrieval (Reimers and Gurevych, 2021; Thakur et al., 2021; Litschko et al., 2022). The goal is to estimate the mutual information between pairs of variables by discriminating between a positive pair and its associated N negative pairs. Given a true context-response pair and N negatives, the noise-contrastive estimation (NCE) loss is computed as:

$$L_{NCE} = -\log \frac{\exp\left(f(c, r_{+})\right)}{\sum_{i=1}^{N+1} \exp\left(f(c, r_i)\right)}\,,$$

where c is the context, r_{+} is the true response and r_i iterates over all responses for the context – the true response r_{+} and N false responses; a function f produces a score meant to indicate whether the response r is a true response of the context c.

By learning to differentiate whether the response is true or false for a given context (RS-Class) or to produce a higher score for a true response than for false responses (RS-Contrast), RS objectives encourage the PLM to adapt to the underlying structure of the conversation. By feeding only in-domain data to it, we impart domain-specific conversational knowledge into the model.

3.1.4.2 Adapter-Based Domain Specialization

Fully fine-tuning the model requires adjusting all of the model's parameters, which can be undesirable due to large computational effort and risk of catastrophic forgetting of the previously acquired knowledge (McCloskey and Cohen, 1989; Pfeiffer et al., 2021). To alleviate these issues, we investigate the use of adapters (Houlsby et al., 2019), additional parameter-light modules that are injected into a PLM before fine-tuning (**C2**; § 2.4.2). In adapter-based fine-tuning only adapter parameters are updated while the pre-trained parameters are kept frozen (and previously acquired knowledge thus preserved). We adopt the Adapter-Transformer architecture proposed by Pfeiffer et al. (2020), which inserts a single adapter layer into each Transformer layer and computes the output of the adapter, a two-layer feed-forward network, as follows:

$$\mathit{Adapter}(\mathbf{h}, \mathbf{r}) = U \cdot g(D \cdot \mathbf{h}) + \mathbf{r}\,,$$

[7] k is uniformly sampled from the set {1, 2, 3}.

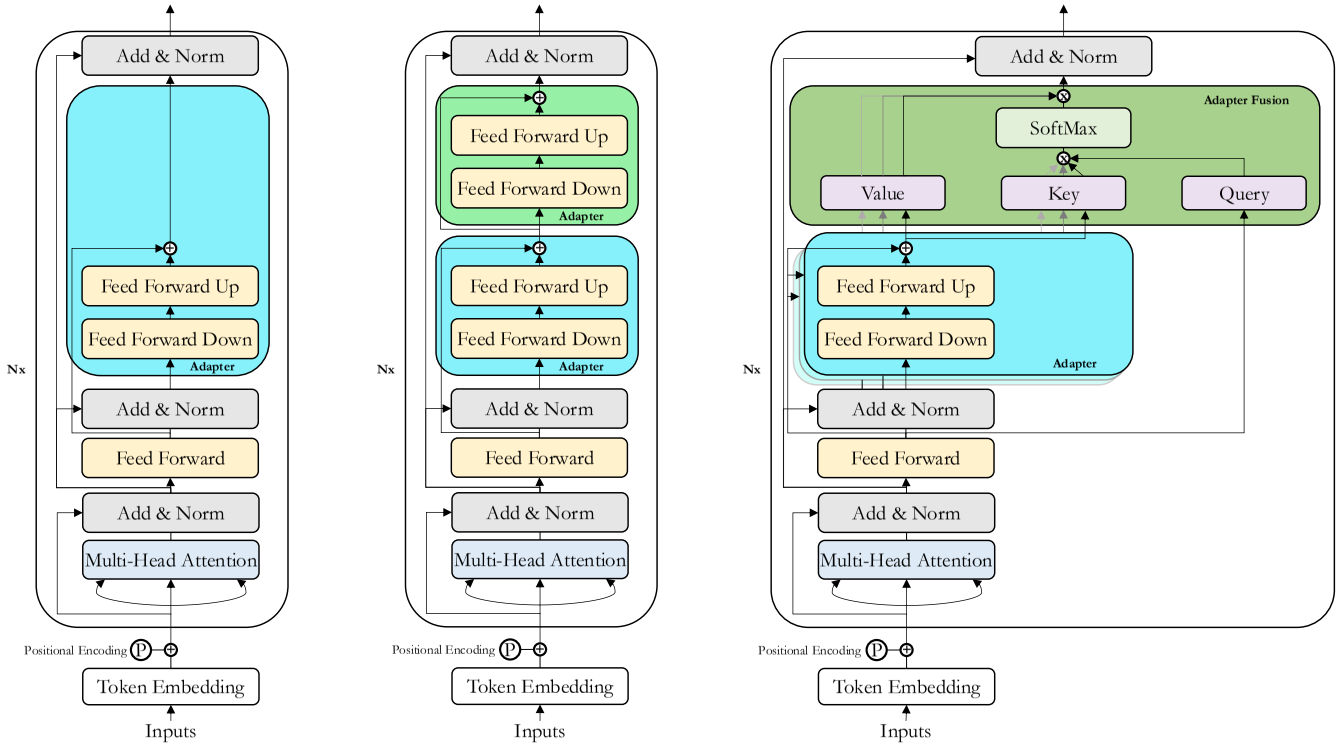


Figure 3.4: Illustration of Adapter-Transformer architecture, featuring Adapter (*single*), Stacking (*multi*), and Fusion (*multi*). (a) **Adapter** (Pfeiffer et al., 2020): a module designed for *single*-domain adaptation within the Transformer architecture (see § 2.1.3; Vaswani et al. (2017)), followed by the configurations for *multi*-domain applied scenarios with. (b) **Stacking** (Pfeiffer et al., 2020): multiple adapters are layered vertically. (c) **Fusion** (Pfeiffer et al., 2021): multiple adapters are fused with a parameterized mechanism for mixing encoded information from each adapter.

with $\mathbf{h}$ and $\mathbf{r}$ as the hidden state and residual of the respective transformer layer. $D \in \mathbb{R}^{m \times h}$ and $U \in \mathbb{R}^{h \times m}$ are the linear down- and up-projections, respectively (h being the transformer's hidden size, and m as the adapter's bottleneck dimension), and $g(\cdot)$ is a non-linear activation function. The residual $\mathbf{r}$ is the output of the Transformer's feed-forward layer, whereas $\mathbf{h}$ is the output of the subsequent layer normalization. The down-projection D compresses token representations to the adapter size $m \ll h$, and the up-projection U projects the activated down-projections back to the Transformer's hidden size h. The ratio h/m captures the factor by which the adapter-based fine-tuning is more parameter-efficient than full fine-tuning.

For multi-domain TOD scenarios (i.e., dialogs covering more than a single domain), we further experiment with combinations of individual domain adapters: (1) **Stacking**: sequential stacking of adapters one on top of the other (Pfeiffer et al., 2020), and (2) **Fusion**, where we compute a weighted average of outputs of individual adapter, with fusion weights as parameters that are learned in the final task-specific fine-tuning (Pfeiffer et al., 2021). An overview of Adapter-Transformer architecture is illustrated in Figure 3.4.

3.1.5 Experimental Setup

We demonstrate the effectiveness of our domain-specialization framework (DS-TOD) by comparing it to non-specialized baseline models and thoroughly compare different specialization methods from § 3.1.4.

Evaluation Task and Measures. We evaluate our domain-specialized models and baselines on two prominent downstream TOD tasks: Dialog State Tracking (DST) and Response Retrieval (RR). DST is treated as a multi-class classification task based on a predefined ontology, where given the dialog history, the goal is to predict the output state, i.e., (domain, slot, value) tuples. For our implementation, we follow Wu et al. (2020), and represent the dialog history as a sequence of utterances. The model then needs to predict slot values for each (domain, slot) pair at each dialog turn. We report the *joint goal accuracy*, in which the predicted dialog states are compared to the ground truth slot values at each dialog turn. The ground truth contains slot values for all the (domain, slot) candidate pairs. A prediction is considered correct if and only if all predicted slot values exactly match its ground truth values. For DST, we show an example in the hotel domain from MULTIWOZ 2.1 dataset (Eric et al., 2020):

Utterance	*I need to book a hotel in the east that has 4 stars.*
Slots	HOTEL-AREA : EAST
	HOTEL-STARS: 4

RR is a ranking problem, relevant for retrieval-based TOD systems (Wu et al., 2017; Henderson et al., 2019c). Following Henderson et al. (2020) and Wu et al. (2020), we adopt recall at top 1 rank given 99 randomly sampled candidates ($R_{100}@1$) as the eval-

uation metric for RR. We demonstrate an example for RR in the hotel domain from MULTIWOZ 2.1 dataset (Eric et al., 2020) as following, where R3 is considered as the *correct response* of the *context*:

Context		*I need to book a hotel in the east that has 4 stars.*
Responses	R1:	*That does not matter as long as it has free wifi and parking.*
	R2:	*If you would like something cheap, I recommend the Allenbell. For something moderately priced, I would recommend the Warkworth House.*
	✓ R3:	*I can help you with that. What is your price range?*
	...	...

Data. In the domain-adaptive pre-training procedure, we use the domain-specific portions of our novel DOMAINCC and DOMAINREDDIT resources (§ 3.1.3). For the MLM training, we randomly sample 200K domain-specific contexts from DOMAINCC and dynamically mask 15% of the subword tokens (Liu et al., 2019c). For RS-Class and RS-Contrast, we randomly sample 200K instances from DOMAINREDDIT. We evaluate the efficacy of the methods on DST and RR using MULTIWOZ 2.1 (Eric et al., 2020). Since we aim to understand the effect of the domain specialization methods, we construct domain-specific training, development, and testing portions from the original data set by assigning them all dialogs that belong to a domain (i.e., both single- and multi-domain dialogs) from respective overall (train, dev, test) portions.

Models and Baselines. We experiment with two PLMs: BERT (Devlin et al., 2019) and its TOD-sibling, TOD-BERT (Wu et al., 2020).[8] As baselines, we report the performance of the non-specialized variants and compare them against our domain-specialized PLM variants, obtained after domain-adaptive MLM-training on DOMAINCC or RS-Class/RS-Contrast training on DOMAINREDDIT.

Hyperparameters and Optimization. During intermediate training (i.e., domain adaptation phase in Figure 3.3), we fix the maximum sequence length to 256 subword tokens (for RS objectives, we limit both the context and response to 128 tokens). We train for 30 epochs, in batches of 32 instances and search for the optimal learning rate among the following values: $\{1 \cdot 10^{-4}, 5 \cdot 10^{-5}, 1 \cdot 10^{-5}, 1 \cdot 10^{-6}\}$. We apply early-stopping based on development set performance (patience: 3 epochs). We minimize the cross-entropy loss using AdamW (Loshchilov and Hutter, 2019). For downstream evaluation (i.e., fine-tuning phase in Figure 3.3), we train for 300 epochs in batches of 6 (DST) and 24 instances (RS) with the learning rate fixed to $5 \cdot 10^{-5}$. We also apply dev-set-based early-stopping (patience: 10 epochs).

[8] We use the pre-trained language models weights loaded from HuggingFace: `bert-base-cased` and `TODBERT/TOD-BERT-JNT-V1`. More details about the models can be referred to § 2.2.1 and § 2.2.3.

Model	Dialog State Tracking						Response Retrieval					
	Taxi	Res.	Hotel	Train	Attr.	Avg.	Taxi	Res.	Hotel	Train	Attr.	Avg.
BERT	23.87	35.44	30.18	41.93	29.77	32.24	23.25	37.61	38.97	44.53	48.47	38.57
TOD-BERT	30.45	43.58	36.20	48.79	42.70	40.34	45.68	57.43	53.84	**60.66**	60.26	55.57
BERT-MLM	23.74	37.09	32.77	40.96	36.66	34.24	31.37	53.08	45.41	51.66	52.23	46.75
TOD-BERT-MLM	29.94	43.14	36.11	47.61	41.54	39.67	41.77	55.27	50.60	55.17	54.62	51.49
TOD-BERT-RS-Class	**36.39**	43.38	37.89	48.82	43.31	41.96	47.01	58.21	**57.05**	59.70	57.72	55.94
TOD-BERT-RS-Contrast	35.03	**44.81**	**38.74**	**49.04**	42.73	**42.07**	48.04	59.82	54.49	60.06	60.63	56.61
BERT-MLM-adapter	22.52	40.49	31.90	42.17	35.05	34.43	32.84	44.01	39.15	38.43	45.05	39.90
TOD-BERT-MLM-adapter	32.06	44.06	36.74	48.84	**43.50**	41.04	49.08	58.18	55.55	59.46	60.26	56.51
TOD-BERT-RS-Class-adapter	33.10	42.57	38.61	49.03	42.35	41.13	**49.59**	**61.26**	56.87	58.88	60.00	**57.32**
TOD-BERT-RS-Contrast-adapter	34.90	44.42	37.52	48.71	42.83	41.68	47.97	58.97	55.41	59.15	**61.95**	56.69

Table 3.5: Results of DS-TOD models on two downstream tasks: Dialog State Tracking (DST) and Response Retrieval (RR) with joint goal accuracy (%) as the metric for DST and R_{100}@1 (Henderson et al., 2020) (%) for RR.

3.1.6 Results and Discussion

Overall Performance. We report downstream DST and RR results in Table 3.5, which is segmented in three parts: (1) at the top we show the baseline results (BERT, TOD-BERT) without any domain specialization; (2) in the middle of the table we show results of PLMs domain-specialized via full fine-tuning; (3) the bottom of the table contains results for adapter-based domain specialization (**C1**; § 2.4.1).

In both DST and RR, TOD-BERT massively outperforms BERT due to its conversational knowledge. Domain specialization brings gains for both PLMs across the board. The only exception is full MLM-fine-tuning of TOD-BERT (i.e., TOD-BERT-MLM vs. TOD-BERT; -4% for RR and -0.8% for DST): we believe that this is an example of negative interference – while TOD-BERT is learning domain knowledge, it is – because of MLM-based domain training – forgetting the conversational knowledge obtained in dialogic pre-training (Wu et al., 2020). This hypothesis is further supported by the fact that adapter-based MLM specialization of TOD-BERT – which prevents negative interference by design – brings slight performance gains (i.e., TOD-BERT-MLM-adapter vs. TOD-BERT; +0.8% for DST and +1.0% for RR) and is consistent with the concurrent findings of Qiu et al. (2021).

Overall, domain specialization with RS seems to be more robust than that via MLM-ing, with the two variants (RS-Class and RS-Contrast) exhibiting similar average performance across evaluation settings. This points to the importance of injecting both the knowledge of dialogic structure as well as domain knowledge for performance gains in TOD tasks in the domain of interest.

Interestingly, the gains from domain specialization are significantly more pronounced for *Taxi* than for other domains. We relate this to the proportion of the single-domain dialogs for a given domain in MULTIWOZ, which is by far the largest (24%, see Table 3.1)

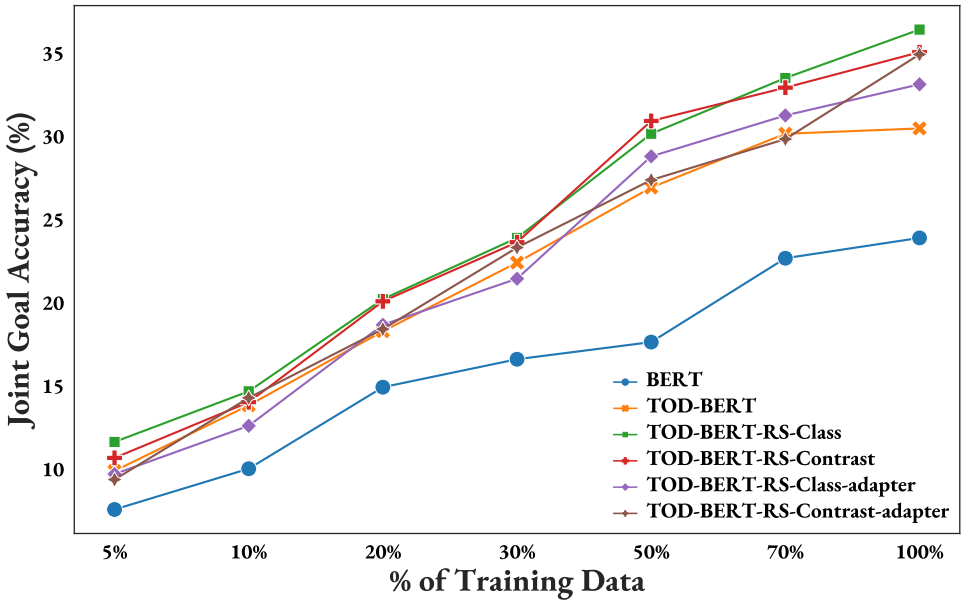


Figure 3.5: Sample efficiency of DS-TOD for DST: joint goal accuracy (%) for randomly sampled sub-portions (5%, 10%, 20%, 30%, 50%, 70%, and 100%) of the downstream training data from the *Taxi* domain.

for the *Taxi* domain. Consequently, successful specialization for that domain is *a priori* more likely to show substantial gains on MULTIWOZ (i.e., less multi-domain influence).

An encouraging finding is that, on average, adapter-based specialization yields similar gains as specialization via full fine-tuning: given that adapter fine-tuning is substantially more efficient, this holds the promise of more sustainable TOD.

Sample Efficiency. To further understand the effect of the injected domain-specific knowledge, we conduct an additional few-shot analysis (Figure 3.5) on DST (**C1**; § 2.4.1). To this end, we select the *Taxi* domain, since we witnessed the largest gains for that domain. We analyze the differences in performance between baseline and domain-specialized PLMs when they are exposed to downstream training portions of different sizes, ranging from 5% to 100% of the whole training dataset.[9] TOD-BERT retains a sizable performance gap over BERT for all settings, pointing to the power of dialogic pre-training. Importantly, for all dataset sizes, the performances of the domain-specialized variants of TOD-BERT-RS-{Class, Contrast} surpass the one of the non-specialized TOD-BERT. Even more interestingly, specialized variants exposed to only 50% of the DST training data manage to surpass the performance of TOD-BERT fine-tuned on all of the training data (100%). This suggests that self-supervised domain specialization has the potential to substantially reduce the amount of annotated TOD data required to reach some performance level.

[9]Note that 5% of the training data in the *Taxi* domain amounts to 83 dialogs.

Figure 3.6: Relative improvements (TOD-BERT-RS-Contrast vs. TOD-BERT) in cross-domain DST transfer.

Cross-Domain Transfer. MULTIWOZ domains are mutually quite related: some are similar, i.e., share vocabulary and slots (e.g., *Taxi* and *Train*) whereas others often appear together in a dialog (e.g., *Train* and *Hotel*; see Table 3.1 for the number of multi-domain MULTIWOZ dialogs). We thus next investigate whether intermediate training for one domain benefits other closely related domains (**C3**; § 2.4.3). To this end, we expose models specialized for one domain (e.g., *Taxi*) to downstream fine-tuning and evaluation in the other domain (e.g., *Restaurant*). Figure 3.6 summarizes the deltas in performance between the non-specialized TOD-BERT and TOD-BERT-RS-Contrast for all domain pairs. Encouragingly, the specialization for one domain seems to generally lead to downstream gains in related domains too: the gains are most prominent for pairs of domains that frequently co-occur in dialogs – *Hotel* pre-training for the *Restaurant* downstream (and vice versa) and *Taxi* pre-training for downstream tasks in the *Restaurant* and *Attraction* domains.

Multi-Domain Specialization. In many real-world scenarios, a single model needs to be able to handle multiple domains because (a) multi-domain dialogs exist, and (b) simultaneous deployment of multiple single-domain models may not be feasible. To simulate this scenario, we conduct an additional analysis, in which we concatenate dialogs from respective MULTIWOZ portions that cover concrete combinations of two or three domains. We choose three domain combinations with the largest number of multi-domain dialogs, namely the two largest 2-domain combinations and the largest 3-domain combination (Figure 3.7): *Hotel*+*Train*, *Attraction*+*Train*, and *Hotel*+*Taxi*+*Restaurant*.

As baselines, we report the performance of BERT and TOD-BERT fine-tuned on the respective multi-domain TOD training sets. We test the effect of multi-domain specialization in two variants: (1) *fully specialized model trained for multiple domains* (**Full-FT**): as

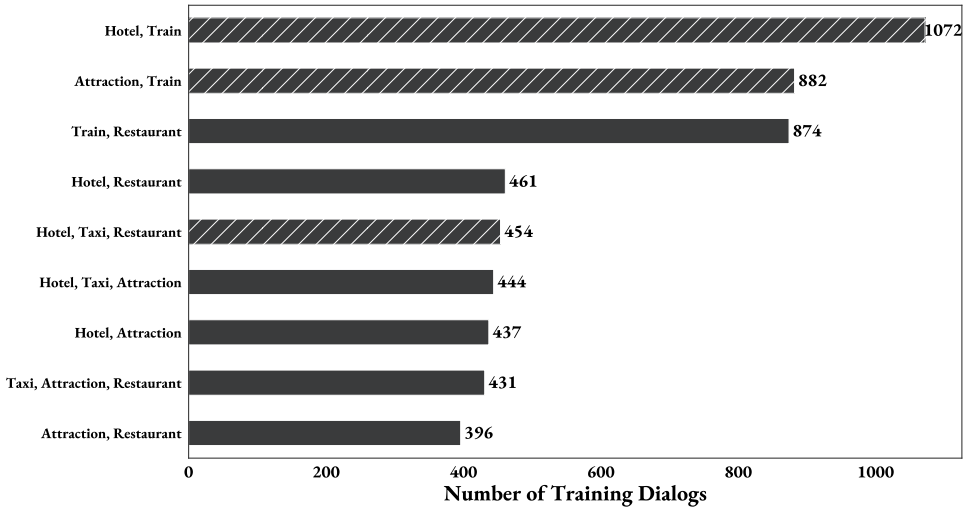


Figure 3.7: Number of dialogs in the MULTIWOZ 2.1 training dataset when joining several domains. Striped bars indicate the domain combinations used for multi-domain specialization.

Model	Specialization Method	Hotel+ Train	Attraction+ Train	Hotel+Taxi+ Restaurant
BERT	–	42.66	45.06	37.00
TOD-BERT	–	46.38	46.40	42.47
TOD-BERT-RS-Class	Full-FT	**47.39**	**47.33**	42.39
	Stacking	47.19	46.68	42.15
	Fusion	44.25	45.57	**44.02**

Table 3.6: DS-TOD performance on DST in multi-domain scenarios. We compare the fully multi-domain-specialized variant (Full-FT) of the TOD-BERT-RS-Class model with its variant that combines readily available single-domain adapters (Stacking and Fusion) on three multi-domain evaluation sets.

RS-Class has proven to be effective in our single-domain specialization experiments, we run RS-Class training on the concatenation of the selected domains from DOMAINREDDIT that correspond to the domains of the joint training sets. Accordingly, the training data is roughly twice (or three times) as big as that used for single-domain specialization; (2) *composition of single-domain adapters for multiple domains*: while for Full-FT, a new intermediate training is necessary for each domain combination, with adapter-based specialization we can simply combine the adapters of relevant domains in downstream fine-tuning. In this setup, we combine the single-domain adapters by sequentially stacking them (Pfeiffer et al., 2020) (**Stacking**) or by fusing them, i.e., interpolating between their outputs (Pfeiffer et al., 2021) (**Fusion**).

The multi-domain specialization results are shown in Table 3.6. Interestingly, combining single-domain adapters in downstream training (via Stacking or Fusion) performs *en par* with full-sized two-domain specialization on DOMAINREDDIT by means of RS-Class training. In contrast to TOD-BERT-RS-Class (Full-FT), which requires full retraining of the model on the unlabelled domain-specific corpora for each combination of the domains, combining single-domain adapters is much more efficient as it does not require any further intermediate domain training for domain combinations. In the 3-domain setup (*Hotel*+*Taxi*+*Restaurant*), the Fusion approach even outperforms the full 3-domain specialization (TOD-BERT-RS-Class Full-FT) by 2 points.

Overall, we find that the adapter compositions provide a simple and effective way to combine information from several domain-specialized adapters, removing the need for additional multi-domain specialization in the face of multi-domain dialogs downstream task (**C1**; § 2.4.1).

3.1.7 Conclusions

In this Section, we introduced **DS-TOD** – a novel framework for domain adaptation of PLMs for task-oriented dialog. Given a collection of in-domain dialogs, we extract domain terms and use them to filter in-domain dialogic corpora. Our experimental study, conducted across five domains of the MULTIWOZ dataset, demonstrates that domain specialization, especially by means of response selection objectives on the dialogic in-domain corpora, leads to consistent gains in TOD tasks: DST and RR.

Our findings reveal the benefits of task-agnostic domain-adaptive pre-training utilizing in-domain dialogic data and response selection objectives could bring substantial gains for TOD downstream tasks. Further, we proposed an adapter-based approach offering a viable solution for multi-domain scenarios. However, questions remain about the applicability of this approach to other domain-specific tasks (e.g., named-entity recognition) and its effectiveness in low-resource settings. This paves the way for investigating more effective and efficient task-agnostic domain adaptation strategies, aiming to catalyze further research for applying a single model for multi-domain usage.

3.2 Efficient Task-Agnostic Domain Adaptation

[*] In Section 3.1, we demonstrated the benefits of domain-adaptive pre-training for TOD, demonstrating how intermediate training of PLMs on domain-specific data leads to substantial gains for downstream TOD tasks. Additionally, we explored the use of *adapters* to enhance training efficiency and prevent catastrophic forgetting alleviated from *full fine-tuning* (i.e., updating all PLM parameters in domain adaptation phase). However, the adapter-based approaches require additional parameters for each layer and are criticized for their limited expressiveness. In this Section, we introduce **TADA**, a novel **T**ask-**A**gnostic **D**omain **A**daptation method which is modular, parameter-efficient, and thus, data-efficient. Within TADA, we retrain the embeddings to learn domain-aware input representations and tokenizers for the Transformer encoder, while freezing all other parameters of the model. Subsequently, task-specific fine-tuning is performed. We extend our approach by conducting experiments with meta-embeddings and newly introduced meta-tokenizers, resulting in a single model per task in multi-domain use cases. Our broad evaluation in 4 downstream tasks for 14 domains across single- and multi-domain setups and high- and low-resource scenarios (**C1**; § 2.4.1) reveals that TADA is an effective and efficient alternative to *full fine-tuning* and *adapters* for domain adaptation, while not introducing additional parameters or complex training steps (**C2**; § 2.4.2).

3.2.1 Introduction

Pre-trained language models (PLMs) (Radford et al., 2018; Devlin et al., 2019) utilizing the Transformer (Vaswani et al., 2017) have emerged as a key technology for achieving impressive gains in a wide variety of NLP tasks. However, these PLMs are trained on massive and heterogeneous corpora with a focus on generalizability without addressing particular domain-specific concerns. In practice, the absence of such domain-relevant information can severely hurt performance in downstream applications as shown in numerous studies (i.a., Zhu and Goldberg, 2009; Ruder and Plank, 2018; Friedrich et al., 2020).

To impart useful domain knowledge, two main methods of domain adaptation leveraging the Transformer have emerged: (1) *Massive pre-training from scratch* (Beltagy et al., 2019; Wu et al., 2020) relies on large-scale domain-specific corpora incorporating various self-supervised objectives during pre-training (see § 2.2). However, the extensive training process is time- and resource-inefficient, as it requires a large collection of (un)labeled domain-specialized corpora and massive computational power. (2) *Domain-adaptive intermediate pre-training* (Gururangan et al., 2020) is considered more light-weight (de-

[*] This Section is adapted from: **Chia-Chien Hung**, Lukas Lange, and Jannik Strötgen. 2023. TADA: Efficient Task-Agnostic Domain Adaptation for Transformers. In *Findings of the Association for Computational Linguistics (ACL 2023)*, pages 487–503, Toronto, Canada, July 2023. Association for Computational Linguistics.

tailed in § 2.3.1), as it requires only a small amount of in-domain data and fewer epochs continually training on the PLM from a previous checkpoint. However, *fully fine-tuning* the model in the adaptation phase (i.e., updating all PLM parameters) may result in catastrophic forgetting and interference (McCloskey and Cohen, 1989; Houlsby et al., 2019), in particular for longer iterations of adaptation. To overcome these limitations, alternatives such as *adapters* (Rebuffi et al., 2017; Houlsby et al., 2019), and *sparse fine-tuning* (Guo et al., 2021a; Ben Zaken et al., 2022) have been introduced. These approaches, however, are still parameter- and time-inefficient, as they either add additional parameters or require complex training steps and/or models.

In this Section, we propose **T**ask-**A**gnostic **D**omain **A**daptation for Transformers (TADA), a novel domain specialization framework. As depicted in Figure 3.8, it consists of two steps: (1) We conduct intermediate training of a pre-trained Transformer-based language model (e.g., BERT) on the unlabeled domain-specific text corpora in order to inject domain knowledge into the Transformer. Here, we *fix* the parameter weights of the encoder while updating only the weights of the embeddings (i.e., embedding-based domain-adaptive pre-training). As a result, we obtain domain-specialized embeddings for each domain with the *shared* encoder from the original PLM without adding further parameters for domain adaptation. (2) The obtained domain-specialized embeddings along with the encoder can then be fine-tuned for downstream tasks in single- or multi-domain scenarios (Lange et al., 2021b), where the latter is conducted with meta-embeddings (Coates and Bollegala, 2018; Kiela et al., 2018) and a novel meta-tokenization method for different tokenizers. An overview of the efficient domain-adaptive pre-training framework is illustrated in Figure 3.9.

Contributions. We advance the field of domain adaptation with the following contributions: (**i**) We propose a modular, parameter-efficient, and task-agnostic domain adaptation method (TADA) without introducing additional parameters for intermediate training of PLMs. (**ii**) We demonstrate the effectiveness of our specialization method on four heterogeneous downstream tasks – Dialog State Tracking (DST), Response Retrieval (RR), Named Entity Recognition (NER), and Natural Language Inference (NLI) across 14 domains. (**iii**) We propose modular domain specialization via meta-embeddings and show the advantages in multi-domain scenarios. (**iv**) We introduce the concept of meta-tokenization to combine sequences from different tokenizers in a single model and perform the first study on this promising topic. (**v**) The code and resources developed for TADA are publicly available.[10]

[10] https://github.com/boschresearch/TADA

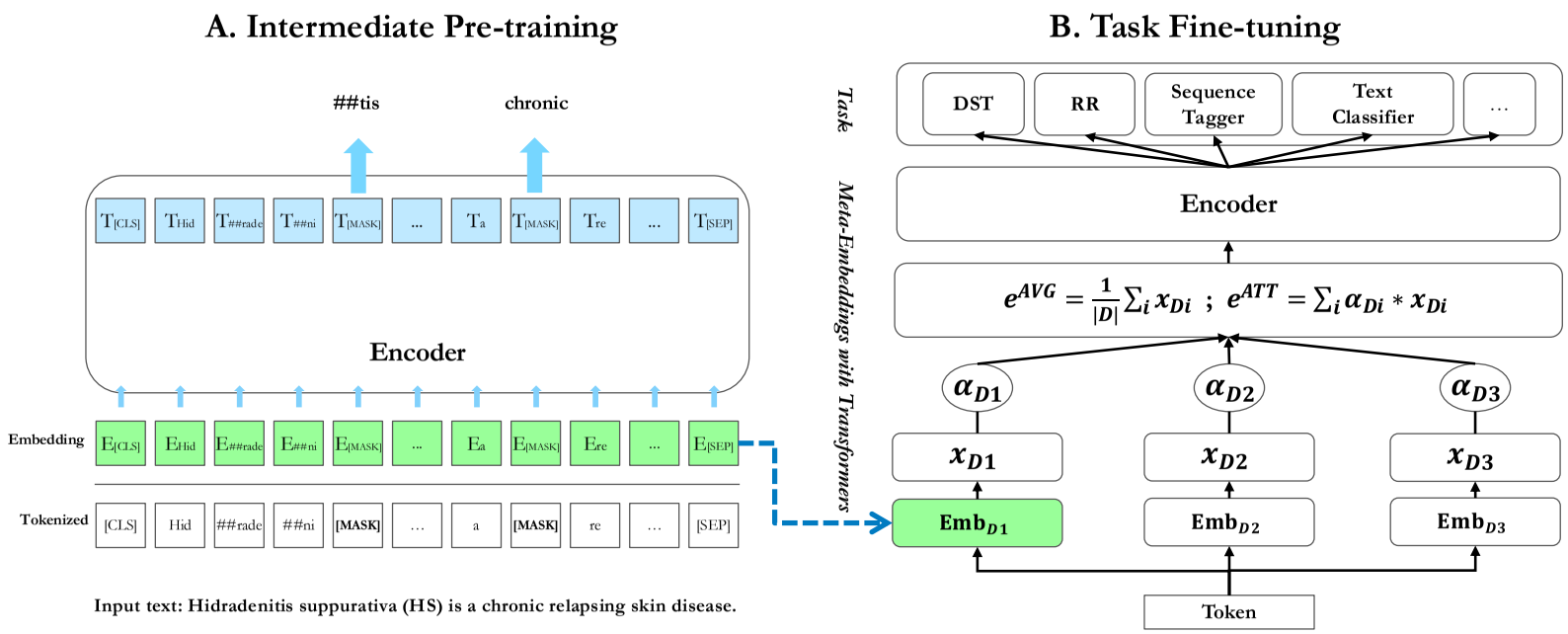


Figure 3.8: Overview of the TADA framework consisting of two steps. Part A: Domain specialization is performed via embedding-based domain-adaptive intermediate pre-training with Masked Language Modeling (MLM) objective on in-domain data. Part B: The domain-specialized embeddings are then fine-tuned for downstream tasks in single- or multi-domain scenarios with two meta-embeddings methods: average (AVG) and attention-based (ATT).

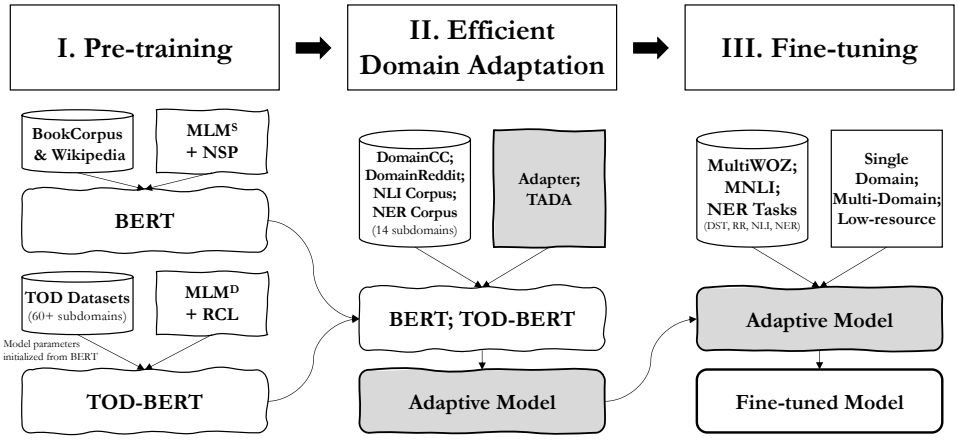


Figure 3.9: Overview of the efficient domain-adaptive pre-training framework. The framework involves a three-stage process: *Pre-training*, *Efficient Domain Adaptation*, *Fine-tuning*.

3.2.2 Related Work

Domain Adaptation. Domain adaptation is a type of transfer learning that aims to enable the trained model to be generalized into a specific domain of interest (Farahani et al., 2021). Recent studies have focused on neural unsupervised or self-supervised domain adaptation leveraging PLMs (Ramponi and Plank, 2020), which do not rely on large-scale labeled target domain data to acquire domain-specific knowledge. Gururangan et al. (2020) proposed domain-adaptive intermediate pre-training, continually training PLM on MLM with domain-relevant unlabeled data, leading to improvements in downstream tasks in both high- and low-resource setups. The proposed approach has been applied to a wide variety of tasks (Glavaš et al., 2020; Lewis et al., 2020b) across languages (Hung et al., 2023c), however, requires *fully* fine-tuning (i.e., update all PLM parameters) during domain adaptation, which can potentially result in catastrophic forgetting and negative interference (Houlsby et al., 2019; He et al., 2021).

Parameter-Efficient Training. Parameter-efficient methods for domain adaptation alleviate these problems. They have shown robust performance in low-resource and few-shot scenarios (Fu et al., 2023), where only a small portion of parameters are trained while the majority of parameters are frozen and shared across tasks. These lightweight alternatives are shown to be more stable than their corresponding fully fine-tuned counterparts and perform *on par with* or better than expensive fully fine-tuning setups, including *adapters*, *prompt-based fine-tuning*, and *sparse subnetworks*. *Adapters* (Rebuffi et al., 2017; Houlsby et al., 2019) are additional trainable neural modules injected into each layer of the

otherwise frozen PLM, including their variants (Pfeiffer et al., 2021), have been adopted in both single-domain (Bapna and Firat, 2019) and multi-domain (Hung et al., 2022a) scenarios. *Sparse subnetworks* (Hu et al., 2022; Ansell et al., 2022) reduce the number of training parameters by keeping only the most important ones, resulting in a more compact model that requires fewer parameters for fine-tuning. *Prompt-based fine-tuning* (Li and Liang, 2021; Lester et al., 2021) reduces the need for extensive fine-tuning with fewer training examples by adding prompts or cues to the input data. These approaches, however, are still parameter- and time-inefficient, as they add additional parameters, require complex training steps, are less intuitive to the expressiveness, or are limited to the multi-domain scenario for domain adaptation. A broader overview and discussion of recent domain adaptation methods in low-resource scenarios is given in the survey of Hedderich et al. (2021).

3.2.3 Methods for Task-Agnostic Domain Specialization

To inject domain-specific knowledge through domain-adaptive pre-training into PLMs, these models are trained on unlabeled in-domain text corpora. For this, we introduce a novel *embedding-based* intermediate training approach as an alternative to *full fine-tuning* and *adapters* (§ 3.2.3.1), and further study the effects of domain-specific tokenization (§ 3.2.3.2) (**C1**, **C2**; § 2.4.1, § 2.4.2). We then utilize multiple domain-specialized embeddings with our newly proposed meta-tokenizers and powerful meta-embeddings in multi-domain scenarios (§ 3.2.3.3 and § 3.2.3.4) (**C1**; § 2.4.1).

3.2.3.1 Domain Specialization

Fully fine-tuning the model requires adjusting all of the model's parameters, which can be undesirable due to time- and resource-inefficiency and can dramatically increase the risk of catastrophic forgetting of the previously acquired knowledge (McCloskey and Cohen, 1989; Ansell et al., 2022). To alleviate these issues, we propose a parameter-efficient approach without adding additional parameters during intermediate domain-specialized adaptation: we freeze most of the PLM parameters and only update the input embeddings weights of the first Transformer layer (i.e., the parameters of the embeddings layer) during MLM. With this, the model can learn domain-specific input representations while preserving acquired knowledge in the frozen parameters. As shown in Figure 3.8, the encoder parameters are fixed during intermediate training while only the embeddings layer parameters are updated.

As a result, after intermediate MLM, multiple embeddings specialized for different domains are all applicable with the *same* shared encoder. As these trained domain-specialized embeddings are easily *portable* to any downstream task, we experiment with their combination in multi-domain scenarios via meta-embeddings methods (Yin and Schütze, 2016; Kiela et al., 2018). We discuss this in more detail in § 3.2.3.3.

3.2.3.2 Domain-Specific Tokenization

Inspired by previous work on domain-specialized tokenizers and vocabularies for language model pre-training (Beltagy et al., 2019; Lee et al., 2019; Yang et al., 2020), we study the domain adaptation of tokenizers for PLMs and train domain-specialized variants with the standard Word Piece Algorithm (WPA) (Schuster and Nakajima, 2012) analogously to the BERT tokenizer. As a result, the domain-specialized tokenizers cover more in-domain terms compared to the original PLM tokenizers. In particular, this reduces the number of out-of-vocabulary tokens, i.e., words that have to be split into multiple subwords, whose embedding quality often does not match the quality of word-level representations (Hedderich et al., 2021).

3.2.3.3 Meta-Embeddings

Given n embeddings from different domains D, each domain would have an input representation $x_{Di} \in \mathbb{R}^E$, $1 \leq i \leq n$, where n is the number of domains and E is the dimension of the input embeddings. Here, we consider two variants: *averaging* (Coates and Bollegala, 2018) and *attention-based* meta-embeddings (Kiela et al., 2018).

Averaging merges all embeddings into one vector without training additional parameters by taking the unweighted average:

$$e^{AVG} = \frac{1}{n} \sum_i x_{Di} \tag{3.3}$$

In addition, a weighted average with dynamic attention weights α_{Di} can be used. For this, the attention weights are computed as follows:

$$\alpha_{Di} = \frac{exp(V \cdot tanh(W x_{Di}))}{\sum_{k=1}^{n} exp(V \cdot tanh(W x_{Dk}))}, \tag{3.4}$$

with $W \in \mathbb{R}^{H \times E}$ and $V \in \mathbb{R}^{1 \times H}$ being parameters that are randomly initialized and learned during training and H is the dimension of the attention vector which is a predefined hyperparameter.

The domain embeddings x_{Di} are then weighted using the learned attention weights α_{Di} into one representation vector:

$$e^{ATT} = \sum_i \alpha_{Di} \cdot x_{Di} \tag{3.5}$$

As *Averaging* simply merges all information into one vector, it cannot focus on valuable domain knowledge in specific embeddings. In contrast, the *attention-based* weighting allows for dynamic combinations of embeddings based on their importance depending on the current input token.

Domain Text : Acetaminophen is an analgesic drug
=> **TOK-1** : Ace #ta #mino #phen is an anal #gesic dr #ug (10 subwords)
=> **TOK-2** : Aceta #minophen is an anal #gesic drug (7 subwords)

Aggregation	TOK-1	TOK-2
SPACE	[Ace #ta #mino #phen] is an [anal #gesic] [dr #ug]	[Aceta #minophen] is an [anal #gesic] drug
DYNAMIC	[Ace #ta] [#mino #phen] is an anal #gesic [dr #ug]	Aceta #minophen is an anal #gesic drug
TRUNCATION	[Ace] [#mino] is an anal #gesic [dr]	Aceta #minophen is an anal #gesic drug

Table 3.7: Examples of our proposed aggregation approaches for meta-tokenization: SPACE, DYNAMIC, TRUNCATION, for a given text and two different tokenizers (TOK-1, TOK-2). The bottom of the table shows the results after aggregation. $[a\ b \ldots z]$ denotes the average of all embedding vectors corresponding to subword tokens a, b, ..., z.

As shown in related works, these meta-embeddings approaches suffered from critical mismatch issues when combining embeddings of different sizes and input granularities (e.g., character- and word-level embeddings) that could be addressed by learning additional mappings to the same dimensions on word-level to force all the input embeddings towards a common input space (Lange et al., 2021a).

Our proposed method prevents these issues by (a) keeping the input granularity fixed, which alleviates the need for learning additional mappings, and (b) locating all domain embeddings in the same space immediately after pre-training by freezing the subsequent Transformer layers. We compare the results of two variants in § 3.2.5. More information on meta-embeddings can be found in the survey of Bollegala and O' Neill (2022).

3.2.3.4 Meta-Tokenization for Meta-Embeddings

To utilize our domain-adapted tokenizers in a single model with meta-embeddings, we have to align different output sequences generated by each tokenizer for the same input. This is not straightforward due to mismatches in subword token boundaries and sequence lengths. We thus introduce three different aggregation methods to perform the meta-tokenization, and the examples for each method are shown in Table 3.7:

(a) SPACE: We split the input sequence on whitespaces into tokens and aggregate for each tokenizer all subword tokens corresponding to a particular token in the original sequence.

(b) DYNAMIC: The shortest sequence from all tokenizers is taken as a reference. Subwords from longer sequences are aggregated accordingly. This assumes that word-level knowledge is more useful than subword knowledge and that fewer word splitting is an indication of in-domain knowledge.

(c) TRUNCATION: This method is similar to the DYNAMIC aggregation, but it uses only the first subword for each token instead of computing the average when a token is split into more subwords.

Task	Dataset	Domain	Background	Train / Dev / Test	License†
DST, RR	MultiWOZ 2.1 (Eric et al., 2020)	Taxi	200 K	1,654 / 207 / 195	MIT
		Restaurant	200 K	3,813 / 438 / 437	
		Hotel	200 K	3,381 / 416 / 394	
		Train	200 K	3,103 / 484 / 494	
		Attraction	200 K	2,717 / 401 / 395	
NLI	MNLI (Williams et al., 2018)	Government	46.0 K	77,350 / 2,000 / 2,000	OANC
		Travel	47.4 K	77,350 / 2,000 / 2,000	OANC
		Slate	214.8 K	77,306 / 2,000 / 2,000	OANC
		Telephone	234.6 K	83,348 / 2,000 / 2,000	OANC
		Fiction	299.5 K	77,348 / 2,000 / 2,000	CC-BY-{3.0; SA-3.0}
NER	CoNLL (Tjong Kim Sang and De Meulder, 2003)	News	51.0 K	14,987 / 3,466 / 3,684	DUA
	I2B2-CLIN (Uzuner et al., 2011)	Clinical	299.9 K	13,052 / 3,263 / 27,625	DUA
	SEC (Salinas Alvarado et al., 2015)	Financial	4.8 K	825 / 207 / 443	CC-BY-3.0
	LITBANK (Bamman et al., 2019)	Fiction	299.5 K	5,548 / 1,388 / 2,973	CC-BY-4.0
	SOFC (Friedrich et al., 2020)	Science	300.1 K	489 / 123 / 263	CC-BY-4.0

Table 3.8: Overview of the selected datasets for 4 tasks (DST, RR, NLI, NER) on 14 domains. For each domain, we report the number of collected in-domain texts for domain-adaptive pre-training, as well as the size and license of the downstream dataset. All selected datasets are applicable for *commercial* usage. †License: Open American National Corpus (OANC), Direct Universal Access (DUA), Creative Commons Attribution Share-Alike (CC-BY-SA), Creative Commons Attribution International License (CC-BY).

Once the token and subword boundaries are determined, we retrieve the subword embeddings from the embedding layer corresponding to the tokenizer and perform the aggregation if necessary, in our case averaging all subword embeddings.

3.2.4 Experimental Setup

We demonstrate the effectiveness of our proposed task-agnostic domain adaptation (TADA) framework on 4 downstream tasks across 14 domains. This includes a detailed description of the downstream tasks, and an overview of the background datasets collected for domain-adaptive pre-training. We further provide details on the models, their hyperparameters, and the comparisons with baseline systems.

Tasks and Evaluation Measures. We evaluate our domain-specialized models and baselines on four prominent downstream tasks: Dialog State Tracking (DST), Response Retrieval (RR), Named Entity Recognition (NER), and Natural Language Inference (NLI) with five domains per task. Table 3.8 shows the statistics of all datasets.

(i) DST is cast as a multi-classification dialog task. As detailed defined in § 3.1.5, given a dialog history (sequence of utterances) and a predefined ontology, the goal is to predict the output state, i.e., (domain, slot, value) tuples (Wu et al., 2020) like (*restaurant*, *pricerange*, *expensive*). The standard joint goal accuracy is adopted as the evaluation measure: at each dialog turn, it compares the predicted dialog states against the annotated ground truth. The predicted state is considered accurate if and only if all the predicted slot values match exactly to the ground truth (see an example in § 3.1.5).

(ii) RR is a ranking task, relevant for retrieval-based task-oriented dialog systems (Henderson et al., 2019c; Wu et al., 2020). As outlined in § 3.1.5, given the dialog context, the model ranks N dataset utterances, including the *true response* to the context (i.e., the candidate set covers one *true* response and $N-1$ *false* responses). Following Henderson et al. (2019c), we report the recall at top rank given 99 randomly sampled false responses, denoted as $R_{100}@1$ (see an example in § 3.1.5).

(iii) NER is a sequence tagging task, aiming to detect named entities within a sentence by classifying each token into the entity type from a predefined set of categories (e.g., PERSON, ORGANIZATION) including a neutral type (O) for non-entities. Following prior work (Tjong Kim Sang and De Meulder, 2003; Nadeau and Sekine, 2007), we report the strict micro F_1 score. An example of NER task from clinical domain (Uzuner et al., 2011) followed BIO format[11] is demonstrated as below:

Tokens	*Subcentimeter*	*attenuation*	*lesion*	*within*	
	the	*lower*	*pole*	*of*	...
Tags	B-PROBLEM	I-PROBLEM	I-PROBLEM	I-PROBLEM	
	I-PROBLEM	I-PROBLEM	I-PROBLEM	O	...

(iv) NLI is a language understanding task testing the reasoning abilities of machine learning models beyond simple pattern recognition. The task is to determine if a *hypothesis* logically follows the relationship from a *premise*, inferred by ENTAILMENT (true), CONTRADICTION (false), or NEUTRAL (undefined). Following Williams et al. (2018), accuracy is reported as the evaluation measure. We show an example of NLI task from government domain (Williams et al., 2018) as following:

Premise	*6 See also Internal Control Management and Evaluation Tool (GAO-01-1008G, August 2001).*
Hypothesis	*The tool is not for Internal Control Management.*
Label	CONTRADICTION

Background Data for MLM-Specialization. We collect unlabeled background datasets from the original or related text sources to specialize our models with domain-adaptive pre-training (details are available in Table 3.9). For MLM training, we randomly sample up to 200K domain-specific sentences[12] (**C2**; § 2.4.2) and dynamically mask 15% of the subword tokens following Liu et al. (2019c).

[11] BIO format is a common tagging format for tagging tokens in a sequence tagging task. B, I, and O stand for BEGIN, INTERIOR, and OUTSIDE respectively.

[12] Except for four low-resource domains. For these, we randomly sample 44K (GOVERNMENT, TRAVEL, NEWS) and 4.5K (FINANCIAL) respectively.

Task	Domain	Background dataset	# Sentences
DST, RR	Taxi	DomainCC corpus from Hung et al. (2022a).	200 K
	Restaurant		200 K
	Hotel		200 K
	Train		200 K
	Attraction		200 K
NLI	Government	The respective part of the OANC corpus.	46.0 K
	Travel		47.4 K
	Slate		214.8 K
	Telephone		234.6 K
	Fiction	The BooksCorpus (Zhu et al., 2015), used as the pre-training data of BERT (Devlin et al., 2019).	299.5 K
NER	News	The Reuters news corpus in NLTK (`nltk.corpus.reuters`). Similar to the training data of CoNLL (Tjong Kim Sang and De Meulder, 2003).	51.0 K
	Clinical	Pubmed abstracts from clinical publications filtered following Lange et al. (2022).	299.9 K
	Financial	The financial phrase bank from Malo et al. (2014).	4.8 K
	Fiction	Same as NLI FICTION, described above.	299.5 K
	Science	Randomly sampled SemanticScholar abstracts from Biology (70%) and Computer Science (30%). Similar to the pre-training data of SciBERT (Beltagy et al., 2019).	300.1 K

Table 3.9: Overview of the background datasets and their sizes as reported in Table 3.8 in the background column. The background datasets are used to train domain-specific tokenizers and domain-adapted embeddings layer.

Models and Baselines. We experiment with the most widely used PLM: BERT (Devlin et al., 2019) for NER and NLI. For DST and RR as dialog tasks, we experiment with BERT and TOD-BERT (Wu et al., 2020) following the previous Section (§ 3.1.6; Hung et al. (2022a)) for comparing general- and task-specific PLMs.[13] We want to highlight that our proposed method can be easily applied to any existing PLM. As baselines, we report the performance of the non-specialized variants and compare them against (a) full fine-tuning (Gururangan et al., 2020), (b) adapter-based models (Houlsby et al., 2019), and (c) our domain-specialized PLM variants trained with TADA.

Hyperparameters and Optimization. During MLM training, we fix the maximum sequence length to 256 (DST, RR) and 128 (NER, NLI) subwords and do lowercasing. We train for 30 epochs in batches of 32 instances and search for the optimal learning rate among the following values: $\{5 \cdot 10^{-5}, 1 \cdot 10^{-5}, 1 \cdot 10^{-6}\}$. Early stopping is applied on the development set performance (patience: 3 epochs) and the cross-entropy loss is minimized using AdamW (Loshchilov and Hutter, 2019). For DST and RR, we follow the hyperparameter setup from Hung et al. (2022a). For NLI, we train for 3 epochs in batches of 32 instances. For NER, we train 10 epochs in batches of 8 instances. Both tasks use a fixed learning rate of $5 \cdot 10^{-5}$. Detailed computational information is in Appendix B.2.

[13] We use the pre-trained language models weights loaded from HuggingFace: `bert-base-uncased` (NLI, NER) and `bert-base-cased`, `TODBERT/TOD-BERT-JNT-V1` (RR, DST). More details about the models can be referred to § 2.2.1 and § 2.2.3.

Model	DST						RR					
	Taxi	Restaur.	Hotel	Train	Attract.	Avg.	Taxi	Restaur.	Hotel	Train	Attract.	Avg.
BERT	23.87	35.44	30.18	41.93	29.77	32.24	23.25	37.61	38.97	44.53	48.47	38.57
TOD-BERT	30.45	43.58	36.20	48.79	42.70	40.34	45.68	57.43	53.84	60.66	60.26	55.57
BERT (MLM-FULL)	23.74	37.09	32.77	40.96	36.66	34.24	31.37	53.08	45.41	51.66	52.23	46.75
TOD-BERT (MLM-FULL)	29.94	43.14	36.11	47.61	41.54	39.67	41.77	55.27	50.60	55.17	54.62	51.49
BERT (MLM-ADAPT)	22.52	40.49	31.90	42.17	35.05	34.43	32.84	44.01	39.15	38.43	45.05	39.90
TOD-BERT (MLM-ADAPT)	32.06	44.06	36.74	**48.84**	43.50	**41.04**	49.08	58.18	55.55	59.46	60.26	56.51
BERT (MLM-EMB)	22.39	31.26	25.75	41.00	34.02	30.88	40.89	54.24	47.30	52.18	56.50	50.22
TOD-BERT (MLM-EMB)	32.00	43.47	36.67	47.34	42.80	40.46	47.08	57.71	55.65	60.72	60.39	56.31
TOD-BERT (MLM-EMBTOK-S)	**33.03**	41.14	36.77	47.50	40.77	39.84	50.41	58.97	56.48	**62.63**	59.56	57.61
TOD-BERT (MLM-EMBTOK-X)	32.55	**44.60**	**36.92**	47.27	**43.58**	40.98	**50.77**	**60.40**	**56.87**	62.11	**60.89**	**58.21**

Model	NLI						NER					
	Govern.	Tele.	Fiction	Slate	Travel	Avg.	Financ.	Fiction	News	Clinical	Science	Avg.
BERT	79.07	78.18	76.63	73.40	77.33	76.92	**90.56**	72.09	90.04	85.91	78.23	83.44
BERT (MLM-FULL)	80.82	**81.43**	76.43	71.97	77.78	77.69	90.53	**72.33**	90.62	**86.18**	78.19	**83.57**
BERT (MLM-ADAPT)	75.58	73.70	72.33	67.11	72.42	72.23	76.62	63.82	89.17	80.64	61.65	74.38
BERT (MLM-EMB)	80.77	80.42	**79.27**	**73.50**	**77.94**	**78.38**	90.38	71.79	**90.67**	85.82	78.82	83.50
BERT (MLM-EMBTOK-S)	80.57	79.15	78.51	72.94	77.28	77.69	87.49	69.90	89.55	85.53	**79.39**	82.37
BERT (MLM-EMBTOK-X)	**81.08**	80.16	78.97	73.15	77.68	78.21	89.27	69.77	89.21	85.31	77.33	82.18

Table 3.10: Results of our single-domain models with domain-specialized embeddings and tokenizers on four downstream tasks (DST, RR, NLI, NER). The evaluation metrics include: joint goal accuracy (%) for DST, $R_{100}@1$ (%) for RR, accuracy (%) for NLI, and F_1 (%) for NER.

3.2.5 Evaluation Results

For each downstream task, we first conduct experiments in a single-domain scenario, i.e., training and testing on data from the same domain, to show the advantages of our proposed approach of task-agnostic domain-adaptive embedding-based pre-training and tokenizers (§ 3.2.5.1). We further consider the combination of domain-specialized embeddings with meta-embeddings variants (Coates and Bollegala, 2018; Kiela et al., 2018) in a multi-domain scenario, where we jointly train on data from all domains of the respective task (§ 3.2.5.2).

3.2.5.1 Single-Domain Evaluation

We report downstream performance for the single-domain scenario in Table 3.10, with each subtable being segmented into three parts: (1) at the top, we show baseline results (BERT, TOD-BERT) without any domain specialization; (2) in the middle, we show results of domain-specialized PLMs via full fine-tuning and the adapter-based approach; (3) the bottom of the table contains results of our proposed approach specializing only the embeddings and the domain-specific tokenization.

As discussed in § 3.1.6, in the TOD tasks of DST and RR, TOD-BERT demonstrates superior performance over BERT, attributed to its specialized training on conversational knowledge. However, when subjected to further domain-adaptive pre-training with full MLM training (MLM-FULL), TOD-BERT's performance decreases (i.e., -4% for RR and -0.8% for DST compared to TOD-BERT). This downturn is believed to result

from the negative interference of full MLM domain specialization: while TOD-BERT is being trained on domain data during intermediate pre-training, the model forgets the conversational knowledge obtained during the initial dialogic pre-training stage (Wu et al., 2020). The hypothesis is further supported by the observations for the adapter-based method which gains slight performance increases.

Our proposed embedding-based domain adaptation (MLM-EMB) yields similar performance gains as specialization with adapters for TOD-BERT on average. Inspired by previous work on domain-specialized subtokens for language model pre-training (Beltagy et al., 2019; Yang et al., 2020), we additionally train domain-specific tokenizers (MLM-EMBTOK) with the Word Piece Algorithm (WPA) (Schuster and Nakajima, 2012). The training corpora are either obtained from only background corpora (**S**) or from the combination of background and training set of each domain (**X**). Further, our domain-specialized tokenizers coupled with the embedding-based domain-adaptive pre-training exhibit similar average performance for DST and outperform the state-of-the-art adapters and all other methods for RR.

Similar findings are observed for NLI and NER. MLM-EMB compared to MLM-FULL results in +0.7% performance gains in NLI and reaches similar average gains in NER. Especially for NLI, the domain-specialized tokenizers (MLM-EMBTOK) are beneficial in combination with our domain-specialized embeddings, while having considerably fewer trainable parameters. Given that TADA is substantially more efficient and parameter-free (i.e., without adding extra parameters), this promises more sustainable domain-adaptive pre-training.

	DST						RR					
Model	**Taxi**	**Restaur.**	**Hotel**	**Train**	**Attract.**	**Avg.**	**Taxi**	**Restaur.**	**Hotel**	**Train**	**Attract.**	**Avg.**
BERT	29.10	39.92	36.67	47.63	42.32	39.13	44.87	51.98	49.11	50.15	54.81	50.18
TOD-BERT	34.65	44.24	39.54	51.66	44.24	42.87	50.99	61.53	56.09	58.94	62.76	58.06
BERT (MLM-FULL)	31.94	42.16	38.48	45.37	41.48	39.89	49.59	55.76	54.66	55.59	59.85	55.09
TOD-BERT (MLM-FULL)	32.26	45.70	39.51	51.31	45.92	42.94	53.51	64.44	59.22	62.14	**66.49**	61.16
(AVG) TOD-BERT (EMB+MLM-EMBs)	**37.65**	46.06	39.61	51.95	46.95	44.44	52.84	62.56	58.54	60.79	64.87	59.92
(ATT) TOD-BERT (EMB+MLM-EMBs)	35.13	46.86	40.73	51.10	44.76	43.72	53.06	63.18	56.94	60.45	64.13	59.55
(AVG) TOD-BERT (MLM-EMBs)	35.42	46.71	40.82	**52.34**	47.30	44.52	**55.20**	**64.58**	**60.39**	**62.84**	66.11	**61.82**
(ATT) TOD-BERT (MLM-EMBs)	37.35	**46.98**	**41.32**	51.92	**47.88**	**45.09**	53.73	64.00	59.89	61.54	65.05	60.84

	NLI						NER					
Model	**Govern.**	**Tele.**	**Fiction**	**Slate**	**Travel**	**Avg.**	**Financ.**	**Fiction**	**News**	**Clinical**	**Science**	**Avg.**
BERT	82.88	**82.10**	80.69	76.01	80.11	80.36	87.68	69.11	89.96	**85.76**	76.14	81.73
BERT (MLM-FULL)	83.29	81.79	81.11	76.32	79.66	80.43	88.71	**69.92**	89.69	85.61	80.03	82.79
(AVG) BERT (MLM-EMBs)	**83.80**	80.87	81.70	**77.60**	**81.30**	**81.05**	87.72	68.78	90.16	85.68	78.22	82.11
(ATT) BERT (MLM-EMBs)	83.50	81.64	**81.74**	76.68	80.36	80.78	**88.89**	69.05	**90.56**	85.43	**80.55**	**82.90**

Table 3.11: Results of our multi-domain models leveraging meta-embeddings on four downstream tasks (DST, RR, NLI, NER).

3.2.5.2 Multi-Domain Evaluation

In practice, a single model must be able to handle multiple domains because the deployment of multiple models may not be feasible (**C1**; § 2.4.1). To simulate a multi-domain setting, we utilize the domain-specialized embeddings from each domain (§ 3.2.5.1) and combine them with meta-embeddings (§ 3.2.3.3).

To train a single model for each task applicable to all domains, we concatenate the training sets of all domains for each task. As baselines for DST and RR, we report the performance of BERT and TOD-BERT and a version fine-tuned on the concatenated multi-domain training sets (MLM-FULL). We test the effect of multi-domain specialization in two variants: *averaging* (AVG) and *attention-based* (ATT) meta-embeddings. We conduct experiments to check whether including general-purpose embeddings from TOD-BERT (EMB+MLM-EMBs) is beneficial compared to the one without (MLM-EMBs). The results in Table 3.11 show that combining domain-specialized embeddings outperforms TOD-BERT in both tasks. In particular, averaging meta-embeddings performs better in RR while attention-based ones work better in DST by 3.8% and 2.2% compared to TOD-BERT, respectively. It is further suggested that combining only domain-specialized embeddings (i.e., without adding general-purpose embeddings) works better for both meta-embeddings variants.

These findings are confirmed by NLI and NER experiments. The meta-embeddings applied in our multi-domain scenarios outperform BERT by 0.7 points for NLI and 1.2 points for NER, respectively. An encouraging finding is that two domains (FINANCIAL, SCIENCE) with the smallest number of training resources benefit the most compared to other domains in NER task. Such few-shot settings are further investigated in § 3.2.6.1.

Overall, we find that the meta-embeddings provide a simple yet effective way to combine several domain-specialized embeddings, alleviating the need of deploying multiple models.

		Government		Telephone		Fiction		Slate		Travel		Avg.	
	Model	**1%**	**20%**	**1%**	**20%**	**1%**	**20%**	**1%**	**20%**	**1%**	**20%**	**1%**	**20%**
SD	BERT	57.62	75.21	49.20	74.45	43.76	72.90	46.70	67.71	54.05	71.55	50.27	72.36
	BERT (MLM-FULL)	**61.92**	76.07	**54.53**	75.07	49.32	**73.21**	45.81	67.26	56.56	72.50	53.63	72.82
	BERT (MLM-ADAPT)	42.88	67.93	41.27	65.80	38.12	59.53	38.91	54.71	40.74	65.89	40.38	62.78
	BERT (MLM-EMB)	61.66	**76.61**	49.86	**75.33**	48.35	72.22	**49.10**	**68.26**	**60.27**	**72.73**	**53.85**	**73.03**
	BERT (MLM-EMBTOK-X)	61.27	75.75	49.20	74.11	**49.74**	72.26	**49.10**	66.51	58.99	72.15	53.66	72.16
MD	BERT	69.56	79.49	64.80	77.72	61.53	76.84	61.43	72.64	66.40	**76.42**	64.74	76.62
	(AVG) BERT (MLM-EMBs)	70.13	**80.00**	64.39	78.28	**62.24**	76.94	**62.61**	71.61	**66.45**	76.21	65.16	76.61
	(ATT) BERT (MLM-EMBs)	**71.21**	79.90	**65.56**	**78.48**	61.33	**77.34**	61.99	**72.69**	66.24	76.32	**65.27**	**76.95**

Table 3.12: Few-shot learning results on NLI task for 1% and 20% of the training data size in single-domain (SD) and multi-domain (MD) scenarios. We report mean of 3 runs with different random seeds (for *brevity*), where the results with both mean and standard deviation are in Appendix B.3.

3.2.6 Analysis

To more precisely analyze the advantages of our proposed embedding-based domain-adaptive pre-training methods and tokenizers, we study the following: few-shot transfer capability (§ 3.2.6.1), the effect of domain-specialized tokenizers on subword tokens (§ 3.2.6.2), and the combinations of multiple domain-specialized tokenizers with meta-tokenizers in multi-domain scenarios (§ 3.2.6.3).

3.2.6.1 Few-Shot Learning

We report few-shot experiments in Table 3.12 using 1% and 20% of the training data for NLI. We run three experiments with different random seeds to reduce variance and report the mean and standard deviation for these limited task data scenarios. MLM-EMB on average outperforms MLM-FULL by 1% in the single-domain scenario, especially for SLATE and TRAVEL domains with the largest improvements (i.e., 3.3% and 2.7%, respectively). In contrast, the adapter-based models (MLM-ADAPT) perform worse in this few-shot setting. This demonstrates the negative interference (-10%) caused by the additional parameters that cannot be properly trained, given the scarcity of task data for fine-tuning. In multi-domain settings, attention-based meta-embeddings, on average, surpass the standard BERT model in both few-shot setups. Overall, these findings demonstrate the strength of our proposed embedding-based domain-adaptive pre-training in limited task data scenarios (**C1**; § 2.4.1).

3.2.6.2 Domain-Specific Tokenizers

To study whether domain-specialized tokenizers better represent the target domain, we select the development sets and count the number of words that are split into multiple

			DST and RR				
Model	**Taxi**	**Restaurant**	**Hotel**	**Train**	**Attraction**	**Avg.**	**Diff.**
TOK-O	856	1597	1530	1659	1310	1390.4	-
TOK-S	715	1338	951	951	946	1048.2	-24.6%
TOK-X	465	959	753	753	740	798.4	-42.6%
			NLI				
Model	**Government**	**Telephone**	**Fiction**	**Slate**	**Travel**	**Avg.**	**Diff.**
TOK-O	4095	4221	3379	5094	5883	4534.3	-
TOK-S	1874	3517	3568	3597	3685	3248.2	-28.4%
TOK-X	1873	3522	2426	3683	3984	3097.6	-31.7%
			NER				
Model	**Finance**	**Fiction**	**News**	**Clinical**	**Science**	**Avg.**	**Diff.**
TOK-O	397	1930	6357	5121	832	2927.4	-
TOK-S	695	1958	8526	3744	653	3115.2	+6.4%
TOK-X	600	1822	5818	2939	463	2328.4	-20.5%

Table 3.13: The number of words that have to be split into multiple tokens (>= subwords) for different tokenizers. The assumption is that the fewer the number of words split into subwords, the more effectively a tokenizer represents the target domain.

tokens for each tokenizer. The assumption is that the domain-specialized tokenizers allow for word-level segmentation, and thus, word-level embeddings, instead of fallbacks to lower-quality embeddings from multiple subword tokens.

We compare three different tokenizers for each setting: (a) TOK-O: original tokenizer from PLMs without domain specialization; (b) TOK-S: domain-specialized tokenizer trained on the in-domain background corpus; (c) TOK-X: domain-specialized tokenizer trained on the concatenated in-domain background corpus plus the downstream training set.

Table 3.13 shows the results on all four tasks averaged across domains. It is evident that TOK-X compared to TOK-O in general significantly reduces the number of tokens split into multiple subwords (-42.6% in DST, RR; -31.7% in NLI; -20.5% in NER). This indicates that the domain-specialized tokenizers cover more tokens on the word-level, and thus, convey more domain-specific information. For domains with smaller background datasets, e.g., FINANCIAL and NEWS, the tokenizers are not able to leverage more word-level information. For example, TOK-S that was trained on the background data performs worse in these domains, as the background data is too small and the models overfit on background data coming from a similar, but not equal source. Including the training corpora helps to avoid overfitting and/or shift the tokenizers towards the dataset word distribution, as TOK-X improves for both domains over TOK-S. The finding is well-aligned with the results in Table 3.10 (see § 3.2.5.1) and supports our hypothesis that word-level tokenization is beneficial (**C3**; § 2.4.3).

Model	DST	RR	NLI	NER
(AVG) BERT‡ (MLM-EMBs)	44.52	**61.82**	**81.05**	82.11
(ATT) BERT‡ (MLM-EMBs)	**45.09**	60.84	80.78	**82.90**
(AVG) BERT‡ (MLM-EMBTOKs-X) dynamic	42.16	59.87	79.10	70.73
(AVG) BERT‡ (MLM-EMBTOKs-X) space	41.57	58.54	79.51	70.63
(AVG) BERT‡ (MLM-EMBTOKs-X) truncation	40.26	58.07	79.47	66.66
(ATT) BERT‡ (MLM-EMBTOKs-X) dynamic	42.73	59.22	79.32	70.83
(ATT) BERT‡ (MLM-EMBTOKs-X) space	41.45	58.95	79.93	70.71
(ATT) BERT‡ (MLM-EMBTOKs-X) truncation	40.82	59.09	79.67	68.41

Table 3.14: Results of meta-tokenizers in multi-domain experiments with meta-embeddings. Here **bold** indicates the best performance and underline indicates the best-performing meta-tokenization aggregation method. ‡BERT variants: TOD-BERT (DST, RR) and BERT (NLI, NER).

3.2.6.3 Study on Meta-Tokenizers

In § 3.2.5.2, we experiment with multiple domain-specialized embeddings inside meta-embeddings. These embeddings are, however, based on the original tokenizers and not on the domain-specialized ones. While the latter are considered to contain more domain knowledge and achieve better downstream single-domain performance (§ 3.2.5.1), it is not straightforward to combine tokenized output by different tokenizers for the same input due to mismatches in subword boundaries and sequence lengths.

Therefore, we further conduct experiments with meta-tokenizers in the context of meta-embeddings setup following § 3.2.3.4. We compare the best multi-domain models with our proposed aggregation approaches. The averaged results across domains are shown in Table 3.14 (per-domain results are available in Appendix B.4). Overall, it is observed that the SPACE and DYNAMIC approaches work better than TRUNCATION. However, there is still a performance gap between using multiple embeddings sharing the same sequence from the original tokenizer compared to the domain-specialized tokenizers. Nonetheless, this study shows the general applicability of meta-tokenizers in PLMs and suggests future work toward leveraging the domain-specialized tokenizers in meta-embeddings.

3.2.7 Conclusions

In this Section, we introduced TADA – a novel task-agnostic domain adaptation method which is modular and parameter-efficient for pre-trained Transformer-based language models. We demonstrated the efficacy of TADA in 4 downstream tasks across 14 domains in both single- and multi-domain settings, as well as high- and low-resource scenarios. An in-depth analysis revealed the advantages of TADA in few-shot transfer and highlighted how our domain-specialized tokenizers take the domain vocabularies into account. We

conducted the first study on meta-tokenizers and showed their potential in combination with meta-embeddings in multi-domain applications. Our work points to multiple future directions, including advanced meta-tokenization methods and the applicability of TADA beyond the studied tasks in this Section.

In the next Chapter, we focus on *language adaptation*, which involves a multifaceted exploration. We create a robust multilingual multi-domain task-oriented dialog dataset, spanning four topologically diverse languages. Further, we introduce a novel language-adaptive pre-training framework utilizing PLMs for multilingual conversational specialization, aiming to facilitate cross-lingual transfer for arbitrary downstream task-oriented dialog tasks.

Chapter 4

Language Adaptation

"The difference between the almost right word and the right word is really a large matter – 'tis the difference between the lightning bug and the lightning."

Mark Twain
«A Letter to George Bainton»

[*] Language diversity is a fundamental aspect of human communication, reflecting the global variety of cultures, perspectives, and knowledge. Effective language adaptation in NLP is essential to dismantle language barriers, foster inclusivity, and ensure that technological advancements cater to a wide spectrum of users. Language adaptation empowers language models to comprehend diverse languages, enabling individuals around the world to access information, and communicate to enhance user experience. Besides, it also extends the research areas and applications to regions and communities that were previously underserved due to language constraints. Research on multi-domain task-oriented dialog (TOD) has predominantly focused on the *English* language, primarily due to the shortage of robust TOD datasets in other languages, preventing the systematic investigation of cross-lingual transfer for this crucial NLP application area. Acknowledging the paramount importance of mitigating language barriers for TOD, this Chapter introduces Multi^{2}WOZ, a new multilingual multi-domain TOD dataset, derived from the well-established English dataset MultiWOZ, that spans four typologically diverse languages: Chinese, German, Arabic, and Russian. In contrast to concurrent efforts (Zuo et al., 2021; Ding et al., 2022), Multi^{2}WOZ contains gold-standard dialogs in target languages that are directly comparable with development and test portions of the English dataset, enabling reliable and comparative estimates of cross-lingual transfer performance for TOD. We then introduce a novel adaptive pre-training framework for

[*] This Chapter is adapted from: **Chia-Chien Hung**, Anne Lauscher, Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. 2022. Multi^{2}WOZ: A Robust Multilingual Dataset and Conversational Pretraining for Task-Oriented Dialog. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2022)*, pages 3687–3703, Seattle, United States, July 2022. Association for Computational Linguistics.

multilingual conversational specialization of PLMs that aims to facilitate cross-lingual transfer for arbitrary downstream TOD tasks (i.e., task agnostic) (**C1**; § 2.4.1). Using such conversational PLMs specialized for concrete target languages, we systematically benchmark a number of zero-shot and few-shot cross-lingual transfer approaches on two standard TOD tasks: Dialog State Tracking (DST) and Response Retrieval (RR). Our experiments show that, in most setups, the best performance entails the combination of (i) conversational specialization in the target language, and (ii) few-shot transfer for the concrete TOD task. Most importantly, we show that our conversational specialization in the target language allows for an exceptionally *sample-efficient few-shot transfer* for downstream TOD tasks.

4.1 Introduction

Task-oriented dialog is arguably one of the most popular NLP application areas (Yan et al., 2017; Henderson et al., 2019c, *inter alia*), with more importance recently given to more realistic, and thus, multi-domain conversations (Budzianowski et al., 2018; Ramadan et al., 2018), in which users may handle more than one task during the conversation, e.g., booking a *taxi* and making a reservation at a *restaurant*. Unlike many other NLP tasks (e.g., Hu et al., 2020; Liang et al., 2020; Ponti et al., 2020, *inter alia*), the progress towards *multilingual multi-domain* TOD has been hindered by the lack of sufficiently large and high-quality datasets in languages other than English (Budzianowski et al., 2018; Zang et al., 2020) and more recently, Chinese (Zhu et al., 2020). This lack can be attributed to the fact that creating TOD datasets for new languages from scratch or via translation of English datasets is significantly more expensive and time-consuming than for most other NLP tasks. However, the absence of multilingual datasets that are comparable (i.e., aligned) across languages prevents a reliable estimate of effectiveness of cross-lingual transfer techniques in multi-domain TOD (Razumovskaia et al., 2022).

In order to address these research gaps, in this Chapter we introduce MULTI^{2}WOZ, a reliable and large multilingual evaluation benchmark for multi-domain task-oriented dialog, derived by translating the monolingual English-only MULTIWOZ data (Budzianowski et al., 2018; Eric et al., 2020) to four linguistically diverse major world languages, each with a different script: Arabic (AR), Chinese (ZH), German (DE), and Russian (RU).

Compared to the concurrent efforts that derive multilingual datasets from English MULTIWOZ (Zuo et al., 2021; Ding et al., 2022), our MULTI^{2}WOZ is: (1) much *larger* – we translate all dialogs from development and test portions of the English MULTIWOZ (in total 2,000 dialogs containing the total of 29.5K utterances); (2) much more *reliable* – complete dialogs, i.e., utterances as well as slot-values, have been manually translated (without resorting to error-prone heuristics), and the quality of translations has been validated through quality control steps; and (3) *parallel* – the same set of dialogs has been translated to all target languages, enabling the direct comparison of the performance of multilingual models and cross-lingual transfer approaches across languages.

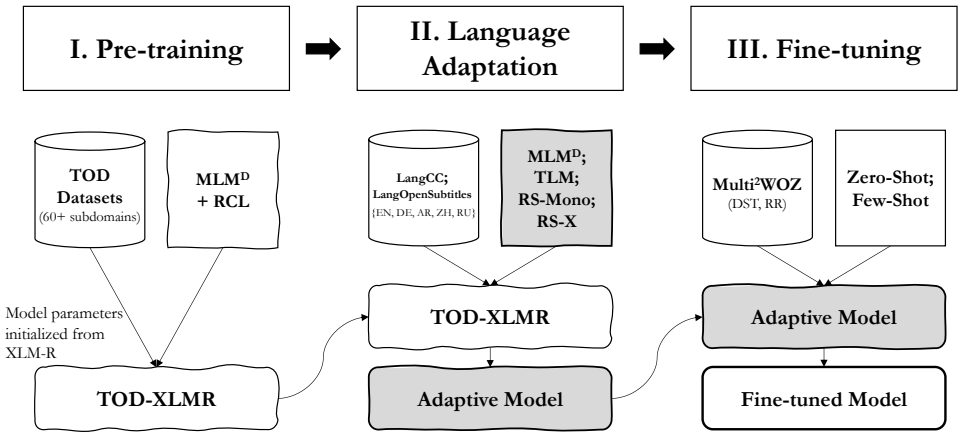


Figure 4.1: Overview of the language-adaptive pre-training framework. The framework involves a three-stage process: *Pre-training*, *Language Adaptation*, *Fine-tuning*. In this Chapter, intermediate training indicates the second (II) stage: language adaptation phase.

We then use MULTI^{2}WOZ to benchmark a range of state-of-the-art zero-shot and few-shot methods for cross-lingual transfer in two standard TOD tasks: Dialog State Tracking (DST) and Response Retrieval (RR). We propose a general framework for improving performance and sample-efficiency of cross-lingual transfer for TOD tasks. We first leverage the parallel conversational OpenSubtitles corpus (Lison and Tiedemann, 2016) to carry out a conversational specialization of a PLM for a given target language, irrespective of the downstream TOD task of interest. We then show that this intermediate conversational specialization in the target language (i) consistently improves the DST and RR performance in both zero-shot and few-shot transfer, and (ii) drastically improves sample-efficiency of few-shot transfer. An overview of language-adaptive pre-training framework is illustrated in Figure 4.1.

Contributions. We advance the field of multilingual multi-domain TOD with the following key contributions: (**i**) We introduce MULTI^{2}WOZ – a robust multilingual muti-domain TOD dataset spanning four typologically diverse languages, and conduct quality control step to ensure reliability. (**ii**) We propose a conversational specialized PLM which can be tailored for cross-lingual transfer in downstream TOD tasks. (**iii**) We examine different objectives for injecting language-specific knowledge into PLMs leveraging two collected corpora: we empirically compare Masked Language Modeling (MLM) applied on the "flat" language dataset LANGCC against Translation Language Modeling (TLM) (Conneau and Lample, 2019) with Response Selection (RS) objectives on dialogic LANGOPENSUBTITLES corpus. We demonstrate the effectiveness of cross-lingual transfer on two downstream TOD tasks – DST and RR. (**iv**) Our proposed language-adaptive pre-training framework for TOD consistently improves task performance in both zero-shot and few-shot transfer scenarios, with exceptionally notable sample efficiency in few-shot transfer for downstream TOD tasks.

4.2 Related Work

TOD Datasets. Research in task-oriented dialog has been, for a long time, limited by the existence of only monolingual English datasets. While earlier datasets focused on a single domain (Henderson et al., 2014a,b; Wen et al., 2017), the focus shifted towards the more realistic multi-domain task-oriented dialogs with the creation of the MULTIWOZ dataset (Budzianowski et al., 2018), which has been refined and improved in several iterations (Eric et al., 2020; Zang et al., 2020; Han et al., 2021). Due to the particularly high costs of creating TOD datasets (in comparison with other language understanding tasks) (Razumovskaia et al., 2022), only a handful of monolingual TOD datasets in languages other than English (Zhu et al., 2020) or bilingual TOD datasets have been created (Gunasekara et al., 2020; Lin et al., 2021). Mrkšić et al. (2017b) were the first to translate 600 dialogs from the single-domain WOZ 2.0 (Mrkšić et al., 2017a) to Italian and German. Concurrent work (Zuo et al., 2021; Ding et al., 2022), which we discuss in detail in § 4.3.2 and compare thoroughly against our MULTI^{2}WOZ, introduces the first multilingual multi-domain TOD datasets, created by translating portions of MULTIWOZ to several languages.

Language Specialization and Cross-lingual Transfer. Multilingual PLMs (e.g., mBERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020a)) are pre-trained on large general-purpose and massively multilingual corpora (over 100 languages) (see § 2.2.2). While this makes them versatile and widely applicable, it does lead to suboptimal representations for individual languages, a phenomenon commonly referred to as the "curse of multilinguality" (Conneau et al., 2020a). Therefore, one line of research focused on adapting (i.e., *specializing*) those models to particular languages (Lauscher et al., 2020; Pfeiffer et al., 2020). For example, Pfeiffer et al. (2020) propose a more computationally efficient approach for extending the model capacity for individual languages: this is done by augmenting the multilingual PLM with language-specific adapter modules. Glavaš et al. (2020) perform language adaptation through additional intermediate masked language modeling in the target languages with filtered text corpora, demonstrating substantial gains in downstream zero-shot cross-lingual transfer for hate speech and abusive language detection tasks. In a similar vein, Moghe et al. (2021) carry out intermediate fine-tuning of multilingual PLMs on parallel conversational datasets and demonstrate its effectiveness in zero-shot cross-lingual transfer for the DST task.

Lauscher et al. (2020) show that few-shot transfer, in which one additionally fine-tunes the PLM on a few labeled task-specific target-language instances leads to large improvements for many task-and-language combinations, and that labelling a few target-language examples is more viable than further LM-specialization for languages of interest under strict zero-shot conditions. This finding is also corroborated in our work for two TOD tasks.

Language	Family	Script	Example
German (DE)	Indo-European (Germanic)	Alphabet (Latin)	Hallo
Russian (RU)	Indo-European (Slavic)	Alphabet (Cyrillic)	Привет (Preevyet)
Chinese (ZH)	Sino-Tibetan	Logographic (Hanzi)	你好 (nǐ hǎo)
Arabic (AR)	Semitic	Abjad	مرحبا (Marhaba)

Figure 4.2: Languages selected for constructing MULTI^{2}WOZ.

4.3 Multi^{2}WOZ

In this Section we describe the construction of the MULTI^{2}WOZ dataset, providing also details on inter-translator reliability. We then discuss two concurrent efforts in creating multilingual TOD datasets from MULTIWOZ and their properties, and emphasize the aspects that make our MULTI^{2}WOZ a more reliable and useful benchmark for evaluating cross-lingual transfer for TOD.

4.3.1 Dataset Creation

Language Selection. We translate all 2,000 dialogs from the development and test portions of the English MULTIWOZ 2.1 (Eric et al., 2020) dataset to German (DE), Russian (RU), Chinese (ZH), and Arabic (AR). We selected the target languages based on the following criteria: (1) linguistic diversity (DE and RU belong to different Indo-European subfamilies – Germanic and Slavic, respectively; ZH is a Sino-Tibetan language and AR Semitic), (2) diversity of scripts (DE and RU use Latin and Cyrillic scripts, respectively, both *alphabet* scripts; AR script represents the *Abjad* script type, whereas the ZH Hanzi script belongs to *logographic* scripts), (3) number of native speakers (all four are in the top 20 most-spoken world languages), and (4) our access to native and fluent speakers of those languages who are proficient in English. The selected languages with language families and script types are displayed in Figure 4.2.

Two-Step Translation. Following the well-established practice, we carried out a two-phase translation of the English data: (1) we started with an *automatic translation* of the dialogs – utterances as well as the annotated slot values – followed by (2) the *manual post-editing* of the translations. We first automatically translated all utterances and slot values from the development and test dialogs from the MULTIWOZ 2.1 (Eric et al., 2020) (1,000 dialogs in each portion; 14,748 and 14,744 utterances, respectively) to our four target languages, using Google Translate.[1] We then hired two native speakers of each target language,[2] all with a University degree and fluent in English, to post-edit the (non-

[1] Relying on its Python API: `https://pypi.org/project/googletrans`

[2] In order to reduce the translation costs, we initially attempted to post-edit the translations via crowdsourcing. We tried this for Russian using the popular platform Toloka (`toloka.yandex.com`); however, the translation quality remained unsatisfactory even after several post-editing rounds.

	Utterance	Value for "attraction-name"
Original	*No hold off on booking for now.* *Can you help me find an attraction called cineworld cinema?*	*cineworld cinema*
Automatic Translation	目前暂无预订。您能帮我找到一个名为*cineworld Cinema*的景点吗?	*Cineworld*电影
Manual Correction	目前暂无预订。您能帮我找到一个名为电影世界电影院的景点吗?	电影世界电影院

Table 4.1: Example utterance (from the dialog MUL0484) with a value for a slot (*"attraction-name"*). We show the original English text, the automatic translation to Chinese and the final translation after manual post-editing.

overlapping sets of) automatic translations, i.e., fix the errors in automatic translations of utterances as well as slot values.

Since we carried out the automatic translation of the utterances independently of the automatic translation of the slot values, the translators were instructed to pay special attention to the alignment between each translated utterance and translations of slot value annotations for that utterance. We show an example utterance with associated slot values after the automatic translation and manual post-editing in Table 4.1.

Quality Control. In order to reduce the translation costs, our human post-editors worked on disjoint sets of dialogs. Because of this, our annotation process contained an additional quality assurance step. Two new annotators for each target language judged the correctness of the translations on the random sample of 200 dialogs (10% of all translated dialogs, 100 from the development and test portion each), containing 2,962 utterances in total. The annotators had to independently answer the following questions for each translated utterance from the sample: (1) *Is the utterance translation acceptable?* and (2) *Do the translated slot values match the translated utterance?* On average, across all target languages, both quality annotators for the respective language answered affirmatively to both questions for 99% of all utterances. Adjusting for chance agreement, we measured the Inter-Annotator Agreement (IAA) in terms of Cohen's κ (Cohen, 1960), observing the almost perfect agreement[3] of $\kappa = 0.824$ for the development set and $\kappa = 0.838$ for test set. The annotation guidelines for *manual post-editing* and *quality control* utilized during the creation of MULTI^{2}WOZ are elaborated in Appendix C.1 and Appendix C.2.

Annotation Duration and Cost. In total, we hired 16 annotators, four for each of our four target languages: two for post-editing and two for quality assessment. The overall effort spanned almost full 5 months (from July to November 2021), and amounted to 1,083 person-hours. With the remuneration rate of 16 $/h, the total cost of creating MULTI^{2}WOZ was $17,328.

[3]According to Landis and Koch (1977), if $\kappa \geq 0.81$.

4.3.2 Comparison with Concurrent Work

Two concurrent works also derive multilingual datasets from MULTIWOZ (Zuo et al., 2021; Ding et al., 2022), with different strategies and properties, discussed in what follows.

GLOBALWOZ (Ding et al., 2022) encompasses Chinese, Indonesian, and Spanish datasets. The authors first create *templates* from dialog utterances by replacing slot-value strings in the utterances with the slot type and value index (e.g., *"...and the post code is cb238el"* becomes the template *"...and the post code is* `[attraction-postcode-1]`*"*. They then *automatically* translate all templates to the target languages. Next, they select a subset of 500 test set dialogs for human post-editing with the following heuristic: dialogs for which the sum of corpus-level frequencies of their constitutive 4-grams (normalized with the dialog length) is the largest.[4] Since this selection step is independent for each language, each GLOBALWOZ portion contains translations of a different subset of English dialogs: this prevents any direct comparison of downstream TOD performance across languages. Even more problematically, the selection heuristic directly reduces linguistic diversity of dialogs chosen for the test set of each language, as it favors the dialogs that contain the same globally most frequent 4-grams. Due to this artificial homogeneity of its test sets, GLOBALWOZ is very likely to overestimate downstream TOD performance for target languages.

Unlike GLOBALWOZ, ALLWOZ (Zuo et al., 2021) does automatic translation of a *fixed small* subset of MULTIWOZ plus post-editing in seven target languages. However, it encompasses only 100 dialogs and 1,476 turns; as such, it is arguably too small to draw strong conclusions about the performance of cross-lingual transfer methods. Its usefulness in joint domain and language transfer evaluations is especially doubtful, since it covers individual MULTIWOZ domains with an extremely small number of dialogs (e.g., only 13 for the *Taxi* domain). Finally, neither Ding et al. (2022) nor Zuo et al. (2021) provide any estimates of the quality of their final datasets nor do they report their annotation costs.

In contrast to GLOBALWOZ, MULTI^{2}WOZ is a parallel corpus – with the exact same set of dialogs translated to all four target languages; as such it directly enables performance comparisons across the target languages. Further, containing translations of *all* dev and test dialogs from MULTIWOZ (i.e., avoiding sampling heuristics), MULTI^{2}WOZ does not introduce any confounding factors that would distort estimates of cross-lingual transfer performance in downstream TOD tasks. Finally, MULTI^{2}WOZ is 20 times larger (per language) than ALLWOZ: experiments on MULTI^{2}WOZ are thus much more likely to yield conclusive findings.

[4] Interestingly, the authors do not provide any motivation or intuition for this heuristic. It is also worth noting that they count the 4-gram frequencies, upon which the selection of the dialogs for post-editing depends, on the noisy automatic translations.

4.4 Cross-lingual Transfer for TOD

The parallel nature and sufficient size of MULTI^{2}WOZ allow us to benchmark and compare a number of established and novel cross-lingual transfer methods for TOD. In particular, (1) we first inject general conversational TOD knowledge into XLM-RoBERTa (XLM-R; Conneau et al., 2020a), yielding TOD-XLMR (§ 4.4.1); (2) we then propose several variants for conversational specialization of TOD-XLMR for target languages, better suited for transfer in downstream TOD tasks (§ 4.4.2); (3) we investigate zero-shot and few-shot transfer for two TOD tasks: DST and RR (§ 4.4.3).

4.4.1 TOD-XLMR: A Multilingual TOD Model

Recently, Wu et al. (2020) demonstrated that specializing BERT (Devlin et al., 2019) on conversational data by means of additional pre-training via a combination of Masked Language Modeling (MLM) and Response Selection (RS) objectives yields improvements in downstream TOD tasks. Following these findings, we first (propose to) conversationally specialize XLM-R (Conneau et al., 2020a), a state-of-the-art multilingual PLM covering 100 languages, in the same manner: applying the RS and MLM objectives on the same English conversational corpus consisting of nine human-human multi-turn TOD datasets (see § 2.2.3 for more details). As a result, we obtain TOD-XLMR – a massively multilingual PLM specialized for task-oriented conversations. Note that TOD-XLMR is not yet specialized (i.e., fine-tuned) for any concrete TOD task (e.g., DST or Response Generation). Rather, it is enriched with general task-oriented conversational knowledge (in English), presumed to be beneficial for a wide variety of TOD tasks.

4.4.2 Target-Language Specialization

TOD-XLMR has been conversationally specialized only in English data. We next hypothesize that a further conversational specialization for a concrete target language X can improve the transfer EN→X for all downstream TOD tasks. Accordingly, similar to Moghe et al. (2021), we investigate several intermediate training procedures that further conversationally specialize TOD-XLMR for the target language X (or jointly for EN and X). For this purpose, we (i) compile target-language-specific as well as cross-lingual corpora from the CCNet (Wenzek et al., 2020) and OpenSubtitles (Lison and Tiedemann, 2016) datasets, and (ii) experiment with different monolingual, bilingual, and cross-lingual training procedures. Here, we propose a novel cross-lingual response selection (RS) objective and demonstrate its effectiveness in cross-lingual transfer for downstream TOD tasks.

Training Corpora. Two types of data are collected for language specialization: (i) LANGCC as *"flat" corpora* (i.e., without any conversational structure): we simply randomly sample 100K sentences for each language from the respective monolingual portion of CCNET (we denote with *Mono-CC* the individual 100K-sentence portions of each language; with *Bi-CC* the concatenation of the English and each of target language *Mono-CCs*, and with *Multi-CC* the concatenation of all five Mono-CC portions); (ii) LANGOPENSUBTITLES as *parallel dialogs corpora* (in EN and target language X) from OpenSubtitles (OS): OS is a parallel conversational corpus spanning 60 languages, compiled from subtitles of movies and TV series. We leverage the parallel OS dialogs to create two different cross-lingual specialization objectives, as described next.

Training Objectives. We directly use the LANGCC portions (Mono-CC, Bi-CC, and Multi-CC) for standard **MLM** training (see § 3.1.4.1). We then leverage the parallel OS dialogs for two training objectives. First, we carry out Translation Language Modeling (**TLM**) (Conneau and Lample, 2019) on the synthetic dialogs which we obtain by interleaving K randomly selected English utterances with their respective target language translations; we then (as with MLM), dynamically mask 15% of tokens of such interleaved dialogs (Liu et al., 2019c); we vary the size of the context the model can see when predicting missing tokens by randomly selecting K (between 2 and 15) for each instance. Second, we use LANGOPENSUBTITLES to create instances for both monolingual and cross-lingual Response Selection (RS) training. RS is a simple binary classification task in which for a given pair of a *context* (one or more consecutive utterances) and *response* (a single utterance), the model has to predict whether the response utterance immediately follows the context (i.e., it is a *true* response) or not (i.e., it is a *false* response). RS pre-training (see § 3.1.4.1) has been proven beneficial for downstream TOD in monolingual English setups (Mehri et al., 2019; Henderson et al., 2019c, 2020; Hung et al., 2022a).

In this work, we leverage the parallel LANGOPENSUBTITLES data to introduce the cross-lingual RS objective, where the context and the response utterance are not in the same language. In our experiments, we carry out both (i) monolingual RS training in the target language (i.e., both the context and response utterance are, e.g., in Chinese), denoted **RS-Mono**, and (ii) cross-lingual RS between English (as the source language in downstream TOD tasks) and the target language, denoted **RS-X**. We create *hard RS negatives*, by coupling contexts with non-immediate responses from the same movie or episode (same `imdbID`), as well as *easy negatives* by randomly sampling $m \in \{1, 2, 3\}$ responses from a different movie of series episode (i.e., different `imdbID`). Hard negatives encourage the model to reason beyond simple lexical cues. Examples of training instances for OpenSubtitles-based training (for EN-ZH) are shown in Table 4.2.

	EN Subtitle	ZH Subtitle	
	- Professor Hall. - Yes. - I think your theory may be correct. - Walk with me. Just a few weeks ago, I monitored the strongest hurricane on record. The hail, the tornados, it all fits. Can your model factor in storm scenarios?	-霍尔教授 -是的 -我认为你的理论正确 -跟我来 上周我观测到史上最大的飓风 雹暴和龙卷风也符合你的理论 你能预测暴风雨的形成吗?	
Translation LM (TLM)	- Professor Hall. - Yes. - I think your theory may be [MASK]. - Walk with...-霍尔教授 -是的 -我认为你的[MASK]正确...		
Response Selection (RS)	*Context:* 上周我观测到史上最大的飓风 雹暴和龙卷风也符合你的理论	**Monolingual (RS-Mono)** *True Response:* 你能预测暴风雨的形成吗? *False Response:* 你有彼得的电脑断层扫描吗?	**Cross-lingual (RS-X)** *True Response:* Can your model factor in storm scenarios? *False Response:* Do you have Peter's CT scan results?

Table 4.2: Examples of training instances from LANGOPENSUBTITLES for conversational specialization for the target language created from OpenSubtitles (OS). Top row: an example of a dialog created from OS, parallel in English and Chinese. Below are training examples for different training objectives: (1) *Translation Language Modeling* (**TLM**) on the interleaved English-Chinese parallel utterances; (2) two variants of *Response Selection* (RS) – (a) monolingual in the target language (**RS-Mono**) and (b) cross-lingual (**RS-X**).

4.4.3 Downstream Cross-lingual Transfer

Finally, we fine-tune the various variants of TOD-XLMR, obtained through the above-described specialization (i.e., intermediate training) procedures, for two downstream TOD tasks (DST and RR) and examine their cross-lingual transfer performance. We cover two cross-lingual transfer scenarios: (1) *zero-shot transfer* in which we only fine-tune the models on the English training portion of MULTIWOZ and evaluate their performance on the MULTI^{2}WOZ test data of our four target languages; and (2) *few-shot transfer* in which we sequentially first fine-tune the models on the English training data and then on the small number of dialogs from the development set of MULTI^{2}WOZ, in similar vein to Lauscher et al. (2020). In order to determine the effect of our conversational target language specialization (§ 4.4.2) on the downstream sample efficiency, we run few-shot experiments with different numbers of target language training dialogs, ranging from 1% to 100% of the size of MULTI^{2}WOZ development portions.

4.5 Experimental Setup

Evaluation Tasks and Measures. We evaluate different multilingual conversational PLMs in cross-lingual transfer (zero-shot and few-shot) for two prominent TOD tasks: Dialog State Tracking (DST) and Response Retrieval (RR).

DST is commonly cast as a multi-class classification task, where given a predefined ontology and dialog history (a sequence of utterances), the model has to predict the output state, i.e., *(domain, slot, value)* tuples (Wu et al., 2020).[5] We adopt the standard *joint goal accuracy* as the evaluation measure: at each dialog turn, it compares the predicted dialog states against the manually annotated ground truth, which contains slot values for

[5]The model is required to predict slot values for each *(domain, slot)* pair at each dialog turn.

all the *(domain, slot)* candidate pairs. A prediction is considered correct if and only if all predicted slot values exactly match the ground truth (see an example in § 3.1.5).

RR is a ranking task that is well-aligned with the RS objective and relevant for retrieval-based TOD systems (Wu et al., 2017; Henderson et al., 2019c): given the dialog context, the model ranks N dataset utterances, including the *true response* to the context (i.e., the candidate set includes the one *true* response and N-1 *false* responses). We follow Henderson et al. (2020) and report the results for $N = 100$, i.e., the evaluation measure is recall at the top 1 rank given 99 randomly sampled false responses, denoted as $\text{R}_{100}@1$ (see an example in § 3.1.5).

Models and Baselines. We briefly summarize the models that we compare in zero-shot and few-shot cross-lingual transfer for DST and RR. As baselines, we report the performance of the vanilla multilingual PLM: XLM-R (Conneau et al., 2020a),[6] and its variant further trained on the English TOD data from Wu et al. (2020): TOD-XLMR (§ 4.4.1). Comparison between XLM-R and TOD-XLMR quantifies the effect of conversational English pre-training on downstream TOD performance, much like the comparison between BERT and TOD-BERT done by Wu et al. (2020); however, here we extend the comparison to cross-lingual transfer setups. We then compare the baselines against a series of our target language-specialized variants, obtained via intermediate training on LangCC (Mono-CC, Bi-CC, and Multi-CC) by means of MLM, and on LangOpenSubtitles jointly via TLM and RS (RS-X or RS-Mono) objectives (see § 4.4.2).

Hyperparameters and Optimization. For training TOD-XLMR (§ 4.4.1), we select the effective batch size of 8. In target-language-specific intermediate training (§ 4.4.2), we fix the maximum sequence length to 256 subword tokens; for RS objectives, we limit the context and response to 128 tokens each. We train for 30 epochs in batches of size 16 for MLM/TLM, and 32 for RS. We search for the optimal learning rate among the following values: $\{10^{-4}, 10^{-5}, 10^{-6}\}$. We apply early stopping based on development set performance (patience: 3 epochs for MLM/TLM, 10 epochs for RS). In downstream fine-tuning, we train in batches of 6 (DST) and 24 instances (RR) with the initial learning rate fixed to $5 \cdot 10^{-5}$. We also apply early stopping (patience: 10 epochs) based on the development set performance, training maximally for 300 epochs in zero-shot setups, and for 15 epochs in target-language few-shot training. In all experiments, we use AdamW (Loshchilov and Hutter, 2019) as the optimization algorithm.

[6] We use the pre-trained language model weights loaded from HuggingFace: `xlm-roberta-base`. More details about the model can be referred to § 2.2.2.

Model	DE	AR	ZH	RU	Avg.
w/o intermediate specialization					
XLM-R	1.41	1.15	1.35	1.40	1.33
TOD-XLMR	1.74	1.53	1.75	2.16	1.80
with conversational target-language specialization					
MLM on Mono-CC	3.57	2.71	3.34	5.17	3.70
Bi-CC	3.66	2.17	2.73	3.73	3.07
Multi-CC	3.65	2.35	2.06	5.39	3.36
TLM on OS	7.80	2.43	3.95	6.03	5.05
TLM + RS-X on OS	**7.84**	**3.12**	4.14	6.13	5.31
TLM + RS-Mono on OS	7.67	2.85	**4.47**	**6.57**	**5.39**

Table 4.3: Performance of multilingual conversational models in zero-shot cross-lingual transfer for Dialog State Tracking (DST) on MULTI^{2}WOZ, with joint goal accuracy (%) as the evaluation metric. Reference English DST performance of TOD-XLMR: 47.86%.

4.6 Results and Discussion

We now present and discuss the downstream cross-lingual transfer results on MULTI^{2}WOZ for DST and RR in two different transfer setups: zero-shot transfer and few-shot transfer (**C1**; § 2.4.1).

4.6.1 Zero-Shot Transfer

Dialog State Tracking (DST). Table 4.3 summarizes zero-shot cross-lingual transfer performance for DST. First, we note that the transfer performance of all models for all four target languages is extremely low, drastically lower than the reference English DST performance of TOD-XLMR, which stands at 47.9%. These massive performance drops, stemming from cross-lingual transfer are in line with findings from concurrent work (Zuo et al., 2021; Ding et al., 2022) and suggest that reliable cross-lingual transfer for DST is much more difficult to achieve than for most other language understanding tasks (Hu et al., 2020; Ponti et al., 2020).

Despite low performance across the board, we do note a few emerging and consistent patterns. First, TOD-XLMR slightly but consistently outperforms the vanilla XLM-R, indicating that *conversational* English pre-training brings marginal gains. All of our proposed models from § 4.4.2 (the lower part of Table 4.3) substantially outperform TOD-XLMR, proving that intermediate conversational specialization for the target language brings gains, irrespective of the training objective.

Expectedly, TLM and RS training on parallel LANGOPENSUBTITLES data brings substantially larger gains than MLM-ing on LANGCC: flat monolingual target-language corpora (Mono-CC) or simple concatenations of corpora from two (Bi-CC) or more languages (Multi-CC). German and Arabic seem to benefit slightly more from the cross-

Model	DE	AR	ZH	RU	Avg.
w/o intermediate specialization					
TOD-XLMR	3.3	2.9	1.9	2.7	2.7
with conversational target-language specialization					
MLM on Mono-CC	22.9	25.5	24.5	33.4	26.6
TLM on OS	**44.4**	30.3	34.1	39.3	37.0
TLM + RS-Mono on OS	44.3	**30.9**	**34.8**	**39.6**	**37.4**

Table 4.4: Performance of multilingual conversational models in zero-shot cross-lingual transfer for Response Retrieval (RR) on MULTI^{2}WOZ with R_{100}@1 (%) as the evaluation metric. Reference English RR performance of TOD-XLMR: 64.75%

lingual Response Selection training (RS-X), whereas for Chinese and Russian we obtain better results with the monolingual (target language) RS training (RS-Mono).

Response Retrieval (RR). The results of zero-shot transfer for RR are summarized in Table 4.4. Compared to DST results, for the sake of brevity, we show the performance of only the stronger baseline (TOD-XLMR) and the best-performing variants with intermediate conversational target-language training (one for each objective type): MLM on Mono-CC, TLM on OS, and TLM+RS-Mono on OS. Similar to DST, TOD-XLMR exhibits a near-zero cross-lingual transfer performance for RR as well, across all target languages. In sharp contrast to DST results, however, conversational specialization for the target language – with any of the three specialization objectives – massively improves the zero-shot cross-lingual transfer for RR. The gains are especially large for the models that employ the parallel OS corpus in intermediate specialization, with the monolingual (target language) RS objective slightly improving over TLM training alone.

Given the parallel nature of MULTI^{2}WOZ, we can directly compare the transfer performance of both DST and RR across the four target languages. In both tasks, the best-performing models exhibit stronger performance (i.e., smaller performance drops compared to the English performance) for German and Russian than for Arabic and Chinese. This aligns well with the linguistic proximity of the target languages to English as the source language.

4.6.2 Few-Shot Transfer and Sample Efficiency

Next, we present the results of few-shot transfer experiments, where we additionally fine-tune the task-specific TOD model on a limited number of target-language dialogs from the development portion of MULTI^{2}WOZ, after first fine-tuning it on the complete English training set from MULTIWOZ (see § 4.4.3). Few-shot cross-lingual transfer results, averaged across all four target languages, are summarized in Figure 4.3. The figure shows the performance for different sizes of the target-language training data (i.e., number of

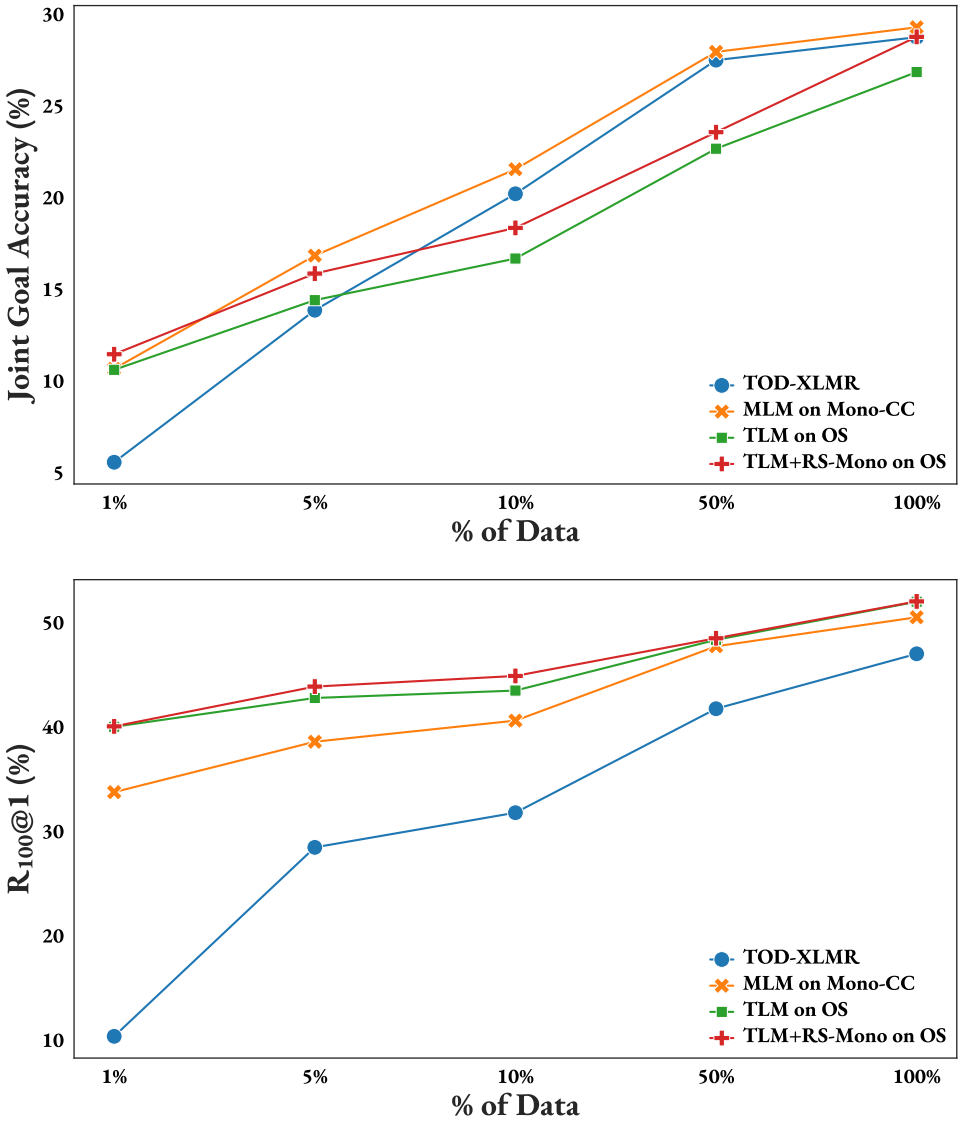


Figure 4.3: Few-shot cross-lingual transfer results for Dialog State Tracking (upper figure) and Response Retrieval (bottom figure), averaged across all four target languages (detailed per-language results available in Appendix C.3). Results are shown for different sizes of the training data in the target language (i.e., different number of *shots*): 1%, 5%, 10%, 50% and 100% of the MULTI^{2}WOZ development sets (of respective target languages).

target-language shots, that is, percentage of the target-language development portion from MULTI^{2}WOZ). Detailed per-language few-shot results are given in Table 4.5, for brevity only for TOD-XLMR and the best target-language-specialized model (TLM+RS-Mono on OS). We provide full per-language results for all specialized models from Figure 4.3 in Appendix C.3.

		DST					RR				
Lang	**Model**	**1%**	**5%**	**10%**	**50%**	**100%**	**1%**	**5%**	**10%**	**50%**	**100%**
DE	TOD-XLMR	7.68	19.26	28.08	33.17	34.10	10.25	32.47	35.56	45.39	49.46
	TLM+RS-Mono on OS	15.88	24.14	28.38	32.57	35.45	46.08	48.94	49.98	53.43	55.72
AR	TOD-XLMR	1.48	1.57	6.18	15.62	17.63	6.36	18.72	23.57	36.04	42.69
	TLM+RS-Mono on OS	4.42	6.79	8.27	14.39	21.48	33.45	37.09	38.01	41.89	47.15
ZH	TOD-XLMR	8.63	12.55	16.40	23.45	25.49	15.69	31.10	33.22	41.97	48.14
	TLM+RS-Mono on OS	11.63	14.90	17.97	22.81	28.84	38.45	43.71	45.27	48.50	51.81
RU	TOD-XLMR	4.34	21.89	30.01	37.58	37.61	8.90	31.31	34.51	43.33	47.45
	TLM+RS-Mono on OS	13.74	17.44	18.63	24.33	29.15	41.97	45.44	46.02	49.90	53.16

Table 4.5: Per-language few-shot transfer performance (sample efficiency results) on DST and RR for the baseline TOD-XLMR and the best specialized model (TLM+RS-Mono on OS).

The few-shot results unambiguously show that the intermediate conversational specialization for the target language(s) *drastically improves the target-language sample efficiency in the downstream few-shot transfer*. The baseline TOD-XLMR – not exposed to any type of conversational pre-training for the target language(s) – exhibits substantially lower performance than all three models (MLM on Mono-CC, TTLMLM on OS, and TLM+RS-Mono on OS) that underwent conversational intermediate training on respective target languages. This is evident even in the few-shot setups where the three models are fine-tuned on merely 1% (10 dialogs) or 5% (50 dialogs) of the MULTI^{2}WOZ development data (after prior fine-tuning on the complete English task data from MULTIWOZ).

As expected, the larger the number of task-specific (DST or RR) training instances in the target languages (50% and 100% setups), the closer the performance of the baseline TOD-XLMR gets to the best-performing target-language-specialized model – this is because the size of the in-language training data for the concrete task (DST or RR) becomes sufficient to compensate for the lack of conversational target-language intermediate training that the specialized models have been exposed to. The sample efficiency of the conversational target-language specialization is more pronounced for RR than for DST. This seems to be in line with the zero-shot transfer results (see Tables 4.3 and 4.4), where the specialized models displayed much larger cross-lingual transfer gains over TOD-XLMR on RR than on DST. We hypothesize that this is due to the intermediate specialization objectives (especially RS) being better aligned with the task-specific training objective of RR than that of DST.

4.7 Reproducibility

To ensure full reproducibility of our results and further fuel research on multilingual TOD, we release the parameters of TOD-XLMR within the Huggingface repository as the first publicly available multilingual PLM specialized for TOD.[7] We also release our code and data and provide the annotation guidelines for *manual post-editing* and *quality control* utilized during the creation of MULTI^{2}WOZ in Appendix C.1 and Appendix C.2. This makes our approach completely transparent and fully reproducible. All resources developed as part of this work are publicly available.[8]

4.8 Conclusions

Task-oriented dialog has predominantly focused on *English*, primarily due to the lack of robust TOD datasets in other languages (Razumovskaia et al., 2022), preventing systematic investigations of cross-lingual transfer methodologies in this crucial NLP application area. To address this gap, in this Chapter, we have presented MULTI^{2}WOZ – a robust multilingual multi-domain TOD dataset. MULTI^{2}WOZ encompasses gold-standard dialogs in four languages (German, Arabic, Chinese, and Russian) that are directly comparable with development and test portions of the English MultiWOZ dataset, thus allowing for the most reliable and comparable estimates of cross-lingual transfer performance for TOD to date. Further, we presented a language-adaptive pre-training framework for *multilingual conversational specialization* of PLMs that facilitates cross-lingual transfer for downstream TOD tasks. Our experiments on MULTI^{2}WOZ for two prominent TOD tasks – Dialog State Tracking and Response Retrieval – reveal that the cross-lingual transfer performance benefits from both (i) intermediate conversational specialization for the target language, and (ii) few-shot cross-lingual transfer for the concrete downstream TOD task. Crucially, we show that our novel conversational specialization for the target language leads to *exceptional sample efficiency* in downstream few-shot transfer.

Subsequent research has expanded on our work, building on the foundation established by MULTI^{2}WOZ dataset and the proposed multilingual conversational specialization framework. Moghe et al. (2023b) leverage MULTI^{2}WOZ dataset to evaluate the effectiveness of automatic metrics in distinguishing between high-quality and low-quality translations at the sentence level. They assess the correlation between machine translation (MT) metrics and task accuracy performance, suggesting that MT metrics should produce labels rather than scores to enable a more informative interaction between MT and multilingual language understanding. Hu et al. (2023) introduce MULTI^{3}WOZ, exploring further refinements to support cultural adaptation through an outline-based

[7] https://huggingface.co/umanlp/TOD-XLMR
[8] https://github.com/umanlp/Multi2WOZ

generation method. The approach aims to address the limitations of machine-translated texts that fail to adapt to cultural nuances. They also provide the training dataset for target languages to improve cross-lingual studies and propose an automatic multilingual evaluation framework for TOD systems (Hu et al., 2024). These advancements focus on two key areas: (i) the quality of multilingual TOD datasets, and (ii) the *proper* evaluation of cross-lingual TOD task performance, both of which drive future research exploration.

In the next Chapter, we focus on adaptive pre-training methods on social dimensions, where we incorporate demographic knowledge into PLMs leveraging the adaptive pre-training framework. We further introduce a series of control experiments to validate the efficacy of the demographic specialization techniques.

Chapter 5

Demographic Adaptation

"不是我要把天堂樂園的籬笆拆掉，我只是想把籬笆往外面再擴張一點。"
"I'm not trying to dismantle the fence in Paradise, I'm trying to extend it a bit further."
李安 Ang Lee

[*]Demographic factors (e.g., gender or age) shape and are reflected in our language. Previous work showed that incorporating demographic factors can consistently improve performance for various NLP tasks. However, the work mainly focused on (1) monolingual English datasets, which limited the multilingual (thus multi-cultural) perspectives; (2) incorporating demographic features in language representations, which rely either on leveraging demographic-specific lexica to specialized word embeddings or relying on text encoders which have not been pre-trained for general-purpose language understanding; and (3) introducing demographic features in *only* task-specific fine-tuning, which limited the benefits of demographic knowledge to tasks at hand (i.e., not task-*agnostic*; see § 2.4.1). In this Chapter, our examination centers on ascertaining whether the previous findings of domain-adaptive pre-training in Chapter 3 and language-adaptive pre-training in Chapter 4 remain valid. Specifically, we investigate whether the practice of integrating demographic factors consistently retains its effectiveness when employing state-of-the-art multilingual PLMs (see § 2.2.2 and § 2.4.1). We use three common specialization methods proven effective for incorporating external knowledge into PLMs (e.g., domain-specific, language-specific, or geographic knowledge) with adaptive pre-training framework (see § 2.3.1). We adapt the language representations for the demographic dimensions of *gender* and *age*, using masked language modeling and dynamic multi-task learning for adaptation, where we couple language modeling objectives with the prediction of demographic classes (**C1**; § 2.4.1). Our results, when employing a multilin-

[*]This Chapter is adapted from: **Chia-Chien Hung**, Anne Lauscher, Dirk Hovy, Simone Paolo Ponzetto, and Goran Glavaš. 2023. Can Demographic Factors Improve Text Classification? Revisiting Demographic Adaptation in the Age of Transformers. In *Findings of the Association for Computational Linguistics (EACL 2023)*, pages 1565–1580, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

gual PLM, show substantial gains in task performance across four languages (English, German, French, and Danish), which is consistent with the results of previous work. However, controlling for confounding factors (i.e., towards interpretability, see **C3** and § 2.4.3) – primarily domain and language proficiency of PLMs – shows that downstream performance gains from our demographic adaptation do *not* actually stem from demographic knowledge. Our results indicate that demographic specialization of PLMs, while holding promise for positive societal impact, still represents an unsolved problem for (modern) NLP.

5.1 Introduction

Demographic factors like social class, education, income, age, or gender, categorize people into specific groups or populations. At the same time, demographic factors both shape and are reflected in our language (e.g., Trudgill, 2000; Eckert and McConnell-Ginet, 2013; Flek, 2020). A large body of work focused on modeling demographic language variation, especially the correlations between words and demographic factors (Bamman et al., 2014; Garimella et al., 2017; Welch et al., 2020, *inter alia*). In a similar vein, Volkova et al. (2013) and Hovy (2015) demonstrated that explicitly incorporating demographic information in language representations improves performance on downstream NLP tasks, e.g., topic classification or sentiment analysis. However, these observations rely on approaches that leverage gender-specific lexica to specialize word embeddings and text encoders (e.g., recurrent networks) that have not been pre-trained for (general purpose) language understanding. To date, the benefits of demographic specialization have not been tested with Transformer-based (Vaswani et al., 2017) PLMs (see § 2.2), which have been shown to excel on the vast majority of NLP tasks and even surpass human performance in some cases (Wang et al., 2018).

More recent studies focus mainly on monolingual English datasets and introduce demographic features in task-specific fine-tuning (Voigt et al., 2018; Buechel et al., 2018), which limits the benefits of demographic knowledge to tasks at hand. In this Chapter, we investigate the (task-agnostic) demographic specialization of PLMs, aiming to impart the associations between demographic categories and linguistic phenomena into the PLMs parameters. If successful, such specialization could benefit any downstream NLP task in which demographic factors (i.e., demographically conditioned language phenomena) matter. For this, we adopt intermediate training paradigms that have been proven effective for the specialization of PLMs for other types of knowledge, e.g., in domain, language, and geographic adaptation (Glavaš et al., 2020; Hung et al., 2022a; Hofmann et al., 2024). To this effect, we perform (i) masked language modeling on text corpora produced by a demographic group and (ii) dynamic multi-task learning (Kendall et al., 2018), wherein we combine language modeling with the prediction of demographic categories.

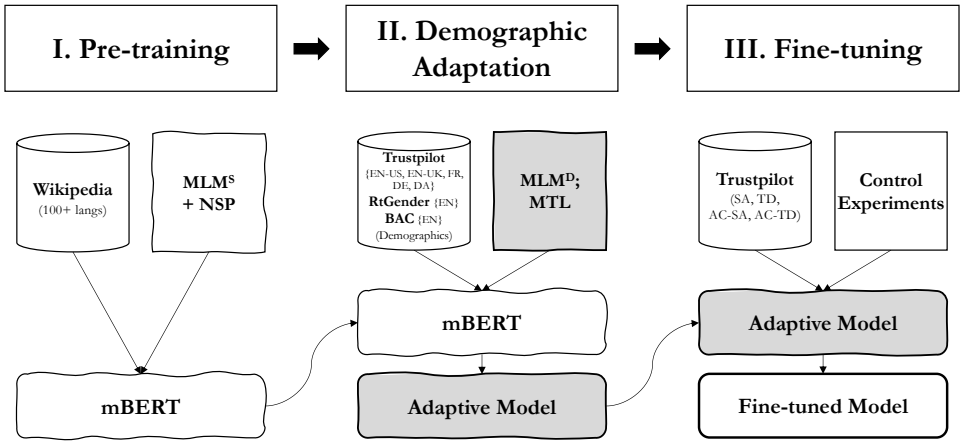


Figure 5.1: Overview of the demographic-adaptive pre-training framework. The framework involves a three-stage process: *Pre-training*, *Demographic Adaptation*, *Fine-tuning*. In this Chapter, intermediate training indicates the second (II) stage: demographic adaptation phase.

We evaluate the effectiveness of the demographic PLM specialization on both intrinsic (demographic category prediction) and extrinsic (sentiment classification and topic detection) evaluation tasks across four languages: English, German, French, and Danish, using a multilingual corpus of reviews (Hovy et al., 2015) annotated with demographic information. In line with earlier findings (Hovy, 2015), our initial experiments based on a multilingual PLM (mBERT; Devlin et al., 2019), render demographic specialization effective: we observe gains in most tasks and settings. Through a set of controlled experiments, where we (1) adapt with in-domain language modeling alone, without leveraging demographic information, (2) demographically specialize *monolingual* PLMs of evaluation languages, (3) carry out a meta-regression analysis over dimensions that drive the performance, and (4) analyze the topology of the representation spaces of demographically specialized PLMs, we show, however, that most of the original gains can be attributed to confounding effects of language and/or domain specialization.

Our findings indicate that specialization approaches, proven effective for other types of knowledge, fail to adequately instill demographic knowledge into PLMs, making demographic specialization of NLP models an open problem in the age of PLMs. An overview of the demographic-adaptive pre-training framework is depicted in Figure 5.1, and our research code and data are publicly available.[1]

[1] https://github.com/umanlp/SocioAdapt

5.2 Related Work

Adaptive Pre-training. Adaptive pre-training (see § 2.3.1), focusing on intermediate language modeling with texts from the same or similar distribution as the downstream data, has been shown to lead to improvements on various NLP tasks (e.g., Gururangan et al., 2020). During this process, the goal is to inject additional information into the pre-trained language model and thus specialize the model for a particular domain (e.g., Aharoni and Goldberg, 2020; Hung et al., 2022a), language (e.g., Glavaš et al., 2020) or to encode other types of knowledge such as argumentation knowledge (e.g., Holtermann et al., 2022), or geographic knowledge (e.g., Hofmann et al., 2024).

For instance, we propose a computationally efficient approach in Chapter 3 by employing domain-specific adapter or embeddings modules, and show the domain adaptation approach leads to improvements in downstream tasks. In Chapter 4, we perform language adaptation through Response Selection (RS) objectives in the target languages with dialogic text corpora (i.e., LangOpenSubtitles), demonstrating substantial gains in downstream cross-lingual transfer for task-oriented dialog tasks. These specialization approaches mainly rely on a single objective (e.g., MLM on "plain" text data, or RS on "dialogic" text corpora). Instead, Hofmann et al. (2024) conduct geoadaptation by coupling MLM with a token-level geolocation prediction in a dynamic multi-task learning setup. In this work, we adopt a similar approach and perform masked language modeling on the text corpora of a specific demographic dimension.

Demographic Specialization. User demographics contribute to the diversity of language preferences (Loveys et al., 2018). Accordingly, several studies have leveraged demographic information (e.g., gender, age, education) to investigate the effect of encoded sociodemographic knowledge in the representations of PLMs (Lauscher et al., 2022a) or obtain better language representations for various NLP tasks (Volkova et al., 2013; Garimella et al., 2017). Recent research studies on demographic adaptation mainly focus on (1) learning demographic-aware word embeddings and do not work with large PLMs (Hovy, 2015), or (2) leveraging demographic information with special PLM architectures specifically designed for certain downstream tasks (e.g., empathy prediction (Guda et al., 2021)). The latter, however, do not consider a task-*agnostic* approach to injecting demographic knowledge into language models, and also focus on a monolingual setup only. Further, what roles the different factors (i.e., domain, language, demographic aspect) in the specialization really play remains unexplored.

5.3 Demographic Adaptation

Our goal is to inject demographic knowledge through intermediate PLM training in a task-agnostic manner (**C1**; § 2.4.1). To achieve this goal, we train the PLM in a dynamic multi-task learning setup (Kendall et al., 2018), in which we couple masked language modeling (MLM) with predicting the demographic category – gender or age group of the text author. Such multi-task learning setup is designed to force the PLM to learn associations between the language constructs and demographic groups, if these associations are salient in the training corpora.

Masked Language Modeling (MLM). Following successful work on pre-training via language modeling for domain-adaptation (Gururangan et al., 2020; Hung et al., 2022a), we investigate the effect of running standard MLM on the text corpora of a specific demographic dimension (e.g., gender-related corpora) (Liu et al., 2019c). We compute the MLM loss L_{mlm} in the common way, as negative log-likelihood of the true token probability (see § 3.1.4.1 and Equation 3.1).

Demographic Category Prediction. In the multi-task learning setup, the representation of the input text, as output by the PLM, is additionally fed into a classification head that predicts the corresponding demographic category: *age* (below 35 and above 45)[2], and *gender* (female and male). The demographic prediction loss L_{dem} is computed as the standard binary cross-entropy loss.

We experiment with two different ways of predicting the demographic category of the text: (i) from the transformed representation of the sequence start token (`[CLS]`), and (ii) from the contextualized representations of each masked token. We hypothesized that the former variant, in which we predict the demographic class from the `[CLS]` token representation, would establish links between more complex demographically condition linguistic phenomena (e.g., syntactic patterns or patterns of compositional semantics that a demographic group might exhibit), whereas the latter – predicting demographic class from representations of masked tokens – is more likely to establish simpler lexical links, i.e., capture the vocabulary differences between the demographic groups.

Multi-Task Learning. Since both losses can be computed from the same input instances, we opt for joint Multi-Task Learning (MTL) and resort to dynamic MTL based on the *homoscedastic* uncertainty of the losses, wherein the loss variances are used to balance the contributions of the tasks (Kendall et al., 2018). The intuition is that more effective MTL occurs if we dynamically assign less importance to more uncertain tasks, as opposed to assigning uniform task weights throughout the whole training. Homoscedastic uncertainty weighting in MTL has been effective in different NLP settings (Lauscher

[2] As suggested by Hovy (2015) the split for the age ranges result in roughly equally-sized data sets for each sub-group and is non-contiguous, avoiding fuzzy boundaries.

et al., 2018; Hofmann et al., 2024). In our scenario, L_{mlm} and L_{dem} are measured on different scales in which the model would favor (i.e., be more confident for) one objective than the other. The confidence level of the model prediction for each task would change throughout the training process: this makes dynamic weighting desirable. We dynamically prioritize the tasks via homoscedastic uncertainties σ_t:

$$\tilde{L}_t = \frac{1}{2\sigma_t^2} L_t + \log \sigma_t \,, \tag{5.1}$$

where σ_t^2 is the variance of the task-specific loss over training instances for quantifying the uncertainty of the task $t \in \{mlm, dem\}$. In practice, we train the network to predict the log variance, $\eta_t := \log \sigma_t^2$, since it is more numerically stable than regressing the variance σ_t^2, as the log avoids divisions by zero. The adjusted losses are then computed as:

$$\tilde{L}_t = \frac{1}{2}(e^{-\eta_t} L_t + \eta_t) \tag{5.2}$$

The final loss is to minimize the sum of two uncertainty-adjusted losses: $\tilde{L}_{mlm} + \tilde{L}_{dem}$.

5.4 Experimental Setup

Here we describe evaluation tasks and provide details on the data used for demographic specialization and downstream evaluation.

Evaluation Tasks. We follow Hovy (2015) and measure the effects of demographic specialization of PLMs on three text-classification tasks, coupling intrinsic demographic *attribute classification* (**AC**) with two extrinsic text classification tasks: *sentiment analysis* (**SA**) and *topic detection* (**TD**). We show an example for each of the **AC**, **SA**, and **TD** task respectively:

(i) Attribute Classification (AC) for SA – United Kingdom (age):

Text *Delivery and dispatch are usually quite quick, and they often have huge discount days.*
Label AGE: <35

(ii) Sentiment Analysis (SA) – Germany (gender-M):

Text *Keine Antwort auf meine Emails, keine Zeichnung.*
Label NEGATIVE

(iii) Topic Detection (TD) – United States (gender-F):[3]

Text *We used housetrip to book an apartment in nyc over spring break.*
Label HOTELS

[3]The five topics for TD task in gender category for United States include: Car Lights, Domestic Appliances, Fashion Accessories, Pets, and Hotels. A more detailed description regarding the selection of topics for different countries is given by Hovy et al. (2015).

Country	Language	*Gender*				*Age*			
		Specialization		**SA, AC-SA**	**TD, AC-TD**	**Specialization**		**SA, AC-SA**	**TD, AC-TD**
		F	**M**	**F / M**		**<35**	**>45**	**<35 / >45**	
Denmark	Danish	1,596,816	2,022,349	250,485	120,805	833,657	494,905	75,300	44,815
France	French	489,778	614,495	67,305	55,570	40,448	36,182	6,570	6,120
Germany	German	210,718	284,399	28,920	30,580	66,342	47,308	5,865	8,040
UK	English	1,665,167	1,632,894	156,630	183,995	231,905	274,528	26,325	22,095
US	English	575,951	778,877	72,270	61,585	124,924	70,015	6,495	12,090

Table 5.1: Number of instances in different portions of the TRUSTPILOT dataset (Hovy et al., 2015) used in our experiments. For each country (Denmark, France, Germany, UK, and US), we report the size of the specialization and fine-tuning portions, the latter for each of the two extrinsic tasks: Sentiment Analysis (SA) and Topic Detection (TD). Note that we use the same SA and TD reviews for the intrinsic AC tasks of predicting the demographic categories (denoted AC-SA and AC-TD, respectively). Numbers are shown separately for the two demographic dimensions: gender and age. For task fine-tuning datasets (for SA/AC-SA, and for TD/AC-TD), we indicate the number of instances in each category (which is the same for both categories: F and M for gender, <35 and >45 for age). We split the fine-tuning datasets randomly into train, development, and test portions in the 60/20/20 ratio.

As an intrinsic evaluation task, AC directly tests if the intermediate demographic specialization results in a PLM that can be more effectively fine-tuned to predict the same demographic classes used in the intermediate specialization: PLMs (vanilla PLM and our demographically specialized counterpart) – are fine-tuned in a supervised fashion to predict the demographic class (gender or age) of the text author. SA is a ternary classification task in which the reviews with ratings of 1, 3, and 5 stars represent instances of *negative*, *neutral*, and *positive* class, respectively. TD classifies texts into 5 different topic categories. We report the F_1-measure for each task following Hovy (2015).

Data. We carry out our core experimentation on the multilingual demographically labeled dataset of reviews (Hovy et al., 2015), created from the internationally popular user review website TRUSTPILOT.[4] For comparison and consistency, we work with exactly the same data portions as Hovy (2015): collections that cover (1) two most prominent demographic dimensions – *gender* and *age*, with two categories in each (gender: male or female; age: below 35 or above 45)[5] and (2) five countries (four languages): United States (US), Denmark, Germany, France, and United Kingdom (UK). Table 5.1 displays the numbers of reviews for each country, demographic aspect, and dataset portion (demographic specialization vs. task fine-tuning).

[4] https://www.trustpilot.com/

[5] As suggested by Hovy (2015), the split for the age ranges results in roughly equally-sized data sets for each sub-group and is non-contiguous, avoiding fuzzy boundaries.

To avoid any information leakage, we ensure that – for each country-demographic dimension collection (e.g., US, gender) – there is zero overlap between the portions we select for intermediate demographic specialization and portions used for downstream task fine-tuning and evaluation (for AC, SA, and TD). For TD, we aim to eliminate the confounding effect of demographically-conditioned label distributions (e.g., female authors wrote reviews for *clothing store* more frequently than male authors; vice-versa for *electronics and technology*). To this effect, we select, for each country, reviews from the five most frequent topics and sample the same number of reviews in each topic for both demographic groups (i.e., *male* and *female* for gender; *below 35* and *above 45* for age). For the intrinsic AC task (i.e., fine-tuning to predict either gender or age category), we report the results for two different review collections: the first is the set of reviews that have, besides the demographic classes, been annotated with sentiment labels (we refer to this as AC-SA), and the second is the reviews that have topic labels (i.e., product/service category; we refer to this portion as AC-TD). For these downstream task datasets, we make sure that two demographic classes (*male* and *female* for gender; *under 35* and *above 45* for age) are equally represented in each dataset portion (train, validation, and test).

For intermediate specialization of the multilingual model, we randomly sample 100K instances per demographic group from the *gender* specialization portion and 50K instances each from the texts reserved for *age* specialization concatenated across all 5 countries (**C2**; § 2.4.2). For the specialization of monolingual PLMs, we randomly sample the same number of instances but from the specialization portions of a *single* country. Following the established procedure (e.g., Devlin et al., 2019; Liu et al., 2019c), we dynamically mask 15% of the tokens in the demographic specialization portions for MLM.

Pre-trained Transformer-based Language Models. Given that we experiment with TRUSTPILOT data in four different languages, in our core experiments, we resorted to multilingual BERT (mBERT)[6] (Devlin et al., 2019) as the starting PLM. This allows us to merge the (fairly large) specialization portions of TRUSTPILOT in different languages (see Table 5.1) and run a single multilingual demographic specialization procedure on the combined multilingual review corpus. We then fine-tune the demographically-specialized mBERT and evaluate downstream task performance separately for each of the five countries (using train, development, and test portions of the respective country). We report the results for two different variants of our dynamic multi-task demographic specialization (DS): (1) when the demographic category is predicted from representations of masked tokens (DS-Tok), and (2) when we predict the demographic category from the encoding of the whole sequence (i.e., review; this version is denoted with DS-Seq). We compare these demographic-specialized PLM variants against two baselines: vanilla PLM and PLM specialized on the same review corpora as our MTL variants but only via MLM (i.e., without providing the demographic signal).

[6]We use the pre-trained language model weights loaded from HuggingFace: `bert-base-multilingual-cased`. More details about the model can be referred to § 2.2.2.

Training and Optimization. In demographic specialization training, we fix the maximum sequence length to 128 subword tokens. We train for 30 epochs in batches of 32 instances and search for the optimal learning rate among the following values: $\{5 \cdot 10^{-5}, 1 \cdot 10^{-5}, 1 \cdot 10^{-6}\}$. We apply early stopping based on the development set performance: we stop if the joint MTL loss does not improve for 3 epochs). For downstream task fine-tuning and evaluation, we train for maximum 20 epochs in batches of 32. We search for the optimal learning rate between the following values: $\{5 \cdot 10^{-5}, 1 \cdot 10^{-5}, 5 \cdot 10^{-6}, 1 \cdot 10^{-6}\}$ and apply early stopping based on the development set performance (patience: 5 epochs). We use AdamW (Loshchilov and Hutter, 2019) as the optimization algorithm.

5.5 Results and Discussion

We first discuss the results of multilingual demographic specialization with mBERT as the PLM (§ 5.5.1). We then provide a series of control experiments in which we isolate the effects that contribute to performance gains of demographically specialized PLMs (§ 5.5.2).

5.5.1 Multilingual Specialization Results

Table 5.2 shows the results of gender- and age-specialized mBERT variants – DS-Seq and DS-Tok – on gender and age classification (AC-SA and AC-TD) as intrinsic tasks together with sentiment analysis (SA) and topic detection (TD) as extrinsic evaluation tasks, for each of the five countries encompassed by the TRUSTPILOT datasets (Hovy et al., 2015). The performance of DS-Seq and DS-Tok is compared against the PLM baselines that have not been exposed to demographic information: vanilla mBERT and mBERT with additional MLM on the same TRUSTPILOT data on which DS-Seq and DS-Tok were trained. Our demographically specialized models generally outperform the vanilla mBERT across the board, both on intrinsic and extrinsic tasks, unsurprisingly with much more prominent gains on the former. The comparison against the domain-adaptation in which mBERT was intermediately trained only MLM on TRUSTPILOT reviews, but without demographic category prediction, however, reveals that much of the gains that DS-Seq and DS-Tok have over vanilla mBERT stem from domain adaptation: somewhat surprisingly, DS models fall behind MLM-based domain adaptation on the intrinsic tasks of gender/age classification (e.g., for age group classification on AC-SA, the DS variants fall short of MLM by 2 F_1 points), while exhibiting small but fairly consistent gains over MLM for extrinsic SA and TD tasks, both in gender and age intermediate specialization. Although the gains are not particularly convincing, the SA and TD still seem to favor intermediate demographic specialization, which is in line with findings from Hovy (2015), who also reported small but (mostly) consistent gains for these two tasks.

		Demographic: ***gender***								*Demographic:* ***age***							
		Gender class.		**SA**			**TD**			**Age class.**		**SA**			**TD**		
Country	**Model**	**AC-SA**	**AC-TD**	**F**	**M**	**X**	**F**	**M**	**X**	**AC-SA**	**AC-TD**	**<35**	**>45**	**X**	**<35**	**>45**	**X**
Denmark	mBERT	64.0	61.8	69.2	64.8	67.2	59.3	58.3	59.0	57.2	64.5	62.7	62.7	62.9	56.1	52.2	53.4
	MLM	**65.2**	63.4	69.5	**65.8**	67.8	59.7	58.8	**59.4**	**65.5**	65.1	63.3	62.1	63.0	**57.1**	52.6	54.1
	DS-Seq	64.9	63.5	**69.9**	65.7	67.7	59.7	57.8	59.1	65.2	**65.2**	63.1	62.9	63.0	56.9	**53.3**	**54.5**
	DS-Tok	65.0	**63.5**	69.1	65.6	**68.0**	**59.9**	**58.9**	59.0	65.3	64.6	**64.2**	**63.3**	**63.2**	56.2	53.2	54.3
Germany	mBERT	59.5	57.9	66.1	63.2	64.5	67.8	65.6	65.8	58.0	56.9	52.6	55.0	55.0	60.1	55.3	57.1
	MLM	61.2	60.1	**67.7**	**65.3**	66.1	**68.6**	67.0	**67.1**	**61.1**	**58.9**	53.6	55.5	56.7	**61.5**	56.5	58.7
	DS-Seq	60.1	**60.3**	66.7	64.0	65.7	67.6	65.7	66.4	56.4	58.2	**53.8**	55.3	55.5	60.8	**57.6**	**59.3**
	DS-Tok	**62.9**	58.3	66.8	64.3	**66.8**	68.3	**67.0**	66.7	56.6	57.4	53.0	**56.5**	**56.7**	59.3	56.5	59.3
US	mBERT	62.6	58.1	66.3	64.4	66.0	71.2	68.4	70.2	62.9	60.7	57.7	57.9	57.8	68.0	64.3	64.3
	MLM	63.3	**59.6**	67.3	66.2	66.9	72.1	69.4	70.3	**63.6**	**61.9**	59.4	57.8	**58.2**	69.0	64.2	65.2
	DS-Seq	**63.8**	59.2	67.2	66.3	67.0	72.3	69.2	70.4	60.7	61.5	59.3	57.9	58.0	**69.8**	64.4	**65.8**
	DS-Tok	62.2	58.8	**68.0**	**66.4**	**67.3**	**72.8**	**69.5**	**70.5**	59.7	61.2	**59.9**	**58.6**	57.8	69.2	**65.4**	64.9
UK	mBERT	61.9	63.1	71.0	69.0	69.7	70.4	67.9	68.9	65.1	65.2	63.8	63.9	63.7	64.7	67.1	66.3
	MLM	63.0	65.3	72.0	70.4	71.0	70.6	67.9	69.8	**65.4**	**65.6**	62.8	62.0	63.0	65.1	67.3	67.3
	DS-Seq	63.4	64.9	72.9	70.9	71.7	70.6	68.2	69.8	65.3	62.8	63.8	64.9	64.9	66.0	**68.1**	66.5
	DS-Tok	**63.5**	**65.6**	**73.0**	**71.0**	**71.9**	**70.8**	**68.2**	**69.9**	64.0	62.8	**64.6**	**65.2**	**65.1**	**66.4**	67.3	**67.6**
France	mBERT	63.9	61.2	69.3	67.0	67.8	44.6	42.4	43.1	55.7	56.6	59.6	57.4	61.5	52.0	47.1	49.0
	MLM	64.6	62.1	69.9	67.1	68.4	45.8	43.3	44.3	**56.8**	**57.2**	59.9	59.5	61.6	**52.5**	47.2	50.3
	DS-Seq	64.1	**63.1**	**70.6**	67.3	68.4	**46.0**	43.4	44.2	55.1	55.5	60.4	**60.3**	**62.8**	51.1	47.3	50.3
	DS-Tok	**65.0**	62.9	70.1	**67.5**	**68.8**	45.5	**43.9**	**44.4**	54.4	55.9	**60.9**	59.8	59.7	50.2	**48.0**	**50.8**
Average	mBERT	62.4	60.4	68.4	65.7	67.0	62.7	60.5	61.4	59.8	60.8	59.3	59.4	60.2	60.2	57.2	58.0
	MLM	63.5	62.1	69.3	**67.0**	68.0	63.4	61.3	**62.2**	**62.5**	**61.7**	59.8	59.4	60.5	**61.0**	57.6	59.1
	DS-Seq	63.3	**62.2**	**69.5**	66.8	68.1	63.2	60.9	62.0	60.5	60.6	60.1	60.3	**60.8**	60.9	**58.1**	59.3
	DS-Tok	**63.7**	61.8	69.4	**67.0**	**68.6**	**63.5**	**61.5**	62.1	60.0	60.4	**60.5**	**60.7**	60.5	60.3	**58.1**	**59.4**

Table 5.2: Results of gender-specialized (age-specialized) multilingual BERT (DS-Seq and DS-Tok) on gender (age) classification (AC-SA and AC-TD) as intrinsic task and sentiment analysis (SA) and topic detection (TD) as extrinsic evaluation tasks. Comparisons against the vanilla mBERT and mBERT additionally trained on the same review corpora but without demographic information, only with masked language modeling (MLM). For SA and TD, we separately report the performance on the test sets consisting of only one demographic class (gender: F and M, age: <35 and >45) as well as on the mixed test sets containing reviews from both demographic classes (X for both gender and age). Bold numbers indicate the best-performing model (between mBERT, MLM, DS-Seq and DS-Tok) for each country-task combination.

5.5.2 Control Experiments

To more precisely measure the contributions of demographic information that DS-* variants incorporate, we design further experiments that control for two key side-effects of demographic specialization: (i) language specialization and (ii) domain adaptation. We then carry out the meta-regression analysis to tease out the individual contributions of

language, domain, and demographic knowledge on the performance difference between vanilla mBERT and respective intermediately specialized models (mBERT or monolingual BERT specialized on the data of the same or different domain with or without demographic signal). Finally, we compare the representation spaces of the PLMs – before and after demographic specialization – along the demographic dimension. The exploration of different controlled conditions is aimed to enhance model transparency, uncover both strengths and limitations, and ultimately contribute to more robust and comprehensive adaptive pre-training methods (**C3**; § 2.4.3).

Controlling for Language Proficiency. Massively multilingual PLMs, like mBERT or XLM-R (Conneau et al., 2020a) (see § 2.2.2) suffer from the *curse of multilinguality* (Conneau et al., 2020a; Lauscher et al., 2020; Pfeiffer et al., 2020): given a fixed capacity of the model, the representations from a multilingual PLM for any individual (high-resource) language will be of lower quality than those of the monolingual PLM, as multilingual PLMs share their limited capacity over many languages. It is thus possible that demographic specialization of mBERT on TRUSTPILOT data in our four languages leads to substantial gains over vanilla mBERT (pre-trained on 104 languages) primarily because of mBERT's acquisition of additional language competencies for these four languages.

To test this, we additionally execute demographic specialization individually for each language (i.e., as opposed to a single multilingual specialization), starting from a monolingual PLM of that language.[7] Monolingual PLMs produce higher quality representations for their respective language than mBERT. Because of this, we hypothesize that subjecting them to demographic specialization on TRUSTPILOT is unlikely to improve their "command" of the language substantially. Consequently, should we still see (downstream) gains from demographic specialization for monolingual PLMs, we can be more confident that they stem from the injected demographic information.

Table 5.3 shows the effects of demographic specialization on monolingual PLMs of the four languages. For brevity (full results in Appendix D.1), we average the demographic attribute classification (AC) results from two different test portions from Table 5.2 (having labels for different downstream tasks, AC-SA and AC-TD); for extrinsic tasks, SA and TD, we report only the score on demographically balanced test sets (denoted "X" in Table 5.2). The results show that, when we control for language proficiency (as monolingual PLMs are more proficient in their respective language than mBERT), the downstream gains of demographic specialization (on SA and TD) vanish. The DS-Seq and DS-Tok still retain marginal numeric (statistically insignificant) gains over MLM in gender-based specialization, but they lag behind in age-based specialization. Also, both DS-* variants

[7] We use the pre-trained language model weights loaded from HuggingFace: `bert-base-cased`, `bert-base-german-cased`, `dbmdz/bert-base-french-europeana-cased`, and `Maltehb/danish-bert-botxo`.

		Gender			*Age*		
Country	**Model**	**AC**	**SA**	**TD**	**AC**	**SA**	**TD**
Denmark	BERT	65.0	70.4	59.9	66.5	66.0	56.3
	MLM	65.1	70.3	60.6	**67.4**	**67.6**	**57.6**
	DS-Seq	**65.2**	**70.6**	60.0	67.1	67.1	56.5
	DS-Tok	65.1	**70.6**	**60.8**	67.2	67.2	56.7
Germany	BERT	59.4	64.3	67.8	58.8	57.1	58.3
	MLM	**60.9**	65.4	67.7	**60.1**	**58.1**	**59.9**
	DS-Seq	60.1	**66.2**	67.8	59.8	55.8	59.1
	DS-Tok	60.6	66.0	**67.9**	58.9	54.0	59.2
US	BERT	61.5	67.1	71.0	64.1	57.2	**67.2**
	MLM	61.7	67.8	71.3	64.1	**60.4**	66.7
	DS-Seq	61.6	**68.0**	**71.6**	**65.2**	59.4	67.1
	DS-Tok	**62.1**	67.9	**71.6**	64.3	59.4	66.7
UK	BERT	64.1	72.3	70.1	65.8	65.5	68.0
	MLM	**64.3**	**72.6**	70.0	**66.5**	66.9	**70.0**
	DS-Seq	64.2	72.4	70.2	65.9	**67.6**	69.4
	DS-Tok	64.1	72.2	**70.3**	66.0	67.1	69.2
France	BERT	63.6	68.6	45.1	**56.5**	60.3	49.6
	MLM	**64.1**	67.6	45.5	56.4	61.6	50.2
	DS-Seq	63.7	69.3	45.3	56.1	**62.0**	50.2
	DS-Tok	63.7	**69.5**	**45.6**	56.3	61.5	**50.3**
Average	BERT	62.7	68.5	62.8	62.3	61.2	59.9
	MLM	**63.2**	68.7	63.0	**62.9**	**62.9**	**60.9**
	DS-Seq	62.9	**69.3**	63.0	62.8	62.4	60.5
	DS-Tok	63.1	69.2	**63.2**	62.5	61.8	60.4

Table 5.3: Results of gender/age-specialized **monolingual** PLMs – DS-Seq and DS-Tok – on demographic attribute classification (AC), sentiment analysis (SA) and topic detection (TD). Bold numbers indicate the best-performing model (between BERT, MLM, DS-Seq and DS-Tok) for each country-task combination.

and MLM display only marginal gains with respect to vanilla monolingual BERT models of the four languages: e.g., in gender-specialization and for SA, DS-Tok has an average advantage of 0.7 F_1 points over the non-specialized vanilla monolingual BERTs; compare this to a gain of 1.6 F_1 points that mBERT-based DS-Tok has over vanilla mBERT (Table 5.2). These results question the downstream usefulness of demographic specialization – suggested by findings from prior work (Hovy, 2015) and our results for multilingual PLMs (Table 5.2) – if one starts from the most proficient PLM for the concrete language at hand, i.e., a monolingual PLM.

Controlling for Domain Knowledge. Both simple additional MLM on TRUSTPILOT data, as well as multi-task demographic specialization training (DS-* variants), inject knowledge about the domain-specific language of reviews into the PLM. As shown by previous work (Glavaš et al., 2020; Diao et al., 2021; Hung et al., 2022a), domain

Figure 5.2: Evaluation results on TRUSTPILOT for Sentiment Analysis (SA) and Topic Detection (TD) when running the intermediate specialization on out-of-domain data (RTGENDER (Voigt et al., 2018) for *gender* and BAC (Schler et al., 2006) for *age*). We report the delta in F_1-score in comparison to the specialization on TRUSTPILOT in-domain data.

adaptation generally leads to better downstream performance on in-domain data for any task. We next investigate to which extent the domain specialization is responsible for performance gains. To this end, we perform demographic specialization on (demographically labeled) training data from a different domain: for *gender* specialization, we use the RTGENDER (Voigt et al., 2018) consisting of social media posts collected from diverse sources, whereas for *age* specialization we resort to the Blog Authorship Corpus (BAC; Schler et al., 2006) containing blogposts from `blogger.com`.

Figure 5.2 displays the effects of out-of-domain specialization of BERT on downstream SA and TD performance (i.e., performance differences w.r.t. corresponding in-domain specialized models). Since RtGender and BAC are English-only datasets, we report the results only for US and UK (for brevity, we report the performance only on the demographically balanced test sets, i.e., setups indicated with "X" in Table 5.2; both DS-* variants exhibit very similar behavior, so for brevity, we only display results for DS-Tok; complete results are in Appendix D.2). Expectedly, the out-of-domain specialization deteriorates the downstream performance for both MLM and DS-Tok. Interestingly, MLM, which is not exposed explicitly to the demographic signal in specialization, tends to suffer less from out-of-domain specialization than the gender-informed DS-Tok. In contrast, age-informed DS-Tok seems to exhibit similar losses as MLM due to out-of-domain specialization. These results further question the hypothesis that demographic information guides downstream gains, suggested by prior work (Hovy, 2015) and our in-domain specialization results (with mBERT) from Table 5.2.

Task	Selected features	all	-D	-M	-S	-C	-A
Gender							
AC-SA	US (1.0); Denmark (0.9); MLM (0.9); DS-Tok (0.9);	0.51	-	0.56	-	0.63	0.62
AC-TD	MLM (1.0); Monoling(1.0) DS-Tok (0.9);	0.51	-	0.73	-	0.54	0.66
SA	France (1.0); DS-Tok (1.0); Denmark (0.8); MLM (0.8); In-domain (0.6)	0.92	0.94	0.95	0.94	0.97	0.98
TD	DS-Tok (0.6); MLM (0.5); In-domain (0.5)	0.33	0.36	0.35	0.34	0.35	0.40
Age							
AC-SA	Denmark (3.0); MLM (1.5); Monoling (0.9)	1.93	-	1.98	-	2.31	2.02
AC-TD	UK (2.1); France (1.4); MLM (0.9);	0.68	-	0.69	-	1.02	0.82
SA	In-domain (1.3); DS-Tok (1.0); MLM (0.9);	0.96	1.03	0.97	0.97	0.98	1.03
TD	Denmark (1.6); <35 (0.7); DS-Seq (0.6); DS-Tok (0.6)	1.52	1.53	1.53	1.55	1.61	1.54

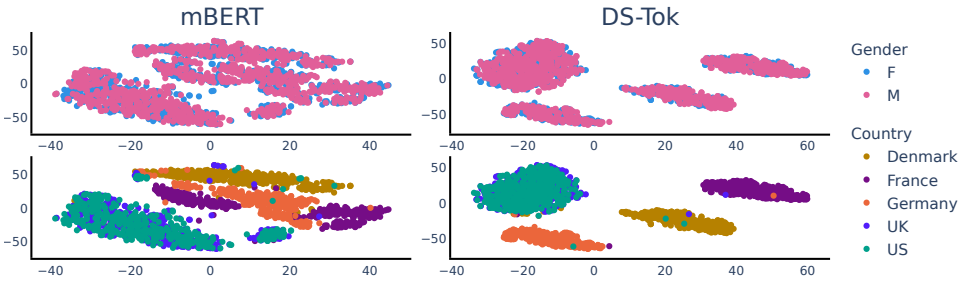
Table 5.4: Results of meta-regression analysis. We report the goodness-of-fit (RMSE) results for predicting deltas in downstream performance between specialized models and their respective vanilla PLM. Results reported for three tasks – intrinsic demographic attribute classification (AC; on datasets AC-SA and AC-TD), Sentiment Analysis (SA), and Topic Detection (TD) with both demographic factors, *gender* and *age*. We compare the results across different feature sets – for all features (**all**), and excluding individual features: domain (**-D**), mono- vs. multilingual (**-M**), fine-tuning demographic setting (e.g., F vs. M vs. X for gender; **-S**), country (**-C**), and the adaptation approach (i.e., MLM vs. DS-Tok vs. DS-Seq; **-A**). For each task, when including all features (column: **all**), we list the most important features, those with weights > 0.5 (*selected features*).

Meta-Regression Analysis. Next, we aim to quantify, via a meta-regression analysis, the contributions of individual factors (country, in-domain vs. out-of-domain specialization, language, specialization approach, test set structure) on the task performance (AC-SA, AC-TD, SA, TD). We use the difference in performance between the specialized model and its corresponding vanilla PLM (mBERT or monolingual PLM) as the label (i.e., output, dependent) variable for the regression. We use the following input features (all one-hot encoded) as prediction variables: (i) country/language of fine-tuning/evaluation data, (ii) specialization method (MLM vs. DS-Tok vs. DS-Seq), (iii) in-domain vs. out-

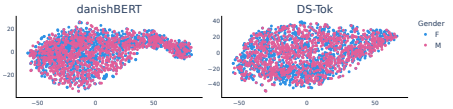
of-domain specialization, (iv) whether the starting/vanilla PLM is monolingual (e.g., French BERT) or multilingual (mBERT), (v) and the demographic group from which the fine-tuning/evaluation data comes from (F vs. M vs. X for gender and <35 vs. >45 vs. X for age). We then fit a linear regressor on all data points, using either the full set of features or, in ablations, excluding certain subsets; we report the goodness of fit as average root mean square error (RMSE).

We summarize the results of our meta-regression analysis in Table 5.4. For each task, we list the selected features paired with the RMSE scores. When we fit regression using all features (**all**), the country of origin of fine-tuning data (i.e., features *Denmark*, *France*, *UK*, etc.) tends to overall explain the variance of specialization effect on model performance as good as or even better than the specialization approach (demographically-informed DS-* variants and demographically-uninformed MLM). The specialization approach features (MLM, DS-Tok, and DS-Seq), however, do appear among the most important features in most settings, suggesting that knowing the specialization approach does help predict the performance of the specialized model. Note, however, that in terms of assessing whether demographic information generally improves specialization, this needs to be combined with actual task performance results from Tables 5.2 and 5.3. For example, feature DS-Tok is among the most important features for SA performance after *gender* specialization: looking at the results for DS-Tok in both Tables 5.2 and 5.3, we see that it achieves, in most cases, scores above MLM – this, in turn, suggests that demographically-informed gender specialization does (regardless of other factors) improve the downstream SA performance. The ablation results offer a complementary view into the importance of individual features: the larger the increase in RMSE when excluding a feature (compared to using all features), the more important the feature is. The regressions in which we exclude the information on the specialization approach (**-A**) result in the highest RMSE for gender specialization on both extrinsic tasks (SA and TD). In all other setups (AC for both gender and age specialization, as well as SA and TD for age), there is another type of information, the removal of which results in a less predictable specialization effect: for instance, AC after age specialization, the **-C** setting increases the RMSE the most, representing that features indicating the demographic composition of the country factor of the fine-tuning dataset jointly have the largest effect on performance.

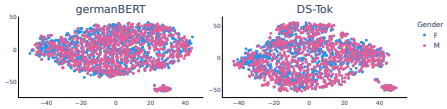
Combining results from Tables 5.2 and 5.3 with findings from the meta-regression analysis leads to the overall conclusion that gender-based language specialization of PLMs generally leads to downstream gains, whereas age-based specialization does not.



(a) Multilingual



(b) Monolingual (Denmark)

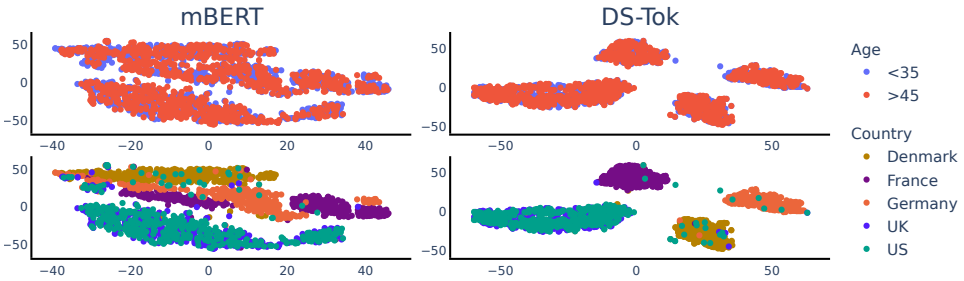


(c) Monolingual (Germany)

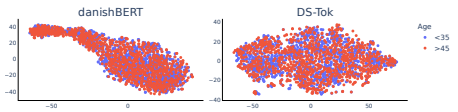
Figure 5.3: Results of our multilingual and monolingual qualitative analysis for *gender*. For multilingual case as plotted in (a), we show a tSNE visualization of review texts embedded with a non-specialized (mBERT) and specialized (DS-Tok) model. Colors indicate the demographic subgroup (upper figures) and countries (lower figures), respectively. For monolingual case as illustrated in (b) and (c) for Denmark and Germany, we show a tSNE visualization of texts embedded with non-specialized (danishBERT, germanBERT) and specialized (DS-Tok) monolingual PLMs. Each subfigure is plotted with 2K instances.

Qualitative Analysis. Finally, we analyze the topology of the PLMs representation space before and after demographic specialization. We encode the reviews from both demographic dimensions – (i) with the vanilla PLM (mBERT or monolingual BERT) and its DS-Tok specialized counterpart – and then compress those representations into two dimensions with t-distributed stochastic neighbor embedding (tSNE; van der Maaten and Hinton, 2008). Figure 5.3 depicts these representation spaces after gender-specialization (the age-specialization effects lead to similar conclusions; see Figure 5.4). The tSNE plots do not show any salient gender specialization effect. In the case of mBERT, gender-specialization (corresponding DS-Tok plot) leads to the separation of representation areas according to review language and not the *gender* of its author.[8] In the monolingual cases (illustrated for Danish and German BERT), the space of the gender-specialized encoder visually largely resembles that of the vanilla one, indicating that the demographic specialization procedure (DS-Tok) does not impart dimensions that allow for easy separation of representation space along the specialization dimension.

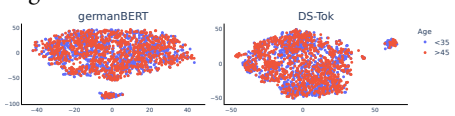
[8] Note that the **green** and **blue** regions, indicating US and UK overlap due to shared language.



(a) Multilingual



(b) Monolingual (Denmark)



(c) Monolingual (Germany)

Figure 5.4: Results of our multilingual and monolingual qualitative analysis for *age*. For multilingual case as plotted in (a), we show a tSNE visualization of review texts embedded with a non-specialized (mBERT) and specialized (DS-Tok) model. Colors indicate the demographic subgroup (upper figures) and countries (lower figures), respectively. For monolingual case as illustrated in (b) and (c) for Denmark and Germany, we show a tSNE visualization of texts embedded with non-specialized (danishBERT, germanBERT) and specialized (DS-Tok) monolingual PLMs. Each subfigure is plotted with 2K instances.

5.6 Conclusions

In this Chapter, we thoroughly examined the effects of demographic specialization of PLMs via straightforward injection methods that have been proven effective for other types of knowledge (**C1**). Initial results on intrinsic and extrinsic evaluation tasks using a multilingual PLM indicated the usefulness of our approach. However, running a series of additional experiments in which we controlled for potentially confounding factors (language and domain) and a meta-analysis indicated that the demographic aspects only have a negligible impact on the downstream task performance. This observation is supported by additional qualitative analysis (**C3**). Overall, our findings point to the difficulty of injecting demographic knowledge into PLMs: we hope that our in-depth analysis and findings catalyze future research on the topic of truly human-centered NLP, especially in multidimensional settings.

5.7 Further Ethical Considerations

In this Chapter, we concentrated on the demographic adaptation of PLMs, specifically addressing *gender* and *age* aspects while acknowledging the existence of other factors like *ethnicity* and *education* that aren't covered here. It is important to recognize that there might be additional effects and intersectional impacts at play. Our work deals with

demographic adaptation from reviews that should be considered sensitive information. We acknowledge that the limitations in data resources and annotations (Schler et al., 2006; Hovy et al., 2015; Voigt et al., 2018) give rise to potential risks of overgeneralizing our findings and applying our methods. We point the reader to the following risks and potential implications: (1) *partial language coverage*, where languages are from Indo-European subfamilies that do not represent typologically diverse languages; (2) *limited cultural coverage* (Joshi et al., 2020), where the countries, although speaking different languages, still belong a culturally relatively homogeneous part of the world, i.e., the West; (3) *simplified gender identities* (Dev et al., 2021), where gender is modeled as a binary variable, which does not reflect the wide variety of possible identities along the gender spectrum and beyond (Lauscher et al., 2022b); (4) *unfair stereotypical biases* (Blodgett et al., 2020), namely potential harms that might arise from unfair stereotypical biases in the data (despite our efforts to balance the sample across demographic groups) or pre-encoded in the model (Lauscher et al., 2021). Further, the sensitive user profile data might bias the model towards additional demographic characteristics and lead to potentially harmful predictions and applications.

In this Chapter, we have focused on advancing NLP research to understand better this fine-grained aspect of the intertwined relationship between demographic adaptation and PLMs in both monolingual and multilingual scenarios. While limited data resources may hinder our ability to fully consider *language coverage*, *cultural coverage*, *gender identities*, and *stereotypical biases*, it is our obligation to be transparent about these constraints and ethical concerns and to continually work towards improving data collection and methodologies to better serve the needs and perspectives of all users. We believe these insights would raise awareness towards establishing fairer and more inclusive language technologies for other demographic factors, other groups within these factors, and also other languages and countries – multidimensional adaptation. In the next and final Chapter, we summarize and wrap up the findings and contributions presented in this thesis.

Part III

Conclusions and Perspectives

Chapter 6

Conclusions and Perspectives

"In a time of drastic change it is the learners who inherit the future.
The learned usually find themselves equipped to live in a world that no longer exists."

Eric Hoffer
«Reflections on the Human Condition»

To address *adaptation barriers* emerging from the discrepancy between the generic knowledge encoded in pre-trained language models and the specific demands of real-world applications across multiple perspectives – domains, languages, and social dimensions, this thesis has undertaken a comprehensive investigation in bridging the critical gap. We targeted three key challenges to adapt language models across multidimensional aspects to enhance effectiveness, efficiency, and interpretability. The main contributions are summarized in the following:

1. Enhance Effectiveness in Adaptive Pre-training (C1). We explored *task-agnostic adaptive pre-training* methods, distinct from task-specific approaches that are limited to single tasks. We developed versatile models capable of handling various tasks across multiple domains and languages, including techniques for self-supervised domain adaptation (Chapter 3) and cross-lingual transfer (Chapter 4), as well as hybrid setups for demographic adaptation (Chapter 5). Further, we addressed the challenges of *multi-domain and multilingual use cases*. Recognizing the impracticality of using separate models for each domain and language, we propose a single-model approach, which enhances the adaptability of PLMs for multi-domain and multilingual scenarios. This involves using adapters (§ 3.1) and novel methods – meta-embeddings and meta-tokenizers (§ 3.2) for multi-domain adaptation, and multi-task learning objectives to incorporate demographic knowledge into multilingual models (Chapter 5). Lastly, we tackled *resource-limited* scenarios, crucial for languages or domains with limited labeled data. We delved into sample efficient few-shot transfer (Chapter 3) and continual few-shot cross-lingual transfer learning (Chapter 4) to enhance model adaptability in resource-lean contexts.

2. Improve Efficiency in Adaptive Pre-training (C2). In the realm of pre-trained language models, the advancement in size and complexity is paralleled by an increase in the computational demands for adaptive pre-training, raising concerns about practical deployment and scalability. We delved into enhancing the efficiency of adaptive pre-training by focusing on two critical areas: *data efficiency* and *parameter efficiency*. *Data-efficient* methods prioritize effective collection and use of limited in-domain or in-language data (around 50K to 200K), addressing the challenges of acquiring diverse datasets and reducing reliance on extensive labeled data. We proposed a term-matching method for efficient data acquisition for task-oriented dialog (DomainCC and DomainReddit) (see § 3.1). In Chapter 4, we introduced a novel multilingual multi-domain TOD datasets: Multi^{2}WOZ, enabling a reliable and robust resource to facilitate cross-lingual transfer studies on TOD. Further, we presented target-language-specific (LangCC) and cross-lingual (LangOpenSubtitles) corpora, streamlining efficient dialogic pre-training for language adaptation. In Chapter 5, we utilized a limited amount of language-specific reviews from two demographic factors of social dimension – gender and age, to conduct demographic adaptation. Regarding *parameter efficiency*, our work underlined the necessity of optimizing resource utilization to minimize computational overhead. We critiqued conventional *full fine-tuning* methods for their resource intensity consumption and introduced *modular-based* approaches for better parameter efficiency. We proposed a novel task-agnostic domain adaptation method using domain-specialized embeddings and tokenizers (see § 3.2). This approach presented a more efficient solution for domain-adaptive pre-training, particularly in situations with limited training data, and offers a more effective alternative to the use of adapters.

3. Enrich Interpretability in Adaptive Pre-training (C3). To understand the behavior and decision-making of language models in optimizing their performance and addressing limitations, we focused on the interpretability and analysis of adaptive pre-training methods across domains, languages, and social dimensions. Our objective was to improve model transparency while systematically identifying the strengths and weaknesses. We investigated the effects of domain and language adaptation through cross-domain transfer and token-level segmentation in Chapter 3 and Chapter 4. The analyses allowed us to gain insights into the benefits of cross-domain knowledge transfer on model behavior and quantify the knowledge encoded within the model through token-level control. Additionally, we explored the impact of demographic factors on a multilingual PLM through controlled experiments in Chapter 5. Our studies presented how the model performed across different demographic subgroups and shifted the research focus from merely assessing performance gains to a deeper understanding of the underlying mechanics and complexities of adaptive pre-training methods across multiple factors. Notably, our findings underscored the open challenge of injecting demographic knowledge into multilingual PLM. This opens up avenues for future research to explore diverse perspectives and insights across various tasks at hand.

To summarize, in this thesis we have emphasized the importance of overcoming the *adaptation barriers* encountered by PLMs across multidimensional aspects. We have described and addressed three core challenges and presented novel corpora, new methods, and comprehensive analyses across various downstream tasks. While our work only scratches the surface of these complex problems, it contributes to the development of more effective, efficient, and interpretable approaches to bridging the adaptation gap in PLMs. We advocate for a more holistic view, as it allows for better integration of the addressed challenges and identifies transferable aspects across different contexts. However, we acknowledge the limitations of our work, which relies on future exploration. First, there is a need for better selection and handling of high-quality data to enhance data efficiency and the reliability of model adaptation across diverse scenarios. Second, our focus was primarily on encoder-based PLMs, and there is significant potential in exploring a broader range of PLMs, particularly through a mixture-of-experts approach (Shazeer et al., 2017; Feng et al., 2024) that integrates expert PLMs and adapters in a task-agnostic manner for multi-domain and multilingual applications. Third, while our work has emphasized quantitative and qualitative analyses to enhance interpretability, incorporating human-centered approaches, particularly through preference alignment, could significantly enhance model transparency and usability, especially in domains like healthcare. The potential paths for future research based on our work are manifold. We outline the following possible directions for future work:

(i) Data Quality and Diversity. Data quality is one of the core factors when focusing on data-efficiency methods for addressing *adaptation barrier* of PLMs. This further highlights the research directions with the reliability of selecting high-quality data (Longpre et al., 2024). The possible directions of research include: (a) employing *influence functions* combined with k-Nearest-Neighbors search to optimize both speed and data quality (Guo et al., 2021b); (b) dynamically optimizing data usage during training, leveraging model gradients on a small subset of clean data (Wang et al., 2021), which are suggested to lead to better adaptation techniques (Grangier and Iter, 2022); or (c) utilizing a PLM to filter the data quality, which is similar as model-as-a-judge approach with learning objectives (Li et al., 2024a,b). Our study encompassed a diverse range of contexts, covering 14 domains, 7 languages, and 2 demographic factors (see § 1.2). However, we acknowledge that the covered selections may not contain the full picture of real-world applications, especially in specialized areas (e.g., legal, education) and resource-lean languages (e.g., Tagalog, Tamil). This acknowledgment underscores the need to expand future research scope to encompass a wider variety of domains and place a deliberate focus on low-resource languages (Chung et al., 2023; Samardzic et al., 2024), thereby extending our future investigations to cover a more comprehensive spectrum of real-world scenarios.

(ii) Task-Agnostic Unified Model for Multi-Domain and Multilingual Use Cases. We presented task-agnostic approaches to address *adaptation barrier* of PLMs, including multi-domain usage in Chapter 3, multilingual usage in Chapter 4, and a hybrid setup for multilingual demographic adaptation usage in Chapter 5. However, research questions are still open regarding the effects of deploying a unified model across both multi-domain and multilingual setups. Kulkarni et al. (2023) proposed a novel unified multi-domain and multilingual NER model, catering combination of language and domain via (a) adapters, and (b) mixture-of-experts (Shazeer et al., 2017) techniques. Their findings suggest that in a *task-specific* (i.e., NER) setting, domain-specific adaptations hold greater significance than language-specific ones. While this insight provides a greater understanding of the interplay between domains and languages in adaptive models, it also raises interesting avenues for further investigation of multidimensional scenarios: (i) exploring more effective and efficient methods for unified models to better handle the complexities of various dimensional adaptations, and (ii) developing *task-agnostic* unified models that are not limited to specific tasks with ensemble methods (e.g., *mixture-of-LoRAs* (Feng et al., 2024)) to enhance generalization across different applications.

(iii) Preference-Alignment Leads to Better Interpretability. We examined the interpretability aspects of adaptive pre-training frameworks across various tasks. However, it is crucial to acknowledge that our investigation primarily revolves around controlled experiments (e.g., zero-shot transfer, few-shot transfer, sample efficiency), leaving room for further exploration in the field of interpretability. Specifically, human-centered interpretability remains an area for future research, concerning both (a) visualization studies; and (b) incorporation of human-preference alignment. For instance, Kwon and Mihindukulasooriya (2023) developed a visual analytics application aimed at enabling users to scrutinize the fairness and biases of PLMs through MLM-scoring (Salazar et al., 2020). This approach holds the potential for extension to multidimensional aspects, with consideration for task-specific evaluation metrics. Furthermore, exploring methods to integrate human preferences into the interpretability process remains a promising avenue for future investigation (Kirk et al., 2023; Hung et al., 2023a). Aligning model behavior with user preferences has the potential to enhance the feasibility and relevance of downstream tasks (Kim et al., 2023; Rafailov et al., 2023; Deng et al., 2024). Nonetheless, the development of task-agnostic mechanisms for injecting specialized knowledge and incorporating user preference alignment for classification tasks remains a direction for future exploration. While efficiency and effectiveness remain primary concerns in assessing the performance of adaptive language models, emphasizing interpretability in human-centered applications holds promise for enhancing the transferability of acquired knowledge and fostering transparent trust between users, thereby maximizing the societal impact of these models.

In order to facilitate future research and ensure reproducibility, all the resources (including scripts, models, and corpora) in the context of the thesis are publicly available (see Appendix A). We hope that our work catalyzes future investigation into overcoming the *adaptation barrier* of PLMs. We aim to highlight the necessity for developing more effective, efficient, and interpretable approaches to adapting language models to multidimensional perspectives, ultimately leading to more robust and reliable NLP applications.

Bibliography

Armen Aghajanyan, Anchit Gupta, Akshat Shrivastava, Xilun Chen, Luke Zettlemoyer, and Sonal Gupta. 2021. Muppet: Massive multi-task representations with pre-finetuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5799–5811, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online. Association for Computational Linguistics.

Alan Ansell, Edoardo Ponti, Anna Korhonen, and Ivan Vulić. 2022. Composable sparse fine-tuning for cross-lingual transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1778–1796, Dublin, Ireland. Association for Computational Linguistics.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *Advances in Neural Information Processing Systems 2016 Deep Learning Symposium*.

David Bamman, Chris Dyer, and Noah A. Smith. 2014. Distributed representations of geographically situated language. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 828–834, Baltimore, Maryland. Association for Computational Linguistics.

David Bamman, Sejal Popat, and Sheng Shen. 2019. An annotated dataset of literary entities. 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)*, pages 2138–2144, Minneapolis, Minnesota. Association for Computational Linguistics.

Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. 2020. PLATO: Pre-trained dialogue generation model with discrete latent variable. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 85–96, Online. Association for Computational Linguistics.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1538–1548, Hong Kong, China. Association for Computational Linguistics.

Pranjali Basmatkar, Hemant Holani, and Shivani Kaushal. 2019. Survey on neural machine translation for multilingual translation system. In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pages 443–448. IEEE.

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset. In *Proceedings of the international AAAI conference on web and social media*, volume 14, pages 830–839.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Eyal Ben-David, Carmel Rabinovitz, and Roi Reichart. 2020. PERL: Pivot-based domain adaptation for pre-trained deep contextualized embedding models. *Transactions of the Association for Computational Linguistics*, 8:504–521.

Elad Ben Zaken, 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)*, pages 1–9, Dublin, Ireland. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. 2000. A neural probabilistic language model. *Advances in neural information processing systems*, 13.

Kasturi Bhattacharjee, Miguel Ballesteros, Rishita Anubhai, Smaranda Muresan, Jie Ma, Faisal Ladhak, and Yaser Al-Onaizan. 2020. To BERT or not to BERT: Comparing task-specific and task-agnostic semi-supervised approaches for sequence tagging. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7927–7934, Online. Association for Computational Linguistics.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Danushka Bollegala and James O' Neill. 2022. A survey on word meta-embedding learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 5402–5409. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Paweł Budzianowski and Ivan Vulić. 2019. Hello, it's GPT-2 - how can I help you? towards the use of pretrained language models for task-oriented dialogue systems. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 15–22, Hong Kong. Association for Computational Linguistics.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Sven Buechel, Anneke Buffone, Barry Slaff, Lyle Ungar, and João Sedoc. 2018. Modeling empathy and distress in reaction to news stories. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4758–4765, Brussels, Belgium. Association for Computational Linguistics.

Steven Cao, Nikita Kitaev, and Dan Klein. 2019. Multilingual alignment of contextual word representations. In *International Conference on Learning Representations*.

Rich Caruana. 1997. Multitask learning. *Machine learning*, 28:41–75.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational*

Linguistics: Human Language Technologies, pages 3576–3588, Online. Association for Computational Linguistics.

John Chung, Ece Kamar, and Saleema Amershi. 2023. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 575–593, Toronto, Canada. Association for Computational Linguistics.

Joshua Coates and Danushka Bollegala. 2018. Frustratingly easy meta-embedding – computing meta-embeddings by averaging source word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 194–198, New Orleans, Louisiana. Association for Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 160–167, New York, NY, USA. ACM.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(ARTICLE):2493–2537.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.

Yang Deng, Lizi Liao, Zhonghua Zheng, Grace Hui Yang, and Tat-Seng Chua. 2024. Towards human-centered proactive conversational agents. In *Proceedings of the 47th*

International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24, page 807–818, New York, NY, USA. Association for Computing Machinery.

Sunipa Dev, Masoud Monajatipoor, Anaelia Ovalle, Arjun Subramonian, Jeff Phillips, and Kai-Wei Chang. 2021. Harms of gender exclusivity and challenges in non-binary representation in language technologies. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1968–1994, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jacob Devlin, 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shizhe Diao, Ruijia Xu, Hongjin Su, Yilei Jiang, Yan Song, and Tong Zhang. 2021. Taming pre-trained language models with n-gram representations for low-resource domain adaptation. 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)*, pages 3336–3349, Online. Association for Computational Linguistics.

Bosheng Ding, Junjie Hu, Lidong Bing, Mahani Aljunied, Shafiq Joty, Luo Si, and Chunyan Miao. 2022. GlobalWoZ: Globalizing MultiWoZ to develop multilingual task-oriented dialogue systems. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1639–1657, Dublin, Ireland. Association for Computational Linguistics.

Finale Doshi-Velez and Been Kim. 2018. Considerations for evaluation and generalization in interpretable machine learning. *Explainable and interpretable models in computer vision and machine learning*, pages 3–17.

Penelope Eckert and Sally McConnell-Ginet. 2013. *Language and Gender*. Cambridge University Press.

Jeffrey L Elman. 1990. Finding structure in time. *Cognitive science*, 14(2):179–211.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 422–428, Marseille, France. European Language Resources Association.

Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R Arabnia. 2021. A brief review of domain adaptation. *Advances in data science and information engineering*, pages 877–894.

Wenfeng Feng, Chuzhan Hao, Yuewei Zhang, Yu Han, and Hao Wang. 2024. Mixture-of-LoRAs: An efficient multitask tuning method for large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11371–11380, Torino, Italia. ELRA and ICCL.

John Firth. 1957. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, pages 10–32.

Lucie Flek. 2020. Returning the N to NLP: Towards contextually personalized classification models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7828–7838, Online. Association for Computational Linguistics.

Robert M. French. 1999. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135.

Annemarie Friedrich, Heike Adel, Federico Tomazic, Johannes Hingerl, Renou Benteau, Anika Marusczyk, and Lukas Lange. 2020. The SOFC-exp corpus and neural approaches to information extraction in the materials science domain. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1255–1268, Online. Association for Computational Linguistics.

Zihao Fu, Haoran Yang, Anthony Man-Cho So, Wai Lam, Lidong Bing, and Nigel Collier. 2023. On the effectiveness of parameter-efficient fine-tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12799–12807.

Philip Gage. 1994. A new algorithm for data compression. *C Users Journal*, 12(2):23–38.

Aparna Garimella, Carmen Banea, and Rada Mihalcea. 2017. Demographic-aware word associations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2285–2295, Copenhagen, Denmark. Association for Computational Linguistics.

Goran Glavaš, Mladen Karan, and Ivan Vulić. 2020. XHate-999: Analyzing and detecting abusive language across domains and languages. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6350–6365, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Goran Glavaš and Ivan Vulić. 2021. Is supervised syntactic parsing beneficial for language understanding tasks? an empirical investigation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3090–3104, Online. Association for Computational Linguistics.

David Grangier and Dan Iter. 2022. The trade-offs of domain adaptation for neural language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3802–3813, Dublin, Ireland. Association for Computational Linguistics.

Bhanu Prakash Reddy Guda, Aparna Garimella, and Niyati Chhaya. 2021. EmpathBERT: A BERT-based framework for demographic-aware empathy prediction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3072–3079, Online. Association for Computational Linguistics.

Chulaka Gunasekara, Seokhwan Kim, Luis Fernando D'Haro, Abhinav Rastogi, Yun-Nung Chen, Mihail Eric, Behnam Hedayatnia, Karthik Gopalakrishnan, Yang Liu, Chao-Wei Huang, et al. 2020. Overview of the ninth dialog system technology challenge: Dstc9. *arXiv preprint arXiv:2011.06486*.

Demi Guo, Alexander Rush, and Yoon Kim. 2021a. 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)*, pages 4884–4896, Online. Association for Computational Linguistics.

Han Guo, Nazneen Rajani, Peter Hase, Mohit Bansal, and Caiming Xiong. 2021b. FastIF: Scalable influence functions for efficient model interpretation and debugging. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10333–10350, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xu Guo and Han Yu. 2022. On the domain adaptation and generalization of pretrained language models: A survey. *arXiv preprint arXiv:2211.03154*.

Raghav Gupta, Harrison Lee, Jeffrey Zhao, Yuan Cao, Abhinav Rastogi, and Yonghui Wu. 2022. Show, don't tell: Demonstrations outperform descriptions for schema-guided task-oriented dialogue. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4541–4549, Seattle, United States. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Ting Han, Ximing Liu, Ryuichi Takanabu, Yixin Lian, Chongxuan Huang, Dazhen Wan, Wei Peng, and Minlie Huang. 2021. Multiwoz 2.3: A multi-domain task-oriented dialogue dataset enhanced with annotation corrections and co-reference annotation. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 206–218. Springer.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jiawei Low, Lidong Bing, and Luo Si. 2021. On the effectiveness of adapter-based tuning for pretrained language model adaptation. 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)*, pages 2208–2222, Online. Association for Computational Linguistics.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2021. A survey on recent approaches for natural language processing in low-resource scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2545–2568, Online. Association for Computational Linguistics.

Matthew Henderson, Paweł Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulić, and Tsung-Hsien Wen. 2019a. A repository of conversational datasets. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 1–10, Florence, Italy. Association for Computational Linguistics.

Matthew Henderson, Iñigo Casanueva, Nikola Mrkšić, Pei-Hao Su, Tsung-Hsien Wen, and Ivan Vulić. 2020. ConveRT: Efficient and accurate conversational representations from transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2161–2174, Online. Association for Computational Linguistics.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014a. The second dialog state tracking challenge. In *Proceedings of the 15th Annual Meeting of the Special Interest*

Group on Discourse and Dialogue (SIGDIAL), pages 263–272, Philadelphia, PA, U.S.A. Association for Computational Linguistics.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014b. The third dialog state tracking challenge. In *2014 IEEE Spoken Language Technology Workshop (SLT)*, pages 324–329.

Matthew Henderson, Ivan Vulić, Iñigo Casanueva, Paweł Budzianowski, Daniela Gerz, Sam Coope, Georgios Spithourakis, Tsung-Hsien Wen, Nikola Mrkšić, and Pei-Hao Su. 2019b. PolyResponse: A rank-based approach to task-oriented dialogue with application in restaurant search and booking. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 181–186, Hong Kong, China. Association for Computational Linguistics.

Matthew Henderson, Ivan Vulić, Daniela Gerz, Iñigo Casanueva, Paweł Budzianowski, Sam Coope, Georgios Spithourakis, Tsung-Hsien Wen, Nikola Mrkšić, and Pei-Hao Su. 2019c. Training neural response selection for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5392–5404, Florence, Italy. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Valentin Hofmann, Goran Glavaš, Nikola Ljubešić, Janet B. Pierrehumbert, and Hinrich Schütze. 2024. Geographic adaptation of pretrained language models. *Transactions of the Association for Computational Linguistics*, 12:411–431.

Carolin Holtermann, Anne Lauscher, and Simone Paolo Ponzetto. 2022. Fair and argumentative language modeling for computational argumentation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7841–7861, Dublin, Ireland. Association for Computational Linguistics.

JJ Hopfield. 1982. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, 79(8):2554–2558.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. In *Advances in Neural Information Processing Systems*, volume 33, pages 20179–20191. Curran Associates, Inc.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Dirk Hovy. 2015. Demographic factors improve classification performance. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 752–762, Beijing, China. Association for Computational Linguistics.

Dirk Hovy, Anders Johannsen, and Anders Søgaard. 2015. User review sites as a resource for large-scale sociolinguistic studies. In *Proceedings of the 24th international conference on World Wide Web*, pages 452–461.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4411–4421. PMLR.

Songbo Hu, Xiaobin Wang, Moy Yuan, Anna Korhonen, and Ivan Vulić. 2024. DIALIGHT: Lightweight multilingual development and evaluation of task-oriented dialogue systems with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: System Demonstrations)*, pages 36–52, Mexico City, Mexico. Association for Computational Linguistics.

Songbo Hu, Han Zhou, Mete Hergul, Milan Gritta, Guchun Zhang, Ignacio Iacobacci, Ivan Vulić, and Anna Korhonen. 2023. Multi 3 WOZ: A multilingual, multi-domain, multi-parallel dataset for training and evaluating culturally adapted task-oriented dialog systems. *Transactions of the Association for Computational Linguistics*, 11:1396–1415.

Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020. Challenges in building intelligent open-domain dialog systems. *ACM Transactions on Information Systems (TOIS)*, 38(3):1–32.

Chia-Chien Hung, Wiem Ben Rim, Lindsay Frost, Lars Bruckner, and Carolin Lawrence. 2023a. Walking a tightrope – evaluating large language models in high-risk domains. In

Proceedings of the 1st GenBench Workshop on (Benchmarking) Generalisation in NLP, pages 99–111, Singapore. Association for Computational Linguistics.

Chia-Chien Hung, Lukas Lange, and Jannik Strötgen. 2023b. TADA: Efficient task-agnostic domain adaptation for transformers. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 487–503, Toronto, Canada. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Dirk Hovy, Simone Paolo Ponzetto, and Goran Glavaš. 2023c. Can demographic factors improve text classification? revisiting demographic adaptation in the age of transformers. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1565–1580, Dubrovnik, Croatia. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Simone Paolo Ponzetto, and Goran Glavaš. 2022a. DS-TOD: Efficient domain specialization for task-oriented dialog. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 891–904, Dublin, Ireland. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. 2022b. Multi2WOZ: A robust multilingual dataset and conversational pretraining for task-oriented dialog. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3687–3703, Seattle, United States. Association for Computational Linguistics.

Hamish Ivison, Noah A. Smith, Hannaneh Hajishirzi, and Pradeep Dasigi. 2023. Data-efficient finetuning using cross-task nearest neighbors. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9036–9061, Toronto, Canada. Association for Computational Linguistics.

Junguang Jiang, Yang Shu, Jianmin Wang, and Mingsheng Long. 2022. Transferability in deep learning: A survey. *arXiv preprint arXiv:2201.05867*.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Daniel Jurafsky and James H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 1st edition. Prentice Hall.

Kaliyaperumal Karthikeyan, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. Cross-lingual ability of multilingual bert: An empirical study. In *International Conference on Learning Representations*.

Nora Kassner, Philipp Dufter, and Hinrich Schütze. 2021. Multilingual LAMA: Investigating knowledge in multilingual pretrained language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3250–3258, Online. Association for Computational Linguistics.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491.

Douwe Kiela, Changhan Wang, and Kyunghyun Cho. 2018. Dynamic meta-embeddings for improved sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1466–1477, Brussels, Belgium. Association for Computational Linguistics.

Hyun Kim, Joon-Ho Lim, Hyun-Ki Kim, and Seung-Hoon Na. 2019. QE BERT: Bilingual BERT using multi-task learning for neural quality estimation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 85–89, Florence, Italy. Association for Computational Linguistics.

Jaehyung Kim, Jinwoo Shin, and Dongyeop Kang. 2023. Prefer to classify: Improving text classifiers via auxiliary preference learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 16807–16828. PMLR.

Su Nam Kim, Timothy Baldwin, and Min-Yen Kan. 2009. Extracting domain-specific words - a statistical approach. In *Proceedings of the Australasian Language Technology Association Workshop 2009*, pages 94–98, Sydney, Australia.

Hannah Kirk, Andrew Bean, Bertie Vidgen, Paul Rottger, and Scott Hale. 2023. The past, present and better future of feedback learning in large language models for subjective human preferences and values. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2409–2430, Singapore. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Mayank Kulkarni, Daniel Preotiuc-Pietro, Karthik Radhakrishnan, Genta Indra Winata, Shijie Wu, Lingjue Xie, and Shaohua Yang. 2023. Towards a unified multi-domain multilingual named entity recognition model. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2210–2219, Dubrovnik, Croatia. Association for Computational Linguistics.

Bum Chul Kwon and Nandana Mihindukulasooriya. 2023. Finspector: A human-centered visual inspection tool for exploring and comparing biases among foundation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 42–50, Toronto, Canada. Association for Computational Linguistics.

Wuwei Lan, Yang Chen, Wei Xu, and Alan Ritter. 2020. An empirical study of pre-trained transformers for Arabic information extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4727–4734, Online. Association for Computational Linguistics.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2021a. FAME: Feature-based adversarial meta-embeddings for robust input representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8382–8395, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2022. CLIN-X: pre-trained language models and a study on cross-task transfer for concept extraction in the clinical domain. *Bioinformatics*, 38(12):3267–3274.

Lukas Lange, Jannik Strötgen, Heike Adel, and Dietrich Klakow. 2021b. To share or not to share: Predicting sets of sources for model transfer learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8744–8753, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Anne Lauscher, Federico Bianchi, Samuel R. Bowman, and Dirk Hovy. 2022a. SocioProbe: What, when, and where language models learn about sociodemographics. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7901–7918, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Anne Lauscher, Archie Crowley, and Dirk Hovy. 2022b. Welcome to the modern world of pronouns: Identity-inclusive natural language processing beyond gender. In

Proceedings of the 29th International Conference on Computational Linguistics, pages 1221–1232, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Anne Lauscher, Goran Glavaš, Simone Paolo Ponzetto, and Kai Eckert. 2018. Investigating the role of argumentation in the rhetorical analysis of scientific publications with neural multi-task learning models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3326–3338, Brussels, Belgium. Association for Computational Linguistics.

Anne Lauscher, Tobias Lueken, and Goran Glavaš. 2021. Sustainable modular debiasing of language models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4782–4797, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Entony Lekhtman, Yftah Ziser, and Roi Reichart. 2021. DILBERT: Customized pre-training for domain adaptation with category shift, with an application to aspect extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 219–230, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. 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*, pages 7871–7880, Online. Association for Computational Linguistics.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020b. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Ming Li, Yong Zhang, Shwai He, Zhitao Li, Hongyu Zhao, Jianzong Wang, Ning Cheng, and Tianyi Zhou. 2024a. Superfiltering: Weak-to-strong data filtering for fast instruction-tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14255–14273, Bangkok, Thailand. Association for Computational Linguistics.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2024b. From quantity to quality: Boosting LLM performance with self-guided data selection for instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7602–7635, Mexico City, Mexico. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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)*, pages 4582–4597, Online. Association for Computational Linguistics.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.

Zhaojiang Lin, Andrea Madotto, Genta Winata, Peng Xu, Feijun Jiang, Yuxiang Hu, Chen Shi, and Pascale N Fung. 2021. Bitod: A bilingual multi-domain dataset for task-oriented dialogue modeling. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).

Robert Litschko, Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. 2022. On cross-lingual retrieval with multilingual text encoders. *Information Retrieval Journal*, pages 1–35.

Jiexi Liu, Ryuichi Takanobu, Jiaxin Wen, Dazhen Wan, Hongguang Li, Weiran Nie, Cheng Li, Wei Peng, and Minlie Huang. 2021a. Robustness testing of language understanding in task-oriented dialog. 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)*, pages 2467–2480, Online. Association for Computational Linguistics.

Qi Liu, Lei Yu, Laura Rimell, and Phil Blunsom. 2021b. Pretraining the noisy channel model for task-oriented dialogue. *Transactions of the Association for Computational Linguistics*, 9:657–674.

Qianchu Liu, Diana McCarthy, Ivan Vulić, and Anna Korhonen. 2019a. Investigating cross-lingual alignment methods for contextualized embeddings with token-level evaluation. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 33–43, Hong Kong, China. Association for Computational Linguistics.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019b. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019c. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zihan Liu, Feijun Jiang, Yuxiang Hu, Chen Shi, and Pascale Fung. 2021c. Ner-bert: a pre-trained model for low-resource entity tagging. *arXiv preprint arXiv:2112.00405*.

Shayne Longpre, Gregory Yauney, Emily Reif, Katherine Lee, Adam Roberts, Barret Zoph, Denny Zhou, Jason Wei, Kevin Robinson, David Mimno, and Daphne Ippolito. 2024. A pretrainer's guide to training data: Measuring the effects of data age, domain coverage, quality, & toxicity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human*

Language Technologies (Volume 1: Long Papers), pages 3245–3276, Mexico City, Mexico. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Kate Loveys, Jonathan Torrez, Alex Fine, Glen Moriarty, and Glen Coppersmith. 2018. Cross-cultural differences in language markers of depression online. In *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*, pages 78–87, New Orleans, LA. Association for Computational Linguistics.

Siwen Luo, Hamish Ivison, Soyeon Caren Han, and Josiah Poon. 2024. Local interpretations for explainable natural language processing: A survey. *ACM Computing Surveys*, 56(9):1–36.

Pekka Malo, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796.

Michael McCloskey and Neal J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. volume 24 of *Psychology of Learning and Motivation*, pages 109–165. Academic Press.

Shikib Mehri, Evgeniia Razumovskaia, Tiancheng Zhao, and Maxine Eskenazi. 2019. Pretraining methods for dialog context representation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3836–3845, Florence, Italy. Association for Computational Linguistics.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, 56(2):1–40.

Nikita Moghe, Evgeniia Razumovskaia, Liane Guillou, Ivan Vulić, Anna Korhonen, and Alexandra Birch. 2023a. Multi3NLU++: A multilingual, multi-intent, multi-domain dataset for natural language understanding in task-oriented dialogue. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3732–3755, Toronto, Canada. Association for Computational Linguistics.

Nikita Moghe, Tom Sherborne, Mark Steedman, and Alexandra Birch. 2023b. Extrinsic evaluation of machine translation metrics. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13060–13078, Toronto, Canada. Association for Computational Linguistics.

Nikita Moghe, Mark Steedman, and Alexandra Birch. 2021. Cross-lingual intermediate fine-tuning improves dialogue state tracking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1137–1150, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. 2017a. Neural belief tracker: Data-driven dialogue state tracking. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1777–1788, Vancouver, Canada. Association for Computational Linguistics.

Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017b. Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the Association for Computational Linguistics*, 5:309–324.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Marinela Parović, Goran Glavaš, Ivan Vulić, and Anna Korhonen. 2022. BAD-X: Bilingual adapters improve zero-shot cross-lingual transfer. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1791–1799, Seattle, United States. Association for Computational Linguistics.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2021. Soloist: Building task bots at scale with transfer learning and machine teaching. *Transactions of the Association for Computational Linguistics*, 9:807–824.

Matthew E. Peters, 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)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Sebastian Ruder, Ivan Vulić, and Edoardo Ponti. 2023. Modular deep learning. *Transactions on Machine Learning Research*. Survey Certification.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online. Association for Computational Linguistics.

Yao Qiu, Jinchao Zhang, and Jie Zhou. 2021. Different strokes for different folks: Investigating appropriate further pre-training approaches for diverse dialogue tasks. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2318–2327, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Osman Ramadan, Paweł Budzianowski, and Milica Gašić. 2018. Large-scale multi-domain belief tracking with knowledge sharing. In *Proceedings of the 56th Annual Meeting of*

the Association for Computational Linguistics (Volume 2: Short Papers), pages 432–437, Melbourne, Australia. Association for Computational Linguistics.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in NLP—A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8689–8696.

Evgeniia Razumovskaia, Goran Glavaš, Olga Majewska, Edoardo M. Ponti, Anna Korhonen, and Ivan Vulić. 2022. Crossing the conversational chasm: A primer on natural language processing for multilingual task-oriented dialogue systems. *Journal of Artificial Intelligence Research*, 74:1351–1402.

Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. 2017. Learning multiple visual domains with residual adapters. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Nils Reimers 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 (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2021. The curse of dense low-dimensional information retrieval for large index sizes. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 605–611, Online. Association for Computational Linguistics.

Andreas Rücklé, Gregor Geigle, Max Glockner, Tilman Beck, Jonas Pfeiffer, Nils Reimers, and Iryna Gurevych. 2021. AdapterDrop: On the efficiency of adapters in transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7930–7946, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sebastian Ruder. 2019. *Neural Transfer Learning for Natural Language Processing*. Ph.D. thesis, National University of Ireland, Galway.

Sebastian Ruder and Barbara Plank. 2018. Strong baselines for neural semi-supervised learning under domain shift. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1054, Melbourne, Australia. Association for Computational Linguistics.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. Association for Computational Linguistics.

Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90, Parramatta, Australia.

Tanja Samardzic, Ximena Gutierrez, Christian Bentz, Steven Moran, and Olga Pelloni. 2024. A measure for transparent comparison of linguistic diversity in multilingual NLP data sets. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3367–3382, Mexico City, Mexico. Association for Computational Linguistics.

Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. 2006. Effects of age and gender on blogging. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Yixuan Su, Lei Shu, Elman Mansimov, Arshit Gupta, Deng Cai, Yi-An Lai, and Yi Zhang. 2022. Multi-task pre-training for plug-and-play task-oriented dialogue system. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4661–4676, Dublin, Ireland. Association for Computational Linguistics.

Zhewei Sun, Qian Hu, Rahul Gupta, Richard Zemel, and Yang Xu. 2024. Toward informal language processing: Knowledge of slang in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1683–1701, Mexico City, Mexico. Association for Computational Linguistics.

Wilson L Taylor. 1953. "cloze procedure": A new tool for measuring readability. *Journalism quarterly*, 30(4):415–433.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Marcos Treviso, Ji-Ung Lee, Tianchu Ji, Betty van Aken, Qingqing Cao, Manuel R. Ciosici, Michael Hassid, Kenneth Heafield, Sara Hooker, Colin Raffel, Pedro H. Martins, André F. T. Martins, Jessica Zosa Forde, Peter Milder, Edwin Simpson, Noam Slonim, Jesse Dodge, Emma Strubell, Niranjan Balasubramanian, Leon Derczynski, Iryna Gurevych, and Roy Schwartz. 2023. Efficient Methods for Natural Language Processing: A Survey. *Transactions of the Association for Computational Linguistics*, 11:826–860.

Peter Trudgill. 2000. *Sociolinguistics: An introduction to language and society*. Penguin UK.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Mina Valizadeh and Natalie Parde. 2022. The AI doctor is in: A survey of task-oriented dialogue systems for healthcare applications. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6638–6660, Dublin, Ireland. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Rob Voigt, David Jurgens, Vinodkumar Prabhakaran, Dan Jurafsky, and Yulia Tsvetkov. 2018. RtGender: A corpus for studying differential responses to gender. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Svitlana Volkova, Theresa Wilson, and David Yarowsky. 2013. Exploring demographic language variations to improve multilingual sentiment analysis in social media. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1815–1827, Seattle, Washington, USA. Association for Computational Linguistics.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. 2018. Generalizing to unseen domains via adversarial data augmentation. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Xinyi Wang, Ankur Bapna, Melvin Johnson, and Orhan Firat. 2021. Gradient-guided loss masking for neural machine translation. *arXiv preprint arXiv:2102.13549*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference*

on Empirical Methods in Natural Language Processing, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Charles Welch, Jonathan K. Kummerfeld, Verónica Pérez-Rosas, and Rada Mihalcea. 2020. Compositional demographic word embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4076–4089, Online. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Taesun Whang, Dongyub Lee, Chanhee Lee, Kisu Yang, Dongsuk Oh, and HeuiSeok Lim. 2020. An effective domain adaptive post-training method for bert in response selection. In *Proc. Interspeech 2020*, pages 1585–1589.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. 2020. TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 917–929, Online. Association for Computational Linguistics.

Han Wu, Kun Xu, Linfeng Song, Lifeng Jin, Haisong Zhang, and Linqi Song. 2021. Domain-adaptive pretraining methods for dialogue understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 665–669, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods*

in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 496–505, Vancouver, Canada. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Zhao Yan, Nan Duan, Peng Chen, Ming Zhou, Jianshe Zhou, and Zhoujun Li. 2017. Building task-oriented dialogue systems for online shopping. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4618–4625. AAAI Press.

Yi Yang, Mark Christopher Siy UY, and Allen Huang. 2020. Finbert: A pretrained language model for financial communications. *arXiv preprint arXiv:2006.08097*.

Wenpeng Yin and Hinrich Schütze. 2016. Learning word meta-embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1351–1360, Berlin, Germany. Association for Computational Linguistics.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. MultiWOZ 2.2 : A dialogue dataset with additional annotation corrections and state tracking baselines. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 109–117, Online. Association for Computational Linguistics.

Yan Zeng and Jian-Yun Nie. 2020. Jointly optimizing state operation prediction and value generation for dialogue state tracking. *arXiv preprint arXiv:2010.14061*.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.

Zhihan Zhang, Wenhao Yu, Mengxia Yu, Zhichun Guo, and Meng Jiang. 2023. A survey of multi-task learning in natural language processing: Regarding task relatedness and training methods. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 943–956, Dubrovnik, Croatia. Association for Computational Linguistics.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. 2023. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415.

Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. 2020. The design and implementation of XiaoIce, an empathetic social chatbot. *Computational Linguistics*, 46(1):53–93.

Qi Zhu, Kaili Huang, Zheng Zhang, Xiaoyan Zhu, and Minlie Huang. 2020. CrossWOZ: A large-scale Chinese cross-domain task-oriented dialogue dataset. *Transactions of the Association for Computational Linguistics*, 8:281–295.

Xiaojin Zhu and Andrew B Goldberg. 2009. Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–130.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

Lei Zuo, Kun Qian, Bowen Yang, and Zhou Yu. 2021. Allwoz: Towards multilingual task-oriented dialog systems for all. *CoRR*, abs/2112.08333.

Appendix A

Published Resources

We provide an overview of the resources (code, data, model) published in the context of this thesis in Table A.1.

Chapter	Resource Name	Type	Access
Chapter 3	DS-TOD (§ 3.1) DomainCC & DomainReddit (§ 3.1.3)	Code Data	GitHub: umanlp/DS-TOD
	TADA (§ 3.2)	Code	GitHub: boschresearch/TADA
Chapter 4	Multilingual Dialog Adaptation (§ 4.1) Multi^{2}WOZ (§ 4.3) LangCC & LangOpenSubtitles (§ 4.4.2)	Code Data Data	GitHub: umanlp/Multi^{2}WOZ
	TOD-XLMR (§ 4.4.1)	Model	HuggingFace: umanlp/TOD-XLMR
Chapter 5	Demographic Adaptation (§ 5.1) Multilingual Demographic Dataset (§ 5.4)	Code Data	GitHub: umanlp/SocioAdapt

Table A.1: Overview of all resources published in the context of this thesis.

Appendix B

Experimental Details for Chapter 3

B.1 Domain-Specific Corpora

In this Section, we show examples from both DomainCC (Table B.1) and DomainReddit (Table B.2). Both resources were created starting from the salient domain terms listed in Table 3.2.

Domain	"Flat" Text
Taxi	***Taxis**: licensed black **cabs** operate a 24-hour, 365 day service from directly outside the arrivals area of the terminal building. Each **taxi** can carry up to five passengers (some can carry up to eight), with luggage and all are able to take wheelchair passengers.*
Restaurant	***Asian food** is very easy to like because it hits your mouth very differently than **European food** does. In **European food**, there may be two things to hit - maybe sweet and salty, maybe salty-savory, but Asian kind of works around, plus you have that distinct in the evening, a five course wine tasting dinner will be served in a gastronomic 2 Michelin starred **restaurant**.*
Train	*Getting to centre **London** is very easy as it take only one underground **train** and it takes only 20-25 minutes to get to Oxford Circus. **Stansted airport** is only 31 minutes away and all major motorways (M1, M11, North circular) is 5-10 minutes away.*
Hotel	*Beautifully restored 1920's **guesthouse**, comfortable and spacious bedrooms, lush gardens to explore, friendly and super helpful host, secure **parking**. What more could you ask for! I would definitely recommend 6 on Kloof.*
Attraction	*On 31 august we travelled to Ely by train from kings cross and visited the Cathedral's interesting stained glass **museum**. We also visited Oliver Cromwell's house nearby and sat outside for lunch, an extra bonus as it was a beautiful summer day. There was also time to look around Ely's **town centre** before heading home.*

Table B.1: Example from DomainCC dataset, where the salient domain terms are marked as **bold**. The texts are displayed in the original version, without correcting typos.

Domain	Dialogic Data	
Taxi	**Context:** *I wager that is majorly low. All the **taxi** drivers around me drive brand new hybrid **Lexus**'s. If you consider the fuel, cars upkeep, the car itself and the insurance. They must be owning a good scoop to make all that worth it.*	**True Response:** *A lot of **taxi** drivers round my way are working two jobs and have it as their second gig filling in what little free time they have from their main job.* **False Response:** *Buying vehicles (converged ones in my fathers case) is a huge expense which I don't think can be fully tax offset.*
Restaurant	**Context:** *Interesting. Thanks for the post and thanks for mentioning Normandie. I will definitely check that out and look at staying somewhere other than Zocalo. Any other recommendations for stuff you really liked? I'm a huge food guy so any awesome **restaurants** (already have a Pujol **reservation** are) welcome.*	**True Response:** *You're welcome. Thanks for reading. Don't get me wrong Zocalo has some historic significance etc. and is nice to visit for the day, but that's about all the time you need there. For some **cheap** but still good tacos, ...* **False Response:** *Zocalo is hectic and filled with tons of people. IMO after 1 day there you'll want out. Roma Norte and Condesa have some beautiful parks and are filled with cool cafes, restaurants and bars ...*
Train	**Context:** *You just need to hope you don't need to walk all the way to the back of the **train**.*	**True Response:** *I have to do I multiple times a day with the TGV's. Those are only 200m short. I don't working on this 400m **train** often. But yes it happens.* **False Response:** *We need a sub for European trains!*
Hotel	**Context:** *Thanks for the info. I didn't book a **hotel** yet and plan to do that by tomorrow. Wasn't aware that most don't have **free parking**. I'll try to find one with **parking** included.*	**True Response:** *You are not likely to find a **hotel** with **free parking** in the old city. And, to be honest, unless budget is a big deal, for a short trip it's entirely worth the experience to stay in either the upper or lower old city ...* **False Response:** *Where is your hotel? Many either have parking, or arrangements for parking in nearby lots and garages. If you're at or near the Frontenac there is a public garage under city hall that is much less expensive than many hotel options.*
Attraction	**Context:** *Thank youuu! I'll better pack a coat to keep myself warm! Hmm you're right I might just skip the day trip! I like history/ **museum**, art, **architecture** and scenery/nature! What are the top few places do you recommend though?*	**True Response:** *In terms of **museums** and history, you're really spoilt for choice in London. The Natural History **Museum**, Imperial War **Museum** and National Maritime **Museum** are my personal favourites. If you like nature go check out the wildlife in Richmond **Park**. It's a ...* **False Response:** *The UK is due to be extremely cold this winter so I'd have some extra warm clothes just in case. November is usually fine, a bit rainy, but this year might be a special case. You can visit Camden but I personally wouldn't spend ...*

Table B.2: Example from DomainReddit dataset, where the salient domain terms are marked as **bold**. The texts are displayed in the original version, without correcting typos.

B.2 Computational Information

Detailed explanations of hyperparameters setups are provided in § 3.2.4. In our conducted experiments, we only search for the learning rate in domain-adaptive pre-training. The best learning rate depends on the selected domains and methods for each task. All the experiments are performed on Nvidia Tesla V100 GPUs with 32GB VRAM and run on a carbon-neutral GPU cluster. The number of parameters and the total computational budget for domain-adaptive pre-training (in GPU hours) are shown in Table B.3.

Model	**# Trainable Parameters**	**MLM Budget (in GPU hours)**
BERT‡ (MLM-FULL)	∼110 M	∼5.5h (NER and NLI), 7.5h (DST and RR)
BERT‡ (MLM-ADAPT)	∼0.9 M	∼2.5h (NER and NLI), 3.5h (DST and RR)
BERT‡ (MLM-EMB)	∼24 M	∼3.5h (NER and NLI), 4.5h (DST and RR)

Table B.3: Overview of the computational information for the domain-adaptive pre-training. ‡BERT variants: BERT (NLI, NER) and TOD-BERT (DST, RR).

B.3 Few-Shot Learning Results for NLI

We provide the results of few-shot learning on NLI (Williams et al., 2018) task in Table B.4. We report the results for 1% and 20% of the training data size in both single-domain and multi-domain scenarios.

		Government		Telephone		Fiction		Slate		Travel		Avg.	
	Model	1%	20%	1%	20%	1%	20%	1%	20%	1%	20%	1%	20%
SD	BERT	57.62±5.4	75.21±.4	49.20±1.9	74.45±.3	43.76±2.2	72.90±.3	46.70±2.1	67.71±.5	54.05±4.0	71.55±.4	50.27±2.4	72.36±.1
SD	BERT (MLM-FULL)	**61.92**±1.8	76.07±.7	**54.53**±1.6	75.07±.7	49.32±1.4	**73.21**±.6	45.81±0.7	67.26±.6	56.56±3.5	72.50±.4	53.63±0.5	72.82±.4
SD	BERT (MLM-ADAPT)	42.88±1.8	67.93±.2	41.27±1.1	65.80±.2	38.12±1.7	59.53±.4	38.91±2.1	54.71±.7	40.74±2.8	65.89±.6	40.38±1.5	62.78±.7
SD	BERT (MLM-EMB)	61.66±1.0	**76.61**±.3	49.86±0.8	**75.33**±.3	48.35±4.1	72.22±.6	**49.10**±2.5	**68.26**±.3	**60.27**±1.6	**72.73**±.6	**53.85**±1.7	**73.03**±.1
SD	BERT (MLM-EMBTOK-X)	61.27±1.8	75.75±.5	49.20±5.5	74.11±.1	**49.74**±0.8	72.26±.8	**49.10**±1.9	66.51±.8	58.99±2.3	72.15±.8	53.66±2.0	72.16±.1
MD	BERT	69.56±3.2	79.49±.7	64.80±2.0	77.72±.2	61.53±2.5	76.84±.7	61.43±2.0	72.64±.4	66.40±2.9	**76.42**±.5	64.74±1.8	76.62±.2
MD	(AVG) BERT (MLM-EMBs)	70.13±1.3	**80.00**±.2	64.39±1.3	78.28±.2	**62.24**±1.7	76.94±.4	**62.61**±1.6	71.61±.3	**66.45**±1.4	76.21±.4	65.16±1.3	76.61±.1
MD	(ATT) BERT (MLM-EMBs)	**71.21**±1.1	79.90±.3	**65.56**±1.4	**78.48**±.1	61.33±1.3	**77.34**±.3	61.99±1.3	**72.69**±.4	66.24±1.7	76.32±.5	**65.27**±1.6	**76.95**±.2

Table B.4: Few-shot learning results on NLI task for 1% and 20% of the training data size in single-domain (SD) and multi-domain (MD) scenarios. We report mean and standard deviation of 3 runs with different random seeds.

B.4 Per-Domain Results for Meta-Tokenizers

We provide the results for each domain in our multi-domain experiments with meta-tokenizers and meta-embeddings in Table B.5 for DST and RR, and in Table B.6 for NLI and NER.

	DST						RR					
Model	**Taxi**	**Restaurant**	**Hotel**	**Train**	**Attraction**	**Avg.**	**Taxi**	**Restaurant**	**Hotel**	**Train**	**Attraction**	**Avg.**
(AVG) TOD-BERT (MLM-EMBs)	35.42	46.71	40.82	**52.34**	47.30	44.52	**55.20**	**64.58**	**60.39**	**62.84**	**66.11**	**61.82**
(ATT) TOD-BERT (MLM-EMBs)	37.35	**46.98**	**41.32**	51.92	**47.88**	**45.09**	53.73	64.00	59.89	61.54	65.05	60.84
(AVG) TOD-BERT‡ (dynamic)	32.06	44.12	40.54	49.89	44.21	42.16	52.84	62.54	58.26	61.24	64.46	59.87
(AVG) TOD-BERT‡ (space)	31.35	44.89	37.27	49.47	44.86	41.57	51.59	62.46	56.44	60.21	61.99	58.54
(AVG) TOD-BERT‡ (truncation)	33.61	43.88	38.20	44.24	41.35	40.26	52.55	61.19	55.55	58.58	62.47	58.07
(ATT) TOD-BERT‡ (dynamic)	34.06	45.01	39.73	50.11	44.73	42.73	51.22	62.08	58.04	61.39	63.35	59.22
(ATT) TOD-BERT‡ (space)	30.19	42.57	40.23	49.84	44.41	41.45	51.51	61.64	57.30	60.91	63.41	58.95
(ATT) TOD-BERT‡ (truncation)	31.45	43.44	37.08	48.13	44.02	40.82	51.59	62.63	57.97	60.66	62.62	59.09

Table B.5: Results of meta-tokenizers in multi-domain experiments with meta-embeddings on two downstream tasks: DST and RR, with joint goal accuracy (%) and $R_{100}@1$ (%) as evaluation metric, respectively. Three meta-tokenization aggregation methods: dynamic, space, truncation, are combined with two meta-embeddings approaches: average (AVG), attention-based (ATT). ‡Domain-specialized tokenizers in use based on MLM-EMBTOKs-X, referring to § 3.2.5.1 and Table 3.10.

	NLI						NER					
Model	**Government**	**Telephone**	**Fiction**	**Slate**	**Travel**	**Avg.**	**Financial**	**Fiction**	**News**	**Clinical**	**Science**	**Avg.**
(AVG) BERT (MLM-EMBs)	**83.80**	80.87	81.70	**77.60**	**81.30**	**81.05**	87.72	68.78	90.16	85.68	78.22	82.11
(ATT) BERT (MLM-EMBs)	83.50	81.64	**81.74**	76.68	80.36	80.78	**88.89**	69.05	**90.56**	85.43	**80.55**	**82.90**
(AVG) BERT‡ (dynamic)	81.08	79.81	80.44	75.35	78.80	79.10	83.26	59.70	75.93	70.42	64.33	70.73
(AVG) BERT‡ (space)	81.90	81.33	80.49	75.14	78.69	79.51	83.68	61.68	76.39	70.78	60.61	70.63
(AVG) BERT‡ (truncation)	81.44	81.38	79.17	75.86	79.50	79.47	77.99	53.53	74.37	67.08	60.33	66.66
(ATT) BERT‡ (dynamic)	81.70	80.62	80.33	74.78	79.15	79.32	84.64	59.98	76.08	71.30	62.17	70.83
(ATT) BERT‡ (space)	83.34	81.43	80.23	74.83	79.81	79.93	83.70	62.03	76.04	71.54	60.22	70.71
(ATT) BERT‡ (truncation)	82.37	81.64	78.81	75.65	79.90	79.67	80.33	58.80	74.49	66.92	61.51	68.41

Table B.6: Results of meta-tokenizers in multi-domain experiments with meta-embeddings on two downstream tasks: NLI and NER, with accuracy (%) and F_1 (%) as the evaluation metric, respectively. Three meta-tokenization aggregation methods: dynamic, space, truncation, are combined with two meta-embeddings approaches: average (AVG), attention-based (ATT). ‡Domain-specialized tokenizers in use based on MLM-EMBTOKs-X, referring to § 3.2.5.1 and Table 3.10.

Appendix C

Experimental Details for Chapter 4

C.1 Annotation Guidelines: Post-Editing of the Translation

C.1.1 Task Description

Multi-domain Wizard-of-Oz dataset (MultiWOZ) (Budzianowski et al., 2018) is introduced as a fully-labeled collection of human-to-human written conversations spanning over multiple domains and topics.

Our project aims to translate the monolingual English-only MultiWOZ dataset to four linguistically diverse major world languages, each with a different script: Arabic (AR), Chinese (ZH), German (DE), and Russian (RU).

In this annotation task, we resort to the revised version 2.1 (Eric et al., 2020) and focus on the development and test portions of the English MultiWOZ 2.1 (in total of 2,000 dialogs containing a total of 29.5K utterances). We first *automatically translate* all the utterances and the annotated slot values to the four target languages, using Google Translate. Next the translated utterances and slot values (i.e., fix the translation errors) will be *post-edited* with manual efforts.

For this purpose, a JSON file for *development* or *test* set will be provided to each annotator. There are two tasks: (1) Fix the errors in automatic translations of translated utterances and the translated slot values. (2) Check the alignment between each translated utterance and the slot value annotations for that utterance.

C.1.2 JSON Representation

The JSON file will be structured as follows, feel free to use any JSON editor tools (e.g., JSON Editor Online) to annotate the files.

Annotation Data

- **dialogID**: An unique ID for each dialog.
- **turnID**: The turn ID of the utterance in the dialog.
- **services**: Domain(s) of the dialog.
- **utterance**: English utterance from MULTIWOZ.
- **SlotValues**: English annotated slot values from MULTIWOZ.
- **transUtterance**: Translated utterance from Google Translate.
- **transSlotValues**: Translated slot values from Google Translate.

Annotation Task

- **fixTransUtterance**: The revised translated utterance with manual efforts.
- **fixTransSlotValues**: The revised translated slot values with manual efforts.
- **changedUtterance**: Whether the translated utterance is changed. Annotate as 1 if the translated utterance is revised, 0 otherwise.
- **changedSlotValues**: Whether the translated slot values is changed. Annotate as 1 if the translated slot values are revised, 0 otherwise.

C.1.3 Annotation Examples

Example 1: Name Correction and Mismatch
The following example in Chinese shows the error fixed with the translated name issue, and also the correctness of the mismatch case between the translated utterance and translated slot values.

dialogID: *MUL0484.json*
turnID: *6*
services: *train, attraction*
utterance: *No hold off on booking for now. Can you help me find an attraction called cineworld cinema?*
slotValues: {attraction-name: *cineworld cinema*}
transUtterance: 目前暂无预订。您能帮我找到一个名为cineworld Cinema的景点吗?
transSlotValues: {attraction-name: Cineworld电影}

---

fixTransUtterance: 目前暂无预订。您能帮我找到一个名为电影世界电影院的景点吗?
fixTransSlotValues: {attraction-name: 电影世界电影院}
changedUtterance: 1
changedSlotValues: 1

Example 2: Grammatical Error
The following example in German shows the error corrected based on the grammatical issue of the translated utterance.

dialogID: *PMUL1072.json*
turnID: *6*
services: *train, attraction*
utterance: *I'm leaving from Cambridge.*
slotValues: {train-departure: *cambridge*}
transUtterance: *Ich verlasse Cambridge.*
transSlotValues: {train-departure: *cambridge*}

---

fixTransUtterance: *Ich fahre von Cambridge aus.*
fixTransSlotValues: {train-departure: *cambridge*}
changedUtterance: 1
changedSlotValues: 0

C.1.4 Additional Notes

There might be some cases of synonyms. For example, in Chinese 周五 and 星期五 both have the same meaning as *Friday* in English, also similarly in Russian regarding the weekdays. In this case, just pick the most common one and stay consistent among all the translated utterances and slot values. Besides there might be some language variations across different regions, please ignore the dialects and metaphors while fixing the translation errors.

If there are any open questions that you think are not covered in this guide, please do not hesitate to get in touch with me or post the questions on Slack, so these issues can be discussed together with other annotators and the guide can be improved.

C.2 Annotation Guidelines: Quality Control

C.2.1 Task Description

Multi-domain Wizard-of-Oz dataset (MULTIWOZ) (Budzianowski et al., 2018) is introduced as a fully-labeled collection of human-to-human written conversations spanning over multiple domains and topics. Our project is aimed to translate the monolingual English-only MULTIWOZ dataset to four linguistically diverse major world languages, each with a different script: Arabic (AR), Chinese (ZH), German (DE), and Russian (RU). In the previous annotation task, we resorted to the revised version 2.1 (Eric et al., 2020) and focused on the development and test portions of the English MULTIWOZ 2.1.

According to the translation process, it was processed in two steps: we first *automatically translated* all the utterances and the annotated slot values to the four target languages, using Google Translate. Next the translated utterances and slot values (i.e., fix the translation errors) were *post-edited* with manual efforts from native speakers of each language.

Additionally, a *quality assurance* step is required to check the quality of the post-edited translation. For this purpose, a JSON file for a random sample 200 dialogs (100 from the development and test set each), containing 2,962 utterances in total will be provided to two annotators for each target language to judge the correctness of the translations. Each annotator has to independently answer the following questions for each translated utterance from the sample: (1) *Is the utterance translation acceptable?* (2) *Do the translated slot values match the translated utterance?*

Annotation Data

- **dialogID**: An unique ID for each dialog.
- **turnID**: The turn ID of the utterance in the dialog.
- **utterance**: English utterance from MULTIWOZ.
- **SlotValues**: English annotated slot values from MULTIWOZ.
- **fixTransUtterance**: The revised translated utterance with manual efforts.
- **fixTransSlotValues**: The revised translated slot values with manual efforts.

Annotation Task

- **UtteranceAcceptable**: Is the utterance translation acceptable? Annotate as 1 if the translated utterance is acceptable, 0 otherwise.
- **SlotValuesMatchAcceptable**: Do the translated slot values match the translated utterance? Annotate as 1 if the translated slot values are acceptable, 0 otherwise.
- **NOTE**: Extra notes of judgement.

C.2.2 Annotation Examples

Small grammatical errors, but still catching the meaning will be considered *acceptable*. However, if the whole meaning regarding the translation changes, it will then be considered as *not acceptable*.

Example 1: Ambiguity
The following example shows the ambiguity issues regarding the translated utterance. In German, *table* can be translated into *Tabelle* as a table form or *Tisch* as a table for reservation. Regarding the contextual information from the utterance, the correct translation should be *Tisch* instead of *Tabelle* in this case. Therefore, the translated utterance will be considered as not acceptable, and annotated as 0.

dialogID: *PMUL2464.json*
turnID: *9*
utterance: *Yes, Bedouin is a restaurant that serves African food in the Centre. It is in the expensive range. Would you like to book a table?*
slotValues: {restaurant-name: *bedouin*}
fixTransUtterance: *Ja, Beduine ist ein Restaurant, das afrikanisches Essen im Zentrum serviert. Es liegt im teuren Bereich. Möchten Sie eine Tabelle reservieren?*
fixTransSlotValues: {restaurant-name: *Beduine*}

UtteranceAcceptable: 0
SlotValuesMatchAcceptable: 1

Example 2: Grammatical Error

The following example shows a slight grammatical issue regarding the translated utterance. This is mainly with the synonym case in Chinese, where the *place* can be translated into 地方 or 位置, while 位置 will be more appropriate in this scenario. However, 地方 still keeps the semantic meaning. Therefore, the translated utterance will be considered as acceptable, and annotated as 1. And further checking with the translated slot values, all are correct, and should be annotated as 1.

dialogID: *PMUL0400.json*
turnID: *12*
utterance: *Please book the place for 7 people at 11:30 on the same day.*
slotValues: {restaurant-people: *7*, restaurant-time: *11:30*, restaurant-day: *Monday*}
fixTransUtterance: 请于当天11:30预订7人的地方。
fixTransSlotValues: {restaurant-people: *7*, restaurant-time: *11:30*, restaurant-day: 周一}

UtteranceAcceptable: 1
SlotValuesMatchAcceptable: 1

C.2.3 Additional Notes

Please ignore the slot values with "dontcare", "not mentioned", and "none", while checking the translation quality. If there are any open questions that you think are not covered in this guide, please do not hesitate to get in touch with me or post the questions on Slack, so these issues can be discussed together with other annotators and the guide can be improved.

C.3 Few-Shot Cross-Lingual Transfer Experiments

We provide the full per-language experimental results of few-shot cross-lingual transfer results for Dialog State Tracking (DST) and Response Retrieval (RR) in Table C.1.

		DST					RR				
Lang	**Model**	**1%**	**5%**	**10%**	**50%**	**100%**	**1%**	**5%**	**10%**	**50%**	**100%**
DE	TOD-XLMR	7.68	19.26	28.08	33.17	34.10	10.25	32.47	35.56	45.39	49.46
	MLM on Mono-CC	13.75	25.15	34.12	38.01	38.26	34.37	42.13	43.51	49.10	52.80
	TLM on OS	14.17	19.45	21.62	27.28	29.91	47.21	48.59	48.96	53.01	55.30
	TLM+RS-Mono on OS	15.88	24.14	28.38	32.57	35.45	46.08	48.94	49.98	53.43	55.72
AR	TOD-XLMR	1.48	1.57	6.18	15.62	17.63	6.36	18.72	23.57	36.04	42.69
	MLM on Mono-CC	4.41	5.74	7.02	14.10	17.22	28.54	31.50	32.82	41.09	44.26
	TLM on OS	4.18	6.33	6.89	13.60	17.77	32.19	35.04	37.02	41.39	47.04
	TLM+RS-Mono on OS	4.42	6.79	8.27	14.39	21.48	33.45	37.09	38.01	41.89	47.15
ZH	TOD-XLMR	8.63	12.55	16.40	23.45	25.49	15.69	31.10	33.22	41.97	48.14
	MLM on Mono-CC	11.64	19.73	25.46	34.93	35.61	34.40	37.65	39.65	48.01	50.97
	TLM on OS	11.48	17.43	21.95	28.52	32.51	38.17	42.82	42.91	49.29	51.63
	TLM+RS-Mono on OS	11.63	14.90	17.97	22.81	28.84	38.45	43.71	45.27	48.50	51.81
RU	TOD-XLMR	4.34	21.89	30.01	37.58	37.61	8.90	31.31	34.51	43.33	47.45
	MLM on Mono-CC	12.70	16.56	19.45	24.58	25.90	37.43	42.80	46.19	52.43	53.73
	TLM on OS	12.45	14.26	16.10	21.13	27.04	42.23	44.40	44.78	49.43	53.76
	TLM+RS-Mono on OS	13.74	17.44	18.63	24.33	29.15	41.97	45.44	46.02	49.90	53.16

Table C.1: Full per-language few-shot cross-lingual transfer results for Dialog State Tracking (DST) and Response Retrieval (RR). Results are shown for different sizes of the training data in the target-language (i.e., different number of *shots*): 1%, 5%, 10%, 50% and 100% of the MULTI^{2}WOZ development sets (of respective target languages).

Appendix D

Experimental Details for Chapter 5

D.1 Control Experiment for Language Proficiency

Experimental results of monolingual BERT and multilingual BERT are provided in five countries with *gender* data (Table D.1) and *age* (Table D.2) data on text classification tasks.

		Gender class.				SA						TD					
		AC-SA		AC-TD		F	M	X	F	M	X	F	M	X	F	M	X
Country	Model	Mono	Multi	Mono	Multi	Mono			Multi			Mono			Multi		
Denmark	BERT	66.1	64.0	63.8	61.8	72.3	67.9	70.4	69.2	64.8	67.2	60.7	59.8	59.9	59.3	58.3	59.0
	MLM	66.0	**65.2**	**64.2**	63.4	72.5	68.3	70.3	69.5	**65.8**	67.8	60.6	**60.6**	60.6	59.7	58.8	**59.4**
	DS-Seq	**66.2**	64.9	64.1	63.5	**72.6**	**68.6**	**70.6**	**69.9**	65.7	67.7	**61.3**	60.5	60.0	59.7	57.8	59.1
	DS-Tok	66.0	65.0	64.1	**63.5**	72.4	68.4	70.6	69.1	65.6	**68.0**	61.1	60.2	**60.8**	**59.9**	**58.9**	59.0
Germany	BERT	59.8	59.5	58.9	57.9	66.5	63.7	64.3	66.1	63.2	64.5	67.9	66.1	67.8	67.8	65.6	65.8
	MLM	62.0	61.2	59.7	60.1	68.1	**65.8**	65.4	**67.7**	**65.3**	66.1	68.5	66.7	67.7	**68.6**	67.0	**67.1**
	DS-Seq	**61.1**	60.1	59.0	**60.3**	**68.8**	64.4	**66.2**	66.7	64.0	65.7	**68.9**	66.4	67.8	67.6	65.7	66.4
	DS-Tok	60.9	**62.9**	**60.3**	58.3	67.9	65.6	66.0	66.8	64.3	**66.8**	68.6	**66.8**	**67.9**	68.3	**67.0**	66.7
US	BERT	64.3	62.6	58.7	58.1	68.6	67.0	67.1	66.3	64.4	66.0	72.5	69.7	71.0	71.2	68.4	70.2
	MLM	64.6	63.3	58.7	**59.6**	68.4	67.6	67.8	67.3	66.2	66.9	73.1	70.1	71.3	72.1	69.4	70.3
	DS-Seq	64.3	**63.8**	58.8	59.2	68.6	**68.0**	**68.0**	67.2	66.3	67.0	73.1	**70.3**	71.6	72.3	69.2	70.4
	DS-Tok	**64.7**	62.2	**59.4**	58.8	**68.9**	67.5	67.9	**68.0**	**66.4**	**67.3**	**73.3**	69.9	**71.6**	**72.8**	**69.5**	**70.5**
UK	BERT	63.2	61.9	65.0	63.1	73.4	71.0	72.3	71.0	69.0	69.7	71.2	69.1	70.1	70.4	67.9	68.9
	MLM	**63.7**	63.0	64.8	65.3	**73.9**	71.0	**72.6**	72.0	70.4	71.0	71.2	**69.4**	70.0	70.6	67.9	69.8
	DS-Seq	63.2	63.4	**65.2**	64.9	73.6	**72.2**	72.4	72.9	70.9	71.7	**71.5**	69.3	70.2	70.6	68.2	69.8
	DS-Tok	63.3	**63.5**	64.8	**65.6**	73.7	72.0	72.2	**73.0**	**71.0**	**71.9**	71.4	69.1	**70.3**	**70.8**	**68.2**	**69.9**
France	BERT	64.1	63.9	63.1	61.2	70.5	67.3	68.6	69.3	67.0	67.8	46.0	**44.5**	45.1	44.6	42.4	43.1
	MLM	**64.9**	64.6	**63.2**	62.1	71.0	67.7	67.6	69.9	67.1	68.4	46.2	44.3	45.5	45.8	43.3	44.3
	DS-Seq	64.2	64.1	63.1	**63.1**	70.5	67.5	69.3	**70.6**	67.3	68.4	**47.1**	44.2	45.3	**46.0**	43.4	44.2
	DS-Tok	64.4	**65.0**	62.9	62.9	**71.7**	**68.3**	**69.5**	70.1	**67.5**	**68.8**	46.9	44.3	**45.6**	45.5	**43.9**	**44.4**
Average	BERT	63.5	62.4	61.9	60.4	70.3	67.4	68.5	68.4	65.7	67.0	63.7	61.8	62.8	62.7	60.5	61.4
	MLM	**64.2**	63.5	62.1	62.1	70.8	68.1	68.7	69.3	**67.0**	68.0	63.9	**62.2**	63.0	63.4	61.3	**62.2**
	DS-Seq	63.8	63.3	62.0	**62.2**	70.8	68.1	**69.3**	**69.5**	66.8	68.1	**64.4**	62.1	63.0	63.2	60.9	62.0
	DS-Tok	63.9	**63.7**	**62.3**	61.8	**70.9**	**68.4**	69.2	69.4	**67.0**	**68.6**	64.3	62.1	**63.2**	**63.5**	**61.5**	62.1

Table D.1: Evaluation results compared with monolingual BERT and multilingual BERT in five countries with *gender* data for intrinsic attribute classification tasks (AC-SA, AC-TD) and extrinsic evaluation tasks: sentiment analysis (SA) and topic detection (TD).

		Age class.				SA						TD					
		AC-SA		AC-TD		<35	>45	X	<35	>45	X	<35	>45	X	<35	>45	X
Country	Model	Mono	Multi	Mono	Multi	Mono			Multi			Mono			Multi		
Denmark	BERT	67.7	57.2	65.3	64.5	67.3	66.2	66.0	62.7	62.7	62.9	58.4	54.4	56.3	56.1	52.2	53.4
	MLM	67.4	**65.5**	**67.4**	65.1	**67.7**	**67.3**	**67.6**	63.3	62.1	63.0	**59.3**	55.3	**57.6**	**57.1**	52.6	54.1
	DS-Seq	67.4	65.2	66.8	**65.2**	67.4	66.2	67.1	63.1	62.9	63.0	58.7	55.0	56.5	56.9	**53.3**	**54.5**
	DS-Tok	**67.8**	65.3	66.6	64.6	67.6	66.1	67.2	**64.2**	**63.3**	**63.2**	59.0	**55.4**	56.7	56.2	53.2	54.3
Germany	BERT	57.9	58.0	59.6	56.9	53.6	57.9	57.1	52.6	55.0	55.0	61.6	57.4	58.3	60.1	55.3	57.1
	MLM	58.1	**61.1**	**62.0**	**58.9**	**58.1**	**58.2**	**58.1**	53.6	55.5	56.7	62.2	57.6	**59.9**	**61.5**	56.5	58.7
	DS-Seq	**58.2**	56.4	61.3	58.2	56.3	57.3	55.8	**53.8**	55.3	55.5	63.5	57.9	59.1	60.8	**57.6**	**59.3**
	DS-Tok	57.2	56.6	60.6	57.4	57.9	58.1	54.0	53.0	**56.5**	**56.7**	**63.5**	**58.2**	59.2	59.3	56.5	59.3
US	BERT	65.2	62.9	63.0	60.7	60.5	58.7	57.2	57.7	57.9	57.8	68.8	64.9	**67.2**	68.0	64.3	64.3
	MLM	65.3	**63.6**	62.9	**61.9**	59.8	**59.5**	**60.4**	59.4	57.8	**58.2**	71.2	65.7	66.7	69.0	64.2	65.2
	DS-Seq	**66.2**	60.7	**64.1**	61.5	**61.6**	58.3	59.4	59.3	57.9	58.0	**72.5**	65.5	67.1	**69.8**	64.4	**65.8**
	DS-Tok	65.7	59.7	62.9	61.2	61.1	58.7	59.4	**59.9**	**58.6**	57.8	69.4	**65.7**	66.7	69.2	**65.4**	64.9
UK	BERT	65.7	65.1	65.8	65.2	65.2	66.3	65.5	63.8	63.9	63.7	68.1	68.1	68.0	64.7	67.1	66.3
	MLM	66.9	**65.4**	**66.1**	**65.6**	**68.2**	**67.2**	66.9	62.8	62.0	63.0	**68.8**	**70.1**	**70.0**	65.1	67.3	67.3
	DS-Seq	67.0	65.3	64.7	62.8	67.8	66.4	**67.6**	63.8	64.9	64.9	67.8	68.9	69.4	66.0	**68.1**	66.5
	DS-Tok	**66.8**	64.0	65.2	62.8	67.6	66.5	67.1	**64.6**	**65.2**	**65.1**	68.2	69.6	69.2	**66.4**	67.3	**67.6**
France	BERT	**56.0**	55.7	**57.0**	56.6	59.7	57.5	60.3	59.6	57.4	61.5	51.9	49.1	49.6	52.0	47.1	49.0
	MLM	55.9	**56.8**	56.9	**57.2**	60.7	59.4	61.6	59.9	59.5	61.6	53.8	48.5	50.2	**52.5**	47.2	50.3
	DS-Seq	55.5	55.1	56.7	55.5	**61.3**	58.7	**62.0**	60.4	**60.3**	**62.8**	53.8	49.0	50.2	51.1	47.3	50.3
	DS-Tok	55.8	54.4	56.7	55.9	60.2	**60.7**	61.5	**60.9**	59.8	59.7	**54.6**	**51.4**	**50.3**	50.2	**48.0**	**50.8**
Average	BERT	62.5	59.8	62.1	60.8	61.3	61.3	61.2	59.3	59.4	60.2	61.8	58.8	59.9	60.2	57.2	58.0
	MLM	62.7	**62.5**	**63.1**	**61.7**	**62.9**	**62.3**	**62.9**	59.8	59.4	60.5	63.1	59.4	**60.9**	**61.0**	57.6	59.1
	DS-Seq	**62.9**	60.5	62.7	60.6	**62.9**	61.4	62.4	60.1	60.3	**60.8**	**63.3**	59.3	60.5	60.9	**58.1**	59.3
	DS-Tok	62.7	60.0	62.4	60.4	**62.9**	62.0	61.8	**60.5**	**60.7**	60.5	62.9	**60.1**	60.4	60.3	**58.1**	**59.4**

Table D.2: Evaluation results compared with monolingual BERT and multilingual BERT in five countries with *age* data for intrinsic attribute classification tasks (AC-SA, AC-TD) and extrinsic evaluation tasks: sentiment analysis (SA) and topic detection (TD).

D.2 Control Experiment for Domain Knowledge

We provide the experimental results on classification tasks compared by specializing on in-domain (TRUSTPILOT (Hovy et al., 2015)) and out-of-domain (RTGENDER (Voigt et al., 2018) for *gender* and BAC (Schler et al., 2006) for *age*) data in Table D.3.

		SA						**TD**					
Gender		F	M	X	F	M	X	F	M	X	F	M	X
Country	**Model**	**RtGender**			**Trustpilot**			**RtGender**			**Trustpilot**		
US	MLM	68.3	67.3	66.9	68.4	67.6	67.8	**72.7**	**69.9**	71.1	73.1	70.1	71.3
	DS-Seq	68.1	**67.4**	**66.9**	68.6	**68.0**	**68.0**	72.7	69.3	**71.2**	73.1	**70.3**	71.6
	DS-Tok	**68.6**	67.2	66.4	**68.9**	67.5	67.9	72.4	69.6	71.2	**73.3**	69.9	**71.6**
UK	MLM	73.3	71.0	71.7	**73.9**	71.0	**72.6**	71.1	**69.3**	**69.8**	71.2	**69.4**	70.0
	DS-Seq	73.3	71.1	**71.9**	73.6	**72.2**	72.4	71.2	69.0	69.5	**71.5**	69.3	70.2
	DS-Tok	**73.4**	**71.1**	71.6	73.7	72.0	72.2	**71.3**	69.2	69.6	71.4	69.1	**70.3**
Average	MLM	70.8	69.2	69.3	71.2	69.3	**70.2**	71.9	**69.6**	**70.5**	72.2	**69.8**	70.7
	DS-Seq	70.7	**69.3**	**69.4**	71.1	**70.1**	**70.2**	**72.0**	69.2	70.4	72.3	**69.8**	70.9
	DS-Tok	**71.0**	69.2	69.0	**71.3**	69.8	70.1	71.9	69.4	70.4	**72.4**	69.5	**71.0**
		SA						**TD**					
Age		<35	>45	X	<35	>45	X	<35	>45	X	<35	>45	X
Country	**Model**	**BAC**			**Trustpilot**			**BAC**			**Trustpilot**		
US	MLM	**59.4**	58.4	**58.9**	59.8	**59.5**	**60.4**	68.4	64.6	66.9	71.2	65.7	66.7
	DS-Seq	58.4	57.3	58.0	**61.6**	58.3	59.4	68.6	64.5	**67.3**	**72.5**	65.5	**67.1**
	DS-Tok	58.6	**58.5**	58.9	61.1	58.7	59.4	**69.3**	**65.0**	67.1	69.4	**65.7**	66.7
UK	MLM	66.2	**66.7**	66.4	**68.2**	**67.2**	66.9	67.8	68.7	68.9	**68.8**	**70.1**	**70.0**
	DS-Seq	66.1	66.6	**66.8**	67.8	66.4	**67.6**	67.8	68.7	68.6	67.8	68.9	69.4
	DS-Tok	**66.6**	66.0	66.3	67.6	66.5	67.1	**68.0**	**68.8**	**69.2**	68.2	69.6	69.2
Average	MLM	**62.8**	**62.6**	**62.7**	64.0	**63.4**	**63.7**	68.1	66.7	67.9	70.0	**67.9**	**68.4**
	DS-Seq	62.3	62.0	62.4	**64.7**	62.4	63.5	68.2	66.6	68.0	**70.2**	67.2	68.3
	DS-Tok	62.6	62.3	62.6	64.4	62.6	63.3	**68.7**	**66.9**	**68.2**	68.8	67.7	68.0

Table D.3: Evaluation results on TRUSTPILOT classification tasks (SA, TD) compared by specializing on out-of-domain data (RTGENDER) for *gender* and BAC for *age* and in-domain data (TRUSTPILOT).