Developing Trustworthy Language Models with Consistent Predictions

Myeongjun Erik Jang
Wolfson College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Hilary 2024

Acknowledgements

I would like to acknowledge and give my sincere gratitude and appreciation to my supervisor Thomas Lukasiewicz, for the opportunity he provided, as well as his unwavering belief and support. His guidance and advice carried me through all the stages of writing this research project. My sincere thanks to Professors Janet B. Pierrehumbert and Isabelle Augenstein for their careful examination of this thesis. I would also like to thank all of my committee collaborators, especially Frank Mtumbuka, Vid Kocijan, and Oana-Maria Camburu, for the many stimulating research discussions. I would like to extend my gratitude to the members of the Intelligent Systems Lab within the Computer Science Department, as well as the Department as a whole, for their valuable feedback, unwavering support, and insightful remarks regarding my work. Special appreciation is due to my parents, siblings, and my entire family, whose patience, love, and support have been indispensable throughout this endeavour. I also would like to extend my heartfelt gratitude to my friends in Oxford Korean Society, particularly to Selina Y. Cho, my life saviour who has provided invaluable advice throughout my entire DPhil studies; Hyunhong J. Min, my gym companion; Yunseul Kim, the wine master and first-born of the Oxford Sibling members; Jaesung Huh, the vice wine master; Chloe H. Lee and Vincent J. Lim, the 3rd and the youngest in the Oxford Sibling members; Yujin Seo, the Canary Wharf Alliance for their support in maintaining my physical and mental well-being. Lastly, my deepest gratitude goes to my beloved partner, my sun, Yoon Sun Choi, who has always glowed brightly in the place where you've stood. To all my numerous and cherished friends, whose names are too many to mention individually and whose kindness is unforgettable, I extend my heartfelt thanks for your enduring support.

Abstract

Transformer-based pre-trained language models (PLMs), which are trained with a vast amount of natural language text, have significantly propelled the rapid progress in the field of natural language processing (NLP). These models have been effectively employed across diverse downstream tasks in various ways, such as fine-tuning or few-shot learning. Notably, they have demonstrated promising performance, even surpassing human capabilities in several downstream tasks. Derived from these outstanding experimental results, a prevailing belief has emerged asserting that PLMs contain extensive knowledge regarding the world and possess the capability to comprehend natural language. However, a number of investigations have delineated the inherently imperfect capability of PLMs for comprehending natural language. Through experiments across diverse spectrums, it was revealed that PLMs exhibit deficiencies in understanding negation expressions and number-related knowledge. An additional avenue of research ascertained that PLMs exhibit logically incorrect behaviours, which greatly diverge from those demonstrated by humans. These fallacious behaviours have presented a substantial concern, as they undermine the trustworthiness of PLMs, potentially impeding their applications across industries, particularly in risk-sensitive areas, such as medical, financial, and legal domains. In this context, this dissertation centres its attention on enhancing the trustworthiness of language models (LMs) in terms of the consistency perspective. Although there have been several attempts that have explored the consistent behaviours of LMs and endeavoured to construct enhanced models equipped with heightened consistency, these studies have critical drawbacks. First, the definition of consistency has diverged across multiple studies, leading to fragmented investigations that hindered the achievement of unified and comprehensive evaluations. The diverse definitions have also resulted in incomprehensive alleviation methods that exclusively focus on certain types of consistency and cannot address other consistency types. Second, methodologies that

have been proposed to improve consistent behaviours require substantial resource allocation. The most widely used techniques are data augmentation, which involves the collection of additional data that adheres to a designated consistency type, and consistency regularisation, where an additional training loss function is introduced to penalise undesirable behaviours through the utilisation of the augmented data. While the automatic collection of supplementary data to address certain consistency types is attainable, as exemplified by the utilisation of symmetry properties, the majority of these strategies mandate ample linguistic resources or significant human endeavours to ensure a high standard of data quality. Consequently, these mitigation methods become comparatively less feasible for languages with limited resources or for researchers operating under resource constraints. Furthermore, the incorporation of the augmented data and auxiliary regularisation terms for updating the parameters of LMs presents a challenge in terms of computational resources required for training. This is particularly pronounced due to the substantial increase in the size of contemporary LMs, as evidenced by GPT-4. This dissertation aims to address the aforementioned drawbacks of previous studies regarding LMs' consistent behaviours. First, I present a comprehensive definition of LMs' consistency based on the notion of behavioural consistency and propose a systematic categorisation that classifies various consistency types addressed in prior studies into three exclusive categories. Second, I introduce a unified benchmark dataset, which is designed to facilitate a comprehensive evaluation encompassing multiple consistency types across diverse downstream tasks. Third, I propose a practically efficient and feasible approach for enhancing LMs' consistent behaviours. Specifically, the method facilitates LMs to capture the precise meaning of language by learning conceptual roles from dictionary data. Subsequently, the LM enhanced with conceptual role information is incorporated with existing LMs, aiming to fully employ acquired knowledge. This is achieved through parameter integration in a resource-efficient manner that requires minimal computational resources. The extensive experimental results indicate two important findings. Above all, the investigation of this dissertation ascertains that, regardless of their size, architecture design, and training objectives, modern LMs exhibited inconsistent behaviours in many test cases, which is distinguished by consistency types and downstream

tasks. None of them demonstrated coherently consistent behaviours in every test case. Subsequent findings demonstrate that the proposed alleviation method concurrently enhances a multitude of consistency types, a capability that previous methods lack. Furthermore, the experimental results attest to the reduction in computational resources achieved by the proposed approach and its wide applicability to low-resource languages beyond English.

Contents

List of Figures

List of Tables

List of Acronyms

PLM pre-trained language model

LM language model

NLP natural language processing

LLM large language model

AI artificial intelligence

MLM masked language modelling

QA question answering

STS semantic textual similarity

NLI natural language inference

NSP next sentence prediction

NLE natural language explanation

MKR-NQ masked knowledge retrieval on negated queries

MWR masked word retrieval

SAR synonym/antonym recognition

POS parts of speech

NLU natural language understanding

MRC machine reading comprehension

TC topic classification

SA semantic analysis

WiC words-in-context

NLG natural language generation

CRM conceptual role model

Chapter 1

Introduction

This chapter outlines the motivation behind this dissertation and presents the research questions that guide this study. Next, the research contributions and the structure of this work are provided. Finally, the publications that originated from this research are attached.

1.1 Motivation

The advent of large Transformer-based [Vaswani et al., 2017] pre-trained language models (PLMs), such as BERT [Devlin et al., 2019], BART [Lewis et al., 2020a], and T5 [Raffel et al., 2020], has been propelling unprecedented rapid progress in the contemporary field of NLP. These models were applied to many downstream tasks through fine-tuning and achieved outstanding results, even exceeding human performance in several datasets [Rajpurkar et al., 2016, Zellers et al., 2018]. Based on their excellent performance on downstream tasks, a claim that PLMs can understand human language has emerged in academic papers [Ohsugi et al., 2019, Qiu et al., 2020] and the popular press like the *Google blog* post.[1] Recently, generative LLMs, such as ChatGPT[2] and LaMDA [Thoppilan et al., 2022], have further bolstered this claim by performing surprisingly well on many professional examination cases, including the United States Medical Licensing Examination [Kung et al., 2023], the US Bar Exam [Bommarito II and Katz, 2022], and a core MBA course [Terwiesch, 2023].

Although PLMs have demonstrated an exceptional performance, there are still doubts regarding their ability to comprehend natural language. It was observed in several studies that PLMs hardly capture the semantic meaning of sentences but

[1] https://www.blog.google/products/search/search-language-understanding-bert/ accessed 2023-05-09

[2] https://openai.com/blog/chatgpt accessed 2023-05.09

heavily rely on statistical cues and syntactic patterns [Habernal et al., 2018, Niven and Kao, 2019, McCoy et al., 2019, Bender and Koller, 2020]. Another line of works argued that the brilliant performance of PLMs is predicated on the memorisation effect on frequently appearing words/phrases/knowledge in pre-training data and, hence, potentially resulting in sub-optimal performance when confronted with unseen expressions [Kassner et al., 2020, Ravichander et al., 2020, Hofmann et al., 2021]. Moreover, various studies ascertained that PLMs lack an understanding of negation expressions [Naik et al., 2018, Hossain et al., 2020, Kassner and Schütze, 2020, Ettinger, 2020, Hosseini et al., 2021, Kalouli et al., 2022] and number-related knowledge [Wallace et al., 2019, Lin et al., 2020a, Nogueira et al., 2021]. Sinha et al. [2021b] observed that PLMs do not conform to the principle of the *sentence superiority effect*, which stipulates that humans exhibit a greater competence in identifying words arranged in grammatically correct orders than those in ungrammatical ones. All the aforementioned works indicate that PLMs are incapable of comprehending natural language perfectly, which undermines people's trust in these models and lowers the likelihood of their practical usage, especially in domains that entail high risk, such as legal, medical, and financial areas.

The lexical definition of "*understand*" is "*to know or realise the meaning of words, a language, what somebody says, etc.*".[3] As humans are capable of understanding language, they perceive natural language in terms of its meanings rather than its textual form, which results in logically consistent behaviours that are not dependent on how the text appears visually but rather on the semantic meanings it delivers. Let us consider the following three example sentences within the context of sentiment analysis of film reviews:

Sentence A: It would be a significant mistake if you choose to watch this film.

Sentence B: If you watch this film, you make a very big mistake.

Sentence C: It would be a significant mistake if you choose <u>not</u> to watch this film.

Provided an individual classifies sentence A into the "negative" category, he/she will make the same decision for sentence B, the paraphrased version of sentence A, because the two sentences convey an identical meaning. Conversely, sentence C, the negated form of sentence A, will be classified as "positive" by the individual, because, despite their high similarity in textual form, the two sentences deliver the opposite

[3]Oxford Learners Dictionaries, accessed 2023-05-09

meaning, which is attributed to the presence of the negation expression "not". Thus, if PLMs possess the ability to understand natural language genuinely, they should exhibit logically consistent behaviours, and such human-like behaviours can assist in establishing trust between humans and artificial intelligence (AI) [De Visser et al., 2016, Jung et al., 2019].

However, many recent studies suggest that PLMs frequently demonstrate logically inconsistent behaviours. When prompted with a masked query, such as "Bill Gates was born in [MASK]" to predict the masked word, PLMs trained with the MLM objective produced different predictions for semantically identical queries, including their paraphrased versions [Elazar et al., 2021] or queries where singular subjects are replaced by their plural forms [Ravichander et al., 2020]. In question answering (QA), Ribeiro et al. [2019] ascertained that state-of-the-art QA and visual QA models are prone to generate logically incorrect predictions. They found that the answers to the questions regarding the same context are contradictory, for example, answering "one" to the question "How many birds are in the photo?" and "yes" to the question "Are there 2 birds in the photo?". In dialogue generation, it was observed that models based on large PLM are likely to generate statements that are inaccurate or contradictory to the dialogue history [Welleck et al., 2019, Li et al., 2020b]. Other studies confirmed that PLMs change their decision when the order of input sentences is switched, even for input-order-invariant tasks, such as natural language inference (NLI) [Li et al., 2019, Wang et al., 2019c] and semantic textual similarity (STS) [Kumar and Joshi, 2022].

In this regard, the consistent behaviours of language models, being of utmost importance, have been widely addressed within the NLP domain. Nonetheless, little attention has been given to the precise definition of consistency, leading to divergent delineations across various studies. Below are examples of different definitions regarding language models' consistency.

- Making consistent decisions in semantically equivalent contexts [Elazar et al., 2021].
- Being consistent on a system's beliefs across various inputs [Li et al., 2019].
- Producing logically or factually accurate statements [Li et al., 2020b].

Due to the lack of an integrated definition, research studies have been conducted individually, focusing only on specific types of consistency or particular tasks, predominantly in NLI and QA. Accordingly, research works that aimed to mitigate the inconsistency issue have been restricted to addressing a single type of consistency

and have no capacity to address the others. For example, the consistency regularisation loss [Elazar et al., 2021, Kim et al., 2021], which compels a model's predictive distribution on the original and its paraphrased inputs to be similar, is incapable of rectifying the violation of the logical negation property or symmetry.

The research presented in this dissertation was motivated by the limitations of PLMs' in language comprehension, as well as the aforementioned shortcomings of prior studies on the consistency of language models. Here, I introduce experiments that validate PLMs' logically flawed behaviours, along with an empirical confirmation that challenges PLMs' language understanding ability and trustworthiness. Subsequently, among various undesirable behaviours, a focus is directed towards enhancing the predictive consistency of PLMs. I propose an integrated definition of a language model's consistency, along with its taxonomy, accompanied by a newly devised benchmark evaluation dataset that facilitates the exploration of multiple types of consistency across various downstream tasks. Finally, I propose a novel strategy based on conceptual role theory which concurrently enhances multiple types of consistency using a computationally efficient parameter integration method.

1.2 Research Questions

In order to investigate PLMs' logically inaccurate behaviours and address the limitations of previous studies highlighted above, several research questions are formulated to guide this investigation. The following research questions serve as a guideline for the work demonstrated in this dissertation:

- RQ1: Many pieces of prior works have explored the logically incorrect behaviours exhibited by PLMs. These investigations include various aspects, such as comparing the predictions of text inputs delivering identical semantic meaning [Ravichander et al., 2020, Elazar et al., 2021] and confirming PLMs' incapacity to comprehend negation expressions [Hossain et al., 2020, Kassner and Schütze, 2020, Ettinger, 2020, Kalouli et al., 2022]. Furthermore, numerous studies have investigated whether PLMs' predictions violate a certain logical property, such as symmetry [Li et al., 2019, Wang et al., 2019c, Kumar and Joshi, 2022] or transitivity [Asai and Hajishirzi, 2020, Lin and Ng, 2022]. Additionally, other research works have confirmed distinctions in behaviours between PLMs and humans, such as being insensitive to correct word ordering grammatically [Pham et al., 2021, Gupta et al., 2021, Sinha et al., 2021b] or generating outputs contradictory to the predictions derived from other input texts [Ribeiro

et al., 2019]. However, these studies primarily concentrated on the behaviour of categorical prediction tasks and did not extensively investigate language generation tasks. Additionally, they studied a restricted range of linguistic characteristics, e.g., exploring a limited number of functional negation expressions such as "no" and "not". **What additional experiments can be further employed to substantiate the existence of faulty behaviours in PLMs that undermine their reliability and trustworthiness?** Two approaches can be considered when selecting additional experiments. Firstly, expanding the scope of previously conducted experiments to encompass broader linguistic characteristics can provide valuable insights. This paper proposes an experiment that enlarges the range of understanding logical negations from functional negation expressions to semantic synonymy and antonymy to achieve this purpose. Secondly, focusing on domains that have received relatively less attention in terms of PLMs' behaviour would be greatly beneficial. Regarding the second approach, this paper explores logical contradictions in natural language explanations generated by PLMs that allow an investigation of the PLMs' logical behaviour on a language generation task, which has been less studied. These approaches can consolidate the argument regarding PLMs' incapability to comprehend language, provided they also display fallacious behaviours in these new experiments.

- RQ2: Despite the promising importance of achieving consistent behaviours in language models (LMs), limited consensus exists regarding the precise definition of an LM's consistency, resulting in individual definitions across numerous studies. The lack of consensus has consequently restricted previous studies to focus exclusively on specific types of consistency. Thus, it becomes imperative to establish a precise definition of an LM's consistency and its systematic categorisation to facilitate a comprehensive evaluation encompassing various perspectives of consistency. **How can a precise definition of LM consistency be formulated that encapsulates the core characteristics of multiple types of consistency that were previously examined and constructs an effective taxonomy?**

- RQ3: Due to the unstandardised definition of LM consistency, many previous works were individually conducted while examining only limited consistency types and specific downstream tasks [Li et al., 2019, Wang et al., 2019c, Ravichander et al., 2020, Elazar et al., 2021, Kumar and Joshi, 2022]. However,

such examinations carry the risk of reaching a distorted conclusion, because LMs may demonstrate a satisfactory performance in certain consistency types while exhibiting significant deficiencies in others. It is also possible that experimental results, whether positive or negative, may be contingent upon the characteristics of specific downstream tasks. As a result, a unified benchmark that allows the evaluation of various consistency types across many downstream tasks is necessary for a comprehensive and objective examination. However, collecting natural, grammatical and high-quality language data is demanding work that requires tremendous effort and resources. Hence, endeavours should be made to implement efficient approaches for data collection. **How can a high-quality unified benchmark dataset be effectively constructed to systematically evaluate various consistency types across multiple downstream tasks?**

- RQ4: Numerous contemporary studies have endeavoured to improve the consistency of LMs through strategies that utilise specially designed consistency loss functions [Elazar et al., 2021, Kim et al., 2021] and data augmentation [Ray et al., 2019, Hosseini et al., 2021]. However, these approaches are constrained in their ability to address solely a solitary type of consistency. For instance, augmenting negation data [Hosseini et al., 2021] may improve compliance with the logical negation property but cannot resolve the violation of symmetry or semantic equivalence. Analogously, symmetry consistency regularisation loss [Kumar and Joshi, 2022] fails to alleviate the breach of semantic equivalence. Furthermore, these approaches require expensive resources. For example, training large-scale LMs with additional loss terms entails significant computational resources. Likewise, collecting auxiliary data for data augmentation, such as negated samples or paraphrased texts, poses a challenge for languages with limited resources. **How can multiple types of consistency be concurrently enhanced while ensuring both data and computational efficiency?**

1.3 Contributions

The work in this thesis constitutes significant contributions to the field of consistency of LMs and their trustworthiness, which is accomplished through an exploration of the research questions demonstrated in Section 1.2. The in-depth contributions will be provided in Chapters 3, 4, and 5. The principal contributions of this dissertation are briefly summarised as follows:

1. **New probing tasks for verifying PLMs undesirable behaviours greatly distinctive to those of humans (Chapter 3).**

   In this study, I present two new probing tasks that enable ascertaining non-human-like behaviours exhibited by PLMs. The first task is expanding previous studies exploring PLMs' ability to negation expression understanding. The logical negation property, when it is applied to natural language, encompasses not only negation expressions but also the opposite meaning. Hence, the first probing task extends the examination scope beyond negation expressions to include antonyms. Specifically, the task explores whether PLMs generate identical predictions for text inputs delivering the opposite meaning, produced by adding negation expressions and substituting a word with its antonym. The second task explores logical contradictions in natural language explanations. It aims to ascertain whether an explainable model that generates natural language explanations produces contradictory explanations with others.

2. **A definition of language model's consistency and its taxonomy (Chapter 4)**

   Behavioural consistency is a core human characteristic, implying being consistent in behavioural patterns by adhering to the same principles.[4] Based on this idea, this dissertation defines the concept of a language model's consistency as the "capacity of a model to make a coherent decision not contradictory to its belief". The definition comprises two pivotal components: *belief*, representing what a model considers to be true, i.e., their behaviours, and *principle*, which refers to the property that determines a coherent decision. Based on these two components, I present a comprehensive taxonomy that categorises the diverse forms of consistency examined in preceding literature into three mutually exclusive categories. The first category is semantic consistency, where belief becomes a *model's decision on texts delivering identical meaning*, and principle becomes *the property of semantic equivalence*. The second category is logical consistency, where the belief and principle are a *model's predictions on examples where a certain logical property holds* and *the certain logical property*, respectively. Finally, the third category is factual consistency. The belief and principle of this consistency type are a *text generated by a model* and *factual correctness*, respectively. The newly introduced categorisation system can serve as a touchstone in the

[4] N., Sam M.S., Behavioral Consistency. PsychologyDictionary.org, April 7, 2013, [link]

LMs' consistency field, enabling the analysis of desirable behaviours exhibited by LMs and enlarging the scope of examination in this domain.

3. **A new benchmark dataset that allows the investigation of multiple consistency types and various downstream tasks (Chapter 4).**

   This work provides a novel benchmark dataset for evaluating consistency, called BECEL (**Be**nchmark for **C**onsistency **E**valuation for **L**anguage models), which includes 19 evaluation scenarios encompassing five consistency types and seven downstream tasks. In contrast to the previous studies, which were confined to a limited scope of consistency types and downstream tasks, this newly developed dataset offers an opportunity to comprehensively assess LMs from various perspectives, thereby mitigating the potential risk of arriving at a distorted conclusion. Additionally, the experimental findings also emphasise the imperative nature of a unified consistency evaluation dataset.

4. **Empirical evidence demonstrating the behavioural mistakes produced by LLMs (Chapters 3 and 4).**

   The experimental results obtained from the two probing tasks presented in Chapter 3 demonstrate that PLMs are prone to display undesirable behaviours, which supports the claim that PLMs often make mistakes and lack the ability to understand natural language. The analysis of the new consistency evaluation benchmark described in Chapter 4 also suggests that none of the contemporary PLMs and LLMs, like ChatGPT, coherently exhibit consistent behaviours across all evaluation scenarios distinguished by different consistency types and downstream tasks. Specifically, both PLMs and LLMs frequently exhibited inconsistent predictions of texts delivering identical meanings. Additionally, the evidence indicates that PLMs often fail to perform correctly on tasks involving the logical negation property, which requires generating different predictions on texts with opposite meanings. Although LLMs demonstrated an enhanced performance over PLMs in tasks involving the logical negation property, they made more inconsistent predictions on tasks measuring the symmetric inference ability. Further comprehensive analysis reveals that the characteristics of a dataset, such as the presence of artefacts within the training data, exert a greater influence in determining PLMs' consistency performance than the attributes of the model itself, including training objectives, model structure, and model size.

5. **A novel strategy that enhances PLMs' consistency predictions in a computationally efficient manner (Chapter 5).**

   In this work, I introduce a novel approach that improves the consistent behaviours of PLMs in a data- and computation-efficient way. Drawing inspiration from the conceptual role theory, a convincing theory that accounts for human language cognition, a new learning framework is proposed. With the use of dictionary data, this framework enables the capture of more precise interrelationships between concepts, extending beyond the distributional theory. The use of dictionary data improves data efficiency, as every language has its own dictionary, which reduces the effort required to collect new data. Subsequently, a computationally efficient parameter integration method is proposed, which enables the integration of parameters of the model mentioned above, existing PLMs, and other models trained with diverse inductive biases. The approach allows further utilisation of the interrelationships between concepts captured by other learning objectives. Experimental results indicate that the proposed approach significantly improves various consistency types across diverse downstream tasks with updates on only a few trainable parameters, which allows a computationally effective training.

1.4 Thesis Structure

In Chapter 2, the background of transformer-based PLMs, which are currently dominant in the NLP field, is first introduced. The chapter then proceeds to review recent studies investigating undesirable and incorrect behaviours of PLMs, indicating their imperfect language understanding ability. Several recent research efforts aimed at alleviating such deficiencies are also discussed.

Chapter 3 investigates logically fallacious behaviours of PLMs that act as evidence of their incapability to understand textual meaning. This chapter proposes two probing tasks designed to ascertain undesirable behaviours exhibited by PLMs, followed by a thorough examination of the results. Finally, potential simple yet efficient solutions to address these issues are discussed.

In Chapter 4, a new benchmark dataset, which is designed to evaluate the consistency of PLMs, is presented. First, a taxonomy of LMs' consistency is proposed based on the concept of *behavioural consistency*. Second, a new benchmark dataset that contains test cases for measuring multiple types of consistency on various downstream tasks is introduced, accompanied by detailed information, including data collection

procedures and evaluation metrics. Finally, thorough experiments and examinations of the consistent behaviours of contemporary PLMs and LLMs are conducted based on the newly created evaluation benchmark.

Chapter 5 proposes a novel approach that can further improve the consistency of PLMs. Specifically, the approach utilises the conceptual role theory to facilitate learning more precise textual meaning and leverage a computationally efficient parameter integration technique to combine models' weights trained with different inductive biases. The outcome of the experiments indicates that improving the textual meaning-understanding can significantly contribute to enhancing PLMs' consistent predictions.

Chapter 6 summarises the principal discoveries outlined in this dissertation and how they address the research questions presented in Section 1.2. Additionally, potential avenues for future research related to this dissertation are discussed.

1.5 Publications from this Work

Table 1.1 shows the research papers that have resulted from this dissertation ((1) to (5)): all papers were published at top-tier peer-reviewed conferences. Further projects that are not directly connected to this dissertation but fall under a similar research theme have been explored with others ((6) to (11)): all these papers were also published in top-tier journals and at top-tier peer-reviewed conferences.

Table 1.1: Research papers from this dissertation and related research.

	Publications	**Related Chapter**
(1)	Myeongjun Jang, Frank Mutumbuka, and Thomas Lukasiewicz. Beyond Distributional Hypothesis: Let Language Models Learn Meaning-Text Correspondence (2022). *In Findings of The NAACL 2022, 2030-2042.* Association for Computational Linguistics.	Chapter 3
(2)	Myeongjun Jang, Bodhisattwa Prasad Majumder, Julian McAuley, Thomas Lukasiewicz, and Oana-Maria Camburu. Know How to Make Up Your Mind! Adversarially Detecting and Remedying Inconsistencies in Natural Language Explanations (2023). *In Proceedings of The ACL 2023, 540-553.* Association for Computational Linguistics.	Chapter 3
(3)	Myeongjun Jang, Deuk Sin Kwon and Thomas Lukasiewicz. BECEL: Benchmark for Consistency Evaluation of Language Models (2022). *In Proceedings of The COLING 2022, 3680-3696.* International Committee on Computational Linguistics.	Chapter 4
(4)	Myeongjun Jang and Thomas Lukasiewicz. Consistency Analysis of ChatGPT (2023). *In Proceedings of The EMNLP 2023, 15970-15985.* Association for Computational Linguistics.	Chapter 4
(5)	Myeongjun Jang and Thomas Lukasiewicz. Improving Language Models' Meaning Understanding and Consistency by Learning Conceptual Roles from Dictionary (2023). *In Proceedings of The EMNLP 2023, 8496-8510.* Association for Computational Linguistics.	Chapter 5
	Other Publications	
(6)	Dohyung Kim, Myeongjun Jang, Deuk Sin Kwon and Eric Davis. KoBEST: Korean Balanced Evaluation of Significant Tasks (2022). *In Proceedings of The COLING 2022, 3697-3708.* International Committee on Computational Linguistics.	
(7)	Myeongjun Jang and Thomas Lukasiewicz. NoiER: An Approach for Training more Reliable Fine-Tuned Downstream Task Models (2022). *IEEE/ACM Transactions on Audio, Speech, and Language Processing, VOLUME 30. 2514-2525.*	
(8)	Chloe H. Lee, Jaesung Huh, Paul R. Buckley, Myeongjun Jang, Mariana Pereira Pinho, Ricardo A. Fernandes, Agne Antanaviciute, Alison Simmons, and Hashem Koohy. A robust deep learning platform to predict CD8+ T-cell epitopes (2023). *Genome Medicine 15, 70.*	
(9)	Vid Kocijan, Myeongjun Jang, and Thomas Lukasiewicz. Pre-training and Diagnosing Knowledge Base Completion Models (2024). *Artificial Intelligence, 329, 104081.*	
(10)	Myeongjun Erik Jang, and Gábor Stikkel. Leveraging Natural Language Processing and Large Language Models for Assisting Due Diligence in the Legal Domain (2024). *In Proceedings of The NAACL 2024: Industry Track.* Association for Computational Linguistics.	
(11)	Myeongjun Erik Jang, Antonios Georgiadis, Yiyun Zhao, Fran Silavong. DriftWatch: A Tool that Automatically Detects Data Drift and Extracts Representative Drifted Examples (2024), *In Proceedings of The NAACL 2024: Industry Track.* Association for Computational Linguistics.	

Chapter 2

Literature Review

This chapter provides a comprehensive overview of the fundamental foundations of current LMs and the associated issues. It first covers the structure of Transformers [Vaswani et al., 2017], which serves as a basic backbone architecture of modern LMs, followed by the details of widely-used PLMs and LLMs. Subsequently, prior studies that analysed undesirable behaviours exhibited by LMs and cast doubts upon their trustworthiness are introduced. Lastly, methodologies devised to alleviate the faulty behaviours of LMs are described.

2.1 Transformer-based Pre-trained Language Models

2.1.1 Transformer

The Transformer model, having an encoder-decoder architecture [Cho et al., 2014, Sutskever et al., 2014], is the most proficient neural sequence model in the contemporary NLP field. The notable characteristic that distinguishes Transformers from the prior neural models, such as long short-term memory (LSTM, Hochreiter and Schmidhuber [1997]) and gated recurrent unit (GRU, Chung et al. [2014]), is its departure from recurrent structures, which precludes efficient parallelisation during training due to their sequential nature. Instead, the Transformer avoids recurrence by introducing a multi-head attention mechanism that enables a fully parallelised training. The overall architecture of the Transformer is shown in Figure 2.1.

Scaled Dot-Product Attention. The *attention mechanism* is a competitive method to draw global dependencies between input and output sequences [Chorowski et al., 2015, Luong et al., 2015]. Vaswani et al. [2017] implemented a novel attention

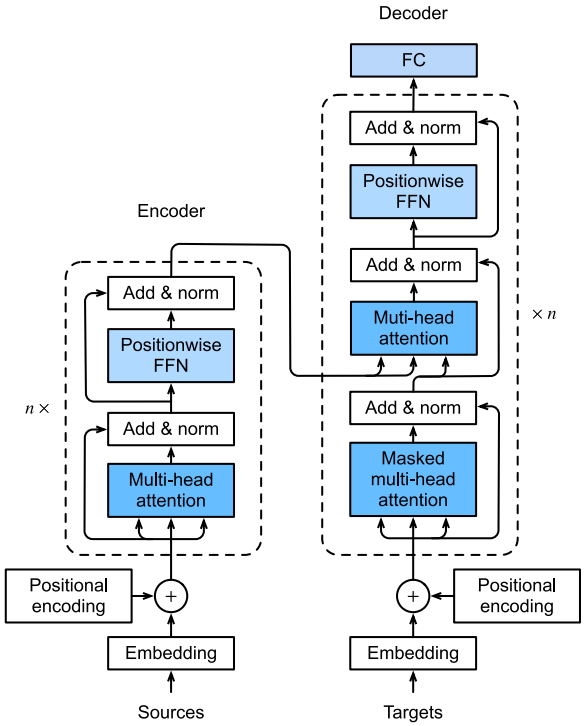


Figure 2.1: The structure of a Transformer (Source: [link]], accessed 2023-05-09)

mechanism called 'Scaled Dot-Product Attention'. Assume that there is a set of queries ($\mathbf{q} = \{q_1, ..., q_n\}$), keys ($\mathbf{k} = \{k_1, ..., k_n\}$), and values ($\mathbf{v} = \{v_1, ..., v_n\}$), where $q_i \in \mathbb{R}^{d_k \times 1}$, $k_i \in \mathbb{R}^{d_k \times 1}$, and $v_i \in \mathbb{R}^{d_v \times 1}$. The scaled-dot product attention obtain the weights on $\mathbf{v}$ by computing the dot products of $\mathbf{q}$ with $\mathbf{k}$, dividing each by $\sqrt{d_k}$, and applying a softmax function:

$$e_{ij} = \frac{q_i \cdot k_j}{\sqrt{d_k}},$$
$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{n} \exp(e_{ik})},$$

where α_{ij} is the attention weight between a vector q_i and k_j. Finally, the i^{th} output vector o_i is calculated as follows:

$$o_i = \sum_{j=1}^{n} \alpha_{ij} v_j.$$

In practice, a set of queries ($\mathbf{q}$), keys ($\mathbf{k}$), and values ($\mathbf{v}$) are packed together into a matrix Q, K, and V, respectively, for efficient computation. The matrix of output O is computed as:

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

where $O_s \in \mathbb{R}^{n \times d_v}$, $Q \in \mathbb{R}^{n \times d_k}$, $K \in \mathbb{R}^{n \times d_k}$, and $V \in \mathbb{R}^{n \times d_v}$. A distinguishing characteristic of the scaled dot-product attention is its inherent ability to compute the attention score without additional trainable weights.

Multi-Head Attention. Vaswani et al. [2017] observed that it is of benefit to project the Q, K, and V multiple times using individual trainable weights. The projected matrices are then fed into the scaled dot-product attention function in parallel, yielding corresponding output values. The resulting outputs are concatenated and projected once again using additional projection weights to produce the final outputs. This technique is called "Multi-Head Attention" mechanism. The overall computation process of the method is as follows:

$$O_m = \text{MultiHead}(Q, K, V) = \text{Concat}[\text{head}_1, ..., \text{head}_h]W^O,$$
$$\text{where} \quad \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V),$$

where h denotes the number of heads, $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$, and $W^O \in \mathbb{R}^{(h \times d_v) \times d_{model}}$.

Positional Encoding. The liberation from the recurrent structure made Transformers less effective in preserving sequence information. To overcome this issue, a 'Positional Encoding' is introduced to inject positional information of tokens in the sequence. The positional encoding (PE) is an $n \times d_{model}$ matrix, where n and d_{model} denote sequence length and the model's hidden dimension, respectively. The value of PE is determined by the sine and cosine functions:

$$PE_{pos,2i} = \sin(pos/10000^{2i/d_{model}}),$$
$$PE_{pos,2i+1} = \cos(pos/10000^{2i/d_{model}}).$$

PE is added to the input and output embeddings before being fed into the encoder and decoder.

Encoder. The encoder of Transformers consists of N layers, each consisting of a multi-head attention layer followed by a single fully-connected feed-forward layer. The input query, key, and value matrices are identical for each multi-head attention layer. In this regard, the multi-head attention of the encoder has another name, called 'Self Attention', because it only leverages an input sentence for computing attention weights. A residual connection [Li et al., 2017] and layer normalisation [Ba et al., 2016] techniques are applied around each of the two sub-layers, i.e., $\text{x}_{new} = \text{LayerNorm}(\text{x} + \text{Sublayer}(\text{x}))$.

Decoder. The decoder also consists of N layers but has two distinguishing attributes compared to the encoder. First, a masking is introduced in the self-attention layer to guarantee that the prediction for position i is only affected by the known outputs at positions less than i. Second, a new sub-layer is inserted that performs multi-head attention over the output of the masked self-attention layer and the output of the encoder stack. In this case, a query Q is the output of the masked self-attention layer, while the final output of the encoder is used as a value V and key K.

2.1.2 Pre-trained Language Models

The emergence of the Transformer architecture facilitated the training of large-sized models at unprecedented speed, leading to a new era of PLMs. PLMs are enormous models pre-trained on extensive volumes of unlabelled textual data. These models are applied to downstream tasks through fine-tuning, few-shot, and zero-shot learning.

Encoder-based Models. PLMs that only leverage the encoder of Transformers are pioneered by BERT [Devlin et al., 2019], which consists of a bi-directional Transformer encoder. The model showcased two unique features. Firstly, it employed a Wordpiece tokenisation [Johnson et al., 2017], wherein sentences are split into frequency-based subwords. Consequently, it facilitated the tokenisation of input sentences with little out-of-vocabulary. Second, BERT introduced segmentation embeddings that indicate the order of a sentence in an input text. In this way, BERT can effectively handle text inputs having multiple sentences.

BERT is a pre-trained self-supervised model with unlabelled corpora through MLM and next sentence prediction (NSP) tasks. MLM is the most renowned self-supervised task. For a sequence of tokens $\mathbf{T} = \{t_1, t_2, ..., t_N\}$, training data are generated by masking a certain portion of tokens. Assume that $\Pi = \{\pi_1, \pi_2, ..., \pi_K\}$ refers to the indexes of the masked tokens in the sentence $\mathbf{x}$. Let $\mathbf{T}_\Pi$ and $\mathbf{T}_{\neg\Pi}$ denote the set of masked and unmasked tokens in $\mathbf{T}$, respectively. The training objective of MLM is to predict the correct token in the masked position, which can be represented as follows:

$$\mathcal{L}_{mlm}(\mathbf{T}_\Pi|\mathbf{T}_{\neg\Pi}, \theta) = \frac{1}{K}\sum_{k=1}^{K} \log p(t_{\pi_k}|\mathbf{T}_{\neg\Pi}, \theta),$$

where θ are BERT's parameters, and K is the number of masked tokens.

NSP is a binary classification task that predicts whether the two given sentences are sequential neighbours. BERT is pre-trained on the two tasks simultaneously,

utilising a shared encoder but distinct classifiers. Following pre-training, the model is fine-tuned on downstream tasks by incorporating a task-specific layer into BERT's contextualised embeddings. BERT has garnered considerable attention due to its outstanding performance in the GLUE benchmark [Wang et al., 2018] and exceeding human-level performance in a QA task [Rajpurkar et al., 2016].

Despite the remarkable success of BERT, its unprecedented number of parameters posed a significant challenge to its practical usage. To address this concern, several studies have focused on training lighter PLMs while maintaining a high performance. Several works introduced structural modifications to BERT. For example, Lan et al. [2020] proposed the ALBERT model, which significantly reduced the size of BERT by introducing two techniques: cross-layer parameter sharing and parameter factorisation. By using these techniques, ALBERT achieved a notable reduction in weight, being approximately 20 times lighter than BERT, while experiencing only a marginal degradation in performance. Alternative studies employed the knowledge distillation technique [Hinton et al., 2014] to train a compact student model like DistilBERT [Sanh et al., 2019] that replicates the behaviours of BERT, a teacher model.

Another line of work endeavoured to train more elaborate PLMs. Liu et al. [2019b] conducted a thorough investigation on BERT, revealing that the impact of NSP is negligible and dynamic MLM is more effective than static MLM used in BERT. Additionally, they empirically found optimum hyperparameters for training. Clark et al. [2020] proposed a model named Electra by jointly training two Transformer encoders: generator and discriminator. The generator is trained using the MLM objective, whereas the discriminator is trained to discern whether the output tokens of the generator are original or replaced. During the fine-tuning process, only the discriminator is employed.

Decoder-based Models. GPT [Radford et al., 2018] is a representative model that only leverages the decoder of Transformers. The model also used WordPiece tokenisation to avoid the out-of-vocabulary issue. However, unlike the BERT family, the model is pre-trained through the autoregressive language modelling objective. For a sequence of tokens $\mathbf{T} = \{t_1, t_2, ..., t_k, ..., t_N\}$, GPT is trained to correctly predict the next word:

$$\mathcal{L}_{alm}(\mathbf{T}|\theta) = \frac{1}{N}\sum_{k=1}^{N} \log p(t_k|\mathcal{C}(t_k), \theta),$$

where $\mathcal{C}(t_k) = \{t_1, t_2, ..., t_{k-1}\}$, θ are GPT's parameters, and N is the total number of tokens.

As follow-up studies, GPT-2 [Radford et al., 2019] and -3 [Brown et al., 2020] have been proposed. GPT-2 shares the same structure as GPT but surpasses its size with 1.5 billion parameters. The model exhibited a decent performance by generating satisfactory answers in few-shot (which appends a few ground-truth examples as a context) and zero-shot settings, suggesting a new promising paradigm beyond fine-tuning. GPT-3 is an enormous model with 175 billion parameters and outperformed GPT-2 in zero-shot performance by a large margin. It also performed far better in a few-shot setting than fine-tuned state-of-the-art models in several tasks, such as the LAMBDA [Paperno et al., 2016] and PhysicalQA [Bisk et al., 2020] datasets.

The success of GPT-3 paved the way for the era of LLMs, and more elaborate LLMs have been proposed. Google's LaMDA [Thoppilan et al., 2022] is an LLM that improves safety and factual correctness through fine-tuning. To improve safety, Thoppilan et al. [2022] curated a human-annotated dataset that scores quality and safety criteria, such as Sensible, Interesting, and Unsafe. The model is fine-tuned to generate a response along with each score. During the application phase, responses are generated as the final output, provided that each criterion exceeds a certain threshold. To improve factual correctness, Thoppilan et al. [2022] first developed a toolset consisting of a calculator, translator, and information retrieval system. Subsequently, they fine-tuned the model for two tasks: 1) generating a special string "TS" to indicate that the input query should be forwarded to the toolset and 2) generating a grounded response by taking the snippet, which is returned by the toolset, and a dialogue statement. Human annotators also annotate the fine-tuning data. InstructGPT [Ouyang et al., 2022], the predecessor of ChatGPT, is trained by using a technique called reinforcement learning from human feedback (RLHF, Christiano et al. [2017], Stiennon et al. [2020]). Specifically, the process involves three distinct phases. In the first phrase, human annotators demonstrate the desired responses from the unlabelled prompt dataset and fine-tune GPT-3 through supervised learning. Subsequently, multiple outputs are sampled for a prompt, and the human annotators rank the outputs from the best to worst to train a reward model. Finally, InstructGPT is trained with reinforcement learning to maximise the reward generated by the reward model. Despite having significantly fewer parameters, the response generated by InstructGPT was favoured over that of GPT-3. InstructGPT greatly contributed to the development of ChatGPT, which astounded the global community by exhibiting a remarkable performance in professional examinations [Bommarito II and Katz, 2022, Kung et al., 2023, Terwiesch, 2023].

Encoder-Decoder Models. In contrast to the previously demonstrated PLMs that solely employ the encoder or decoder of the Transformer, PLMs that belong to this category leverage the complete Transformer architecture, i.e., both encoder and decoder. Lewis et al. [2020a] proposed an encoder-decoder Transformer named BART. The training objective of BART is *denoising*, where the model strives to restore the original uncorrupted form of perturbed textual inputs. For the noise generation, five methods are used: *token masking*, *token deletion*, *text infilling*, *sentence permutation*, and *document rotation*. After the pre-training, the model is fine-tuned on numerous tasks, such as machine translation, sentence classification, and sequence generation. Raffel et al. [2020] introduced a unified encoder-decoder framework called T5, which converts all NLP tasks into a text-to-text format. The Colossal Clean Crawled Corpus (C4) dataset takes about 750GB and is used during the pre-training phase. The training process is akin to that of BERT; approximately 15% of tokens are dropped, and the model is trained to correctly generate the removed spans of tokens. A fundamental characteristic of T5 is its approach of considering every NLP task as a text-generation problem. During fine-tuning, a task-specific symbol is appended to the input sentence, and the model is trained to generate a free-text label, e.g., "entailment" for an NLI task and "positive" for a sentiment analysis. This novel framework simplified the training of a multi-task model significantly, enabling its application in diverse text-generation tasks, including dialogue systems and explanation generation.

2.2 Pre-trained Language Models Lack Reliability

In contrast to the prevailing perspective that praises PLMs' language understanding ability based on their exceptional performance across various downstream tasks, a number of studies have uncovered PLMs' limitations in comprehending natural language. These findings have concurrently sparked concerns regarding the trustworthiness of such models.

Niven and Kao [2019] claimed that BERT heavily relies on exploiting spurious statistical cues that exist in the training set. To investigate this, they modified the English Argument Reasoning Comprehension dataset [Habernal et al., 2018] by adding adversarial examples where a single original token was negated. The finding revealed that BERT's performance on the modified dataset was far behind expectations, indicating the excessive usage of statistical cues presented in the original training dataset. Similarly, McCoy et al. [2019] discovered that BERT is only highly competent in leveraging syntactic heuristics. To show this, they introduced a novel

dataset called HANS, designed to diagnose whether models leverage fallible structural heuristics when making predictions on NLI tasks. Three heuristics used in the examination are as follows:

1. Lexical overlap: A premise entails all words in a hypothesis.
2. Subsequence: A premise entails the contiguous subsequence in a hypothesis.
3. Constituent: A premise entails a complete subtree in a hypothesis.

Intuitively, the heuristics result in the high textual similarity between the premise and hypothesis, consequently causing the model to assign the label *entailment* despite the true label being *not-entailment*, provided the model heavily depends on the heuristics. The experimental findings revealed that state-of-the-art NLI models, including a fine-tuned BERT, performed poorly on instances with a *not-entailment* label, recording less than 10% accuracy in most test cases. McCoy et al. [2019] also showed that introducing HANS-like training instances can greatly improve performance. Bender and Koller [2020] claimed the predominant belief that PLMs understand or comprehend natural language is hype. Through a carefully crafted thought experiment called the *octopus test*, they demonstrated that learning the meaning conveyed in natural language solely through the textual form is almost impossible. Consequently, Bender and Koller [2020] asserted that PLMs, which only utilise textual form during the pretraining stage, inherently lack the capacity to comprehend natural language, which subsequently undermines the reliability of such models.

Another line of research ascertained that PLMs process natural language in a distinct manner compared to human cognition. Multiple studies confirmed that PLMs demonstrate insensitivity to word order, unlike humans, who display a phenomenon known as the sentence superiority effect [Pham et al., 2021, Sinha et al., 2021b, Gupta et al., 2021]. These studies confirmed that PLMs generate accurate predictions on instances where the word order entirely deviates from grammatical correctness, often with higher confidence scores than those having grammatically correct word orders.

Furthermore, numerous works revealed that PLMs do poorly understand negation expressions. Ettinger [2020] introduced a diagnostic dataset that includes the examination of understanding the meaning of negation. They generated a masked sentence using a hypernym relation and its corresponding negated sentence by adding negation expressions "not", for example, "A robin is a [MASK]" and "A robin is not a [MASK]". Next, they let PLMs fill in the missing word through MLM and observed a substantial overlap in the predicted outcomes of the two sentences. This

finding suggested that PLMs cannot correctly understand the meaning of negation. Hossain et al. [2020] performed similar experiments regarding negation by negating the LAMA dataset [Petroni et al., 2019] and confirmed analogous experimental outcomes. In addition, they conducted a novel experiment, called *mispriming*, drawing inspiration from priming methods in human psychology. Specifically, they appended automatically produced misprimes to masked sentences[1] and performed MLM. The experimental results revealed that PLMs are prone to be misled to fill in the masked word with a misprime word, in contrast to humans, who can easily ignore it. [Kassner and Schütze, 2020] introduced a new benchmark for NLI where negation expressions are pivotal in making predictions. They subsequently confirmed that state-of-the-art transformer-based models encounter challenges in making correct decisions with this new dataset. [Kalouli et al., 2022] also performed MLM-based probing experiments based on theoretical linguistic insights. Specifically, they investigated PLMs' understanding of functional words, including negation (not/no/without), coordination (and/or), and quantifiers (all/some/no). Following a similar approach to the aforementioned studies, they compared the predictions of the original and perturbed masked queries. They observed the presence of wrong answers within these outputs. Other studies revealed that PLMs lack basic knowledge and understanding regarding numerical expressions and mathematical calculations [Wallace et al., 2019, Nogueira et al., 2021, Frieder et al., 2023].

An additional body of research delved into exploring whether PLMs exhibit logically consistent behaviours. Ribeiro et al. [2020] proposed an evaluation dataset called CheckList, which includes data for the *invariance test* that is designed to assess a model's robustness against meaning-preserving perturbations. To achieve this, they collected data for sentiment analysis by substituting named entities within input sentences, as these entities do not inherently change the polarity of the sentence. Ribeiro et al. [2020] confirmed that PLM-based state-of-the-art commercial sentiment analysis models are not perfectly robust against the meaning-preserving perturbations. Ravichander et al. [2020] conducted MLM-based experiments by asking PLMs to fill in the masked token within the original query and its perturbed counterpart, where an object is replaced with its plural form.[2]

Similarly, Elazar et al. [2021] carried out the same experiment but employed a different perturbation method that paraphrased the original query.[3] Both studies

[1]For instance, "Talk? Birds can [MASK]" where "Talk?" is a misprime.

[2]For instance, "A car is a [MASK]" vs "Cars are [MASK]".

[3]For instance, "Homeland originally aired on [MASK]" and "Homeland premiered on [MASK]".

confirmed that PLMs make distinctive predictions for the original inputs and their meaning-preserving perturbations. Research on textual adversarial attacks [Jin et al., 2020, Garg and Ramakrishnan, 2020, Li et al., 2020a, 2021, Ivgi and Berant, 2021], which aims to generate adversarial samples that deliver the same meaning as their original counterparts but confuse the model, also demonstrated the susceptibility of PLMs to meaning-preserving perturbations. Wang et al. [2019c] introduced a symmetric property to consistent behaviour testing of PLMs on NLI tasks. The symmetric property implies that a model should be insensitive to the input sentence order for downstream tasks that are inherently order-invariant. Wang et al. [2019c] suggested that the symmetric property applies to examples labelled *contradiction* and *neutral*. They investigated models' predictions after switching the premise and hypothesis of these examples.

On the contrary, Li et al. [2019] argued that the property is only applicable to instances that are labelled as a *contradiction*. Both studies observed a notable shift in predictions of fine-tuned modern PLMs when the premise and hypothesis are switched. Kumar and Joshi [2022] further expanded the symmetric investigation to STS tasks and confirmed the violation of the symmetry property. Another logical property used for PLMs' consistent behaviour testing is transitivity, represented as $X \rightarrow Y \wedge Y \rightarrow Z$ then $X \rightarrow Z$ [Gazes et al., 2012, Asai and Hajishirzi, 2020]. Li et al. [2019] constructed a new NLI evaluation benchmark by defining four transitive inference rules. For example, suppose the relation between sentences A and B is *entailment* and between B and C is *entailment*. In that case, the relation between the sentences A and C should also be *entailment*. In QA, Asai and Hajishirzi [2020] collected data to explore the transitivity of QA models. They combined two questions, q_1 and q_2, where the effect of q_1 is the cause of q_2.[4] Lin and Ng [2022] explored PLMs' transitive inference ability on word senses through MLM. They employed the **Is-A** relation in WordNet to construct the evaluation data. For example, if the two knowledge triplets are {poodle, Is-A, dog} and {dog, Is-A, animal}, the new triplet would be {poodle, Is-A, animal}. All three studies ascertained that PLMs do not always make logically consistent decisions.

[4]For example, assume Q1: "If it is silent, does the outer ear collect less sound waves?" and "Q2: If the outer ear collects less sound waves, is less sound being detected?". The new question would be "If it is silent, is less sound being detected?".

2.3 Alleviating Inconsistency Issues of Pre-trained Language Models

Two principal avenues of research have been conducted to mitigate the inconsistency challenges inherent in PLMs. The first approach involves data augmentation, which entails collecting augmented training data that adhere to specific consistency types. These new data are further integrated with the original dataset during the pre-training or fine-tuning stages. For instance, Asai and Hajishirzi [2020] automatically generated supplementary training data by leveraging the logical negation and transitive property in QA tasks, thereby improving the negational and transitive consistency.[5]. Similarly, Hosseini et al. [2021] amassed data instances containing negation expressions to facilitate training an enhanced PLM endowed with heightened proficiency in negation comprehension. In the context of semantic consistency, adversarial training is employed. This approach involves the generation of adversarial samples by introducing minor perturbations to vector representations [Zhu et al., 2020, Kitada and Iyatomi, 2021, Pan et al., 2022] or textual structures [Ray et al., 2019, Jin et al., 2020] in a manner that preserves their intrinsic meaning. These generated samples are subsequently integrated into the training process alongside the original instances, thereby fostering the advancement of semantic consistency.

The second strategy involves incorporating supplementary loss functions in conjunction with the data augmentation process. These auxiliary loss functions are specially devised terms aimed at imposing penalties upon undesired behaviours demonstrated by PLMs. For example, Asai and Hajishirzi [2020] introduced negational and transitive inconsistency loss terms, which are defined as an absolute loss of predictive probabilities. Suppose two QA instance pairs $(q, p, a^*) \leftrightarrow (q_{aug}, p, a_{aug})$, where p denotes a paragraph, q and q_{aug} are the antonyms of each other, and a_{aug} is the opposite of the ground-truth answer a^*. Then, the consistency loss term is defined as follows:

$$\mathcal{L}_{aug} = |\log p(a|q, p) - \log p(a_{aug}|q_{aug}, p)|.$$

When it comes to the transitive consistency, they projected the logical conjunction operation $(q_1, p, a_1) \wedge (q_2, p, a_2)$ to a product of probabilities of two operations, i.e., $p(a_1|q_1, p)p(a_2|q_2, p)$, and defined the transitive consistency loss term as follows:

$$\mathcal{L}_{tran} = |\log p(a_1|q_1, p) + \log p(a_2|q_2, p) - \log p(a_{tran}|q_{tran}, p)|.$$

[5]Note that the phrase "symmetric consistency", as used by Asai and Hajishirzi [2020] in the context of QA, is classified as negational consistency from the perspective of this dissertation.

In a similar context, Elazar et al. [2021] implemented a consistency loss function that compels a model to generate similar predictive distributions between the original instance and its rephrased version using two-sided KL-divergence. They employed augmented paraphrase data to achieve this purpose. Kumar and Joshi [2022] utilised the same approach to enhance symmetric consistency. Li et al. [2020b] employed the unlikelihood training technique [Welleck et al., 2020], which penalises the generation of unfavourable predictions, as a preventive measure against the generation of contradictory predictions.

Chapter 3

Exploration of Undesirable Behaviours Exhibited by Pre-trained Language Models

In this chapter, I will explore two distinct undesirable behaviours exhibited by PLMs that undermine their trustworthiness. The first behaviour involves PLMs' deficiency in understanding negation expressions and antonyms, while the second one addresses the generation of logically contradictory explanations. Following these investigations, concise and readily applicable methodologies that can alleviate the issues will be presented. Lastly, a comprehensive analysis of the findings will be provided.

3.1 Introduction

Contemporary Transformer-based PLMs like BERT [Devlin et al., 2019], which are trained on vast corpora in a self-supervised manner, have played a pivotal role in the rapid progress of the field of NLP. These models exhibited a remarkable performance across numerous downstream tasks and even surpassed human-level performance in several benchmark datasets, such as GLUE [Wang et al., 2018] and SuperGLUE [Wang et al., 2019b]. These promising achievements have solidified PLMs as the foundation of modern NLP systems.

Despite their remarkable accomplishments, however, there has been an escalating concern regarding the trustworthiness of PLMs. Numerous studies have ascertained through various probing tasks that PLMs demonstrate behaviours that deviate from human-like patterns and exhibit logically incorrect behaviours. These include insensitivity to grammatical word order [Pham et al., 2021, Gupta et al., 2021, Sinha et al., 2021b], difficulties in comprehending number-related representations [Wallace et al.,

2019, Lin et al., 2020a, Nogueira et al., 2021], inadequate understanding of semantic content [Ravichander et al., 2020, Elazar et al., 2021], and limitations in comprehending negation expressions [Ettinger, 2020, Kassner and Schütze, 2020, Hossain et al., 2020]. The presence of these fallacious behaviours gave rise to concerns regarding PLMs' trustworthiness, thereby undermining their suitability for practical applications, particularly in risk-sensitive domains.

This chapter focuses on the examination of PLMs' trustworthiness. Trustworthiness is a general concept, and its definition varies in different contexts. Throughout this dissertation, the definition of Huang et al. [2020] is adopted, where the concept consists of *Certification* and *Explanation* as follows:

$$\textit{Trustworthiness} = \{\textit{Certification, Explanation}\}.$$

A certification is a process that ascertains where an AI system behaves correctly, while the explanation involves providing explanations to elucidate actions undertaken by the AI system. In other words, a user will trust an AI system if it successfully undergoes the certification examination and its behaviour can be explained well. This chapter utilises two probing tasks for a comprehensive examination to ascertain the trustworthiness of PLMs based on these two concepts.

Regarding *certification*, I centre on assessing the ability of PLMs to adhere to the logical negation property (p is true iff $\neg p$ is false; see [Aina et al., 2018]). When applied to NLP tasks where the property holds, this implies that PLMs should demonstrate different decisions when presented with inputs delivering the opposite meaning. The investigation of the logical negation property in PLMs includes previous studies that examined PLMs' incapability to understand negation expressions. However, these works restricted their scope of investigation solely to the functional words, such as the addition or removal of negation expressions (e.g., "no" and "not"). In contrast to these works, I propose the probing task that extends the boundary of the logical negation property to encompass lexical semantics, i.e., synonyms and antonyms. Experimental results suggest that PLMs are prone to violate the logical negation property in terms of both functional words and lexical semantics.

When it comes to *explanation*, I concentrate on the generation of contradictory NLEs. Contradictory NLEs refer to explanations generated by a model under the same context yet contradictory to each other. Figure 3.1 presents an example of contradictory NLEs. Assume a self-driving car stops in a given traffic environment (the context). A passenger asks the car "Why did you stop?" and receives NLE1 as a response: "Because the traffic light is red.". However, if the passenger poses a different question in the same context, "Why did you decide to stop here?" and if the

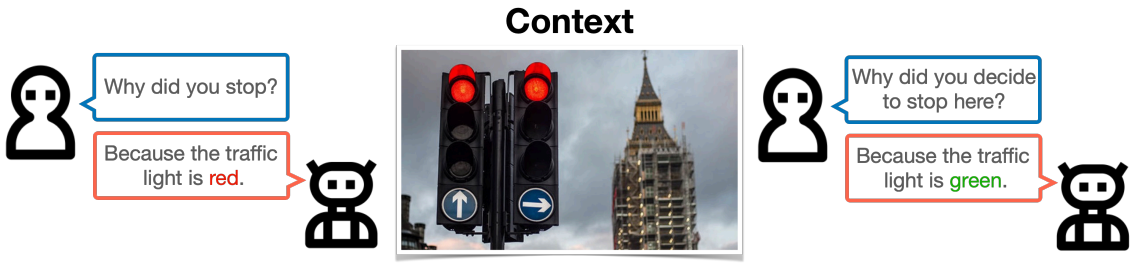


Figure 3.1: An example of contradictory natural language explanations. The VQA model generated two distinct answers, which are inherently contradictory, when presented with the same context (image) but different variables (question).

car provides NLE2 as the explanation, stating "Because the traffic light is green.", then NLE1 and NLE2 form a contradiction.

Although the availability of human-written NLE datasets [Wiegreffe and Marasovic, 2021] facilitated the development of models capable of providing NLEs for their predictions, the introduction of **e**xplanation **C**ontradictory **A**ttack (eCA, Camburu et al. 2020) revealed that early NLE generation models [Camburu et al., 2018] were susceptible to generating contradictory NLEs. In order to investigate the trustworthiness in terms of *explanation*, I propose an enhanced attack method that offers advantages in terms of speed, efficiency, and task generalisability compared to the eCA method. The proposed attack method is applied to high-performing NLE generation models utilising PLMs as a backbone architecture. Experimental results indicate that even high-performing models are still likely to generate contradictory explanations.

On top of the exploration of PLMs' trustworthiness through the two probing tasks, in this chapter, I present simple yet effective methods that alleviate the undesirable behaviours exhibited by PLMs. A novel task called **I**ntermediate-training on **M**eaning-**M**atching (IM^2) is introduced to address the violation of the logical negation property. It is hypothesised that the leading cause of the issue lies in the MLM training objective, which assumes the distributional hypothesis for learning textual meaning. Instead, the IM^2 task enables direct learning of the association between words and their semantic contents. Regarding mitigating the issue of contradictory NLE generation, this chapter introduces an approach that extracts relevant background knowledge from a knowledge base and grounds NLE generation models to the extracted knowledge.

The main contributions of this chapter are briefly summarised as follows:

- The investigation scope of the logical negation property is expanded from functional words to lexical semantics.

- An enhanced adversarial NLE attack method, which is much more efficient than the previously proposed eCA method, is proposed and applied to high-performing NLE generation models.
- The results of the probing tasks indicate that PLMs are susceptible to violating the logical negation property, and high-performing NLE generation models are likely to produce contradictory NLEs.
- I propose the IM^2 method to mitigate the violation of the logical negation property by enhancing the capability of PLMs to capture more accurate textual semantics.
- I propose a method that grounds NLE generation models to background knowledge to produce fewer contradictory NLEs.
- The experimental results suggest that the proposed approaches contribute to alleviating the undesirable behaviours exhibited by PLMs.

3.2 Probing Tasks for Exploring PLMs' Undesirable Behaviours

3.2.1 Assessment on the Logical Negation Property

The logical negation property implies "p is true iff $\neg p$ is false" [Aina et al., 2018]. When applied to NLP tasks, the property indicates that the model should generate different outputs when presented with two inputs that convey semantically opposite meanings. Note that the property does not universally apply to all tasks.

The property applies exclusively to tasks and data instances where the target label is altered when the original text is modified to convey an opposite meaning. For instance, when it comes to the STS task, the property applies if the label assigned to a pair of sentences is "equivalent," indicating that the two texts convey an identical meaning. If one of the given sentences is modified to deliver an opposite meaning, the answer for the modified text pair should be "not_equivalent". Similarly, regarding the NLI task, this property applies to instances labelled "entailment". However, suppose instances are labelled "not_equivalent" in the STS task. In that case, the property is inapplicable, because the target label may remain "not_equivalent" even when one sentence is modified to deliver an opposite meaning. Similarly, the logical negation property does not apply to topic classification, where a sentence can pertain to the

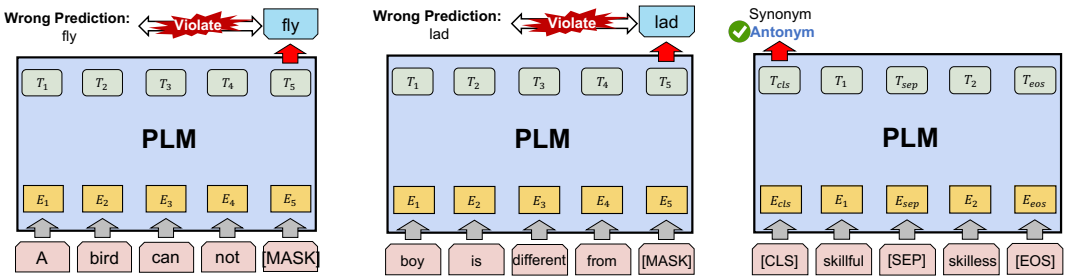
same topic despite the presence of contradictory content. For example, "Rishi Sunak was elected as prime minister" and "Rishi Sunak was not elected as prime minister" both address the same topic.

Three probing tasks are introduced to investigate whether PLMs satisfy the logical negation property: MKR-NQ, MWR, and SAR. Brief illustrations of each task are presented in Figure 3.2.

3.2.1.1 Masked Knowledge Retrieval on Negated Queries

The MKR-NQ task aims to measure the proficiency of PLMs in comprehending negation expressions by evaluating their tendency to produce incorrect answers for negated queries. Following the research of Kassner and Schütze [2020], the evaluation dataset is created by negating the LAMA dataset [Petroni et al., 2019], which consists of masked free-text forms of ConceptNet [Speer et al., 2017] triplets along with their corresponding answers. For example, the LAMA data instance ("A bird can [MASK]", fly)) can be derived from a knowledge triplet (bird, CapableOf, fly). The task's objective is to measure how likely PLMs are to generate a correct answer through MLM.

According to the logical negation property, it is expected that a model should not generate the original answer when the query is negated. To assess how likely PLMs are to produce incorrect predictions for negated queries, I collected the pairs of (negated_query, wrong_predictions). From the instances in the LAMA dataset, those having a particular relation were selected to ensure mutual exclusivity of the original and negated queries. For instance, the *HasProperty* relation is unsuitable to be used, as sentences like "Some adults are immature" and "Some adults are not immature" are not mutually exclusive, i.e., adults can be both mature and immature. The relations employed for data collection encompass {*IsA, CapableOf, PartOf, HasA, UsedFor, MadeOf, NotDesires*}. Subsequently, instances containing multiple verbs were eliminated within the chosen LAMA data points. Spacy's parts of speech (POS) tagger [Honnibal and Johnson, 2015] was used to identify the verbs existing in a sentence. Then, negation expressions, such as "not" and "don't" were added to the sentences or removed if such expressions already existed. Finally, the wrong predictions, which correspond to the correct answers of the original queries, were collected from ConceptNet by conducting searches using the head entity and relation. As a result, 3,340 instances were created for the MKR-NQ task. The examples of the collected dataset are presented in Table A.2 in Appendix A.4.



(a) MKR-NQ (b) MWR (c) SAR

Figure 3.2: Illustration of the MKR-NQ, MWR, and SAR tasks. The MKR-NQ and MWR tasks generate a word through MLM, whereas the SAR task is a binary classification task.

3.2.1.2 Masked Word Retrieval

The MWR task is designed to expand the investigation boundary of the logical negation property from functional words to lexical semantics. In this task, PLMs are presented with a masked query to generate the synonym or antonym of a given target word using the MLM (e.g., "happy is the synonym of [MASK]").

Let s_w and a_w represent the masked queries that ask about the synonym and antonym of the target word w, respectively. Also, let A_s and A_a denote the list of correct predictions for s_w and a_w, respectively. Intuitively, A_a can be regarded as the list of incorrect predictions for s_w, because s_w and a_w possess the opposite meaning. Similarly, A_s can be considered as the wrong prediction for s_a. Therefore, similarly to the evaluation conducted in the MKR-NQ task, it is possible to assess the violation of the logical negation property by investigating whether a PLM produces incorrect predictions for a given query.

In order to construct a dataset for the MWR task, a selection of frequently used English nouns, adjectives, and adverbs was first made. Candidates were identified based on their occurrence of more than five times in the SNLI dataset [Bowman et al., 2015]. Spacy's POS tagger [Honnibal and Johnson, 2015] was utilised to identify the POS tags. Subsequently, the filtering process was conducted to identify words containing synonym or antonym relations in ConceptNet [Speer et al., 2017]. Finally, free-text masked queries were generated by employing the templates defined by Camburu et al. [2020]. As a consequence, 27,000 data instances were created for the MWR task. The templates used to create the data and examples are demonstrated in Table A.4 in Appendix A.4.

3.2.1.3 Synonym/Antonym Recognition

The SAR task is a classification task that aims to distinguish whether two given words are in a synonym or antonym relation. The primary objective of the SAR task is to assess the extent to which the contextualised representations of PLMs reflect the lexical meaning of words. To achieve this purpose, a parametric probing model [Adi et al., 2017, Liu et al., 2019a, Belinkov and Glass, 2019, Sinha et al., 2021a] was employed for the experiment. Specifically, the encoder of PLMs remained fixed, and only the final layer of each PLM, i.e., the classifier, was trained throughout the experimental process.

Synonyms and antonym relations in ConceptNet [Speer et al., 2017] were used for building data instances. However, the knowledge base contains many more synonym triplets compared to antonyms, which can cause a class imbalance problem. Consequently, a balanced distribution of synonym and antonym classes was achieved by randomly sampling synonym triplets. To that end, 33,000, 1,000, and 2,000 data instances were created for the training, validation, and test sets, respectively.

3.2.1.4 Evaluation Metrics

To evaluate the performance on the MKR-NQ and MWR tasks, the top-k hit rate (HR@k) was introduced. Assume the set of predictions for a given data point x as $P = \{(p_1, c_1), (p_2, c_2), \ldots, (p_n, c_n)\}$, where p_t and c_t represent the predicted word and confidence score of the t-th prediction, respectively. Then, the top-k hit rate for a data point x can be defined as follows (Eq. 3.1):

$$HR@k(x) = \frac{\sum_{i=1}^{k} \mathbb{1}(p_i \in \mathcal{W}_x)}{k}, \tag{3.1}$$

where $\mathcal{W}_x$ is the set of incorrect predictions of x. The metric has a value ranging from 0 to 1. Intuitively, the metric quantifies the proportion of the top-k predicted words that belong to the set of wrong predictions.

In addition to the top-k hit rate, the weighted top-k hit rate (WHR@k), which leverages the predictive confidence score as weights, was employed to reflect the confidence score to the evaluation metric. Similar to the top-k hit rate, the weighted top-k hit rate also has a value ranging from 0 to 1. It is worth mentioning that lower metrics indicate better performance in both metrics, as they evaluate how likely the models make wrong answers that they must avoid. The weighted top-k hit rate can be defined as follows (Eq. 3.2):

$$WHR@k(x) = \frac{\sum_{i=1}^{k} c_i \times \mathbb{1}(p_i \in \mathcal{W}_x)}{\sum_{i=1}^{k} c_i}. \tag{3.2}$$

Regarding the SAR task, a simple accuracy was utilised as an evaluation metric, because the label distribution is perfectly balanced due to the random synonym sampling strategy.

3.2.2 Assessment of Contradictory Explanation Generation

To examine how likely PLM-based high-performing NLE generation models produce contradictory NLEs, I present the **e**xplanations **Know**ledge-grounded **C**ontradictory **A**ttack (eKnowCA) method, which detects more contradictory NLEs in a faster and more general manner compared to the previously proposed eCA method [Camburu et al., 2020]. Note that the eCA and eKnowCA methods were exclusively designed to identify contradictory NLEs and not intended as label attacks.

3.2.2.1 Original eCA method

Basic Setting. Given an input instance x, Camburu et al. [2020] separated the input into: the *context* part ($\mathrm{x_c}$), which remains fixed, and the *variable* part ($\mathrm{x_v}$), which is changed during the adversarial attack. In the NLI task, for example, $\mathrm{x_c}$ and $\mathrm{x_v}$ would be a *premise* and a *hypothesis*, respectively. Let $\mathrm{e}_m(\mathrm{x})$ refer to the NLE generated by a model m for the input $\mathrm{x} = (\mathrm{x_c}, \hat{\mathrm{x}}_\mathrm{v})$. The objective of the adversarial attack is to identify $\hat{\mathrm{x}}_\mathrm{v}$ such that $\mathrm{e}_m(\mathrm{x})$ and $\mathrm{e}_m((\mathrm{x_c}, \hat{\mathrm{x}}_\mathrm{v}))$ are logically contradictory.

Steps. The eCA method performs the attack based on the following steps:

1. Train a reverse explainer (RevExpl). The model takes $\mathrm{x_c}$ and $\mathrm{e}_m(\mathrm{x})$ as input and generates $\mathrm{x_v}$: $\mathrm{x_v} = \textsc{RevExpl}(\mathrm{x_c}; \mathrm{e}_m(\mathrm{x}))$.
2. For each model-generated NLE ($\mathrm{e}_m(\mathrm{x})$):
   (a) Automatically create a set of statements $\mathcal{I}_e$ that are logically contradictory with $\mathrm{e}_m(\mathrm{x})$.
   (b) For each $\hat{e} \in \mathcal{I}_e$, create a variable part ($\hat{\mathrm{x}}_\mathrm{v}$) by using the context part and reverse explainer: $\hat{\mathrm{x}}_\mathrm{v} = \textsc{RevExpl}(\mathrm{x_c}; \hat{e})$.
   (c) Ask the NLE generation model m on $\hat{x} = (\mathrm{x_c}, \hat{\mathrm{x}}_\mathrm{v})$ to get a reverse explanation $\mathrm{e_m}(\hat{\mathrm{x}})$.
   (d) Verify the contradiction between $\mathrm{e_m}(\hat{\mathrm{x}})$ and $\mathrm{e_m}(\mathrm{x})$ by confirming whether $\mathrm{e_m}(\hat{\mathrm{x}})$ is included in $\mathcal{I}_e$.

A brief illustration of the attack process is described in Figure 3.3.

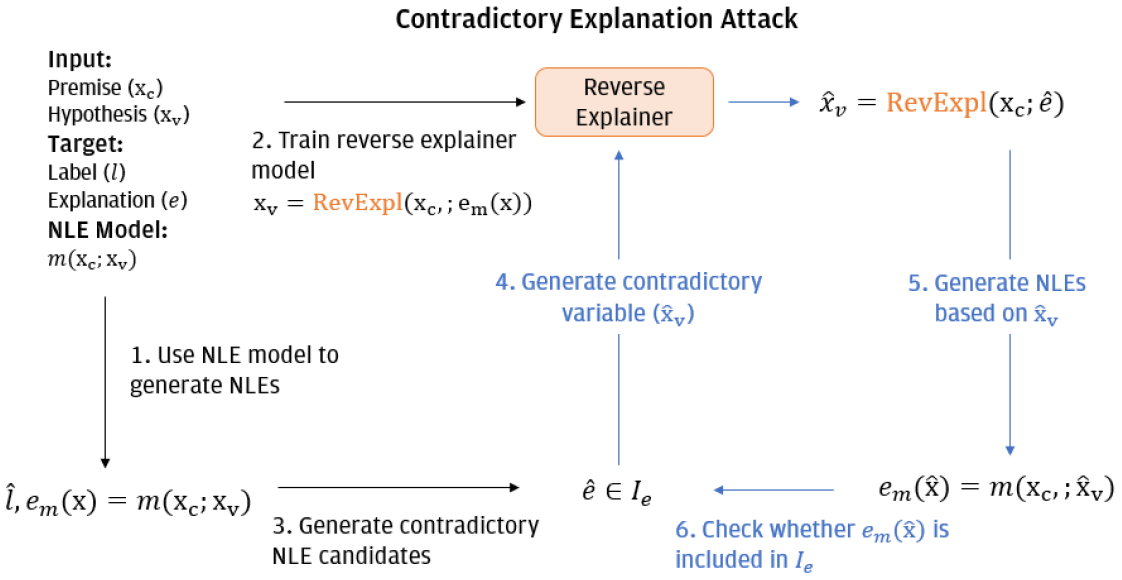


Figure 3.3: The overall framework of the eCA method.

Creating $\mathcal{I}_e$. The key process in the eCA method is the automated generation of $\mathcal{I}_e$. Camburu et al. [2020] employed two heuristics for this procedure. First, they eliminated negation expressions presented in the statement ("not" or "n't"). Second, they constructed task-specific templates to generate $\mathcal{I}_e$. Camburu et al. [2020] applied the eCA attack only on the e-SNLI dataset [Camburu et al., 2018], which contains NLEs of the SNLI dataset [Bowman et al., 2015], where the task aims to determine the relationship between a *premise* and a *hypothesis*, specifically whether it exhibits *entailment* (if the premise entails the hypothesis), *contradiction* (if the hypothesis contradicts the premise), or *neutral* (if neither entailment nor contradiction hold). Camburu et al. [2020] manually created a collection of label-specific templates for NLE to form contradictions. The instance-specific term s of an NLE from one template were introduced into any template from another label, resulting in the creation of a contradiction. As a result, the template-based approach only applies to the NLI task. Examples of the templates are: "[X] is [Y]" (for *entailment*) and "[X] cannot be [Y]" (for *contradiction*) where a statement "A dog is an animal.", a contradictory statement of "A dog cannot be an animal" is obtained ([X] = "A dog", [Y] = "an animal") based on the templates. Camburu et al. [2020] manually created an average of 10 templates per label.

3.2.2.2 eKnowCA method

While eCA is a well-designed framework for the contradictory explanation attack, it contains several drawbacks regarding generating $\mathcal{I}_e$. Regarding the negation rule, its current implementation solely eliminates negation expressions. However, there is room for improvement by additionally considering adding the negation expressions if they are absent in a given statement. Next, the template-based approach is a task-specific method, which is unsuitable for other downstream tasks. Implementing new templates is necessary to apply eCA to other tasks, which requires human effort to identify a comprehensive collection of templates for each dataset, diminishing its overall applicability across diverse downstream tasks. Moreover, the approach entails a significant time commitment for generating $\mathcal{I}_e$ due to the substantial number of template combinations that form a contradiction. Considering an average of 10 templates per label, the expected size of the set $\mathcal{I}_e$ of a data instance is 100, which would substantially amplify the overall time consumption for conducting eCA. To mitigate these limitations, three heuristics were adopted, significantly enhancing the speed and generalisability of the generation of $\mathcal{I}_e$.

Negation. The improved negation rule not only removes negation tokens from negated statements but also inserts them for non-negated sentences. Provided a statement contains only one negation token, it is eliminated from the statement. Regarding the insertion of negation tokens, the rule adds a single negation token per statement only if the statement corresponds to one of the following templates to avoid possible grammatical errors:

- **A** is **B**, **A** are **B** (add "not").
- **A** has **B**, **A** have **B** (add "does/do not" only if **B** is a noun[1]).

Antonym replacement for adjectives/adverbs. This rule involves the substitution of adjectives or adverbs with their respective antonyms. Only one adjective or adverb is replaced at a time for each statement. For instance, if a statement contains two adjectives and one adverb, three potential candidates for contradictory statements can be generated. This guideline is taken to prevent the deterioration of contradictory meanings that may arise from simultaneously replacing multiple adjectives or adverbs. Antonyms were collected from ConceptNet, and the Spacy POS tagger was utilised to identify adjectives and adverbs.

[1]Spacy's POS tagger [Honnibal and Johnson, 2015] is used to identify POS tags.

Unrelated noun replacement. This rule replaces a noun in a statement with an unrelated one, e.g., "human" with "plant". Like above, the Spacy POS tagger was used to verify nouns. The rule only applies to the noun at the end of each statement due to occasional inaccurate predictions made by the POS tagger for words in the middle of a sentence, potentially leading to false contradictions. To get unrelated nouns, *DistinctFrom* and *Antonym* relations in ConceptNet were employed. However, it was confirmed that ConceptNet contains noisy knowledge triplets where the subject and objects are hardly recognised as antonyms.[2] Examples of such cases include {"man", Antonym, "person"} and {"people", "Antonym", "person"}. To avoid these, a list of ConceptNet triplets to be disregarded was created through a manual investigation of a random subset of 3,000 generated contradictory statements. The list is provided in Table A.5 in Appendix A.4. Although this process involves a modest amount of human effort, it is worth highlighting that this is due to the inherent characteristics of ConceptNet and implementing more accurate and enhanced knowledge bases could mitigate the issue. Finally, it was revealed that replacing with an unrelated noun does not lead to contradiction if both the original and generated explanations contain negation expressions. Hence, a supplementary rule was introduced to filter out such cases, ensuring the preservation of contradictions. Examples corresponding to this case are described in Table A.6 in Appendix A.4.

3.2.2.3 Evaluation Metrics

Let $\mathcal{I}_e$ denotes a set of contradictory NLEs generated at Step 2a for each instance in a test set $\mathcal{D}_{test}$, and let $\mathcal{I}_s \subseteq \mathcal{I}_e$ denote the set of detected contradictory NLEs after Step 2d. For each instance, both eCA and eKnowCA can detect multiple contradictions by employing multiple variable parts. To measure how likely NLE generation models produce contradictory NLEs, two evaluation metrics: hit-rate ($\mathcal{H}_r$) and success-rate ($\mathcal{S}_r$) were introduced, which are calculated as follows (Eq. 3.3):

$$\mathcal{S}_r = \frac{N_c}{|\mathcal{D}_{test}|} \text{ and } \mathcal{H}_r = \frac{|\mathcal{I}_s|}{|\mathcal{I}_e|}, \tag{3.3}$$

where N_c denotes the count of distinct test instances for which the attack successfully detected at least one contradiction. Both metrics have a value ranging from 0 to 1. Intuitively, $\mathcal{S}_r$ represents the ratio of the test instances in which the attack achieves success, whereas $\mathcal{H}_r$ represents the ratio of detected contradictory NLEs to the proposed contradictory NLE candidates. For both metrics, higher values indicate

[2] A pair of words with *opposite* meanings (from Wikepedia).

Model	MKR-NQ					MWR				
	HR@1	HR@3	WHR@3	HR@5	WHR@5	HR@1	HR@3	WHR@3	HR@5	WHR@5
BERT-*base*	9.57	6.38	8.42	5.00	7.81	35.03	18.83	28.71	13.26	26.03
BERT-*large*	13.33	7.70	11.17	6.03	10.51	36.66	20.45	29.68	14.60	26.56
RoBERTa-*base*	11.52	6.85	9.63	5.30	8.91	13.02	8.47	10.50	6.54	9.32
RoBERTa-*large*	15.72	9.25	13.31	6.86	12.24	27.92	17.07	23.06	12.84	20.82
ALBERT-*base*	4.24	3.75	4.24	3.26	4.09	26.37	14.95	22.10	10.75	20.32
ALBERT-*large*	9.22	6.38	7.96	4.94	7.30	50.77	25.05	42.67	17.03	39.09

Table 3.1: Overall results of the MKR-NQ and MWR experiments. Top-1,3,5 hit rates and weighted hit rates are provided. For each value, 100 is multiplied to improve readability. Note that the lower the values, the better.

superior attack performance, signifying the increased success of the attack method. Conversely, from the perspective of the NLE generation model, the metrics are lower the better, indicating a greater robustness against the attack.

3.3 Experiments of the Undesirable Behaviours of PLMs

3.3.1 Experiments on the Logical Negation Property

The following widely-used PLMs pre-trained with the MLM training objective were selected for the experiments: BERT-*base/large* [Devlin et al., 2019], RoBERTa-*base/large* [Liu et al., 2019b], and ALBERT *base/large* [Lan et al., 2020]. ELECTRA-*small/base/large* models were additionally incorporated for the SAR task, while they were excluded from the MKR-NQ and MWR experiments. This decision was made, because ELECTRA models employ a discriminator for downstream tasks, which are trained using a replaced token prediction training objective and have no MLM classifier. For the SAR task, each PLM was fine-tuned for 10 epochs with a batch size of 32. An early stopping strategy was applied to stop training if there was no improvement in validation performance for three consecutive training epochs. The AdamW optimiser [Loshchilov and Hutter, 2019] with a learning rate of $5e^{-6}$ was used for the training. No additional training was required for the MKR-NQ and MWR tasks.

3.3.1.1 Results for MKR-NQ

The second column in Table 3.1 provides the results of the MKR-NQ task. In general, the results demonstrate that all PLMs exhibit mistakes in generating incorrect

	BERT		RoBERTa		ALBERT	
	base	*large*	*base*	*large*	*base*	*large*
$\mathcal{R}_{syn}$	37.79	36.58	17.13	46.01	11.37	76.10
$\mathcal{R}_{ant}$	41.44	41.79	13.57	30.21	33.26	62.65

Table 3.2: The ratios of instances in which PLMs regenerate the word present in the input sentence. $\mathcal{R}_{syn}$ and $\mathcal{R}_{ant}$ are the ratios of synonym and antonym-asking questions, respectively.

predictions for negated queries, which is consistent with previous studies [Ettinger, 2020, Kassner and Schütze, 2020]. Furthermore, the experimental results unveil three important characteristics.

First, it was discovered that the large models consistently produce a higher hit rate compared to their corresponding base-size models across all three PLMs, exhibiting an average increase of approximately 1.5 times. The observation implies that large-size models are prone to generating incorrect predictions for negated queries despite their superior performance to small-size models in various benchmark downstream tasks. The results indicate that relying solely on quantitative metrics of downstream tasks to evaluate a model's performance poses the risk of both underestimating and overestimating the PLMs' language understanding capability. Hence, a comprehensive evaluation from multiple perspectives should be conducted to achieve a more precise assessment. Second, it was observed that the hit rate diminishes with the increase of k, suggesting that the majority of top predictions (e.g., k=1 or k=2) made by the PLMs are inaccurate. Finally, the weighted hit rates were higher than their corresponding normal hit rate, indicating that the incorrect predictions ranked in the top k accompany high confidence scores. All these findings can serve as compelling evidence substantiating the violation of the logical negation property.

3.3.1.2 Results for MWR

The results of the MWR task are summarised in the right column in Table 3.1. Interestingly, the three characteristics observed in the MKR-NQ task were also found in the MWR task. Moreover, two distinctive patterns were additionally observed in the MWR task.

PLMs lack knowledge of antonyms. It was observed that the hit rates and weighted hit rates of the MWR task are significantly higher compared to the MKR-NQ task across all three PLMs. During the analysis of predictions produced by the PLMs,

Model	Encoder-fixed		Fine-tune	
	$\mathcal{A}_{val}$	$\mathcal{A}_{test}$	$\mathcal{A}_{val}$	$\mathcal{A}_{test}$
BERT-*base* (108M)	53.1	55.0	84.0	85.6
BERT-*large* (333M)	54.4	53.5	92.1	92.5
RoBERTa-*base* (124M)	71.1	70.1	87.2	87.8
RoBERTa-*large* (355M)	69.7	69.1	93.7	94.2
ALBERT-*base* (11M)	56.6	58.1	81.5	84.0
ALBERT-*large* (17M)	54.7	56.6	86.9	88.0
ELECTRA-*small* (13M)	64.1	63.9	80.2	80.9
ELECTRA-*base* (109M)	67.9	70.6	93.3	92.9
ELECTRA-*large* (334M)	69.4	72.7	95.9	95.4

Table 3.3: Results of the SAR experiment. $\mathcal{A}_{val}$ and $\mathcal{A}_{test}$ are the accuracy of the *validation* and *test* dataset, respectively. The values are the average of five repetitions.

it was found that incorrect predictions were predominantly produced in antonym-asking queries. Specifically, the average HR@1 for the antonym-asking queries was 41.9%, whereas for the synonym-asking queries, it was only 1.4%. This significant difference can be attributed to the fact that PLMs tend to reproduce the target word presented in the input query. For example, the models regenerated the word "boy" for a query "boy is an antonym of [MASK]". Table 3.2 illustrates the proportion of instances in which each PLM replicated the target word. Although the proportions are considerably high for both synonym-asking and antonym-asking queries, the issue is more critical in the latter case, because the generated predictions correspond to wrong predictions. Based on the experimental results, it is inferred that the contextualised representations of PLMs lack lexical semantic information. The conclusion aligns with the findings of Liu et al. [2019a], which revealed that encoder-frozen PLMs are unsuitable for tasks that demand intricate linguistic knowledge.

The violation of the logical negation property is more severe with nouns. While analysing the results, it was found that the hit rates are higher when the POS tag of a target word in a query is a noun. The average HR@1 of nouns, adjectives, and adverbs are 35.1%, 27.4%, and 11.8%, respectively. It is interesting that PLMs exhibit a higher error rate when handling nouns despite being trained with a tremendous amount of written English corpus, where nouns comprise the largest portion (at least 37%) among all POS tags [Hudson, 1994, Liang and Liu, 2013]. This suggests that merely collecting additional training data cannot be the ultimate solution to resolve the violation of the logical negation property.

3.3.1.3 Results for SAR

As described in Section 3.2.1.3, the SAR task adopted a parametric probing model that keeps the encoder of PLMs fixed and only updates the classifier layer. Additionally, the entire set of parameters for each PLM was fine-tuned on the SAR task as part of the comparison. Table 3.3 presents the results of the SAR task. The results demonstrate a substantial gap between fine-tuned and encoder-fixed models' performance. While the fine-tuned models achieved a high accuracy (on average of 0.87 and 0.92 for the *base* and *large* models, respectively), the encoder-fixed models failed to meet expectations, even exhibiting almost random guessing performance in BERT models. Moreover, consistent with a widely accepted belief, large models demonstrated a significantly superior performance in comparison to their corresponding base-sized models. However, in the case of encoder-fixed models, the difference was statistically insignificant, except for the Electra models, which exhibited only a marginal improvement. These two phenomena indicate that the exceptional performance of PLMs heavily relies on updating a large number of parameters to learn syntactic associations present in training data [Niven and Kao, 2019, McCoy et al., 2019]. Still, their contextualised representations do not possess ample lexical semantic information, which leads to the violation of the logical negation property.

3.3.2 Experiments on Contradictory Explanation Generation

Two datasets were employed for conducting experiments on the generation of contradictory explanations: e-SNLI [Camburu et al., 2018] for the NLI task and the Cos-E 1.0 dataset [Rajani et al., 2019] for the commonsense QA task. The e-SNLI dataset aims to predict the relationship between a *premise* and *hypothesis* accompanied by NLE to support the answer. The instance in the Cos-E dataset consists of a *question*, three *answer candidates*, and a human-generated NLE regarding the correct answer. The Cos-E dataset aims to select an answer from the three candidates for a given question and to generate an NLE that provides justification for the selected answer. Following the work of Camburu et al. [2020], in the e-SNLI dataset, the *premise* and *hypothesis* were set as context and a variable part, respectively. In the case of Cos-E, the question and the correct answer were designated as the context, and the remaining two answer candidates were considered as the variable part. This design choice was made to prevent situations where the correct answer may be omitted when all three answer candidates are treated as the variable part.

Dataset	Method	Time	$\mathcal{S}_r$	$\mathcal{H}_r$
e-SNLI	eCA	10 days	2.19	384/24M
	eKnowCA	**40 min**	**12.88**	**1,494/88K**
Cos-E	eCA	2.5 days	0.32	5/5M
	eKnowCA	**5 min**	**0.95**	**13/11K**

Table 3.4: Comparison between eCA and eKnowCA on WT5-base. Success rate ($\mathcal{S}_r$), hit rate ($\mathcal{H}_r$), and the duration for each attack method are reported. The best results are in bold; $\mathcal{S}_r$ is given in %; $\mathcal{H}_r$ values are in fractions to emphasise the high denominators of eCA.

When it comes to the model candidates, the following high-performing NLE generation models were considered: NILE [Kumar and Talukdar, 2020] for the NLI task, CAGE [Rajani et al., 2019] for commonsense QA task, and WT5-base [Narang et al., 2020] (220M parameters) for both tasks. These models leverage widely-used generative PLMs as the explanation generator. Specifically, CAGE employs OpenAI GPT [Radford et al., 2018], NILE utilises GPT-2 [Radford et al., 2019], and WT5-base is based on T5-base [Raffel et al., 2020]. WT5 models with a larger number of parameters, such as WT5-11B, were excluded from this experiment due to the significant increase in computational resources but with only marginal improvements in the quality of NLEs. The WT5-base model was implemented based on the HuggingFace transformers package.[3] The replicated performance was close to the results reported by Narang et al. [2020], which can be confirmed in Appendix A.1. The implementations provided by the respective authors were used for the other models.

As depicted in Figure 3.3, training a reverse explainer, which utilises a context part and model-generated explanation as input, is necessary for the contradictory explanation attack process to generate contradictory variable parts. As a model structure, T5-base [Raffel et al., 2020] was adopted and trained for 30 epochs with a batch size of 8. Early stopping was applied for efficient training, which was stopped if the validation loss increased for 10 consecutive logging steps. The AdamW optimiser [Loshchilov and Hutter, 2019] was used with a learning rate of $1e^{-4}$. Gradient clipping to a maximum norm of 1.0 and a linear learning rate schedule decaying from $5e^{-5}$ were also applied. For Cos-E, 10% of the training data was used as the validation set for the early stopping strategy.

[3]https://github.com/huggingface/transformers

3.3.2.1 Comparison between eKnowCA and eCA.

To verify the effectiveness of eKnowCA over eCA, a performance comparison was conducted between the two approaches, where the results are demonstrated in Table 3.4. The comparison specifically focused on the WT5-base model, as the time required for employing the eCA method was prohibitive.

It was observed that the eCA approach generates an extensive number of contradictory candidates ($\mathcal{I}_e$), amounting to 24 million for the e-SNLI dataset and 5 million for the Cos-E dataset. Consequently, this significantly slowed down the attack process, requiring approximately 10 days and 2.5 days for the e-SNLI and Cos-E datasets, respectively, when using a single CPU. On the contrary, the eKnowCA approach proved to be much faster, requiring less than an hour to complete the process, by virtue of the significantly smaller size of $\mathcal{I}_e$, which amounted to less than 0.1 million. The reduced size of $\mathcal{I}_e$ also played a role in achieving a higher $\mathcal{H}_r$ by decreasing the denominator. Moreover, eKnowCA successfully attacked more test instances than eCA. Specifically, it achieved approximately six times higher in the e-SNLI dataset and three times higher in the Cos-E dataset. As a result, eKnowCA obtained superior $\mathcal{S}_r$ and $\mathcal{H}_r$ than eCA. The results demonstrate that eKnowCA is a faster and more efficient attack method than eCA.

3.3.2.2 Applying eKnowCA on NLE generation models

To evaluate the potential of generating contradictory NLEs by high-performing NLE generation models, the eKnowCA approach was applied to the aforementioned NLE generation models. However, the reverse explainer, responsible for generating the reverse variable part, was observed to contain the risk of producing unnatural statements [Camburu et al., 2020]. Therefore, the manual verification process performed by Camburu et al. [2020] was also conducted in this experiment, and unnatural reverse variable parts were not counted for the experiment. Detailed information regarding the naturalness evaluation is provided in Appendix A.2.

A human-evaluated e-ViL score [Kayser et al., 2021] was introduced to measure the quality of generated NLEs. To calculate the e-ViL score, a human annotator assesses the extent to which the generated NLEs support the model's decision using four response options: yes, weak yes, weak no, and no. Then, each response is subsequently converted to numeric values of 1, 1/3, 1/3, and 0, respectively. This metric is introduced as automatic evaluation metrics that heavily rely on syntactic

Model	e-SNLI				Cos-E			
	Acc.	$\mathcal{S}_r$	$\mathcal{H}_r$	e-ViL	Acc.	$\mathcal{S}_r$	$\mathcal{H}_r$	e-ViL
NILE	90.7	3.13	2.27	0.80	-	-	-	-
CAGE	-	-	-	-	61.4	0.42	0.06	0.43
WT5-base	90.6	12.88	1.70	0.76	65.1	0.95	0.12	0.55

Table 3.5: Results of the eKnowCA attack. Accuracy (Acc.), success rate ($\mathcal{S}_r$), hit rate ($\mathcal{H}_r$), and e-ViL score (e-ViL) are provided; $\mathcal{S}_r$ and $\mathcal{H}_r$ are given in % to improve the readability.

PREMISE: A man is riding his dirt bike through the air in the desert.	
HYPOTHESIS: A man is on a motorbike PREDICTED LABEL: entailment EXPLANATION: A dirt bike is a motorbike.	HYPOTHESIS: The man is riding a motorbike. PREDICTED LABEL: contradiction EXPLANATION: A dirt bike is not a motorbike.
QUESTION: John knew that the sun produced a massive amount of energy in two forms. If you were on the surface of the sun, what would kill you first?	
CHOICES: heat, light, life on earth PREDICTED LABEL: heat EXPLANATION: the sun produces heat and light.	CHOICES: heat, light, darkness PREDICTED LABEL: heat EXPLANATION: the sun produces heat and darkness.

Table 3.6: Examples of contradictory NLEs detected by eKnowCA for WT5 on e-SNLI and CAGE on Cos-E. The first column shows the original variable part, and the second column shows the adversarial one.

similarity and only weakly reflect human judgements. Details of the human evaluation process are described in Appendix A.3.

The experimental results of eKnowCA on NILE, CAGE, and WT5-base are presented in Table 3.5. It was found that all models are vulnerable to the contradictory attack. In the case of the e-SNLI dataset, the value of $\mathcal{S}_r$ is approximately 3% in the NILE model and exceeds 10% in the WT5-base model, despite models achieving an NLI accuracy of over 90%. The $\mathcal{H}_r$ of the NILE model is higher than that of the WT5-base model, suggesting that more contradictory NLEs per test instance were detected in the NILE model. Regarding the Cos-E dataset, both $\mathcal{S}_r$ and $\mathcal{H}_r$ are much lower than the results on the e-SNLI dataset. However, it is important to note that the variable parts of the Cos-E dataset were limited to only two answer candidates, significantly reducing the degrees of freedom compared to the e-SNLI dataset. This limitation restricts the diversity of generated statements and ultimately decreases the possibility of a successful attack. Achieving nearly 1% of $\mathcal{S}_r$ despite such restriction should not be overlooked. In addition, the results suggest that higher NLE quality

may only sometimes ensure fewer contradictions. For instance, although WT5-base achieved a better NLE quality than CAGE on the Cos-E dataset, more contradictions were detected by eKnowCA for WT5-base than for CAGE, considering $\mathcal{S}_r$. It was observed that the detected contradictory NLEs usually contradict commonsense knowledge, which is in line with previous research showing that LMs often suffer from factual hallucinations [Mielke et al., 2022, Zhang et al., 2021]. Examples of detected contradictory NLEs are provided in Table 3.6. More examples are available in Tables A.7–A.9 in Appendix A.4.

3.4 Methodologies to Alleviate Undesirable Behaviours

3.4.1 Intermediate Training on Meaning Matching Task: IM^2

The experiments conducted in Section 3.3.1 showed that PLMs lack information on negation words and lexical semantics, which leads to the violation of the logical negation property. A leading cause was hypothesised to lie in the training objective of PLMs: the language modelling objective, which serves as a fundamental pre-training task for nearly all modern PLMs.

The language modelling objective generates words based on the provided contexts. The fundamental assumption of the language modelling objective is the distributional hypothesis [Sinha et al., 2021a], which assumes that words with similar meanings or semantically related words will occur in similar contexts [Harris, 1954, Mrkšić et al., 2016]. Under this assumption, a model learns the semantic meaning of texts through their correlation with others, providing a great advantage in learning the meaning of texts solely from their textual form, enabling unsupervised and self-supervised training. Based on this benefit, many unsupervised text representations, such as Word2Vec [Mikolov et al., 2013] and Glove [Pennington et al., 2014], and contemporary PLMs have been developed.

However, despite its efficiency, the distributional hypothesis is limited in capturing a word's meaning, because words delivering different or even opposite meanings can appear in similar or identical contexts. Consider the words "boy" and "girl", for example. It is not difficult to imagine sentences in which the two words appear under the same context, e.g., "the little boy/girl cuddled the teddy bear closely". Although the distributional hypothesis facilitates capturing their shared functional meanings, such as "young human beings" by relating the two words with "little" and "teddy bear", it hardly allows capturing their distinctive meanings, such as gender. The same issue arises with words exhibiting semantic antonymy, such as "happy" and

"unhappy". This drawback also applies to negation expressions, as negated sentences have nearly identical contexts to their original forms. Consequently, LMs are unable to effectively have a comprehensive understanding of the meanings of words and negation expressions, provided they rely solely on the distributional hypothesis.

With the objective of better comprehending the meaning of the text and mitigating the violation of the logical negation property, I propose a methodology that incorporates the *meaning-text* theory as a supplementary hypothesis beyond the distributional hypothesis. The meaning-text theory assumes that there exists a correspondence between linguistic expressions (*text*) and semantic contents (*meaning*) [Mel'čuk and Žolkovskij, 1970, Milićević, 2006]. Based on the theory, I propose the new learning task called *meaning-matching* task, which facilitates direct learning of the correspondence. Specifically, meaning matching is a classification task where a word and a sentence are provided as inputs, and the objective is to ascertain whether the sentence accurately defines the word. I assumed that a model can learn meaning-text correspondences through this task.

When it comes to training PLMs on the meaning-matching task, the intermediate training technique [Phang et al., 2018, Wang et al., 2019a, Liu et al., 2019a, Pruksachatkun et al., 2020, Vu et al., 2020] was employed, which initially fine-tunes PLMs on an intermediate task, followed by a subsequent round of fine-tuning the model again on the desired target task. Recent findings have indicated that training PLMs on intermediate tasks, which require a high-level linguistic knowledge and inference ability, can enhance the performance of downstream tasks [Liu et al., 2019a, Pruksachatkun et al., 2020]. Moreover, this method offers a greater efficiency in terms of time and resource allocation compared to training models from scratch using large corpora, such as the BERTNOT model [Hosseini et al., 2021].

Dataset. For training, I collected a dataset comprising roughly 150,000 free-text definitions that depict the meaning of English words from the **WordNet** [Miller, 1995] and the **English Word, Meaning, and Usage Examples** dataset.[4] In cases where a word was present in both datasets, the definitions were combined through concatenation. Table A.3 in Appendix A.4 presents several examples of the data.

Training details. It is necessary to generate false word-definition pairs where the word and its corresponding definition do not align to fine-tune PLMs on the meaning-matching task. In order to achieve this purpose, a negative sampling technique was

[4]https://data.world/idrismunir/english-word-meaning-and-usage-examples/

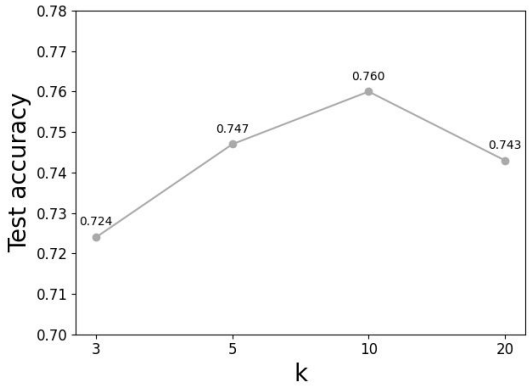


Figure 3.4: The test accuracy of the RoBERTa-*base* (IM^2) model with different k values, which denote the number of false word-definition pairs used for a negative sampling. The reported values are the average of five repetitions for each experiment.

employed that generates k false word-definition pairs for each word. The proper k was investigated in the range of 3, 5, 10, and 20. The performance of the RoBERTa-*base* model on the SAR task was used as a criterion for deciding the best k. Figure 3.4 shows the performance of the RoBERTa-*base* model on the SAR task with different k values. Intuitively, it seems that using a larger k value would enhance the model's performance, allowing the exploration of more word-meaning combinations. However, it was observed that the model achieves an optimal performance when k is set to 10, with a subsequent decline in performance with a larger k value. Consequently, I conjectured that too large k values may amplify the likelihood of the model mistakenly recognising the meanings of such similar words as different.

Generating k negative samples for each word may result in the class-imbalance issue, because the number of instances labelled as *false* (i.e., word and definition not aligned) becomes k times greater than that of examples labelled as *true*. Hence, to address this class-imbalance issue, the correct word-definition pairs were duplicated k times when constructing the training data to ensure a balance between labels. For training, 5% of data points were used for validation, and the early stopping strategy was used to prevent overfitting. The PLMs were fine-tuned for 15 epochs, and the AdamW optimiser [Loshchilov and Hutter, 2019] with a learning rate of $5e^{-6}$ and a weight decay rate of $1e^{-2}$ was employed.

3.4.2 KNOW-Method for Alleviating Contradictory NLEs

As stated in Section 3.3.2.2, the detected contradictory NLEs predominantly conflicted with commonsense knowledge. Therefore, I introduce an approach involving

the incorporation of relevant knowledge information, aiming to mitigate the generation of contradictory NLEs. Specifically, the process consists of two steps: (1) extraction of knowledge triplets related to the input statements and (2) knowledge injection.

Extracting related knowledge. To find out knowledge triplets relevant to given statements, a heuristic proposed by Xu et al. [2021] was employed, which extracts the related knowledge triplets from a given input statement as follows:

1. Extract entities from the given input statements.
2. Find all knowledge triplets that contain the entities.
3. For each entity, calculate a weight s_j for each extracted triplet as in Eq. 3.4:

$$s_j = w_j \times N/N_{r_j} \text{ and } N = \sum_{j=1}^{K} N_{r_j}, \tag{3.4}$$

   where w_j is the weight of the j-th triplet pre-defined by ConceptNet, N_{r_j} is the number of extracted triplets having the relation r_j, and K is the total number of triplets containing the entity. The intuition behind the weight for the triplets is to favour rarer relation types and higher importance scores, which are pre-defined by ConceptNet.
4. For each entity, extract the triplet with the highest score s_j.

Grounding with the extracted knowledge. After extracting the triplet with the highest weight for each entity within an instance, I transformed the extracted triplets into natural language form. This transformation was achieved by utilising templates used by Petroni et al. [2019], which convert a relation in knowledge triplet into its corresponding free-text form. For example, using the templates, the relation *IsA* would be transformed into the free-text "is a". Subsequently, the transformed free-text knowledge information was concatenated to the input statements with the separator "Context" between the input statements and the extracted knowledge information.

Model	Encoder-fixed		Fine-tune	
	$\Delta\mathcal{A}_{val}$	$\Delta\mathcal{A}_{test}$	$\Delta\mathcal{A}_{val}$	$\Delta\mathcal{A}_{test}$
BERT-*base* (108M)	+5.5*	+5.1*	+3.9*	+3.0*
BERT-*large* (333M)	+3.1*	+6.3*	+1.0	+0.2
RoBERTa-*base* (124M)	+4.5*	+5.9*	+1.3*	+1.6*
RoBERTa-*large* (355M)	+15.0*	+17.1*	+0.6	+0.5
ALBERT-*base* (11M)	-2.6	+2.5	+4.7*	+3.3*
ALBERT-*large* (17M)	+1.3	+1.4	+1.2	+1.6
ELECTRA-*small* (13M)	-4.1*	-2.7*	+1.1	+1.1
ELECTRA-*base* (109M)	+3.8*	+3.2*	-0.2	+0.7
ELECTRA-*large* (334M)	+14.0*	+10.2*	+0.4	+0.5

Table 3.7: The change of PLMs' accuracy in the SAR task when IM^2 is applied. The experiments were repeated 5 runs, and their average score is reported in the table. The proposed methods show a statistically significant difference using a two-sample t-test with p-value < 0.05 (*) compared to the baseline results in Table 3.3.

3.5 Experiments of the Alleviation Methods

3.5.1 Results of the IM^2 Method

This section presents evidence of the improved adherence to the logical negation property following the application of the IM^2 method. To sum up, the performance showed improvement across all three probing tasks. Furthermore, the proposed approach maintained the performance of various natural language understanding (NLU) downstream tasks and outperformed the BERTNOT model in the negation understanding task. These results indicate that meaning-matching serves as a safe intermediate task.

3.5.1.1 SAR Results

After the intermediate training, all models were fine-tuned on the SAR task under the same hyperparameter settings stated in Section 3.3.1. The experimental results are summarised in Table 3.7.

Improved lexical semantic information. The observation was made that, particularly for large-sized PLMs, there were marginal or no statistically significant improvements when fine-tuning the entire set of parameters after applying the IM^2 method. However, when considering the fixed encoder case, it was found that the performance of PLMs with over 100M parameters exhibited statistically significant enhancements, especially for large-sized PLMs. The experimental results indicate

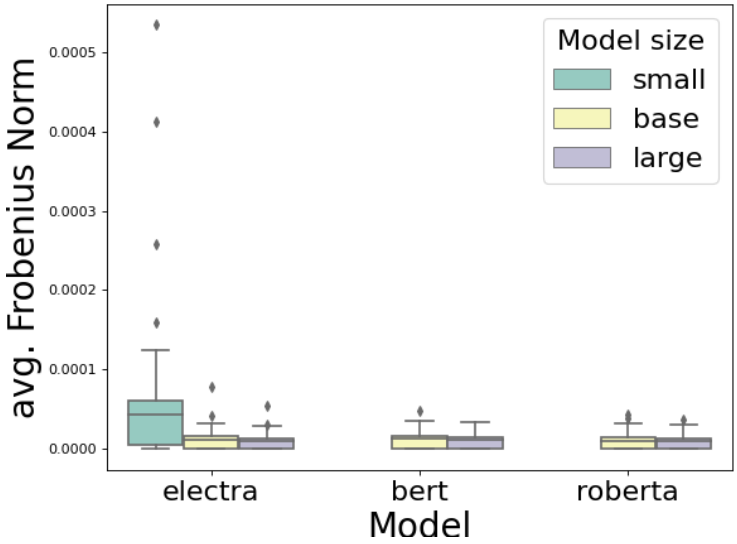


Figure 3.5: The box plots of the Frobenius norm of PLMs' layers after intermediate training on the meaning-matching task. The horizontal bars within each box denote, from bottom to top, the minimum value, the 25% quantile, the median, the 75% quantile, and the maximum value. Dots refer to outlier values.

that the proposed approach facilitates PLMs acquiring enriched representations encompassing a wider range of linguistic semantic information.

Catastrophic forgetting. The investigation revealed that small-sized PLMs, such as ELECTRA-*small* and ALBERT, generally demonstrated no significant performance improvement or even negative impacts. Given that all PLMs achieved a satisfactory accuracy on the meaning-matching task, a hypothesis was formulated attributing this phenomenon to the occurrence of the *catastrophic forgetting* issue [Pruksachatkun et al., 2020, Wallat et al., 2020], which refers to a model's tendency to forget previously acquired knowledge from pre-training to accept new information from the intermediate task. In order to verify this hypothesis, I measured the change in parameter values after applying the IM^2. Specifically, let M_i and M_i^{mm} denote the parameter of i-th layer before and after IM^2, respectively. The average of the Frobenius norm for each layer was calculated as follows (Eq. 3.5):

$$\mathcal{F}_i = \frac{1}{|M_i|}|M_i - M_i^{mm}|_F. \tag{3.5}$$

The boxplots of $\mathcal{F}_i$ for each PLM are presented in Figure 3.5. It was observed that the parameters of ELECTRA-*small*, which experienced a negative impact, were changed substantially compared to other PLMs having parameter sizes exceeding

Model	MKR-NQ					MWR				
	HR@1	HR@3	WHR@3	HR@5	WHR@5	HR@1	HR@3	WHR@3	HR@5	WHR@5
BERT	13.33	7.70	11.17	6.03	10.51	36.66	20.45	29.68	14.60	26.56
BERT (IM^2)	**11.41**	**7.01**	**9.86**	**5.57**	**9.14**	**18.92**	**13.07**	**15.78**	**10.30**	**14.14**
RoBERTa	15.72	9.25	13.31	6.86	12.24	27.92	17.07	23.06	12.84	20.82
RoBERTa (IM^2)	**6.56**	**4.97**	**6.05**	**3.99**	**5.67**	**22.08**	**12.68**	**18.94**	**9.20**	**17.63**

Table 3.8: The MKR-NQ and MWR results of BERT-*large* and RoBERTa-*large* after applying the IM^2 approach. Top-1,3,5 hit rates and weighted hit rates are provided. For each value, 100 is multiplied for better readability. Note that the lower the values, the better.

Query: demand is an antonym of [MASK]	
RoBERTa-large demand	RoBERTa-large (IM^2) supply
Query: tomorrow is the opposite of [MASK]	
BERT-large tomorrow	BERT-large (IM^2) today
Query: question is an antonym of [MASK]	
BERT-large question	BERT-large (IM^2) answer

Table 3.9: Examples of top-1 predictions on MWR queries. Unlike the original PLMs, models trained on the meaning-matching task did not reproduce a word in a query and make quite accurate predictions.

100M, which exhibited significant improvements in the SAR task when the encoder was kept fixed. The parameters of larger models with more than 100M parameters were barely changed, where the average value was statistically significantly lower than the ELECTRA-*small* model. The experimental results indicate that catastrophic forgetting is more likely to occur in small-sized models and is a prominent factor contributing to the adverse effects observed following the IM^2 method, implying that the size of PLMs plays a key role in preventing catastrophic forgetting issues.

3.5.1.2 MKR-NQ and MWR Results

For the experiments, the original PLMs' encoder was substituted with the encoder of the models that underwent fine-tuning on the meaning-matching task while reusing the MLM classifier. This is because the models were not trained with the MLM objective when applying the IM^2 method. BERT-*large* and RoBERTa-*large* were used

	COLA	MNLI	QNLI	RTE	QQP	MRPC	SST2
Batch-size	16	64	64	8	64	8	64
Learning rate	$2e^{-5}$	$1e^{-5}$	$1e^{-5}$	$2e^{-5}$	$1e^{-5}$	$1e^{-5}$	$1e^{-5}$

Table 3.10: Batch size and learning rates used for the GLUE benchmark experiments.

Model	COLA	MNLI-m	MNLI-mm	QNLI	RTE	QQP	MRPC	SST2
BERT	59.6±1.1	85.5±0.4	85.3±0.5	**91.7**±0.1	65.5±2.6	89.9±0.2	80.9±2.0	92.3±0.3
BERT (IM^2)	**61.5**±1.0	**85.7**±0.1	**85.5**±0.1	91.6±0.2	**66.8**±1.0	**90.0**±0.1	**82.8**±1.1	**92.4**±0.3
RoBERTa	62.9±1.9	90.2±0.1	**90.0**±0.2	**94.5**±0.1	81.7±1.8	90.9±0.4	87.2±1.1	**95.7**±0.1
RoBERTa (IM^2)	**64.8**±2.1	**90.3**±0.1	89.9±0.1	94.4±0.1	**83.1**±1.4	**91.0**±0.0	**88.2**±1.5	95.4±0.3
ELECTRA	68.4±2.3	**90.9**±0.1	90.7±0.2	**94.5**±0.3	86.9±2.2	91.6±0.5	88.9±1.5	**96.7**±0.1
ELECTRA (IM^2)	**69.1**±0.7	90.8±0.1	**90.7**±0.1	94.3±0.2	**87.0**±1.3	**91.7**±0.3	**89.5**±0.5	96.4±0.4

Table 3.11: GLUE benchmark validation performance of PLMs before and after intermediate training on the meaning-matching task. Matthew's correlation for the COLA and accuracy for the other tasks are used as an evaluation metric. The mean and standard deviation across 5 runs were reported. The best values for each PLM are in bold.

for the experiment for two reasons: (1) unlike ELECTRA models, they are pre-trained based on the MLM objective, and (2) parameter values were hardly changed after applying the IM^2 method, as illustrated in Figure 3.5. The results are summarised in Table 3.8.

Interestingly, substantial decreases in the hit rates of incorrect predictions were observed in both PLMs. When it comes to the MWR task, the application of the IM^2 method greatly mitigated the issue of generating a target word within a given query. Specifically, the portion of such instances dropped from 40.3% to 19.6% and from 33.8% to 25.2% for BERT-*large* and RoBERTa-*large*, respectively. Table 3.9 provides several examples of the predicted results. The results provide empirical support for the claim that the IM^2 method offers advantages in learning lexical semantic information and comprehension of negated expressions.

3.5.1.3 Fine-Tuning on GLUE Benchmark

A significant limitation of intermediate training is the potential negative impact on the target performance when the intermediate task lacks relevance to the target task [Liu et al., 2019a, Pruksachatkun et al., 2020]. Hence, the performance of BERT-*large*, RoBERTa-*large*, and ELECTRA-*large* on 7 GLUE benchmark datasets [Wang et al., 2018] were compared with their IM^2 counterparts to ascertain whether this issue

Model	SNLI		MNLI	
	dev	w/neg	dev	w/neg
BERTNOT	89.0±0.1	46.0±0.4	**84.3**±2.3	60.9±0.3
BERT-IM^2	**90.3**±0.2	**48.00**±0.5	83.1±0.3	**61.8**±0.6

Table 3.12: Accuracies on the original development dataset (dev) and the NegNLI (w/neg) dataset for the SNLI and MNLI tasks. The results of our approach are averaged across 5 runs. The best values are in bold.

occurs. For each dataset, the models were trained for 10 epochs, and the early stopping strategy was employed, with a patience number of 3. The batch size per GPU and learning rates used for each downstream task are provided in Table 3.10. It was found that downstream tasks with large training sets, such as MNLI, QNLI and QQP, were not sensitive to the hyperparameter settings.

Table 3.11 summarises the experimental results. No substantial performance disparity was observed for tasks with large training sets, such as MNLI, QNLI, QQP, and SST2. Conversely, a slight performance improvement was noted for MRPC and RTE, which have much smaller training sets. The obtained results were consistent with previous studies [Pruksachatkun et al., 2020, Vu et al., 2020], demonstrating that downstream tasks with small training sets benefit more from intermediate training. Furthermore, contrary to the prior research that reported a negative transfer on the COLA task [Phang et al., 2018, Pruksachatkun et al., 2020], the IM^2 approach demonstrated improved performance. The results indicate that meaning-matching serves as a safe and reliable intermediate task, promoting positive transfer across various NLU downstream tasks.

3.5.1.4 Experiments on the NegNLI Dataset

Lastly, experiments were conducted on the NegNLI dataset [Hossain et al., 2020], a task in which negation holds significant importance for the NLI task, to compare the performance between the IM^2 method and BERTNOT model [Hosseini et al., 2021]. The latter is a PLM implemented to improve PLMs' ability to comprehend negation by incorporating unlikelihood training technique [Welleck et al., 2020] with additional negation data. As Hosseini et al. [2021] leveraged BERT-*base* as a backbone structure, the IM^2 method was also applied to BERT-*base* for a fair comparison.

The results are summarised in Table 3.12. For both SNLI and MNLI tasks, we observed that the IM^2 method significantly outperforms BERTNOT in the NegNLI

Model	e-SNLI				Cos-E			
	Acc.	$\mathcal{S}_r$	$\mathcal{H}_r$	e-ViL	Acc.	$\mathcal{S}_r$	$\mathcal{H}_r$	e-ViL
NILE	90.7	3.13	2.27	0.80	-	-	-	-
KnowNILE	**90.9**	**2.42**†	**1.99**†	**0.82**	-	-	-	-
CAGE	-	-	-	-	61.4	0.42	0.06	0.43
KnowCAGE	-	-	-	-	**62.6**	**0.11**†	**0.01**†	**0.44**
WT5-base	90.6	12.88	1.70	0.76	65.1	0.95	0.12	0.55
KnowWT5-base	**90.9**	**11.45**	**1.19**†	**0.80**†	**65.5**	**0.84**†	**0.09**†	**0.56**

Table 3.13: Results of applying eKnowCA and Know-method for mitigating contradictory NLEs. The experiments were conducted 5 times, and their average score is reported in the table. The best results for each pair of (model, Know-model) are in bold; $\mathcal{S}_r$ and $\mathcal{H}_r$ are given in %; † indicates that Know-models showed statistically significant difference with p-value < 0.05 (†) using a two-sample t-test.

dataset while achieving comparable results in the original development sets. Interestingly, the IM^2 method achieved an improved understanding of negation expressions in both MKR-NQ and NegNLI tasks, even though it does not directly leverage negated datasets. I conjectured that a leading cause is the presence of negation expressions within the definitions of the meaning-matching dataset, which allows the proposed meaning to be captured by a model. The experimental results suggest that IM^2 is more efficient than the BERTNOT model, as it requires less time and resources for training compared to the latter, which involves pre-training PLMs from scratch with an expanded training objective.

3.5.2 Results of KNOW-Method

In order to validate the effectiveness of incorporating related knowledge information in mitigating the generation of contradictory NLEs, the Know approach was applied to the NILE, CAGE, and WT5-base models, referred to as KnowNILE, KnowCAGE, and KnowWT5-base, respectively.

The findings presented in Table 3.13 provide empirical evidence that incorporating commonsense knowledge reduces the occurrence of contradictory NLEs across all models and tasks. The Know-models successfully defended against 58% of the examples attacked by eKnowCA while being unsuccessful in the remaining 42%. The successfully defended examples imply that the eKnowCA method did not discover an adversarial instance with the Know-models, whereas the attack did find at least one such adversarial instance for the original model. Also, among the contradictory examples identified in the Know-models, it was observed that an average of 20% of them constituted newly introduced instances. Examples that were not successfully

		BLEU	R-1	R-2	R-L	Meteor	BERT-S
e-SNLI	WT5-base	28.4	45.8	22.5	40.6	33.7	89.8
	KnowWT5-base	**30.6**	**48.2**	**24.6**	**43.0**	**38.0**	**90.5**
	NILE	22.3	41.7	18.7	36.3	30.2	90.0
	KnowNILE	**22.4**	**42.0**	**18.9**	**36.5**	**30.5**	**90.1**
Cos-E	WT5-base	7.3	25.0	8.3	21.6	20.2	86.3
	KnowWT5-base	**7.9**	**26.7**	**9.6**	**22.9**	**21.8**	**86.7**
	CAGE	3.0	9.7	**1.1**	9.0	6.3	85.1
	KnowCAGE	3.0	**9.8**	1.0	9.0	**6.4**	85.1

Table 3.14: Automatic evaluation results on generated NLEs on the e-SNLI and Cos-E datasets. R-1, R-2, R-L, and BERT-S denote the ROUGE-1, ROUGE-2, ROUGE-L score, and BERT-Score, respectively. The best results are in bold.

defended, as well as newly introduced contradictory NLEs, are given in Tables A.10-A.11 in Appendix A.4. The examples of successfully defended cases are provided in Table 3.15, and more examples are in Tables A.12-A.13 in Appendix A.4.

Based on the analysis of the obtained results, it was found that in cases where the KNOW-models were unsuccessful in defending against the eKnowCA attack, the extracted knowledge triplets were found to be unrelated to generating decent NLEs, while the majority of successfully defended examples involved knowledge triplets that were directly associated with generating correct NLEs. The finding suggests that employing a more enhanced knowledge search algorithm can further prevent the generation of contradictory NLEs. The analysis also revealed that even if the extracted knowledge may be partially related to correctly generating NLEs, the model can still benefit from it. For example, in the first sample in Table A.13 in Appendix A.4, the most suitable knowledge triplet would be {dog, DistinctFrom, bird}. However, despite the given indirect knowledge,[5] i.e., {dog, DistinctFrom, cat}, the model can successfully counter contradictory NLEs with the information that dogs are distinct from other animals.

An argument could be made that the increased correct NLEs observed in KNOW-models may result from *knowledge leakage*, which refers to utilising the same knowledge triplets in both the KNOW mitigation method and the eKnowCA attack. However, it was found that for the e-SNLI dataset, a mere 0.3% of the knowledge triplets used for training KNOW-models were reused for the eKnowCA attack. In the case of the Cos-E dataset, no overlap was detected, suggesting that the leakage was not a prevailing issue.

[5]ConceptNet does not contain {dog, DistinctFrom, bird}.

PREMISE: A man is riding his dirt bike through the air in the desert.	
HYPOTHESIS: A man is on a motorbike PREDICTED LABEL: entailment EXTRACTED KNOWLEDGE: {dirt bike, IsA, motorcycle}, {desert, MannerOf, leave}, {air, HasA, oxygen} EXPLANATION: A dirt bike is a motorbike.	HYPOTHESIS: The man is riding a motorbike. PREDICTED LABEL: entailment EXTRACTED KNOWLEDGE: {dirt bike, IsA, motorcycle}, {desert, MannerOf, leave}, {air, HasA, oxygen} EXPLANATION: A dirt bike is a motorbike.
QUESTION: John knew that the sun produced a massive amount of energy in two forms. If you were on the surface of the sun, what would kill you first?	
CHOICES: heat, light, life on earth PREDICTED LABEL: heat EXTRACTED KNOWLEDGE: {light, IsA, energy} ,{heat, IsA, energy} EXPLANATION: light and heat are two forms of energy.	CHOICES: heat, light, darkness PREDICTED LABEL: heat EXTRACTED KNOWLEDGE: {light, Antonym, dark}, {heat, IsA, energy} EXPLANATION: the sun produces heat and light.

Table 3.15: Examples of successfully defended instances by KnowWT5 on e-SNLI and KnowCAGE on Cos-E. This table should be read with Table 3.6 to appreciate the defense.

Table 3.13 illustrates that the KNOW-models exhibit comparable e-Vil scores to their original counterparts. This finding is consistent with the results obtained from evaluating NLEs on automatic metrics in Table 3.14. These results indicate that the KNOW-method maintains NLE quality while decreasing contradictions.

3.6 Related Work

Negation understanding of PLMs. In addition to many analyses investigating PLMs' understanding ability of natural language, Ettinger [2020] and Kassner and Schütze [2020] revealed that PLMs often fail to comprehend *negation*, a crucial language property in many NLU tasks. Ettinger [2020] assessed PLMs' capability of comprehending the meaning of negation in given contexts. They evaluated the sensitivity of PLMs in completing sentences that either include *negation* or not. Under normal circumstances, it was anticipated that the completions would exhibit variability depending on the presence or absence of negation within the provided sentences. However, their findings revealed that PLMs are insensitive to the influence of negations when completing sentences. Kassner and Schütze [2020] constructed the negated LAMA dataset by inserting negation expressions (e.g., "not") in the LAMA cloze questions [Petroni et al., 2019]. They leveraged the pairs of negated and original questions to query PLMs and ascertained that PLMs exhibit an equal tendency

to generate identical predictions for both the original and negated questions. This observation revealed that PLMs encounter challenges in comprehending negation.

In light of PLMs' struggle to comprehend negation, Hosseini et al. [2021] proposed a remedy to mitigate the issue. In their methodology, they trained PLMs from scratch by augmenting the language modelling objective with an unlikelihood objective [Welleck et al., 2020] based on negated sentences from the training corpus. A syntactic augmentation method was used to create negated sentences. This method utilises the dependency parse of sentences (i.e., POS tags) and morphological information of each word as input. The negation of sentences was accomplished by employing sets of dependency tree regular expression patterns, such as Semgrex [Chambers et al., 2007]. During the training, Hosseini et al. [2021] substituted objects within the negated sentences with [MASK] tokens and employed unlikelihood training to reduce the likelihood of the masked-out tokens under the PLM distribution. To guarantee the factual inaccuracy of the negated sentences, they leveraged the corresponding positive sentences as contextual information for the unlikelihood prediction task.

The studies above primarily confined the scope of the logical negation property to functional negation expressions, such as "no" and "not". Although Welleck et al. [2020] incorporated the use of dependency tree regular expression patterns to negate sentences, thus expanding the scope of negation, because the patterns are not restricted to several negation expressions, the investigation still remained primarily focused on the realm of functional words. However, it is important to recognise that the core spirit of the logical negation property is the opposite meaning, which is not limited to negation. The experiments conducted in this chapter successfully extended the investigation scope from functional words to lexical semantics.

NLE generation models and their evaluation. A growing number of works focused on building NLE generation models in different areas such as NLI [Camburu et al., 2018, Kumar and Talukdar, 2020], QA [Rajani et al., 2019, Narang et al., 2020], visual-textual reasoning [Hendricks et al., 2018, Kayser et al., 2021, Majumder et al., 2022], medical imaging [Kayser et al., 2022], self-driving cars [Kim et al., 2018], and offensiveness classification [Sap et al., 2020]. The assessment of these models' performance was commonly centred on measuring the similarity between the generated NLEs and the gold standard NLEs created by humans using automatic evaluation metrics, such as BLEU and ROUGE, or the plausibility of the explanations through human evaluation.

Several studies attempted to evaluate the quality of the explanations from other perspectives. Based on all available information, the work of Camburu et al. [2020] stands as the only study that endeavoured to investigate contradictions in NLEs. Besides contradictory NLEs, other studies assessed the NLEs with regard to faithfulness, which pertains to how effectively the explanation elucidates the model's decision-making process [Wiegreffe et al., 2021, Atanasova et al., 2023] and simulatability [Hase et al., 2020].

3.7 Discussion

This chapter examined the trustworthiness of PLMs in terms of certification and explanation, which are the concepts that constitute the definition of trustworthiness [Huang et al., 2020]. The investigation from a certification perspective centred on ascertaining the correct behaviour of PLMs. Specifically, the exploration was conducted concerning the logical negation property by evaluating their ability to comprehend negations and semantic antonymy through three probing tasks: MKR-NQ, MWR, and SAR. With regards to the exploration from an explanation perspective, this chapter concentrated on the generation of contradictory NLEs by proposing an adversarial explanation attack method and assessing PLM-based high-performing NLE generation models using the proposed attack framework. The investigation revealed that PLMs are prone to violating the property of logical negation and are susceptible to attacks that aim to make the model generate contradictory explanations. These findings indicate that PLMs exhibit undesirable behaviours in terms of both certification and explanation perspectives and, therefore, lack trustworthiness.

The limitation of the proposed probing tasks lies in the fact that the test instances are predominantly generated by specific rules, which restricts the scope of the investigation. However, it is believed that modern LLMs can effectively address this issue. These models can be directly utilised to construct a greater variety of test instances through methods such as paraphrasing [Fang et al., 2023, Cegin et al., 2023], synthetic data augmentation [Patel et al., 2024], and in-context learning. For example, a model can be instructed with a command like "Rephrase the given sentence to convey an opposite meaning:". Although a certain degree of human inspection may still be necessary, these approaches can significantly enhance the diversity of test instances, thereby enabling a more systematic evaluation of the proposed probing tasks.

Subsequently, this chapter proposed two remedy methods to alleviate the undesirable behaviours exhibited by PLMs. To enhance the preservation of the logical

negation property, I proposed a novel intermediate task called "meaning-matching" inspired by the meaning-text theory. The task aims to overcome the limitation of the distributional hypothesis in capturing the precise meaning of words. The experimental findings substantiated that meaning-matching is a safe intermediate task that maintains the performance of various NLU downstream tasks while mitigating the violation of the logical negation property. In alleviating the generation of contradictory NLEs, a defense mechanism incorporating relevant background knowledge was introduced. The findings showed that the proposed method effectively reduces the occurrence of contradictory NLEs when appropriate knowledge triplets are extracted. However, in cases where the extracted knowledge is completely unrelated to generating correct NLEs, the defense mechanism failed to defend against producing contradictory NLEs. The finding suggests that employing a more accurate knowledge search algorithm can resolve the issue of generating contradictory NLEs. Given that the problem in certification is inherent to PLMs, whereas the contradictions in explanation can be addressed by introducing external components, the subsequent chapters of this dissertation will primarily focus on the issues that occurred in the certification process.

Chapter 4

Benchmark for Consistency Evaluation of Language Models

This chapter will explore the inconsistencies exhibited by LMs. It begins by proposing a definition of LMs' consistency based on the idea of *behavioural consistency*. Additionally, a taxonomy will be established to comprehensively categorise various consistency types explored in previous studies. Next, a new benchmark dataset will be introduced, allowing an evaluation across 19 distinct test cases encompassing multiple types of consistencies and downstream tasks. Finally, extensive experiments and analyses will be conducted to assess various LMs, including PLMs and LLMs, using the newly proposed benchmark suite.

4.1 Introduction

The emulation of human-like behaviour is an important characteristic that can enhance users' trust in an AI agent [De Visser et al., 2016, Jung et al., 2019], as it improves the certification process, which validates the correct behaviour of a system [Huang et al., 2020].[1] Accordingly, despite the remarkable performance of transformer-based PLMs on NLU benchmark datasets, concerns have been raised regarding their trustworthiness, stemming from their non-human-like behaviours, such as insensitivity to sentence order [Pham et al., 2021, Gupta et al., 2021, Sinha et al., 2021b] and inadequate understanding of negation expressions [Hossain et al., 2020, Kassner and Schütze, 2020, Ettinger, 2020, Hosseini et al., 2021]. In this respect, the characteristic of *behavioural consistency*, one of the core properties of humans, plays a crucial role in determining the trustworthiness of an LM.

[1] Trustworthiness = {Certification, Explanation} [Huang et al., 2020].

In response to the significance of consistency, its concept has been extensively discussed within the field of NLP. Nevertheless, despite its notable importance, a consensus regarding the precise and unified definition of consistency remains elusive. Below are the examples of distinct consistency definitions across many studies:

- Making consistent decisions in semantically equivalent contexts [Elazar et al., 2021].
- Being consistent on a system's beliefs across various inputs [Li et al., 2019].
- Producing logically or factually accurate statements [Li et al., 2020b].

The presence of multiple definitions has led to constrained investigations that are limited to specific types of consistency and certain downstream tasks, predominantly focusing on NLI and QA. These fragmented explorations pose the risk of reaching biased conclusions due to the restricted scope of the investigation.

To this end, this chapter presents a definition of the consistency of an LM inspired by the concept of *behavioural consistency*. Subsequently, a taxonomy of LM's consistency is established by categorising prior studies according to the proposed definition. Following the definition and taxonomy of LM's consistency, I contruct a new benchmark suite named **be**nchmark for **c**onsistency **e**valuation of **l**anguage models (**BECEL**), which is a unified evaluation dataset for assessing LM's consistency across multiple consistency types and six distinct NLP tasks: NLI, STS, words-in-context (WiC), semantic analysis (SA), machine reading comprehension (MRC), and topic classification (TC). Finally, extensive experiments and thorough analyses are performed on the new benchmark suite to assess the consistent behaviour of LMs, including widely used PLMs and LLMs like ChatGPT. The experimental results reveal that none of the LMs coherently exhibits consistent behaviours across all test cases.

The main contributions of this chapter can be summarised as follows:

- This chapter proposes the concept of LM's consistency and its taxonomy that categorises multiple definitions of consistency in previous studies into three exclusive categories.
- This chapter introduces a novel benchmark suite named **BECEL** that enables an analysis across multiple types of consistencies and downstream tasks.
- Thorough experiments and analyses reveal that none of PLMs and even LLMs coherently show consistent behaviours across all test cases.

4.2 Language Models' Consistency

Behavioural consistency implies being consistent in behavioural patterns by adhering to the same principles.[2] In accordance with this concept, I defined the *consistency of an LM* as follows:

Definition: The consistency of an LM is its capacity to make a coherent decision that is not contradictory to its underlying belief.

This definition of consistency comprises two fundamental components. The first one is *belief*, which implies something a model considers true. The second component is *principle*, the property that is used to determine what a coherent decision is. Based on these two components, various types of consistency addressed in the previous literature are classified into three large categories: semantic, logical, and factual consistency.

4.2.1 Semantic Consistency

The fundamental concept of meaning-text theory assumes that the correspondence between linguistic expressions (*text*) and semantic contents (*meaning*) is characterised as *many-to-many*, implying that the meaning can be conveyed through various textual forms [Mel'čuk and Žolkovskij, 1970, Milićević, 2006]. Within this context, it is expected that a model possessing a heightened level of NLU proficiency will effectively grasp the core meaning and consistently arrive at identical decisions when encountered with semantically equivalent text, as the definition of "language understanding" is "to focus on the meaning and not the text" [Krashen, 1982]. Hence, the belief and principle of semantic consistency become a *model's predictions on semantically identical texts* and *semantic equivalence*, respectively. In this regard, the concept of *semantic consistency of an LM* can be defined as its ability to make identical decisions on semantically equivalent texts.

Semantic consistency is an essential and indispensable attribute of LMs, irrespective of the tasks and data, as it arises from the semantic meaning, a universal language characteristic. It is the most extensively employed concept in numerous studies that addressed the consistency of LMs. Research on text adversarial attacks revealed that many PLMs are vulnerable to adversarial samples, which are crafted to deliver a similar meaning but have different textual forms compared to their original counterparts [Jin et al., 2020, Garg and Ramakrishnan, 2020, Li et al., 2020a, Ivgi and

[2]N., Sam M.S., Behavioral Consistency. PsychologyDictionary.org, April 7, 2013, [link].

Berant, 2021, Li et al., 2021]. Other works ascertained a disparity in the MLM predictions of PLMs when encountering queries involving the substitution of an object with its plural from [Ravichander et al., 2020] and paraphrased queries [Elazar et al., 2021]. Raj et al. [2022] proposed an improved framework that enables the semantic consistency evaluation of natural language generation (NLG) outputs by introducing agreement functions that measure the semantic similarity of two generated NLG outputs. Moreover, another line of work adopted the concept of semantic consistency to consistency regularisation to train LMs with improved inductive bias [Wang and Henao, 2021, Zheng et al., 2021, Kim et al., 2021].

4.2.2 Logical Consistency

Many NLP tasks require adherence to specific logical properties, rendering predictions that violate these properties invalid. In this case, the belief and principle become a *model's predictions regarding instances where the logical property holds* and a *logical property*, respectively. Hence, the concept of *an LM's logical consistency* can be defined as its ability to make decisions without logical contradiction.

The concept of logical consistency can be further categorised based on the required logical properties. This chapter delivers four types of logical consistencies: *negational*, *symmetric*, *transitive*, and *additive* consistency. However, more sub-categories of logical consistency can be defined by employing alternative logical properties.

Negational consistency. The core property of negational consistency is the logical negation property (p is true $\Leftrightarrow \neg p$ is false; Aina et al. 2018). In other words, for tasks where the property is held, the predictions of an LM should differ for texts delivering the opposite meanings. The preceding studies outlined in Section 3.6 that investigated the comprehension of negation expressions in PLMs pertain to the body of research that explores negational consistency.

Symmetric consistency. Assuming a function f that takes two variables, a symmetric inference is formally defined as follows: $f(x, y) = f(y, x)$. Intuitively, this implies that an LM's prediction should remain unchanged when the input texts are swapped for NLP tasks where the property is held. Previous studies regarding symmetric consistency are mainly focused on the NLI task. Wang et al. [2019c] proposed that symmetric consistency applies to instances having *contradiction* and *neutral* as labels and examined the variation of accuracy after switching the order of premise and hypothesis. In contrast, Li et al. [2019] suggested that the property holds exclusively

when the label corresponds to a *contradiction* and explored the symmetric consistency of NLI models using a newly collected dataset derived from MS-COCO [Lin et al., 2014]. More recently, Kumar and Joshi [2022] extended the investigation scope from NLI to STS and measured the symmetric consistency more conservatively by additionally employing the difference in prediction confidence score. All these works ascertained that PLMs easily violate the symmetric consistency.

Transitive consistency. Considering the three predicates X, Y, and Z, transitive inference is formally expressed as follows: if $X \rightarrow Y \wedge Y \rightarrow Z$, then $X \rightarrow Z$ [Gazes et al., 2012, Asai and Hajishirzi, 2020]. Li et al. [2019] applied this property to NLI tasks by defining four transitive inference rules. Specifically, for the three related sentences P, H, and Z, the rules are defined as below (Eqs 4.1 – 4.4):

$$E(P,H) \wedge E(H,Z) \rightarrow E(P,Z), \tag{4.1}$$
$$E(P,H) \wedge C(H,Z) \rightarrow C(P,Z), \tag{4.2}$$
$$N(P,H) \wedge E(H,Z) \rightarrow \neg C(P,Z), \tag{4.3}$$
$$N(P,H) \wedge C(H,Z) \rightarrow \neg E(P,Z), \tag{4.4}$$

where E, N, and C refer to entailment, neutral, and contradiction, respectively. Based on these rules, Li et al. [2019] constructed a new evaluation set from MS-COCO dataset [Lin et al., 2014] to assess the transitive consistency of NLI models. In the domain of QA, Asai and Hajishirzi [2020] employed the transitive property to test QA models on new data, which is made by combining two questions (q_1, q_2) where the effect of q_1 corresponds to the cause of q_2. For example, consider the two questions and corresponding answers below:

Q1: If it is silent, does the outer ear collect fewer sound waves? A1: more

Q2: If the outer ear collects fewer sound waves, is less sound being detected? A2: more

The new question would be "If it is silent, is less sound being detected?", and the model should answer "more" if it preserves the transitive consistency. More recently, Lin and Ng [2022] investigated the transitive inference ability of PLMs on WordNet word senses and the **IS-A** relation, i.e., if A *is-a* B and B *is-a* C, then A *is-a* C. These works observed that PLMs do not completely adhere to the transitive consistency.

Additive consistency. In this dissertation, I introduce a novel type of logical consistency referred to as additive consistency, which has yet to be addressed in previous studies. For a function f, the additive inference is expressed as follows: $f(x) = f(y) = c \rightarrow f(x+y) = c$, where c is a predicted label. When it comes to NLP tasks, additive consistency applies to any single-sentence classification task (e.g., SA and TC). Intuitively, the property implies that if a model produces an identical prediction for distinct input sentences, the prediction for the combined sentence should also remain unchanged.

Specificity of logical consistency. Notably, unlike semantic consistency, logical consistency is contingent upon the specific task and data. In other words, logical consistency cannot be universally applied in cases where certain logical properties are invalid. For example, negational consistency does not apply to TC, because it is evident that negating the sentence below belonging to the "Sports" topic does not change its category.

Tottenham forward Son Heung-min has signed a new four-year contract.

4.2.3 Factual Consistency

The fundamental idea underlying factual consistency is that a model should produce outputs that align with factual accuracy. Hence, the belief is the *model's output*, and the principle becomes *factual correctness*, respectively. In this regard, the *factual consistency of an LM* can be defined as its ability to generate outputs that are not contradictory to the common facts and given context.

Due to its inherent nature of generating correct facts, factual consistency has exhibited a close relationship with *knowledge grounding* and *mitigating hallucinations*. Consequently, most research regarding factual consistency has primarily focused on NLG tasks, such as text summarisation [Kryscinski et al., 2020, Maynez et al., 2020, Wang et al., 2020, Pagnoni et al., 2021], generative open-domain QA [Lewis et al., 2020b, Izacard and Grave, 2021], and dialogue generation [Li et al., 2020b, Shuster et al., 2021, Komeili et al., 2022].

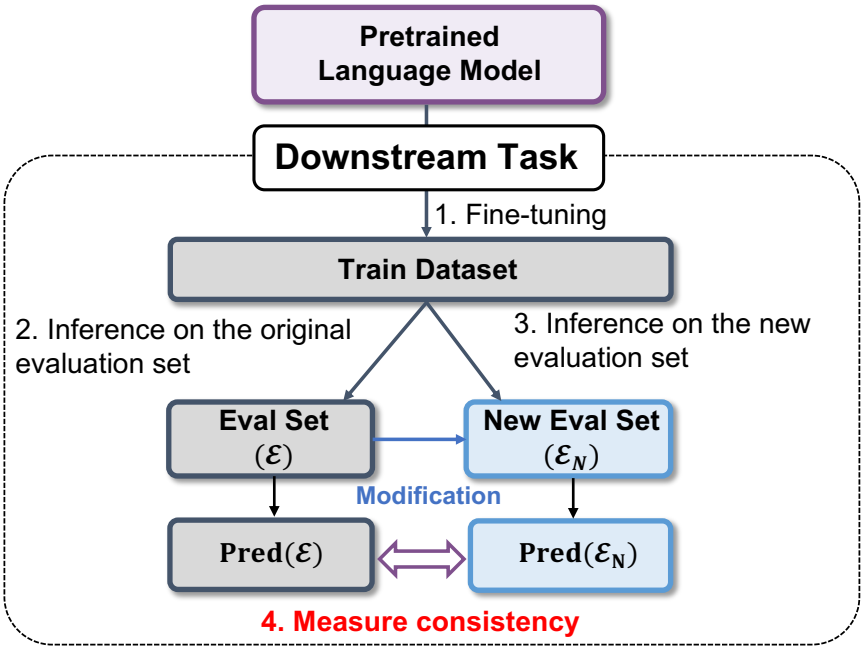


Figure 4.1: Overall evaluation framework for evaluating an LM's consistency. The comparison is conducted solely between models' predictions, not with gold labels.

4.3 BECEL Dataset

4.3.1 Overview

The BECEL dataset was designed to evaluate semantic consistency and four types of logical consistencies. Factual consistency was removed from the evaluation scope, as benchmark datasets and evaluation frameworks dedicated to assessing factual consistency were extensively studied across various downstream tasks, such as text summarisation [Kryscinski et al., 2020, Wang et al., 2020, Pagnoni et al., 2021], QA [Choi et al., 2018, Rajpurkar et al., 2018, Reddy et al., 2019], and dialogue generation [Dinan et al., 2019, Komeili et al., 2022].

Figure 4.1 illustrates the overall framework for evaluating an LM's consistency through the BECEL dataset. First, a model is fine-tuned on a downstream task. Subsequently, it proceeds to generate predictions for both the original evaluation set ($\mathcal{E}$) and a newly constructed evaluation set ($\mathcal{E}_N$), specially designed based on $\mathcal{E}$ to evaluate a certain type of consistency. Finally, the predictions on $\mathcal{E}$ and $\mathcal{E}_N$ are compared to measure the consistency.

The BECEL dataset encompasses $\mathcal{E}_N$ across multiple NLP downstream tasks for evaluating various consistency types. Specifically, it includes six downstream tasks: SNLI [Bowman et al., 2015] and RTE [Candela-Quinonero et al., 2006] for NLI,

	BoolQ	SNLI	RTE	MRPC	WiC	SST2	AG-News
semantic	1,076	4,406	248	202	140	187	540
negational	401	2,204	153	290	-	-	-
symmetric	-	3,237	1,241	3,668	5,428	-	-
transitive	-	2,375	-	-	3,162	-	-
additive	-	-	-	-	-	53K	53K

Table 4.1: The number of new test data instances for each downstream task and consistency type.

MRPC [Dolan and Brockett, 2005] for STS, BoolQ [Clark et al., 2019] for MRC, SST-2 [Socher et al., 2013] for SA, AG-News [Zhang et al., 2015] for TC, and WiC [Pilehvar and Camacho-Collados, 2019] for evaluating semantic consistency and four distinct logical consistencies. Brief descriptions of each downstream task are provided in Appendix B.1. Several data examples are presented in Figures B.2 and B.3 in Appendix B.5.

Table 4.1 demonstrates the size of the newly created 19 test sets for each task and consistency type. Test cases where a particular logical property cannot be applied to a specific task were excluded from the evaluation scope. The viability of each consistency type across diverse downstream tasks is demonstrated in Appendix B.3. The test sets of each dataset were utilised as the original evaluation sets ($\mathcal{E}$) for collecting the new evaluation sets ($\mathcal{E}_N$), given that gold labels were provided. If not, development sets (i.e., validation sets) were used instead of original evaluation sets ($\mathcal{E}$). However, if the following two conditions are satisfied, training sets were employed as the original evaluation sets ($\mathcal{E}$): (1) the development/test set sizes are extremely small, and (2) new evaluation data can be automatically gathered. Specifically, the RTE, MRPC, and WiC tasks for assessing symmetric and transitive consistencies fall within this case.

4.3.2 Data Collection Schema

This section provides comprehensive descriptions of methodologies used for constructing various consistency evaluation datasets.

Semantic consistency data. The new evaluation set ($\mathcal{E}_N$) for semantic consistency consists of paraphrased versions of the sentences in the original evaluation set ($\mathcal{E}$). For downstream tasks where the input consists of only one sentence (i.e., SA and TC), the sentence was paraphrased to construct $\mathcal{E}_N$. For other tasks where the input

	Fixed Variable	Modified Variable
BoolQ	passage	question
SNLI	premise	hypothesis
RTE	premise	hypothesis
MRPC	sentence1	sentence2
WiC	word, sentence1	sentence2
SST2	-	text
AG-news	-	text

Table 4.2: Modified variables of each dataset for constructing $\mathcal{E}_N$ of semantic and negational consistency evaluation sets.

consists of more than one sentence, only one text input was paraphrased. Table 4.2 illustrates each task's fixed and modified text inputs for constructing $\mathcal{E}_N$. In order to generate paraphrased sentences, the publicly available Quilbot[3] was employed due to its capability to generate more natural paraphrases and cover a wider range of linguistic variations compared to model-driven paraphrasing approaches, such as text adversarial attacks. For example, Quilbot is able to change the grammatical structure of a given sentence, such as converting passive voice to active voice, where text adversarial attack is incapable of. Subsequently, a human evaluation was conducted on the generated paraphrased sentences using Amazon MTurk to enhance the overall data quality. For each data instance, three annotators were assigned to score the textual similarity between the original and paraphrased sentences on a scale from 1 to 5. Only instances where the average similarity score is greater or equal to 4 were finally included in $\mathcal{E}_N$. More detailed information regarding the human annotation process is provided in Appendix B.2. In contrast to other downstream tasks, the WiC dataset presents a risk of target word omission when employing the paraphrasing tool. Consequently, new data points were eliminated if the target word did not appear in the paraphrased sentence.

Negational consistency data. To collect the new evaluation set ($\mathcal{E}_N$) for negational consistency assessment, the opposite-meaning sentences of the modified variables listed in Table 4.2 were produced by utilising two heuristics: *negation* and *antonym replacement*. Regarding the former, sentences containing a single verb were negated by inserting negation expressions like "not". For the latter, adjectives and adverbs were extracted from the sentences and substituted with their respective antonyms using ConceptNet [Speer et al., 2017]. Only one word was replaced at a

[3]https://quillbot.com/

time. Next, the same human annotation process, which was conducted during the collection of semantic consistency evaluation sets, was carried out. In this case, however, instances in which the average textual similarity score falls below 2 were included in $\mathcal{E}_N$. Finally, a manual review was conducted to remove ambiguous or grammatically incorrect data points.

As previously mentioned, negational consistency is a data-specific consistency type. In the SNLI dataset, the label changes from "entailment" to "contradiction" when the hypothesis is replaced with its opposite-meaning sentence. However, the assurance of label alteration does not hold uniformly for the other labels, particularly for instances labelled as "neutral". For this reason, only the instances with the "entailment" label were considered to construct $\mathcal{E}_N$ of the SNLI task. For the same reason, only the data instances with the label "entailment" for RTE, "equivalent" for MRPC, and "true" for BoolQ were utilised to construct $\mathcal{E}_N$.

Symmetric consistency data. The order of two text inputs was switched for the downstream tasks where the symmetric consistency is applicable. Regarding the WiC and MRPC tasks, the symmetric consistency is valid for every data instance. On the contrary, in the context of NLI tasks, it only pertains to instances with the "contradiction" label [Li et al., 2019] or the "neutral" label when the hypothesis is less specific than the premise [Wang et al., 2019c]. In the RTE task, it was ascertained that the premise is always more specific than the hypothesis, so instances with the "not_entailment" label were utilised to construct $\mathcal{E}_N$. In the SNLI task, however, it is not guaranteed. As a result, only the data points with the "contradiction" label were used.

Transitive consistency data. The $\mathcal{E}_N$ for transitive consistency evaluation was constructed on two downstream tasks: SNLI and WiC. For the SNLI task, it is a prerequisite that two data instances must possess an identical hypothesis to be eligible for the application of the transitive inference rules described in Section 4.2.2. However, it was verified that only a premise is shared in the SNLI dataset. As a result, the symmetric consistency, which applies to instances labelled as "contradiction", was utilised, allowing us to reformulate the rules Eqs 4.2 and 4.4 to Eqs 4.5 and 4.6, respectively, in the following manner:

$$E(P, H) \wedge C(P, Z) \rightarrow C(H, Z), \tag{4.5}$$

$$N(P, H) \wedge C(P, Z) \rightarrow \neg E(H, Z). \tag{4.6}$$

Rules 4.1 and 4.3 were deprecated, because symmetric consistency is not ensured for data instances with the "entailment" label. By using the modified rules, $\mathcal{E}_N$ of the SNLI task was collected automatically. However, it was noted that the modified rules did not hold for several data points, because the hypothesis is less specific than the premise for several instances in the SNLI dataset [Wang et al., 2019c]. Therefore, a human annotation process was conducted through Amazon MTurk to eliminate such instances. Each instance was assigned to three annotators, and it was included in $\mathcal{E}_N$ if at least two annotators agreed that the instance complied with the rules.

For the WiC task, given a target word w and three sentences A, B, and C, the following transitive rules (Eqs 4.7 – 4.9) are universally applicable:

$$T(A, B|w) \wedge T(B, C|w) \rightarrow T(A, C|w), \tag{4.7}$$

$$T(A, B|w) \wedge F(B, C|w) \rightarrow F(A, C|w), \tag{4.8}$$

$$F(A, B|w) \wedge T(B, C|w) \rightarrow F(A, C|w), \tag{4.9}$$

where T/F denotes that the meaning of the word w is used identically/differently in the two given sentences. These rules were used to automatically collect $\mathcal{E}_N$ of the WiC task.

Additive consistency data. The additive consistency holds for tasks that involve a single-text input. For each such task, all possible combinations of two data points that share the same label were generated, and a data instance was produced by combining their respective text inputs to construct $\mathcal{E}_N$ for each task. Any new data instance where the length of a merged text surpassed the 75% quantile of the token length observed in the training data was removed. This decision was made to avoid potential out-of-distribution instances that could exaggerate the inconsistency issues.

4.3.3 Inconsistency Evaluation Metrics

The inconsistency evaluation metrics are defined and utilised to assess the inconsistent behaviours of LMs. These metrics intuitively measure the proportion of test cases in which LMs produce an inconsistent prediction, i.e., an error rate. As the metrics indicate an error rate, they are scaled between 0 and 1.

Semantic/Symmetric consistency. Assume that $e_i \in \mathcal{E}$, $e_i^N \in \mathcal{E}_N$, and e_i^N is a perturbed version of e_i, and therefore, $|\mathcal{E}| = |\mathcal{E}_N|$. Provided a model M is able to make consistent decisions, it should generate the same predictions for e_i and e_i^N.

Hence, inspired by the idea of robust accuracy [Tsipras et al., 2019, Ivgi and Berant, 2021], I defined the inconsistency metric (τ) for semantic and symmetric consistency as follows (Eq. 4.10):

$$\tau = 1 - 1/|\mathcal{E}_N| \sum_{i=1}^{|\mathcal{E}_N|} \mathbb{1}(M(e_i) = M(e_i^N)). \tag{4.10}$$

Negational consistency. Let $e_i \in \mathcal{E}$, $e_i^N \in \mathcal{E}_N$, and e_i^N is a negated version of e_i (i.e., $|\mathcal{E}| = |\mathcal{E}_N|$). In contrast to semantic and symmetric consistency, model M is expected to generate different predictions for e_i and e_i^N, where $e_i^N \in \mathcal{E}_N$ represents a newly designed instance intended for assessing negational consistency. Therefore, I defined the inconsistency metric for negational consistency as follows (Eq. 4.11):

$$\tau = 1 - 1/|\mathcal{E}_N| \sum_{i=1}^{|\mathcal{E}_N|} \mathbb{1}(M(e_i) \neq M(e_i^N)). \tag{4.11}$$

Transitive/Additive consistency. For both transitive and additive consistency, a new instance e_i^N is derived from two data points of $\mathcal{E}$. Assume that $e_i^N \in \mathcal{E}_N$, and e_i^N originates from $e_{i,1}, e_{i,2} \in \mathcal{E}$. In the evaluation process, the inclusion of e_i^N in which the antecedent condition is not satisfied, meaning that model M makes incorrect predictions for $e_{i,1}$ or $e_{i,2}$, has a potential risk of overestimating the inconsistency problem. Hence, I introduce a conditional inconsistency metric for the transitive and additive consistency measurement as an evaluation metric (Eq. 4.12):

$$\tau = 1 - 1/|\mathcal{C}| \sum_{i=1}^{|\mathcal{C}|} \mathbb{1}(M(c_i) = l_i), \tag{4.12}$$

where l_i is the label of $c_i \in \mathcal{C}$, and $\mathcal{C} \subset \mathcal{E}_N$ denotes the set of e_i where model M makes correct predictions for both $e_{i,1}$ and $e_{i,2}$. Intuitively, the metric only considers new instances for assessment if model M generates accurate predictions for two instances from which the new instance originated.

Conditional consistency metric. It is noted that LMs generally produce an imperfect performance on downstream tasks, potentially leading to an incorrect estimation of conditional consistency types, such as negational and symmetric consistencies where the consistency applies to data instances having specific labels. For instance, consider the below example of the MRPC task:

S1: In the evening, he asked for six pepperoni pizzas and two six-packs of soft drinks, which officers delivered.

S2: In the evening, he asked for six pizzas and soda, which the police delivered.

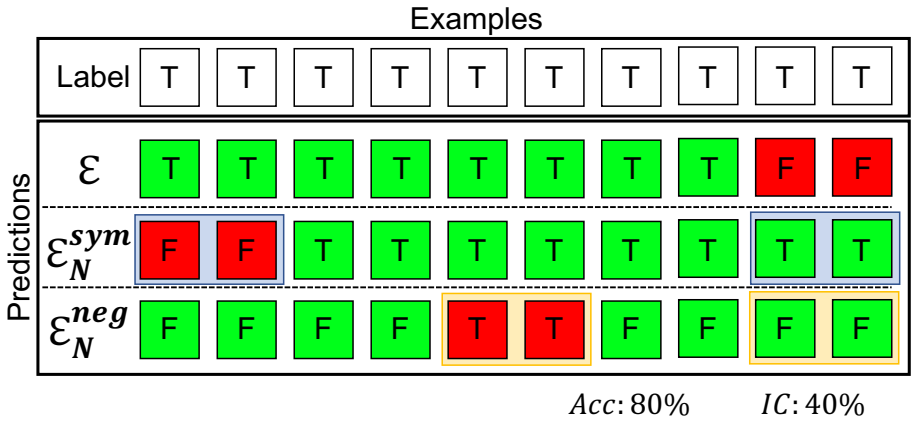


Figure 4.2: A graphical representation of accuracy, symmetric consistency, and negational consistency for binary classification. The blue and yellow boxes represent inconsistent symmetric and negational consistency cases, respectively.

S2-neg: In the evening, he asked for six pizzas and soda, which the police did not deliver

Given that the gold label of the **S1**-**S2** pair is "Equivalent", predicting the relation between **S1**-**S2-neg** as "Equivalent" violates the negational consistency. However, suppose the model deems the answer of the **S1**-**S2** pair as "Not Equivalent". In that case, generating "Equivalent" as an answer for the **S1**-**S2-neg** pair may not be regarded as violating the negational consistency, because the model's belief does not satisfy the condition of the negational consistency, i.e., the label should be "Equivalent". Incorporating such instances in calculating inconsistency metrics may lead to imprecise evaluations, particularly when the model's performance is not sufficiently satisfactory. In order to mitigate the risk of misestimating the consistency of LMs, I employed conditional consistency metrics eventually. These metrics only take into account data instances where LMs make correct predictions when calculating the inconsistency metric. Specifically, these metrics are applied to BoolQ, SNLI, RTE, and MRPC for the negational consistency evaluation and SNLI and RTE for the symmetric consistency evaluation.

Importance of Measuring Consistency Previous benchmark suites for assessing the model's understanding of the opposite meanings [Naik et al., 2018, Hossain et al., 2020] or symmetry property [Wang et al., 2019c] measured the accuracy of the new test suite but not the other metrics. Indeed, models with low accuracy on the new test suite are likely to exhibit inconsistency, but it is important to note that high accuracy

	BoolQ	SNLI	RTE	MRPC	WiC	SST2	AG-News
b-size	8	64	8	8	64	32	32
tok-len	512	128	256	128	128	128	256
lr	$2e^{-5}$	$1e^{-5}$	$1e^{-5}$	$2e^{-5}$	$1e^{-5}$	$1e^{-5}$	$1e^{-5}$

Table 4.3: Batch size (b-size), maximum token length (tok-len), and learning rates (lr) used for BECEL benchmark experiments.

does not necessarily guarantee high consistency. Figure 4.2 illustrates a representative example case well. Despite 80% in the original test set and two types of $\mathcal{E}_N$ for the symmetric and negational consistencies, which suggest a decent level of robustness on unseen data, the inconsistency remains at 40%. In this regard, consistency should be treated as an independent evaluation metric and be measured along with accuracy.

4.4 Experiments and Analysis on Pre-trained Language Models

4.4.1 Experimental Design

For the experiments, the BECEL benchmark suite was evaluated using widely-used PLMs, as described below. Both *base* and *large*-size models were used for each PLM.

- **Encoder models**: BERT [Devlin et al., 2019], RoBERTa [Liu et al., 2019b], ELECTRA [Clark et al., 2020], and ERNIE 2.0 [Sun et al., 2020].
- **Decoder models**: GPT2 [Radford et al., 2019].
- **Encoder-Decoder models**: BART [Lewis et al., 2020a] and T5 [Raffel et al., 2020].

During fine-tuning, the AdamW optimiser [Loshchilov and Hutter, 2019] and a linear learning rate scheduler decaying from $1e$-3 were employed. All models were trained for 10 epochs, and the early stopping strategy was adopted during the training to avoid overfitting. Table 4.3 shows each downstream task's batch size per GPU, maximum number of tokens, and learning rates for training PLMs. Consistent with prior research, it was ascertained that downstream tasks with substantial training data (e.g., SNLI, SST2, and AG-news) exhibited insensitivity to hyperparameter values.

In the case of the T5 model, the text-to-text multitask training strategy was applied using the free-text input format proposed by Raffel et al. [2020], as multitask

Model		BoolQ		MRPC		RTE		SNLI		SST2		WiC	AG-News	
		τ_{sem}	τ_{neg}	τ_{sem}	τ_{neg}	τ_{sem}	τ_{neg}	τ_{sem}	τ_{neg}	τ_{sem}	τ_{add}	τ_{sem}	τ_{sem}	τ_{add}
BERT	base	20.5	87.5	16.6	88.0	15.8	70.1	11.0	14.9	5.2	0.2	7.1	2.8	1.6
	large	16.5	73.6	12.5	93.0	12.3	69.7	9.9	10.2	3.3	0.1	8.4	3.0	1.7
RoBERTa	base	13.5	32.3	13.2	81.3	12.8	48.1	9.6	8.1	4.5	0.1	10.1	3.1	3.1
	large	10.2	41.5	8.4	82.9	9.8	13.0	7.9	4.7	2.3	0.1	9.3	2.7	1.1
Elelctra	base	7.1	69.9	8.8	84.4	9.4	17.8	9.2	6.3	3.0	0.0	10.1	2.8	2.4
	large	6.8	39.9	5.5	72.8	8.9	8.2	7.9	4.0	4.0	0.1	8.9	2.6	1.0
ERNIE2.0	base	13.3	53.6	6.3	77.4	13.2	17.6	10.1	11.9	5.2	0.1	5.1	3.2	2.7
	large	7.6	64.9	7.3	54.8	9.8	26.8	9.0	6.1	3.5	0.0	9.0	3.5	1.7
GPT2	base	12.8	86.8	18.4	82.6	18.1	68.3	28.8	30.0	19.6	0.8	14.1	2.7	2.2
	large	23.3	73.9	14.6	82.0	13.9	35.5	12.0	13.9	6.2	0.1	13.4	3.0	4.3
BART	base	13.4	63.7	12.2	82.6	11.4	68.3	10.8	9.6	4.7	0.1	8.7	3.0	3.5
	large	7.9	55.7	11.4	82.0	10.2	25.5	8.7	5.3	3.0	0.1	6.9	2.5	3.4
T5	base	12.9	22.3	8.1	32.8	11.7	7.5	10.9	5.3	4.1	0.2	16.0	2.1	0.3
	large	10.9	13.3	4.5	17.4	8.6	8.1	9.3	3.9	3.0	0.1	8.6	1.7	0.2

Table 4.4: The average of semantic (τ_{sem}), negational (τ_{neg}), and additive inconsistency (τ_{add}). All the metrics are lower the better. An average of five repetitions is reported.

training is a prominent advantage of the T5 model over other PLMs. The experiments were repeated five times for each model. It was confirmed that the best validation performance for each downstream task was almost close to the reported results in previous studies, demonstrated in Table B.2 in Appendix B.4.

4.4.2 Semantic Consistency Results

The experimental results of the semantic consistency evaluation are presented in Table 4.4. It was ascertained that PLMs exhibit different levels of consistency across diverse downstream tasks. Specifically, τ_{sem} values were exceedingly low in the SST-2 and AG-News tasks. I conjectured that a leading cause for this observation is that these tasks demonstrate a strong correlation between labels and specific words, such as sentiment words and proper nouns, resulting in minimal impact from paraphrasing. Regarding the remaining tasks, it was observed that PLMs make numerous errors, recording inconsistencies of approximately 10% or higher. An intriguing observation was that the inconsistencies in the MRPC task are comparable to those observed in other downstream tasks, even though the STS task is explicitly designed to concentrate on semantic equivalence.

Additionally, a notable observation was made regarding early-stage PLMs, such as

	BoolQ	MRPC	SST2	RTE
BAE	12.4 (-1.1)	7.9 (-5.3)*	5.9 (+3.6)*	11.7 (-1.1)
TextFooler	11.2 (-2.3)*	8.9* (-6.7)*	6.4 (+4.2)*	11.3 (-1.5)

Table 4.5: The inconsistency results of the adversarial training experiments. The value in parenthesis implies the difference compared to the original RoBERTa-base model. The difference is statistically significant with p value < 0.05 (*) using a two-sample t-test.

GPT2 and BERT, which demonstrated higher inconsistency levels than other PLMs. T5 and ELECTRA, on the other hand, exhibited the lowest τ_{sem} values, but the difference to the other PLMs was marginal, except for GPT2 and BERT. The findings suggest that a PLM's training objective does have some influence on its semantic consistency, albeit not to a critical extent.

Can adversarial training be a solution? Adversarial training is a widely used method to improve robustness and semantic consistency. The approach involves providing models with original and adversarial samples [Jin et al., 2020] that have different textual forms but convey identical meanings as their original counterparts. To investigate the potential benefits of adversarial training in enhancing semantic consistency, I conducted supplementary experiments by generating adversarial samples using text-attack methods and incorporating them into the training dataset. Two text attack methods, BAE [Garg and Ramakrishnan, 2020] and TextFooler [Jin et al., 2020], were employed to generate adversarial samples. Subsequently, the RoBERTa-base model was fine-tuned on four downstream tasks, BoolQ, MRPC, SST2, and RTE, using the original training set integrated with the generated adversarial samples. Each data instance had five adversarial samples produced for it.

The experimental results are summarised in Table 4.5. It was confirmed that adversarial training is not always beneficial. The enhancement in semantic consistency was generally marginal, except for the MRPC task, and the approach even resulted in a negative impact in the SST-2 task. A leading cause was speculated to be the likelihood of adversarial attack methods generating incorrect paraphrase sentences. Several examples of incorrectly generated paraphrased sentences by the adversarial attack methods are demonstrated in Table 4.6. It is obvious that the label of the adversarial samples cannot be considered the same as the original label. Furthermore, adversarial training possesses another disadvantage in that it is vulnerable to instances for which the attack method cannot generate. It was observed that

Original sample Sentence 1: The stupendous power of the Tevatron made possible the 1995 discovery of the top quark - the last of six flavors of quarks predicted by the standard model theory of particle physics. Sentence 2: The top quark is the last of six flavors of quarks predicted by the standard model theory of particle physics. **Adversarial sample** Sentence 1: The stupendous power of the Tevatron made possible the 1995 discovery of the top quark - the top of six flavors of quarks predicted by the standard model theory of particle physics. Sentence 2: The top quark is the last of six flavors of quarks predicted by the standard model theory of particle physics.

Original Sample Label	Adversarial Sample Label
entailment	entailment

Original sample Sentence 1: Rockweed has been harvested commercially in Nova Scotia since the last 1950's and is currently the most important commercial seaweed in Atlantic Canada. Sentence 2: Marine vegetation is harvested. **Adversarial sample** Sentence 1: Rockweed has been introduced commercially in Nova Scotia since the last 1950's and is currently the most important commercial seaweed in Atlantic britain. Sentence 2: Marine vegetation is harvested.

Original Sample Label	Adversarial Sample Label
entailment	entailment

Table 4.6: Examples of degenerated adversarial samples of BAE [Garg and Ramakrishnan, 2020] on the RTE dataset. The words that changed in the adversarial samples are underlined in both the original and adversarial samples.

approximately 45% of inconsistent predictions include examples with distinct sentence structures (e.g., transformations from active to passive voice), which synonym-replacement-based methods like BAE and TextFooler are unable to generate. These limitations can be addressed by employing recent LLMs, which enable the generation of more elaborate and diverse paraphrased sentences [Fang et al., 2023, Cegin et al., 2023]. However, as these models are still prone to errors, a certain degree of human inspection is required. Consequently, the findings indicate that adversarial training cannot serve as a fundamental solution for enhancing semantic consistency, as the process cannot be entirely automated and still necessitates human intervention.

4.4.3 Negational Consistency Results

Table 4.4 shows the experimental results of the negational consistency evaluation. It was discovered that τ_{neg} exhibits a significantly high value across all tasks, with the exception of the SNLI task, indicating that the fine-tuned PLMs completely fail to

Model	BoolQ		MRPC		RTE	
	base	large	base	large	base	large
BERT	1.63	1.80	6.95	2.81	1.38	1.10
GPT2	1.32	1.78	7.53	2.82	1.07	1.10
BART	2.18	1.99	15.31	3.09	1.58	1.31

Table 4.7: The ratio of the effect of model design over that of superficial cues (ρ) of BERT, GPT2, and BART.

comprehend the opposite meaning. Regarding the SNLI task, I strongly posited that the low τ_{neg} values are attributed to superficial cues present in the training data. It is widely acknowledged that a strong correlation exists between negation expressions and "contradiction" labels in the SNLI data [Gururangan et al., 2018]. I also confirmed in this study that nearly 68% of training instances containing negation expressions in the hypothesis are labelled as "contradiction". Hence, obtaining low τ_{neg} values in the SNLI task is relatively easy, because the new evaluation set for assessing the negational consistency originates from instances with "entailment" labels, as illustrated in Section 4.3.2. The low τ_{neg} values exhibited by the T5 models across all downstream tasks further support this claim, as the T5 model can benefit from the SNLI task through the multi-task training strategy.

Model design vs. superficial cues. Similar to the semantic consistency evaluation, GPT2 and BERT generally exhibited the worst performance, raising the question of which attribute, among the model design (e.g., training objectives, model structure) or superficial cues existing in training data, has a greater influence on the negational consistency. In Eq. 4.13, I defined the following metric for model M on task T to measure the impact of superficial cues over model designs.

$$\rho_T^M = (\tau_T^M - \tau_{SNLI}^M)/(\tau_T^M - \tau_T^*), \tag{4.13}$$

where τ_T^M denotes the negational inconsistency of model M on the task T. τ_T^* indicates the best inconsistency of task T among similar-sized models (e.g., *base*). The metric intuitively implies that the performance gap with SNLI (i.e., the impact of superficial cues) is ρ times larger than that with the best PLM (i.e., the impact of model designs). Therefore, if the abovementioned assumption is correct, ρ values should be larger than 1.

The ρ metric was computed for BERT, GPT2, and BART, whose performance did not rank at the top across all downstream tasks. T5 was excluded from deciding τ_T^*,

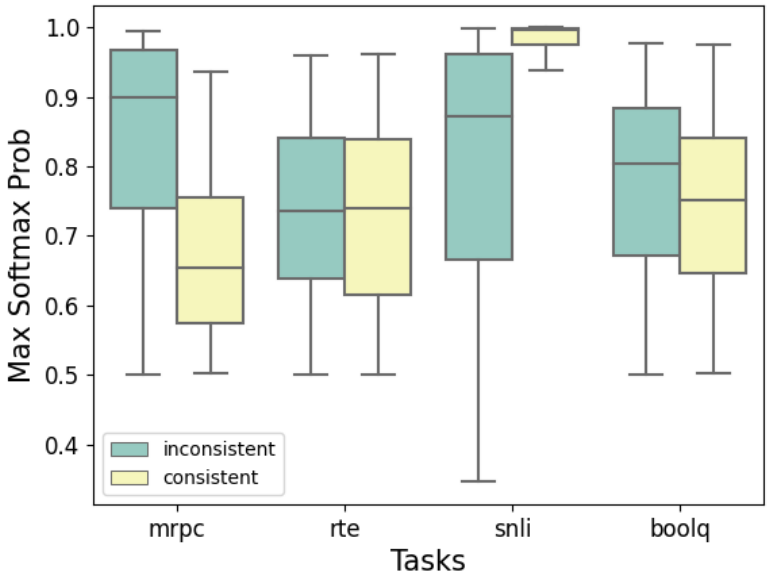


Figure 4.3: Box plot of the maximum softmax probability of RoBERTa-base for negational consistency evaluations. The horizontal bars within each box denote, from bottom to top, the minimum value, the 25% quantile, the median, the 75% quantile, and the maximum value. The figure illustrates that PLMs generate prediction probabilities for inconsistent predictions that are comparable to or higher than those for consistent predictions.

because it was trained in a multi-task fashion, which bestows advantages over other PLMs, thus complicating the precise measurement of the impact of superficial cues and model designs. The results are summarised in Table 4.7. It was observed that the ρ value is greater than 1 in every case. All three models exhibited relatively low ρ values in the RTE task, sharing similar superficial cues with the SNLI task. The findings suggest that superficial cues possess a greater impact than model designs in determining PLMs' negational consistency.

Overconfident inconsistent predictions. The negational inconsistency problem will be less concerned if the predictions are made by chance, indicating a high entropy in the predictive distribution. However, it was observed that PLMs exhibited high confidence regarding their inconsistent decisions, producing similar or higher maximum softmax probabilities, i.e., confidence score, than the consistent predictions in most cases. For example, Figure 4.3 displays the box plot illustrating the maximum softmax probability of the RoBERTa-base model for its consistent and inconsistent predictions. The confidence scores of consistent predictions were higher than those

Model		MRPC	RTE	SNLI		WiC	
		τ_{sym}	τ_{sym}	τ_{sym}	τ_{trn}	τ_{sym}	τ_{trn}
BERT	base	7.6	3.3	8.7	4.0	8.9	46.8
	large	6.8	3.2	7.1	3.6	7.0	49.3
RoBERTa	base	4.7	1.7	6.6	3.3	6.9	50.8
	large	4.3	2.2	7.6	3.5	7.3	46.6
ELECTRA	base	7.1	1.8	6.8	3.3	5.1	48.0
	large	5.3	1.1	3.9	2.5	7.9	46.5
ERNIE2.0	base	6.6	1.7	6.4	3.3	5.1	51.1
	large	6.4	5.4	4.7	3.0	6.9	46.7
GPT2	base	14.5	3.7	15.3	10.4	13.1	47.4
	large	10.6	1.5	9.0	4.9	12.5	49.8
BART	base	5.6	3.7	11.6	4.7	7.7	53.0
	large	4.6	2.6	5.9	2.8	5.4	53.1
T5	base	3.7	6.0	7.3	3.6	7.9	46.3
	large	4.2	2.9	6.1	2.9	6.3	45.3

Table 4.8: The average of symmetric (τ_{sym}) and transitive inconsistency (τ_{trn}). All the metrics are lower the better. An average of five repetitions is reported.

of inconsistent predictions only in the SNLI task, wherein the issue of superficial cues exists. For the other downstream tasks, the confidence scores of inconsistent predictions were comparable to or higher than those of consistent predictions. The experimental results indicate that fine-tuned PLMs are hard to trust regarding negational consistency, given their overconfidence in producing incorrect and inconsistent predictions.

4.4.4 Additive Consistency Results

The results of additive consistency evaluations are provided in Table 4.4. It is worth noting that this evaluation is a very easy task, as the input comprises a combination of two sentences that belong to the same category, providing the model with ample evidence to make the correct decision. It was observed that PLMs were generally highly consistent in terms of additive consistency. All the PLMs exhibited low τ_{add} values in the STS task. However, except for the T5 models, other PLMs made some mistakes in the AG-News task, which attained an average τ_{add} of 2.3. Given the low level of task difficulty, an error rate of approximately 2% should not be overlooked, indicating that PLMs need to be more consistent on the additive property.

4.4.5 Symmetric Consistency Results

Table 4.8 presents the experimental results of the symmetric consistency evaluation. In terms of the model architecture, it was observed that GPT2, once again, exhibited the least favourable performance in most cases, suggesting that decoder-only auto-regressive models are less suitable for achieving a high consistency through fine-tuning. In general, ELECTRA models demonstrated a superior performance than the other PLMs, yet no substantial difference in τ_{sym} values with the other PLMs was observed.

One could argue that around or lower than 10% of symmetric inconsistency is deemed an acceptable performance. Nevertheless, this should be noticed, because symmetry is a straightforward property that demands a simple reasoning ability. For this reason, humans are prone to exhibit an exceedingly low level of symmetric inconsistency. Through a concise human evaluation conducted on the MRPC task, asking 30 questions to each of the five human annotators, it was observed that humans display a high level of consistency on symmetry, achieving an average τ_{sym} of 0.7 across the four downstream tasks. This observation implies that substantial efforts should be invested to enhance PLMs in order to achieve a symmetric consistency at a level comparable to human performance.

4.4.6 Transitive Consistency Results

Table 4.8 describes the transitive consistency evaluation results. Interestingly, an entirely different tendency was observed between the two tasks: SNLI and WiC. Overall, all PLMs demonstrated a strong performance in the SNLI task, designed to infer the logical relationship between two given sentences. However, in the WiC task, which centres on word meaning, the inconsistency was remarkably high despite the fact that the new evaluation data were derived from the training set. The results indicate that the transitive reasoning ability strongly depends on the objectives of downstream tasks.

Does the training data size matter? The SNLI and WiC tasks have two significant distinctions: (1) task objective and (2) training data size, with approximately 500,000 instances for SNLI and 6,000 instances for WiC. To investigate whether a larger training dataset is beneficial to achieving a higher transitive consistency, I conducted a supplementary experiment by down-sampling the training data size of SNLI to 6,000 instances. The ELECTRA models that record the best τ_{trn} were used for

Model		SNLI		SNLI-6K		WiC	
		$\mathcal{A}_{val}$	τ_{trn}	$\mathcal{A}_{val}$	τ_{trn}	$\mathcal{A}_{tr}$	τ_{trn}
ELECTRA	base	91.8	3.3	64.8	10.0	81.6	48.0
	large	93.5	2.5	85.6	3.3	80.7	46.5

Table 4.9: The transitive inconsistency (τ_{trn}) observed ELECTRA models during the down-sampled SNLI experiments. $\mathcal{A}_{val}$ and $\mathcal{A}_{tr}$ denote the validation and training accuracy, respectively.

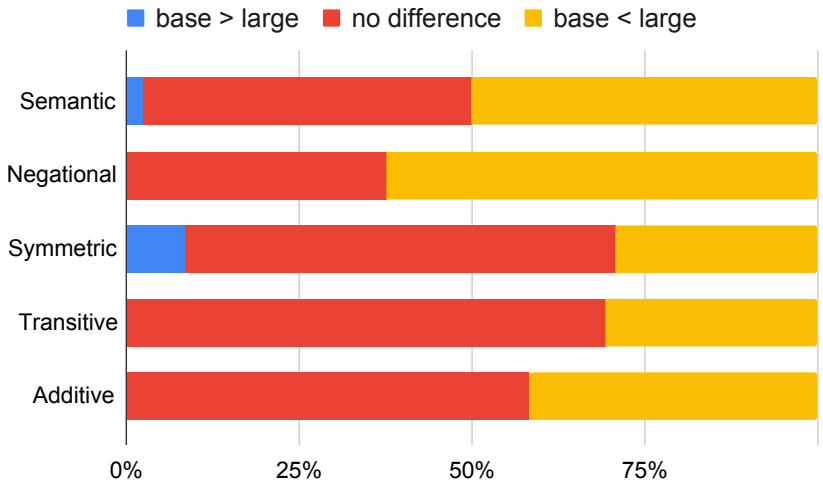


Figure 4.4: The portion of experimental cases where the large-size models are more or less consistent than the base-size models. A two-sample t-test under the significance level of 0.05 was employed to ascertain the statistical difference in performance. It was observed that large models do not necessarily perform better than small models in terms of consistency.

this experiment. The results are summarised in Table 4.9. The training performance of the WiC task was reported instead of the validation performance, because the new transitive consistency evaluation set of the WiC task originated from the training data. It was observed that the inconsistency increased after the down-sampling, but it still remained lower than that of the WiC task. Additionally, the validation accuracy was impaired, particularly in the case of the ELECTRA-base model. These observations indicate that a small training dataset can exacerbate transitive inconsistency due to decreased model accuracy, but the task objective has a significantly greater impact than the size of the training data.

4.4.7 Overall Analysis

Are large models more consistent? It is a widely recognised phenomenon that, across various downstream tasks in terms of accuracy metrics, large-size PLMs generally outperform their corresponding small-size models, i.e., the models have the same training objective and structure but only differ in the number of parameters. The question arises of whether the same trend holds true from a consistency perspective. Figure 4.4 illustrates the portion of the three cases: the performance of the large PLMs was superior, inferior, or exhibited no statistical difference compared to their corresponding small-sized PLMs. Interestingly, the cases where no statistically significant difference in consistency between the large- and base-size PLMs accounted for the largest portion across all five consistency types. In some cases, even the base-size PLMs exhibited a superior performance compared to their large-size counterparts. This phenomenon was hardly observed in accuracy-based evaluation metrics. The result highlights the significance of considering additional evaluation metrics like consistency, which offers a different perspective on the model's performance compared to accuracy-based metrics to achieve a more precise and comprehensive assessment of model performance. Our experimental results are consistent with recent findings demonstrating that smaller models frequently achieve a comparable performance to larger models [Li et al., 2023]. Our findings also align with studies on *inverse scaling*, a phenomenon that observes a decline in output quality as model size increases, particularly in areas such as negation understanding [Truong et al., 2023], social biases [McKenzie et al., 2023], and programming [Miceli Barone et al., 2023].

Necessity of a unified benchmark. The experimental results on the BECEL benchmark highlight the importance of comprehensively assessing models' performance in a wide spectrum. The findings verify that none of the PLMs exhibited a high consistency performance in every test case, indicating that concentrating solely on a specific downstream task or consistency type may lead to a biased conclusion. For instance, if only semantic consistency were considered for the evaluation, one might conclude that PLMs exhibit satisfactory consistent behaviours. Provided the evaluation solely focused on the NLI tasks, similar to previous studies, the conclusion might be significantly distorted, because the results of all consistency types appear reasonable in the NLI tasks, particularly in the case of the SNLI task. The BECEL dataset, however, mitigates the risk of drawing such distorted conclusions by enabling

researchers to evaluate models across multiple consistency types and tasks. The finding demonstrates the importance of employing a unified benchmark encompassing diverse perspectives.

4.5 Experiments and Analysis on Large Language Models

The emergence of LLMs such as GPT-3 [Brown et al., 2020] has facilitated in-context learning, which allows for the execution of NLP tasks without fine-tuning by utilising a few or no examples. Recently, more improved LLMs like ChatGPT have gained a huge attention due to their notable performance in human professional exams. This section investigates whether such LLMs demonstrate correct behaviours regarding consistency.

4.5.1 Experimental Design

Evaluation scope. Due to the competitive usage of OpenAI's LLM API, the scope of the evaluation was reduced to ensure an effective assessment. Among the five consistency types in the BECEL dataset, additive consistency was excluded from the evaluation, as it was observed in Section 4.4.4 that most PLMs were highly consistent with additive consistency. As for downstream tasks, the SST2 and AG-News tasks were not included due to their exclusive focus on semantic and additive consistency.

Generating LLMs Predictions. The experiments were conducted on the *20th July 2023* version of OpenAI API. For test cases where the data size exceeds 1,000, 200 data instances were sampled due to the heavy traffic of OpenAI API. The zero-shot experiments were conducted by using two different prompt versions designed by Eleuther AI[4] and Wei et al. [2022] in order to verify how consistency changes with different prompt designs. The prompts for each downstream task and examples are described in Tables B.3 and B.4 in Appendix B.5. The primary focus from Sections 4.5.2 – 4.5.5 will be the performance of ChatGPT utilising the Eleuther AI prompt. The results of other prompts and GPT-4 will be discussed in Section 4.5.6.

In general, LLMs generated structured answers, such as "Yes/No (Equivalent/Not Equivalent)" for the MRPC task, along with/without explanations for their decisions.

[4]https://github.com/EleutherAI/lm-evaluation-harness

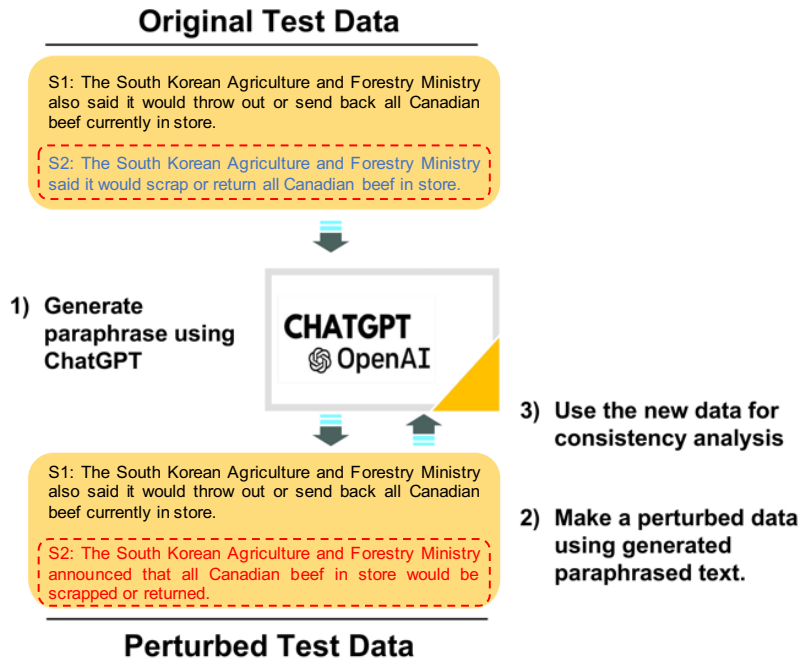


Figure 4.5: The overall procedure of measuring self-semantic consistency through paraphrases generated by LLMs themselves.

Nonetheless, a few instances were noted where the outputs did not adhere to the structured formats. To address this issue, a manual review was conducted to revise and align these cases with the desired format.

4.5.2 Semantic Consistency Results

It is widely recognised that LLMs are capable of performing various NLP tasks, such as summarisation, question answering, and paraphrasing. In this context, an additional experiment was conducted, which involved generating paraphrased sentences using LLMs and subsequently employing the generated paraphrases for the semantic consistency evaluation in addition to the original BECEL benchmark. The overall procedure of this evaluation is depicted in Figure 4.5.

The experimental results are summarised in the second column of Table 4.10. The BECEL dataset performances of two PLMs, ELECTRA-large [Clark et al., 2020] and T5-large [Raffel et al., 2020]), which performed the best in general, are taken from Table 4.4. In the SNLI task, it was observed that LLMs frequently encountered difficulties distinguishing the "Neutral" and "Contradiction" classes, potentially resulting in underestimating their consistency. Hence, the two classes were merged into

Model	Semantic								Negation		
	MRPC		RTE		SNLI		WiC		MRPC	RTE	SNLI
	τ_B	τ_S	τ_B	τ_S	τ_B	τ_S	τ_B	τ_S	τ_N	τ_N	τ_N
ChatGPT (EAI)	9.4	14.4	11.3	13.7	20.0	20.5	14.3	18.4	7.0	**2.1**	1.3
ChatGPT (Wei)	10.9	11.4	15.3	17.3	16.5	16.5	14.3	17.2	**6.7**	5.2	3.4
GPT-4 (EAI)	16.3	13.4	12.5	12.5	19.0	22.0	12.9	13.8	7.6	6.8	1.7
GPT-4 (Wei)	11.9	10.4	9.7	10.5	23.5	25.5	6.4	7.3	9.8	6.3	**0.5**
ELECTRA-large	5.5	-	8.9	-	**7.9**	-	8.9	-	72.9	8.2	4.0
T5-large	**4.5**	-	**8.6**	-	9.3	-	**8.6**	-	17.4	8.1	3.9

Table 4.10: Experimental results of the semantic and negation consistency evaluation on LLMs. τ_B and τ_S denote the inconsistency of the BECEL dataset and paraphrases generated by LLMs, respectively. τ_N refers to the negation inconsistency. "EAI" implies the prompt design of Eleuther AI, and "Wei" denotes the prompt design of Wei et al. [2022]. The best performance is in bold.

Task: SNLI, Paraphrase Type: BECEL	
Original Inputs	Perturbed Inputs
Premise: Kids play in the water in the middle of the street.	Premise: Kids play in the water in the middle of the street.
Hypothesis: Kids are running from zombies.	Hypothesis: Children are fleeing from zombies.
Prediction: Not Entailment (Contradiction)	Prediction: Entailment
Task: MRPC, Paraphrase Type: ChatGPT	
Original Inputs	Perturbed Inputs
S1: Looking to buy the latest Harry Potter ?	S1: Looking to buy the latest Harry Potter ?
S2: Harry Potter 's latest wizard trick ?	S2: The newest magical feat of Harry Potter ?
Prediction: Not Equivalent	Prediction: Equivalent

Table 4.11: Examples of semantic consistency violation of ChatGPT.

a unified "Not Entailment" class to address this concern. The experimental results revealed that ChatGPT exhibited significantly higher levels of inconsistency in the BECEL dataset when compared to fine-tuned PLMs, thereby indicating ChatGPT's limited capability to comprehend semantic meanings. Furthermore, it was confirmed that ChatGPT demonstrated self-contradiction, meaning that it generates inconsistent outputs for self-generated paraphrase sentences with a semantic inconsistency exceeding 10%. The findings indicate that ChatGPT failed to generate proper paraphrased sentences or to capture the meaning of texts conveying identical meaning; either case undermines its reliability. Table 4.11 presents several examples where ChatGPT violates semantic consistency. More examples are listed in Table B.5 in Appendix B.5.

Task: MRPC, Consistency Type: Negation	
Original Inputs	Perturbed Inputs
S1: He arrives later this week on the first state visit by a US President .	S1: He arrives later this week on the first state visit by a US President .
S2: Mr Bush arrives on Tuesday on the first state visit by an American President .	S2: Mr Bush doesn't arrive on Tuesday on the first state visit by an American President.
Prediction: Equivalent	Prediction: Equivalent
Task: SNLI, Consistency Type: Symmetric	
Original Inputs	Perturbed Inputs
Premise: There is a man climbing as the boy holds the rope	Premise: A man holds a rope for a boy who's about to climb a wall.
Hypothesis: A man holds a rope for a boy who's about to climb a wall.	Hypothesis: There is a man climbing as the boy holds the rope
Prediction: Not Entailment (Contradiction)	Prediction: Entailment

Table 4.12: Examples of negational and symmetric consistency violations of ChatGPT.

4.5.3 Negational Consistency Results

The third column of Table 4.10 presents the experimental results of the negation consistency evaluation. It was observed that ChatGPT attained significantly lower negation consistency than the best-performing fine-tuned PLMs across all three downstream tasks while exhibiting nearly perfect consistency on the SNLI task. The findings suggest that ChatGPT showed an improved understanding of negation expressions and antonyms, which has been a significant concern for PLMs trained in a self-supervised manner [Kassner and Schütze, 2020, Ettinger, 2020, Hossain et al., 2020, Hosseini et al., 2021]. I conjectured that involving human feedback in ChatGPT training [Ouyang et al., 2022] is pivotal for learning the meaning of negation expressions and antonyms, surpassing previous PLMs that infer their meaning solely from context information based on the distributional hypothesis. Investigating the impact of providing human feedback on learning textual meaning will be an intriguing avenue for future research. Several examples of the violation of negational consistency are presented in Table 4.12. More examples are available in Table B.6 in Appendix B.5.

4.5.4 Symmetric Consistency

The results of the symmetric consistency evaluation are described in the second column of Table 4.13. Compared to the best-performing PLM, ChatGPT exhibited approximately three times higher symmetric inconsistency in the MRPC and WiC

Model	Symmetric				Transitive	
	MRPC	**RTE**	**SNLI**	**WiC**	**SNLI**	**WiC**
	τ_S	τ_S	τ_S	τ_S	τ_T	τ_T
ChatGPT (EAI)	13.0	19.7	1.4	25.8	**2.1**	**12.3**
ChatGPT (Wei)	14.5	33.3	3.6	24.2	2.6	14.9
GPT-4 (EAI)	11.0	20.7	**1.0**	4.1	2.3	3.6
GPT-4 (Wei)	9.0	6.9	2.0	8.1	2.9	**4.3**
ELECTRA-large	5.3	**1.1**	3.9	7.9	2.5	46.5
T5-large	**4.2**	2.9	6.1	**6.3**	2.9	45.3

Table 4.13: Experimental results of the symmetric and transitive consistency evaluation. τ_S and τ_T denote the symmetric and transitive inconsistency, respectively. The best performance is in bold.

tasks and more than ten times higher in the RTE task. ChatGPT performed better than the best-performing PLM in the SNLI task, but the difference was marginal, considering the huge gap in the other three downstream tasks. Interestingly, a huge variation in performance across different downstream tasks was observed, recording the lowest symmetric inconsistency in the SNLI task. However, despite the relatively low inconsistency rate for the SNLI task, it should not be overlooked, given the straightforward nature of the symmetry property. Consider ChatGPT, which operates in the medical domain, generating prescriptions by taking a list of symptoms as input. If the model were to produce completely different prescriptions solely due to the variation in the order of listed symptoms, its trustworthiness would be severely damaged and even cause detrimental results, even if the probability of such an occurrence is extremely low. Therefore, it is imperative to ensure that LLMs meet logical consistencies to enhance their trustworthiness and facilitate their safe implementation in real-world applications. Table 4.12 presents several examples of symmetric consistency violations. In the case of the upper example, ChatGPT predicted the answer for the left instance as "entailment." However, in the right instance, despite S2 being modified to convey the opposite meaning, the model produced the same answer as in the left instance, thereby demonstrating a violation of negational consistency. In the case of the lower example, the predicted label for the left instance is "contradiction." Consequently, when S1 and S2 are switched in the right instance, the model should produce the same prediction according to the symmetric property. However, it produced "entailment" instead. Thus, this example demonstrates a violation of symmetric consistency. More examples can be found in Table B.6 in Appendix B.5.

Task: WiC, Consistency Type: Transitive	
Original Inputs	
Sentence1: Strike a medal.	Sentence1: Strike coins.
Sentence2: Strike coins.	Sentence2: A bullet struck him.
Word: strike	Word: strike
Prediction: Equivalent	Prediction: Not Equivalent
Newly Created Inputs	
Sentence1: Strike a medal Sentence2: A bullet struck him. Word: strike	
Prediction: Equivalent	
Task: SNLI, Consistency Type: Transitive	
Original Inputs	
Premise: A performance group is staged in one collective motion.	Premise: A performance group is staged in one collective motion.
Hypothesis: The group is separate from one another.	Hypothesis: There's a performance group doing something together.
Prediction: Contradiction	Prediction: Entailment
Newly Created Inputs	
Premise: The group is separate from one another.	
Hypothesis: There's a performance group doing something together.	
Prediction: Entailment	

Table 4.14: Examples of transitive consistency violations of ChatGPT.

4.5.5 Transitive Consistency

The third column of Table 4.13 displays the transitive consistency evaluation results. In contrast to other consistency types where ChatGPT demonstrated comparable or worse performance than fine-tuned PLMs, it exhibited better results than fine-tuned PLMs, particularly attaining noteworthy improvements in the WiC task. It is intriguing to note that ChatGPT excelled in higher-level logical reasoning, such as transitive inference, while encountering challenges in simpler logical properties, such as the symmetry property. It is inferred that incorporating diverse logical properties into the RLHF process for training LLMs can open avenues for the development of LMs that exhibit greater logical consistency. Several examples that violate transitive consistency are presented in Table 4.14.

4.5.6 Overall Analysis

Can Prompt Design to be a Solution? A prompt is a textual input comprising task descriptions and, in the case of a few-shot task, additional examples [Lester et al., 2021], which was demonstrated as an effective approach for controlling the behaviour of GPT-3 [Brown et al., 2020]. In this regard, it has been that seeking an optimal

prompt for each task might enhance LLMs' consistency. However, the experimental results offer sceptical evidence for this claim. When comparing the performance of ChatGPT using prompts designed by Eleuther AI and Wei et al. [2022], no statistically significant difference in performance was observed across all consistency types and downstream tasks at a confidence level of 0.05. In the case of GPT-4, analogous outcomes were demonstrated, where a statistically significant difference at a confidence level of 0.05 was observed solely in symmetric consistency in the RTE task. I posited that a primary factor contributing to this phenomenon is an inherent characteristic of machine learning: inductive reasoning. The underlying idea of prompt design is that prompts created by experimenters may not be optimal, because language models might have acquired target information from entirely different contexts [Jiang et al., 2020]. In other words, prompt design can address the inconsistency issue only if numerous consistency properties are explicitly reflected in the inductive bias of LLMs, which, from a technical standpoint, is not practically feasible. Furthermore, even if different prompt designs were to improve consistency, it would lead to another violation of semantic consistency, as prompts would essentially deliver identical semantic meanings. In this regard, it is challenging to anticipate that prompt design can serve as a fundamental solution for resolving the inconsistency issue.

Can Increasing the Size of Data and Model be a Solution? It could be posited that training larger LLMs with more abundant data containing high-quality instances may lead to an enhanced performance. Nevertheless, the experimental findings shed light on the constraints of this approach. It was revealed that GPT-4 did not necessarily outperform ChatGPT across every downstream task and consistency type when identical prompt designs were utilised. GPT-4 demonstrated statistically significant performance improvements in several test cases. With Eleuther AI's prompt, the symmetric and transitive consistency of the WiC task was improved. Meanwhile, with prompts devised by Wei et al. [2022], the improvements were discernible in symmetric consistency for both the RTE and WiC tasks, transitive consistency in the WiC task and negational consistency in the SNLI task. However, statistically notable enhancements were not observed in the remaining test cases.

Furthermore, increasing the size of the data and model is an unsustainable approach from a longer-term perspective. Firstly, the data collection procedure demands a considerable effort and poses challenges in encompassing all potential variations, particularly in complex consistency types, such as transitive and semantic consistency. Secondly, even in the event of successful data expansion, the affordability

of updating an LLM on the new dataset is questionable. Given the ever-changing nature of information, the process of data expansion and continuous updating of an LLM becomes imperative, because the presence of outdated and incorrect information may potentially lead to adverse effects on the model and hinder its practical applications [Wen and Wang, 2023, Zhuo et al., 2023]. However, training an LLM entails substantial financial and, particularly, environmental costs [Bender et al., 2021]. For instance, training a BERT-base model without hyperparameter tuning results in a CO2 emission of 640kg, a quantity comparable to that of a passenger flying from New York to San Francisco [Strubell et al., 2019]. Consequently, a simple estimation of the CO2 emissions for re-training ChatGPT and GPT-4 is expected to be 1,033t and 10,240t, respectively, as those models are 1,590 times and 16,000 times larger than BERT-base. This is a huge amount of emissions, considering a person is responsible for 5t CO2 per year. The persistent release of such a significant volume of greenhouse gases would exert a deleterious effect on the environment, particularly in modern society confronting the global climate crisis.

4.6 Related Work

Consistency Investigation of Pre-trained Language Models Despite the remarkable performance of PLMs in various NLP downstream tasks, they have been observed to make inaccurate decisions, a mistake that humans barely exhibit. These errors have facilitated research that investigates and evaluates the performance of PLMs from other perspectives beyond simplistic metrics like accuracy. Consistency is one of these alternative perspectives that assess PLMs in terms of their behavioural correctness. The concept of semantic consistency can be traced back to the field of computer vision, wherein it was discovered that negligible perturbations could lead to substantial changes in the model's decision [Szegedy et al., 2014]. Similar ideas have been applied in the domain of NLP, where small perturbations are introduced to the input text, preserving its meaning while altering the model's decisions [Jin et al., 2020, Garg and Ramakrishnan, 2020, Li et al., 2020a, Ivgi and Berant, 2021, Li et al., 2021]. Unlike the aforementioned studies that primarily focused on the behaviour of models fine-tuned for specific downstream NLP tasks, another strand of research delved into the inherent semantic consistency of PLMs through the employment of MLM. Ravichander et al. [2020] modified a masked query by substituting an object in the query with its plural form. Although PLMs are expected to generate similar outputs, given that the perturbation does not alter the overall meaning of the query,

it was observed that PLMs frequently generate different outputs. Elazar et al. [2021] adopted an alternative modification method, where a masked query is rephrased into its paraphrased version. They observed the analogous outcomes where PLMs generate different predictions when perturbation is applied. Another line of research endeavours has investigated the capacity of PLMs to adhere to specific logical properties. Studies regarding this area include the examinations of PLMs' comprehension regarding negation expressions [Hossain et al., 2020, Ettinger, 2020, Kassner and Schütze, 2020, Hosseini et al., 2021] and the assessment of their logical reasoning ability, which was evaluated based on logical properties such as symmetry [Wang et al., 2019c, Li et al., 2019, Kumar and Joshi, 2022] and transitive properties [Li et al., 2019, Asai and Hajishirzi, 2020, Lin and Ng, 2022]. Other studies have centred their focus on factual consistency, which pertains to a model's capacity to produce factually accurate outputs. Given the inherent nature of generating factual information, most of these investigations have concentrated on NLG tasks [Kryscinski et al., 2020, Pagnoni et al., 2021, Lewis et al., 2020b, Izacard and Grave, 2021, Li et al., 2020b, Shuster et al., 2021, Komeili et al., 2022].

Large Language Models GPT-3 [Brown et al., 2020], which has 175 billion parameters, initiated the new era of generative LLMs. In contrast to the formerly predominant PLMs, where LMs are fine-tuned on downstream NLP tasks, LLMs are distinguished by their in-context learning approach, which obviates the need for fine-tuning. Instead, LLMs generate outputs only with task descriptions and a few examples (few-shots) or even without examples (zero-shots) written in natural language form. More recently, the emergence of ChatGPT [Ouyang et al., 2022], which is trained with the RLHF technique [Christiano et al., 2017, Stiennon et al., 2020], has captivated the world with its remarkable performance across many downstream NLP tasks and its ability to pass human professional exams, such as the United States Medical Licensing Examination [Kung et al., 2023], four real exams at the University of Minnesota Law School [Choi et al., 2023], and Operation Management exam questions, which is a core MBA course [Terwiesch, 2023]. The astonishing capability demonstrated by ChatGPT has fostered the competitive development of more sophisticated and enhanced LLMs, such as LlaMA2 [Touvron et al., 2023], PaLM2 [Anil et al., 2023], and GPT-4 [OpenAI, 2023]. However, the investigation of these LLMs' consistency behaviours has yet to be conducted.

4.7 Discussion

This chapter commenced by presenting the definition of the consistency of an LM. Subsequently, it established a comprehensive taxonomy, classifying the concept into three major categories: semantic, logical, and factual consistencies derived from the definition mentioned above. The logical consistency was further divided into four sub-categories: negational, symmetric, transitive, and additive consistencies. These divisions were established based on the logical property employed to define desirable behaviours of PLMs. Following that, the BECEL benchmark was created, consisting of 19 test cases designed to evaluate LMs on five consistency types across seven downstream tasks. The new benchmark suite was then utilised to evaluate widely-used PLMs and LLMs.

The comprehensive experiments and thorough analyses of the experimental results revealed several important insights. First, it was ascertained that none of the PLMs, characterised by their training objectives, architectures, and model size, exhibited consistent behaviours across all test cases. While some extent of differences in degree was observed, inconsistent behaviours were exhibited by all PLMs irrespective of their architectures (i.e., Encoder-only, Decoder-only and Encoder-Decoder structures) and distinct training objectives (i.e., MLM and auto-regressive language modelling). Additional experiments unveiled that the influence of data characteristics on consistent behaviours surpasses that of the model's design, thus implying that the data play a more pivotal role in shaping the model's inductive bias. This is a critical issue, as the huge influence of data evokes a significant concern about uncontrollable AI. While humans can exert control over the model design, they lack control over the inherent artefacts and data characteristics, reviewing and manipulating the vast volume of data instances are demanding. The lack of data control can result in the potential risk of faulty behaviour of the model, such as producing inaccurate information or generating ethically problematic outputs [Nangia et al., 2020]. Second, the impact of the size of LMs on their consistent behaviours was found to be marginal. It was verified that large-sized PLMs are not necessarily more consistent than their corresponding base-sized counterparts. Moreover, it was noteworthy that even LLMs like ChatGPT demonstrate inconsistent behaviours, which, in certain consistency types and downstream tasks, are worse than fine-tuned PLMs. The findings suggested that increasing the size of LMs, which involves significant financial and environmental costs, cannot serve as a fundamental solution to enhance the consistent behaviours of LMs. Finally, the experimental results provided evidence regarding the limitations

of widely used existing methods, which were developed to improve the performance of LLMs, such as using more training data and prompt engineering. All the findings and insights from this chapter strongly emphasised the necessity of a sustainable and feasible approach that can fundamentally enhance the consistent behaviours of LMs.

Chapter 5

Improving Language Models' Meaning Understanding and Consistency by Learning Conceptual Roles

This chapter will propose a practical and efficient methodology that effectively alleviates inconsistency concerns across multiple consistency types simultaneously.

5.1 Introduction

AI systems that exhibit human-like behaviours can be considered to be more reliable and trustworthy [De Visser et al., 2016, Jung et al., 2019]. However, despite the remarkable progress that modern PLMs have achieved in various NLP tasks, recent evidence revealing their non-human-like behaviours [Hossain et al., 2020, Kassner and Schütze, 2020, Hosseini et al., 2021, Gupta et al., 2021, Sinha et al., 2021b] raises concerns regarding their trustworthiness, suggesting that PLMs' language understanding ability is significantly below that of humans.

The indicative evidence highlighting the inadequacy of PLMs in understanding textual meanings lies in their *inconsistent* behaviours [Mitchell et al., 2022]. Contrary to humans, these models manifest the incapability of textual understanding or logically contradictory behaviours in various aspects, such as generating different predictions for texts with semantically identical meanings [Ravichander et al., 2020, Elazar et al., 2021] or violating certain logical properties, such as negation [Kassner and Schütze, 2020, Ettinger, 2020, Asai and Hajishirzi, 2020] and symmetry [Wang et al., 2019c, Li et al., 2019, Kumar and Joshi, 2022], or transitivity [Asai and Hajishirzi, 2020, Lin and Ng, 2022]. Such inconsistent behaviours undermine the trust-

worthiness of the models and impose constraints on their practical usage, especially in risk-sensitive industries where even minor undesirable actions could potentially result in catastrophic consequences.

In prior research, attempts were made to alleviate the inconsistency issues primarily through two main approaches: data augmentation [Ray et al., 2019, Hosseini et al., 2021] or the introduction of specialised loss functions [Elazar et al., 2021, Kim et al., 2021, Kumar and Joshi, 2022]. However, these approaches possess inherent limitations, as they can solely target a specific consistency type and are incapable of addressing others. For example, the utilisation of consistency regularisation loss, which aims to enforce similarity between a model's predictive distribution on the original and its paraphrased inputs [Elazar et al., 2021], cannot effectively mitigate the violations of the logical negation property or symmetry property. Analogously, augmented negation data [Hosseini et al., 2021] can exclusively address the logical negation property. Furthermore, these approaches necessitate costly resources, leading to practical inefficiency. For instance, training large PLMs or LLMs with an additional regularisation term or augmented training data requires considerable computing resources, making it impractical. Moreover, collecting auxiliary data, such as paraphrased texts or negated texts, poses significant challenges, especially for low-resource languages lacking sufficient infrastructure for amassing extensive datasets.

To this end, this chapter presents a comprehensive approach that can concurrently enhance multiple types of consistency in a resource-efficient manner. Based on the findings from prior research [Sinha et al., 2021b] and Chapter 3 that revealed the limitations of the distributional hypothesis, a fundamental theory underpinning modern PLMs, I postulated a hypothesis that attributes the inconsistent behaviour of PLMs to their inability to accurately capture the precise meaning of language. Thus, the alleviation method proposed in this chapter is centred around addressing PLMs' deficiencies in meaning understanding by employing an auxiliary model specifically trained to imitate the language comprehension of humans, thereby enhancing the model's awareness of the semantic meaning of language. The underlying assumption of this approach is the conceptual role theory, a compelling theory that explains human cognition. According to the theory, the meaning of a word is predominantly determined by the interrelationship between concepts [Deacon, 1998, Santoro et al., 2021, Piantadosi and Hill, 2022]. In accordance with the conceptual role theory, the proposed alleviation approach commences by training a model that learns abundant symbolic meanings by tracing more precise interconnections between concepts

through word-definition pairs in a dictionary. However, this approach entails the potential downside that it is restricted to covering interrelationships between concepts provided in dictionary data, although it captures the interrelationship beyond the distributional hypothesis. Hence, general knowledge information is hardly reflected in the model. For example, the relationship between individuals such as Donald Trump and Ivanka Trump may not be correctly captured if their names and biographies are not listed in the dictionary used for training. To address this limitation, the second approach is followed, which is a training-efficient parameter integration method that effectively combines the parameters of the aforementioned model with those of existing PLMs. The rationale behind this parameter integration is to maximally exploit both word knowledge embedded in PLMs and the newly acquired meaning information, which is achieved by capturing precise interconnections between concepts. Moreover, the second approach facilitates fine-tuning by enabling only a few parameters to update. The contributions of this chapter can be concisely summarised as follows:

1. This chapter verifies that the enhancement of PLMs' meaning awareness can concurrently improve multiple types of consistency.

2. This chapter introduces a training-efficient parameter integration method that enables practical fine-tuning for large-sized PLMs.

3. I ascertained that the proposed approach can be readily applied to low-resource languages.

4. The experimental results indicate that the proposed approach facilitates additional consistency improvements in a simple yet effective manner by aggregating the parameters of PLMs possessing distinct inductive biases.

5.2 Methodology

5.2.1 Learning Conceptual Roles from Dictionary

Conceptual role theory is a convincing theory that accounts for human cognition. The theory views that the meaning of linguistic expressions is determined by the contents of the concepts or conceptual thoughts they are used to express [Harman, 1982, Greenberg and Harman, 2009]. In other words, the theory posits that meanings are derived from use and rejects the idea that meanings possess intrinsic content.

For better understanding, let's consider an example of a physics statement used by Block [1998]: $f = m \cdot a$. The equation does not illustrate an exact definition of f (force), m (mass) or a (acceleration); rather, it demonstrates the interrelationship of the three factors. As another example, consider the word "water" in the context of language. According to the theory, a term's meaning has no intrinsic definition and is instead determined by its closely interrelated concepts, such as "liquid", "without smell", "hydrogen", and "oxygen".

According to this view, the key to discovering a concept's meaning in language learning models is understanding the interrelationship between concepts [Deacon, 1998, Santoro et al., 2021, Piantadosi and Hill, 2022]. The idea has become a prevailing assumption for learning models in recent NLP research, and many studies have attempted to encapsulate the structure of interrelationships within the relational geometry of vector representations. Word2vec [Mikolov et al., 2013] was a pioneering approach in this field, training vector representations based on the distributional hypothesis, assuming that semantically analogous words are likely to appear in similar contexts [Harris, 1954]. In essence, the distributional hypothesis can be viewed as a subset of conceptual role theory, as it seeks to identify the interrelationships of words that frequently co-occur in similar contexts [Piantadosi and Hill, 2022]. Although there was a certain degree of discrepancy with human judgement, word representation vector spaces trained in this manner have exhibited promising results in modelling the relational geometry of bilingual languages [Lample et al., 2018] and multiple object features, such as size, temperature, and intelligence [Grand et al., 2022]. The recent success of large-size PLMs also supports the efficacy of the distributional hypothesis [Piantadosi and Hill, 2022]. PLMs are the modern NLP techniques that represent language into contextualised representations by utilising the distributional hypothesis as their learning objective [Sinha et al., 2021a],[1] i.e., they are trained to define concepts' meaning by relating them to other concepts that co-occur in similar contexts. Many studies have demonstrated the PLMs' ability to capture the relational geometry through various downstream tasks [Devlin et al., 2019, Liu et al., 2019b, Brown et al., 2020] and direct analyses of the relational information in colour space [Abdou et al., 2021] and ontological knowledge [Petroni et al., 2019, Wu et al., 2023].

The distributional hypothesis has served as an efficient assumption, enabling self-supervised training with general corpora. However, as stated above, it is a subset of conceptual role theory. This implies that the distributional hypothesis cannot fully encapsulate the relational geometry, as it can only reflect partial interrelationships

[1]See Appendix C.1 for more details.

that can be found in similar contexts. For example, as illustrated in Section 3.4.1, capturing semantic antonymy through the distributional hypothesis poses a challenge. To this end, under the hypothesis that the major cause of inconsistent behaviours is the lack of ability to comprehend concepts' meanings, the new approach that I propose in this chapter endeavours to enhance PLMs' understanding of meaning by enabling them to capture more precise interrelationships between concepts. Recent studies have revealed that training data play a pivotal role in determining the inductive bias of a model, surpassing the influence of its structure or learning objective [Furrer et al., 2020, Wang et al., 2022]. Hence, I formulated a novel learning task to overcome the limitation imposed by the distributional hypothesis. As the limitation of the hypothesis arises from the constrained interrelations between concepts presented in the training contexts, the training instances of the new learning task consist of the contexts where a concept and other closely interconnected concepts are jointly presented. To achieve this purpose, *word-definition* pairs from a dictionary were utilised. This choice was made on the fact that a definition represents a composition of concepts that explains the target concepts, making dictionaries widely used as vocabulary learning tools, particularly for second and foreign language learners [Takahashi, 2012, Zhang et al., 2020a]. Specifically, a training instance for language modelling was created by concatenating a target word and its definitions as a single text. With the language modelling objective, this approach enables the model to learn a word's meaning based on concepts constituting its' meaning rather than solely relying on concepts that appear in similar contexts.

For example, consider the words "happy" and "unhappy" in Table 5.1. Distributional models typically produce high similarity scores of the two words, as they are emotional expressions that can appear in similar contexts. For this reason, Skip-gram vectors [Mikolov et al., 2013], a representative distributional model trained with an English Wikipedia dataset and BERT representations [Devlin et al., 2019] demonstrate a high similarity between the embeddings of the two words despite the two words being antonyms. On the contrary, the proposed learning task is expected to improve the limitation by focusing on the relationships between words and their definitions and thereby capturing more precise interconnections between concepts: "unhappy" with "not happy", "sad", and "not pleased". Other examples in Table 5.1 that are not antonyms include "dining hall" and "dorm". These words can appear in similar contexts, as they both relate to illustrate school life. Consequently, they will convey similar meanings if the model is trained based on the distributional hypothesis. This is evident from the high similarity in the representations of the two

Raw Dictionary Data	
Word	Definition
happy	1) feeling or showing pleasure; pleased 2) giving or causing pleasure
unhappy	1) not happy; sad 2) not pleased or satisfied with somebody or something
dorm	1) a large room containing many beds, for example, in a boarding school 2) a large building at a college or university where students live
dining hall	1) a large room in a school or other building, where many people can eat at the same time
Training Instances	
happy feeling or showing pleasure; pleased giving or causing pleasure	
unhappy not happy; sad not pleased or satisfied with somebody or something	
dorm a large room containing many beds, for example, in a boarding school a large building at a college or university where students live	
dining hall a large room in a school or other building, where many people can eat at the same time	

Table 5.1: Examples of word-definition pairs in a dictionary and their concatenated version for training.

words generated by BERT and Skip-gram. In contrast, the proposed new learning task can associate "dorm" with concepts like "bed" and "students live" and "dining hall" with "people can eat", thereby learning more precise meanings by capturing the distinctions between two words.

The intermediate training technique [Phang et al., 2018, Wang et al., 2019a, Liu et al., 2019a, Pruksachatkun et al., 2020, Vu et al., 2020] was applied, which first trains PLMs on an intermediate task and employs it for other downstream tasks. As a result, PLMs are trained on the new word-definition dataset, and the resulting trained model is referred to as CRM. A primary reason for adopting the intermediate training technique is the insufficient number of word-definition pairs to train large PLMs from scratch. Also, the pre-trained weights of PLMs can be used as good initial values for training the models on the new dataset. CRM offers practical advantages, as (1) it can be applied to any PLM trained with language modelling objectives, and (2) it is easily applicable to other languages due to the relative ease of collecting dictionary data.

Several studies have been proposed to train word representations by relating words and their definitions using dictionary data [Eisner et al., 2016, Tissier et al., 2017, Bosc and Vincent, 2018, Shu et al., 2020]. The details regarding these previous studies can be found in Section 5.4. The primary distinction between these studies and the approach proposed by myself is that the latter is capable of producing contextualised

representations, where the representation of each word varies based on its context. In contrast, the aforementioned studies can only generate static word representations and cannot reflect the contextual sensitivity of words, which is a significant disadvantage compared to the contextualised representations [Ethayarajh, 2019]. Furthermore, the inductive bias that the previous studies can learn is restricted to dictionary data. In contrast, our approach can utilise multiple inductive biases of different models, which will be described in the next section.

5.2.2 Training-efficient Parameter Integration

Subsequently, I propose a parameter integration method to effectively incorporate the previously acquired knowledge of PLM with the improved meaning awareness of CRM.

Let $\mathrm{W}_p \in \mathcal{R}^{d \times l}$ and $\mathrm{W}_c \in \mathcal{R}^{d \times l}$ denote the parameter matrices of PLM and CRM, respectively. It is important to note that both PLM and CRM share the same model architecture; in other words, they have a parameter matrix of identical size. The purpose of the parameter integration process is to learn an integrated parameter matrix, denoted as $\mathrm{W}_{new} \in \mathcal{R}^{d \times l}$, during fine-tuning by incorporating W_p and W_c. Hence, the process can be defined as follows (Eq. 5.1):

$$\mathrm{W}_{new} = \mathrm{W}_o \begin{bmatrix} \mathrm{W}_p \\ \mathrm{W}_c \end{bmatrix}, \tag{5.1}$$

where $\mathrm{W}_o \in \mathcal{R}^{d \times 2d}$ is a trainable parameter matrix, while W_p and W_c remain fixed during fine-tuning. By decomposing W_o into $\mathrm{W}_1 \in \mathcal{R}^{d \times d}$ and $\mathrm{W}_2 \in \mathcal{R}^{d \times d}$, Equation 5.1 can be reformulated as follows (Eq. 5.2):

$$\begin{aligned} \mathrm{W}_{new} &= \begin{bmatrix} \mathrm{W}_1 & \mathrm{W}_2 \end{bmatrix} \begin{bmatrix} \mathrm{W}_p \\ \mathrm{W}_c \end{bmatrix} \\ &= \mathrm{W}_1\mathrm{W}_p + \mathrm{W}_2\mathrm{W}_c \\ &= \mathrm{W}'_p + \mathrm{W}'_c, \end{aligned} \tag{5.2}$$

where W'_p and W'_c refer to the matrices after fine-tuning. Finally, by decomposing W' into a fixed component (W) and an updated component (ΔW), Equation 5.2 can be rewritten in the following manner (Eq. 5.3):

$$\begin{aligned} \mathrm{W}_{new} &= (\mathrm{W}_p + \Delta\mathrm{W}_p) + (\mathrm{W}_c + \Delta\mathrm{W}_c) \\ &= \mathrm{W}_p + \mathrm{W}_c + \Delta\mathrm{W}_t. \end{aligned} \tag{5.3}$$

where $\mathrm{W}_t = \mathrm{W}_p + \mathrm{W}_c$. As a result, W_{new} is expressed as the addition of the fixed matrices W_p, W_c, and a trainable matrix $\Delta\mathrm{W}_t \in \mathcal{R}^{d \times l}$. Performing updates on the

entire parameters of $\Delta \mathrm{W}_t$ is equivalent to fine-tuning with the pre-trained weights of $\mathrm{W}_p + \mathrm{W}_c$, which is an impractical process, especially for large-size PLMs. Moreover, as extensively studied in previous research [Pruksachatkun et al., 2020, Wallat et al., 2020], fine-tuning carries the risk of catastrophic forgetting, which entails losing knowledge acquired during the pre-training stage. To address this concern, a low-rank adaptation technique [Hu et al., 2022] was introduced for the parameter integration process. This technique involves fixing the pre-trained weights and updating only a limited number of parameters based on the intrinsic dimension of the PLMs [Aghajanyan et al., 2021]. Specifically, $\Delta \mathrm{W}_t$ is transformed into a multiplication of two matrices $\mathrm{A} \in \mathcal{R}^{d \times r}$ and $\mathrm{B} \in \mathcal{R}^{r \times l}$, where $r \ll \min(d, l)$ (Eq. 5.4):

$$\mathrm{W}_{new} = \mathrm{W}_p + \mathrm{W}_c + \mathrm{AB}. \tag{5.4}$$

As a result, the number of trainable parameters can be significantly reduced, facilitating efficient fine-tuning for large-size PLMs. Additionally, in comparison to other adapter modules [Houlsby et al., 2019, Pfeiffer et al., 2021] that introduce additional adapter components between layers, i.e., increasing the size of the model, the approach merely adds AB without increasing the number of parameters. Consequently, this approach avoids the need for employing additional time or resources during the inference phase, thereby enabling practical and efficient inference [Hu et al., 2022].

Aggregating $\mathbf{W}_p$ and $\mathbf{W}_c$. The addition of W_p and W_c illustrated in Equation 5.3 leads to a magnification of the weight scale, making it unsuitable for conducting hyperparameter search from the values used in previous studies, resulting in much more effort for conducting hyperparameter search. Thus, to control the scale of the integrated parameters, a simple averaging aggregation method [Wortsman et al., 2022] was employed (Eq. 5.5):

$$\mathrm{W}_{new} = \frac{\mathrm{W}_p + \mathrm{W}_c}{2} + \mathrm{A}'\mathrm{B}'. \tag{5.5}$$

The rationale for adopting the simple averaging approach is to demonstrate the effectiveness of CRM by achieving notable improvements in consistency through the most straightforward aggregation method. However, it is worth noting that alternative aggregation methods, such as the Fisher-Weighted Average [Matena and Raffel, 2022] or RegMean [Jin et al., 2023], can also be considered.

Additional Knowledge Add-up. It is important to mention that the aforementioned aggregation approach also enables the integration of weights from any other models that share the same architecture with PLM and CRM. Consequently, Equation 5.5 can be further extended as follows (Eq. 5.6):

$$\mathrm{W}_{new} = \frac{1}{|S|+2} \sum_{i \in \{p,c,S\}} \mathrm{W}_i + \mathrm{A}'\mathrm{B}', \tag{5.6}$$

where S is the set of additional PLMs' weights that are going to be integrated. Note that, due to the parameter integration occurring through addition, no supplementary training or inference resources are required during fine-tuning, which confers computational benefits.

Applying the integration method to PLMs. Given that contemporary PLMs typically employ the Transformer [Vaswani et al., 2017] as a backbone architecture, the same low-rank adaptation approach proposed by Hu et al. [2022], which limits its usage solely to the self-attention weights while excluding MLP weights from consideration, was adopted for the parameter integration. For the sake of experimental efficiency, the same hyperparameter values for the low-rank adaptation, which was determined by Hu et al. [2022], were used. Specifically, for a hidden representation x, the scaling of $\Delta\mathrm{W}'_t x$ was performed by a factor of $\frac{\alpha}{r}$, where α and r were both set to 16 and 8, respectively.

5.3 Experiments and Analysis

5.3.1 Training CRM

Dataset. Frequently used English words were collected from the List of English Words[2] and Wikipedia Word Frequency data[3] to collect word-definition pairs for training CRM. The meanings of the collected words were then obtained using the Wikipedia English Dictionary API. Consequently, approximately 455,000 word-definition pairs were collected for training CRM. When it comes to training CRM in the Korean language, 1.6 million word-definition pairs were collected from the Korean Dictionary made by the National Institute of Korea Language.[4].

[2]https://github.com/dwyl/english-words, accessed 20th January, 2023
[3]https://github.com/IlyaSemenov/wikipedia-word-frequency, accessed 20th January, 2023
[4]https://github.com/spellcheck-ko/korean-dict-nikl, accessed 31st January 2023

Model		COLA	MRPC	RTE	QQP	SST2	MNLI	QNLI	Avg
RoB_B	P only	62.2	89.1	77.1	**89.8**	94.2	**87.0**	**92.6**	84.5
	P+C	**63.2***	**89.5**	**79.1***	**89.8**	**94.5**	**87.0**	**92.6**	**85.1**
RoB_L	P only	67.3	**90.5**	86.6	90.9	95.9	**90.4**	**94.8**	88.0
	P+C	**68.4***	**90.5**	**86.9**	**91.1†**	**96.0**	**90.4**	**94.8**	**88.3**

Table 5.2: Average GLUE performance of RoBERTa-base (RoB_B) and RoBERTa-large (RoB_L) models when the parameter integration is applied. The best performance is in bold. The figures of the proposed approach (P+C) show a significant difference compared to our counterpart (P only) with p-value < 0.05 (*) and p-value < 0.1 (†) using a two-sample t-test.

Model and training details. RoBERTa [Liu et al., 2019b] was chosen as the backbone PLM in experiments involving English datasets. For Korean dataset experiments, the Korean-RoBERTa model [Park et al., 2021] was employed. Both base and large-size models were employed for the experiments. The base-size models were trained for 500K steps, while the large-size models underwent training for 700K steps. The AdamW optimiser [Loshchilov and Hutter, 2019] with a learning rate of $1e$-6 and a linear learning rate scheduler decaying from $1e$-2 were used. The β_1, β_2, and ϵ values were set to 0.9, 0.98, and $1e$-6, respectively. Four GeForce GTX TITAN XP GPUs were used for the training.

5.3.2 Experiments on English Dataset

For the English dataset experiments, the proposed approach was evaluated on two benchmark datasets: GLUE [Wang et al., 2018] to measure the performance of basic NLU tasks and BECEL, introduced in Chapter 4, to measure consistency.

5.3.2.1 Experiments on GLUE Benchmark

The proposed approach was applied to GLUE downstream tasks to validate its performance on widely used NLU tasks. The models that exclusively utilise the weights of PLM, i.e., without integrating the weights of CRM, were also trained for comparison. The experiments were repeated five times for downstream tasks where the size of a training set exceeds 100,000 data instances (e.g., MNLI, QNLI, QQP, and SST2) and 10 times for the other tasks. The maximum sequence length and batch size were set to 256 and 32, respectively. All models were trained for 15 epochs, and the models with the best validation accuracy were finally selected. The AdamW optimiser [Loshchilov and Hutter, 2019] was used with a linear learning rate scheduler

Model			BoolQ		MRPC			RTE			SNLI				WiC		
			τ_{sem}	τ_{neg}	τ_{sem}	τ_{neg}	τ_{sym}	τ_{sem}	τ_{neg}	τ_{sym}	τ_{sem}	τ_{neg}	τ_{sym}	τ_{trn}	τ_{sem}	τ_{sym}	τ_{trn}
RoB_B	FT	P	11.3	43.2	7.4	82.2	**2.7**	11.9	22.9	1.0	10.8	8.3	12.0	3.8	14.8	**4.0**	13.6
	(125M)	P+C	**10.7**†	**35.4***	**6.7**	**78.9**†	**2.7**	**11.4**	**18.3**†	**0.5***	**10.6**	**7.4***	**11.2**†	**3.7**	**13.8**	4.2	**12.9**
	PI	P	12.2	59.9	10.0	82.2	6.5	12.8	30.9	**1.4**	**9.8**	8.3	**11.8**	4.1	15.1	**3.5**	13.9
	(0.8M)	P+C	**11.4***	**50.7***	**7.8***	**71.1***	**6.3**	**11.9**†	**26.4**	1.7	10.0	**7.3***	12.1	**4.0**	**11.9***	**3.5**	**13.5**
RoB_L	FT	P	8.4	45.9	**6.2**	78.9	2.6	7.8	7.0	2.1	9.2	7.8	9.2	3.3	10.6	5.9	11.5
	(355M)	P+C	**8.0**†	**45.8**	**6.2**	**75.2**	**2.0**	**6.2***	**7.0**	**1.3**†	**9.0**	**5.9***	**8.8**	**3.2**	**8.9**†	**4.8**	**9.6**†
	PI	P	9.6	43.3	6.8	77.3	4.7	7.7	**7.0**	2.2	**8.6**	7.3	7.3	3.2	11.7	3.0	10.6
	(2.6M)	P+C	**8.7***	**37.8***	**6.7**	**77.0**	**4.4**	**6.5**†	7.2	**1.6**†	**8.6**	**5.2***	**7.1**	**3.0***	**10.9**	**2.8**	**9.5**†

Table 5.3: Average BECEL performance of RoBERTa-base (RoB_B) and RoBERTa-large (RoB_L). τ_{sem}, τ_{neg}, τ_{sym}, and τ_{trn} denote semantic, negational, symmetric, and transitive inconsistency, respectively. The figure is highlighted in bold if the proposed approach (P+C) outperforms its counterpart (P). The values of P+C show a significant difference compared to P with p-value < 0.05 (*) and p-value < 0.1 (†) using a two-sample t-test.

decaying from $1e^{-2}$ and a warm-up ratio of 0.06. The learning rate was set to $5e^{-4}$. Matthew's correlation was used for the evaluation metric for the COLA task, while accuracy was used for the other downstream tasks.

Table 5.2 shows the performance of GLUE validation sets. Regarding the RoBERTa-base model, the proposed approach outperformed the sole usage of the PLM in COLA, MRPC, and RTE tasks, demonstrating comparable results in other tasks. In the case of the RoBERTa-large model, both approaches yielded similar performances in most downstream tasks, except for COLA, where the proposed approach showed a statistically significant improvement. The notable finding is that the integration of the CRM weights leads to a significant performance improvement in COLA, which is designed to assess grammatical errors and thus demands linguistic knowledge. Moreover, the improvements are particularly pronounced in the RoBERTa-base model, where the PLM retains less pre-trained information. The experimental outcomes indicate that CRM weights offer a richer source of linguistic knowledge, which can contribute to enhancing the comprehension of textual meaning.

5.3.2.2 Experiments on BECEL Benchmark

Subsequently, the proposed approach was assessed on the BECEL dataset, which was proposed in Chapter 4 and allows assessing five types of consistencies across seven downstream tasks. For each task, a model was fine-tuned using the original training set and subsequently tested on BECEL test sets specially designed to assess diverse

consistency types. The additive consistency was not included in this experiment, because it was verified in Section 4.4.4 that most PLMs were highly consistent regarding the additive consistency. In accordance, two downstream tasks, SST and AG-News, were also excluded from the evaluation scope, as they contain evaluation sets for only semantic consistency with the exclusion of additive consistency. To ascertain the impact of CRM and the parameter integration technique, four cases were examined, which are defined by (1) whether only a PLM is used (P) or a CRM is utilised with a PLM (P+C) and (2) whether the whole parameters are fine-tuned (FT) or the parameter integration is applied (PI). For clarification, P+C with FT represents a model that fine-tunes all the parameters, initialising the weights as the aggregation of PLM and CRM weights. P with PI denotes a model that solely adopts a low-rank adaptation technique to PLM, i.e., the work of Hu et al. [2022]. The same hyperparameter values used in GLUE benchmark experiments were employed, with the only variation in the learning rate of FT models, which was set to $2e^{-5}$. The experiments were repeated 5 times for the SNLI task, which has a large training set, and 10 times for the rest of the downstream tasks. For semantic and transitive consistency evaluation, the evaluation metrics proposed in Section 4.3.3 were used. When it comes to the negational and symmetric consistency, the conditional inconsistency metrics described in Section 4.5.1, which were used to measure the performance of LLMs, were employed for more conservative assessment. Table 5.3 presents the average performance.

Influence of CRM. The experimental findings demonstrated that incorporating the CRM weights leads to an enhanced consistency of multiple types across various downstream tasks. Specifically, among 15 consistency test cases, the models employing both the PLM and CRM weights exhibited lower inconsistencies than their counterparts using only the PLM weights in general. Through the application of the t-test, it was ascertained that the improvement was statistically significant in seven cases for FT and PI in the RoBERTa-base model. The figures corresponding to the RoBERTa-large models were 6 and 7, respectively. The notable aspect lies in the compelling observation that the improvements exhibited a particular significance in consistency types requiring the comprehension of textual meanings, such as semantic and negational consistency. The findings substantiate the hypothesis posited in this chapter, which postulates that acquiring precise conceptual roles can improve PLMs' meaning awareness and, consequently, lead to improved consistencies.

Fine-tuning vs. parameter integration. The primary reason behind adopting PI resides in improving the training/inference efficiency of large-size PLMs. The application of PI leads to considerable decreases in the number of training parameters, reducing from 125 million to 0.8 million for RoBERTa-base and from 355 million to 2.6 million for RoBERTa-large. Regarding the RoBERTa-base model, several cases were observed where the incorporation of PI led to an increment in the inconsistency metrics for both P and P+C methods, 4.5 over 15 test cases on average under the confidence level of 0.1 using a t-test. The exacerbation primarily occurred in downstream tasks having a small training set, namely, BoolQ, MRPC, and RTE. Nonetheless, it is noteworthy to highlight that this limitation is not significantly detrimental, given that fine-tuning the base-size model on a small training set does not entail substantial resource consumption. Conversely, the experimental results demonstrated different characteristics in the large-size model compared to those observed in the base-size model. For the RoBERTa-large model, a degradation in inconsistency metrics was observed in only 2 out of 15 test cases, on average, under the confidence level of 0.1 using a t-test. Notably, this degradation was considerably lower when compared to the RoBERTa-base model. Moreover, an average improvement of 5.5 out of 15 test cases was observed. This presents a notable contrast to RoBERTa-base, where no statistically significant enhancements were observed upon employing PI. The findings indicate that employing PI yields greater benefits for large-size models, as it not only significantly reduces the number of training parameters but also consistently generates higher or comparable levels of consistency compared to FT. This advantageous outcome aligns perfectly with the objective of incorporating PI.

Comparison with baseline methods. Finally, the performance of the proposed method was compared with three baseline approaches. The first approach is DictBERT [Yu et al., 2022], a model that incorporates the definition of infrequent words from the dictionary during the pre-training phase. The second approach involves paraphrased data augmentation (Sem-Aug), also recognised as adversarial training [Yoo and Qi, 2021]. This method is widely used to improve consistency by introducing supplementary paraphrased versions of the original data during fine-tuning. The last approach is consistency regularisation (Sem-CR) [Elazar et al., 2021]. This methodology trains a model to produce nearly identical predictive distributions for both the original and paraphrased data instances by incorporating a consistency regularisation term. Referring to prior works [Elazar et al., 2021, Kumar and Joshi, 2022], the loss

Model	BoolQ			MRPC				RTE			
	$\mathcal{A}$	τ_{sem}	τ_{neg}	$\mathcal{A}$	τ_{sem}	τ_{neg}	τ_{sym}	$\mathcal{A}$	τ_{sem}	τ_{neg}	τ_{sym}
PI (P)	85.2	9.6*	43.3	90.5	6.8*	77.3	4.7*	86.6	7.7	7.0*	2.2
PI (P+C)	85.6*	8.7*	37.8†	90.5	6.7†	77.0	4.4*	86.9	6.5	7.2*	**1.6***
PI (P+C+M)	**85.7***	8.3	**33.9†**	90.3	6.3	**69.3***	4.3*	**87.6***	6.5	**5.3***	1.8*
DictBERT	74.6	8.7	68.1	87.6	11.2	86.5	5.7	74.6	15.0	38.0	2.5
Sem-Aug	84.6	8.2	44.1	90.3	**5.3**	80.4	**1.2**	86.2	**6.3**	12.1	3.7
Sem-CR	84.3	**7.8**	67.6	**90.7**	5.8	78.7	2.7	83.3	7.1	13.8	2.8

Table 5.4: Performance comparison of the proposed approaches and baseline methods on BoolQ, MRPC, and RTE tasks. M denotes a RoBERTa-large model trained on the meaning-matching task. τ_{sem}, τ_{neg}, τ_{sym}, and τ_{trn} denote semantic, negational, symmetric, and transitive inconsistency, respectively. $\mathcal{A}$ refers to the validation accuracy. The best figure is in bold. The values of the proposed approaches show a significant difference compared to the best-performing baseline with p-value < 0.05 (*) and p-value < 0.1 (†) using a two-sample t-test.

function of the Sem-CR method is defined as follows (Eq. 5.7):

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda JS(o||p), \tag{5.7}$$

where $\mathcal{L}_{ce}$ is the cross-entropy loss, and JS refers to Jensen-Shannon divergence. o and p are the original and corresponding paraphrased data, respectively. Following the work of Elazar et al. [2021], the tuning of the hyperparameter ($\lambda \in 0.1, 0.5, 1$) was conducted to pick the best model. To generate paraphrased data for Sem-Aug and Sem-CR, the EMBEDDING method of TextAttack [Morris et al., 2020] was utilised, which involves replacing a certain portion of words (15% in this experiment) with their respective neighbours in the embedding space. Sem-Aug and Sem-CR were applied to RoBERTa-large. In order to verify the impact of additional knowledge add-up, the weights of a RoBERTa-large model trained on the meaning-matching task, described in Section 3.4.1, were incorporated with PLM and CRM. It was hypothesised that adding the parameters trained with different inductive biases could further enhance specific consistencies while maintaining the overall performance.

The experimental results are summarised in Tables 5.4 and 5.5. First, it was observed that the proposed approach outperforms DictBERT significantly in terms of both accuracy and consistency by a large margin. To address the possibility that the performance disparity could be attributed to the differences in the size of the backbone models, a comparison was made between the performance of DictBERT and RoB$_B$-PI-(P+C) in Table 5.3. Regarding accuracy, the latter displayed a superior performance compared to the former in all five downstream tasks, with statistical significance at a

Model	SNLI					WIC			
	$\mathcal{A}$	τ_{sem}	τ_{neg}	τ_{sym}	τ_{trn}	$\mathcal{A}$	τ_{sem}	τ_{sym}	τ_{trn}
PI (P)	**93.0**	8.6*	7.3*	7.3*	3.2*	71.9	11.7*	3.0	10.6
PI (P+C)	**93.0***	8.6*	**5.2***	**7.1***	3.0*	73.0	10.9*	**2.8**	9.5*
PI (P+C+M)	**93.0***	8.3*	**5.2***	7.2*	2.9*	**74.2***	8.6†	3.0	**8.9***
DictBERT	90.7	10.5	10.8	10.0	3.7	67.3	13.3	3.4	16.6
Sem-Aug	56.3	7.6	17.9	10.2	1.1	72.5	8.6	4.7	11.2
Sem-CR	55.5	**6.6**	16.9	11.8	**1.3**	69.9	**6.6**	3.8	11.6

Table 5.5: Performance comparison of the proposed approaches and baseline methods on SNLI and WiC tasks. M denotes a RoBERTa-large model trained on the meaning-matching task. τ_{sem}, τ_{neg}, τ_{sym}, and τ_{trn} denote semantic, negational, symmetric, and transitive inconsistency, respectively. $\mathcal{A}$ refers to the validation accuracy. The best figure is in bold. The values of the proposed approaches show a significant difference compared to the best-performing baseline with p-value < 0.05 (*) and p-value < 0.1 (†) using a two-sample t-test.

Tasks	$\Delta\tau_{sem}$	$\Delta\tau_{neg}$	$\Delta\tau_{sym}$	$\Delta\tau_{trn}$
BoolQ	+0.2	-23.6*	-	-
MRPC	-2.8*	-6.4*	-0.8*	-
RTE	-3.1*	-18.1*	-0.8*	-
SNLI	-0.5*	-4.7*	+0.1	0.0
WIC	-0.2	-	0.0	-3.4*

Table 5.6: The difference of consistency performance between the proposed approach applied to BERT-base and DictBERT reported in Table 5.5. $\Delta\tau_{sem}$, $\Delta\tau_{neg}$, $\Delta\tau_{sym}$, and $\Delta\tau_{trn}$ denote the difference of semantic, negational, symmetric, and transitive inconsistency between the proposed approach and DictBERT. The values below 0, highlighted in blue, indicate that the proposed approach exhibits more consistent behaviours compared to DictBERT, as lower inconsistency metrics signify better performance. The proposed methods show a statistically significant difference using a two-sample t-test with p-value < 0.05 (*) compared to DictBERT.

p-value < 0.05. In terms of consistency, the latter exhibited improved results in 9 out of 15 test cases, with a p-value < 0.05, while the former outperformed the latter in only 2 test cases. These results illustrate that the proposed approach can establish a more favourable inductive bias toward consistency and accuracy than DictBERT. To ensure a fairer comparison, I conducted an additional experiment between DictBERT and the proposed approach, applying to BERT-base, i.e., the same model, and using the same Wiktionary data containing approximately one million word-definition pairs. The inconsistency difference values between the proposed approach and DictBERT are reported in Table 5.6. The experimental results reveal that the proposed method

Premise: An older man is drinking orange juice at a restaurant.	
Original Hypothesis: A man is drinking juice. Label: Entailment	Paraphrased Hypothesis: A man is alcohol juice. Label: Entailment
Premise: A person on skis on a rail at night.	
Original Hypothesis: They are fantastic skiiers. Label: Neutral	Paraphrased Hypothesis: They are terrific skiiers. Label: Neutral
Premise: A woman in white in the foreground and a man slightly behind walking with a sign for John's Pizza and Gyro in the background.	
Original Hypothesis: A man is advertising for a restaurant. Label: Entailment	Paraphrased Hypothesis: A man is advertising for a lunchroom. Label: Entailment

Table 5.7: Examples of the imperfect paraphrased hypothesis of SNLI task generated by TextAttack.

produces a lower inconsistency compared to DictBERT under identical conditions, i.e., the same model and dictionary dataset. Specifically, the proposed approach significantly reduced the inconsistency in 10 out of 15 test cases while maintaining a comparable performance in the remaining test cases.

Secondly, it was verified that both Sem-Aug and Sem-CR achieved an enhanced semantic consistency compared to the proposed approach in most test cases, displaying statistical significance in 3 out of 5 test cases on average with a p-value < 0.1. Nevertheless, it is noteworthy that these approaches yielded considerably inferior results in other consistency types. These findings furnish evidence that substantiates the claim that data augmentation and consistency regularisation are confined in their ability to address solely a specific consistency type. In contrast, the proposed approach demonstrates the capability to enhance multiple consistency types simultaneously. Moreover, both the Sem-Aug and Sem-CR methods demonstrated a reduced accuracy across all downstream tasks and encountered a complete failure in the SNLI task, thereby offsetting their lower transitive inconsistency. I conjectured that a primary factor contributing to this phenomenon is the imperfect performance of the paraphrasing method, which can lead to confusion for models during fine-tuning. Several instances of such cases are presented in Table 5.7. Due to the significantly larger training set size in the SNLI dataset compared to the other downstream tasks, it is much more susceptible to being influenced by incorrectly paraphrased instances. The findings elucidate the drawback of data augmentation and consistency regulari-

	BoolQ	COPA	WiC	HellaSwag	SentiNeg
max-token-len	256	128	128	176	128
epochs	10	10	10	10	10
batch-size	16	16	16	8	16
lr	$5e^{-4}$	$5e^{-4}$	$5e^{-4}$	$5e^{-4}$	$5e^{-4}$
weight decay	0.1	0.1	0.1	0.1	0.1

Table 5.8: Hyperparameters used for training the PI models on KoBEST downstream tasks.

sation, as their efficacy relies heavily on the automatic tool for generating additional data instances. Although human-generated data can ensure a high data quality, its practical feasibility is limited due to the substantial resources required for employing human labour to collect vast amounts of data.

Finally, it was confirmed that the P+C+M model demonstrates a comparable or superior performance than the P+C model. Specifically, the P+C+M model exhibits statistically significant improvements in accuracy for 2 out of 5 tasks and achieves lower consistency in 5 out of 15 test cases compared to the P+C model under the confidence level of 0.1 using a t-test. These improvements were predominantly evident in the negational consistency, where the P+C+M model outperforms the P+C model in 3 out of 4 test cases. In the remaining test cases, no statistically significant differences are observed. These results validate the aforementioned hypothesis regarding the impact of incorporating additional parameters trained with distinct inductive bias.

5.3.3 Experiments on Korean Datasets

As previously mentioned, the primary advantage of the proposed approach lies in its flexible scalability to other low-resource languages. Consequently, the method was employed to assess its applicability to the Korean language. It is worth mentioning that the experiments did not include Korean baseline methods corresponding to DictBERT and consistency regularisation. This omission was due to the absence of a publicly available Korean DictBERT model, source code for training DictBERT, and a paraphrase generation tool in the Korean language at the time of the experiments.

5.3.3.1 Experiments on KoBEST Dataset

The proposed approach was evaluated on the KoBEST dataset [Kim et al., 2022], a recently introduced challenging benchmark known for its increased difficulty com-

Model			KB-BoolQ $\mathcal{A}_{test} \uparrow$	KB-COPA $\mathcal{A}_{test} \uparrow$	KB-WiC $\mathcal{A}_{test} \uparrow$	KB-HellaSwag $\mathcal{A}_{test} \uparrow$	KB-SentiNeg $\tau_{neg} \downarrow$
KoRoB$_B$	FT (111M)	P	78.2	60.4	79.1	**78.2**	8.5
		P+C	**78.7**	**65.3***	**81.3***	78.1	**7.4***
	PI (1.2M)	P	77.2	74.1	79.4	77.1	7.1
		P+C	**78.4**	**75.0**†	**80.7***	**77.4**	**6.5***
KoRoB$_L$	FT (338M)	P	86.5	75.7	85.1	83.4	5.5
		P+C	**86.9**	**82.5***	**86.3***	**83.6**	**4.6***
	PI (2.6M)	P	86.7	83.2	85.5	84.8	5.5
		P+C	**87.6***	**83.9**†	**85.7**	**85.1**	**4.5***

Table 5.9: Average KoBEST performance of KoRoBERTa-base (KoRoB$_B$) and KoRoBERTa-large (KoRoB$_L$). $\mathcal{A}_{test}$ and τ_{neg} represent the accuracy and negational consistency on the test set, respectively. The notation "↑" denotes that higher values are better, and "↓" stands for lower values are better. The best performance is in bold. The figures of the proposed approach (P+C) show a significant difference compared to our counterpart (P) with p-value < 0.05 (*) and p-value < 0.1 (†) using a two-sample t-test.

pared to the Korean-GLUE benchmark [Park et al., 2021]. The KoBEST dataset comprises five distinct downstream tasks, namely, Boolean Question (KB-BoolQ), Choice of Plausible Alternative (KB-COPA), Words in Context (KB-WiC), HellaSwag (KB-HellaSwag), and Sentiment Negation (KB-SentiNeg). Each task is intentionally designed to assess the model's proficiency in comprehending context information, causality, words' meanings, temporal sequences, and negation expressions, respectively. The SentiNeg task entails a sentiment analysis objective, but the test set comprises negated training sentences wherein the inclusion of negation expressions switches the polarity. As a result, this dataset can serve as the evaluation set for assessing negational consistency, which constitutes a key rationale for selecting this dataset for the experiment. Table 5.8 displays the hyperparameter values employed during training the PI models on KoBEST downstream tasks. For training the FT models, slight adjustments were made to the learning rate, with $1e^{-5}$ for the KB-WiC and KB-HellaSwag tasks and $5e^{-6}$ for the remaining tasks.

The experimental results are summarised in Table 5.9. In general, the results demonstrated a comparable trend to that observed in the English experiments. Primarily, the utilisation of the CRM weights positively impacted on both FT and PI scenarios, leading to statistically significant performance enhancements in an average of 3 out of 5 tasks compared to employing only the weight of the PLM. A subtle distinction was that, in GLUE benchmark experiments, the integration of the CRM

weights demonstrated a comparable performance to solely utilising the PLM weights in the case of a large-size model. However, in the KoBEST experiments, the former generally surpasses the latter in performance.

The prevailing factors behind these observations were presumed to be as follows: (1) A larger volume of training data was available for Korean CRM compared to its English counterpart, with the former being approximately four times higher, and (2) the KoBEST downstream tasks demand a higher level of language understanding ability, making the effects of CRM more pronounced in this context. Furthermore, although only one test case for examining negational consistency was available due to the absence of Korean benchmarks designed for assessing multiple consistency types, incorporating CRM weights led to a significant enhancement in negational consistency. This result is in line with the experimental results on English datasets. The experimental findings support that the proposed approach is highly adaptable and performs effectively in other low-resource languages beyond English.

5.4 Related Work

Adaptation technique. Adapter modules are devised to facilitate the efficient fine-tuning of PLMs by keeping pre-trained weights fixed and introducing only a small number of additional trainable parameters. This approach proves particularly advantageous when training multiple downstream tasks, as it enables training a limited number of new parameters per task without the need to re-update previously trained weights. Numerous previous studies have incorporated a low-rank residual adapter between layers, employing both down-projections and up-projections [Rebuffi et al., 2017, Houlsby et al., 2019, Lin et al., 2020b]. In contrast, Karimi Mahabadi et al. [2021] proposed the COMPACTER method, which utilises the Kronecker product for a low-rank parameterisation instead of the down- and up-projections used by the previous approaches. In recent work, Hu et al. [2022] introduced an approach named LoRA. A notable difference between LoRA and the previously mentioned methods lies in the fact that the trainable weights of the former are combined with pre-trained weights during the inference phase through addition, thereby avoiding an increase in latency. Conversely, this is not the case for the previous works, as they introduce additional adapter modules that contribute to increased latency.

Representation learning beyond the distributional hypothesis. Several studies have attempted to overcome the limitation of the distributional hypothesis by employing cognitive word theory to represent words. Word Sheriff [Parasca et al., 2016, Stuczyński et al., 2017] is a simple interactive word game designed as an effort to collect data that define a word using a brief sequence of words. The result is a pair consisting of a word and its closely related words, such as ("wasabi", {"japanese", "spice"}). However, the papers did not propose word representations trained with the collected data. Another line of research sought to train word vectors in a similar way to the proposed approach, i.e., by relating a word to words comprising of its definition. Dict2vec [Tissier et al., 2017] adopted the same training strategy as Word2vec [Mikolov et al., 2013] but utilised word-definition pairs from a monolingual dictionary rather than a general corpus. Eisner et al. [2016] trained the vector representations of emoji similarly by employing emojis and their descriptions. Shu et al. [2020] proposed an enhanced version of Dict2vec that incorporates the higher-order relations of words in a dictionary by transforming the dictionary into a relational graph based on the co-occurrence of the entries and words within definitions. Bosc and Vincent [2018] proposed a word embedding method distinct from the aforementioned approaches by leveraging an LSTM autoencoder. The core idea is to make the entry word representation similar to the autoencoder's hidden representation of its definition.

5.5 Discussion

Trustworthiness is a highly recommended property that a competent NLP model is expected to possess, particularly in risk-sensitive domains like the legal or medical field. However, recent studies have revealed that modern PLMs exhibit many mistakes that humans rarely commit, which renders them less reliable. One of the problematic aspects are the inconsistent behaviours displayed by PLMs. To address this concern, several studies have proposed remedies involving data augmentation and the design of special loss functions that penalise inconsistent behaviours. However, these methods have drawbacks, including limited applicability to specific consistency types, limited suitability for low-resource languages, and high computational demands when employed with large-scale PLMs.

To this end, this chapter introduces an approach to improve overall consistencies practically. The underlying hypothesis posits that a compelling reason for inconsistent behaviours lies in the deficiency of meaning-understanding ability. Therefore, an

endeavour was made to enhance PLMs' meaning awareness by incorporating insights from the conceptual role theory. Specifically, I designed a novel learning task, which utilises word-definition pairs in a dictionary to capture more precise interrelationships between concepts. The new model trained on the proposed task was named CRM. Subsequently, I proposed a parameter integration method using low-rank adaptation to combine the weights of CRM and PLMs in order to exploit both previously- and newly learned knowledge.

The experimental results revealed that CRM improves multiple consistency types concurrently, particularly in semantic and negational consistency. Furthermore, the integration of CRM led to an improved performance on COLA and WiC tasks, both of which necessitate linguistic knowledge and comprehension of textual meaning. These findings suggest evidence that CRM genuinely enhances the model's meaning awareness. Secondly, it was ascertained that the parameter integration technique significantly reduces the number of training parameters while delivering a comparable or superior performance compared to fine-tuning the entire set of parameters. This outcome was particularly pronounced for large-size PLMs, suggesting that the approach enables efficient training of large-size PLMs. Thirdly, the proposed approach outperformed previous methods by a large margin, which were widely used to improve consistency. Notably, unlike data augmentation and consistency regularisation methods that are only capable of addressing semantic consistency, it was confirmed that the proposed approach is able to enhance multiple types of consistency concurrently. Moreover, it was verified that further improvement across various consistency types can be achieved by integrating additional parameters of PLMs that share the same architecture but possess a distinctive inductive bias, which is highly resource-efficient, as it does not introduce any additional training or inference latency. Finally, the experiments conducted on the Korean language demonstrated that the proposed approach is readily applicable to low-resource languages beyond English. The experimental findings suggest that exploring human cognition and incorporating its principles into the learning objectives of language models could serve as a key pathway to achieving trustworthy language AI by fundamentally drawing their language understanding ability closer to human-level performance.

Chapter 6

Conclusion

This dissertation encompasses a comprehensive investigation into the inconsistent behaviours exhibited by modern PLMs and explores potential strategies to mitigate the issue, aiming to enhance their trustworthiness. This research has been driven by undesirable behaviours demonstrated by PLMs and constitutes a valuable contribution to the field of language models' reliability. The following sections summarize this research and discuss recommendations for future work.

6.1 Summary

Certification and explanation are two pivotal concepts that constitute trustworthiness. This dissertation was initiated by investigating the behaviours of PLMs with respect to these two properties to examine their trustworthiness.

To commence, from the perspective of certification, I carried out an investigation regarding PLMs' capacity to adhere to the logical negation property. According to this property, PLMs were expected to make distinct predictions for two inputs delivering opposite meanings. In this regard, I proposed three probing tasks to examine PLMs' understanding of negation expressions and antonyms. The experiments on the proposed probing tasks demonstrated that PLMs lack an understanding of negation expressions and antonyms and, hence, easily violate the logical negation property. They frequently generated identical predictions when presented with an input query and its negated version. They also retrieved synonyms or reproduced the given words in response to antonym-asking queries. Moreover, it was revealed that pre-trained weights of PLMs lack adequate knowledge of semantic antonymy. These findings provided evidence indicating that PLMs do not exhibit correct behaviours, thereby impeding the certification property. To address this concern, I introduced a mitigation method named IM^2 based on a linguistic theory called meaning-matching theory.

The approach involved fine-tuning PLMs on an intermediate task where the training objective was to predict whether a given word and its definition matched correctly. The experimental results verified that the proposed approach significantly enhances the performance of the three probing tasks and produces a superior performance on the NegNLI dataset compared to a previously proposed alleviation method, which is pre-trained on corpora containing negation expressions. These results imply that an improved comprehension of negation expressions and antonyms can be achieved by enabling the model to learn the correct word's meaning. Moreover, the approach preserved the performance of general NLU tasks in the GLUE benchmark, indicating that the meaning-matching task is a safe intermediate task.

Second, from an explanation perspective, I performed an investigation to explore the generation of inconsistent NLEs, indicating the phenomenon of producing contradictory NLEs under the same context. I introduced the method for enhancing adversarial attacks on explanations, named eKnowCA, to attain this objective. This approach efficiently leveraged external knowledge bases and demonstrated an improved time and performance efficiency compared to the previously proposed attack method called eCA. Moreover, experimental results showed that high-performing NLE generation models, typically based on PLMs, tend to produce contradictory NLEs, with a majority of them being counterfactual statements. To alleviate this issue, I proposed a method that extracts related background knowledge information and incorporates it for generating an NLE. The findings demonstrate that the occurrence of inconsistent NLEs was significantly reduced when appropriate knowledge triplets were extracted, whereas it remained susceptible when unrelated knowledge triplets were utilised. These results highlighted the importance of developing more accurate knowledge extraction engines, as they can substantially diminish the generation of contradictory NLEs. Also, the findings guide the direction of this dissertation towards the certification property, as the contradictions in NLEs can be resolved by incorporating external components, whereas the issues in the certification process are inherent to PLMs.

Third, I proposed the overall definition of language model consistency in this dissertation. Previous studies on consistency proposed distinct definitions, leading to multiple definitions of consistency. Consequently, experiments were conducted separately and individually, with a limited scope focused on each definition and investigating a few downstream tasks. Thus, I introduced a unified definition of language model consistency, drawing from the idea of behavioural consistency, a core characteristic of human behaviours. The definition encompasses two core components: *belief*

and *principle*. Based on these components, a taxonomy was formulated to categorise previous studies into three exclusive categories: semantic, logical, and factual consistency. The logical consistency was subsequently separated into four sub-categories: negational, symmetric, transitive, and additive consistency, which are defined based on the specific logical property used to delineate desirable behaviours.

Fourth, I introduced a unified benchmark suite called BECEL to investigate the consistent behaviours of PLMs comprehensively. In contrast to previous studies that focused on limited consistency types and a few downstream tasks, typically in the context of NLI tasks, this dataset comprises 19 test cases that encompass the examination of 5 consistency types across 7 downstream tasks. The test sets of the original task were modified to evaluate specific consistency types, such as paraphrasing a statement for semantic consistency and switching the order of two inputs for symmetric consistency. Next, all the generated instances underwent human annotation to ensure the preservation of high data quality, a prerequisite for an accurate evaluation.

Fifth, contemporary widely-used PLMs and LLMs were extensively evaluated on the proposed BECEL dataset, and several important insights were attained from the experimental results. Above all, none of the PLMs exhibited consistent behaviours across all test cases, regardless of their training objectives, architectures, and number of parameters. In contrast to previously conducted experimental results, where large-size PLMs typically exhibited a superior performance in various NLU downstream tasks, it was confirmed that large-sized PLMs do not necessarily show a greater consistency compared to their corresponding base-sized counterparts. Despite their exceptional performance in human-professional examinations, even LLMs demonstrated many inconsistent behaviours, worse than fine-tuned PLMs in certain consistency types and downstream tasks. The findings indicate that scaling up the size of LMs, which involves substantial financial and environmental costs, does not fundamentally resolve the issue of inconsistent behaviour. Furthermore, I verified that the impact of data characteristics surpasses that of the model's architecture, highlighting the critical role played by data in forging the model's inductive bias. Finally, the experimental results furnished evidence regarding the limitations of alleviation methods, such as data augmentation and prompt engineering. These findings highlight the need for sustainable and technically feasible methods that are capable of fundamentally improving LMs' consistent behaviours.

Sixth, I proposed an alleviation method to address multiple types of consistency simultaneously. Based on the conceptual role theory, the approach first attempts

to capture precise interconnections between concepts by leveraging word-definition pairs available in a dictionary. The model produced through this process was named CRM. Due to the ubiquitous presence of dictionaries in any language, the approach holds a practical advantage, because it can be readily applied to low-resource languages. Subsequently, the parameters of CRM were integrated with those of PLM by using a low-rank adaptation technique, thereby leveraging the knowledge information existing in both CRM and PLM. The parameter integration method offered practical advantages, as it does not introduce any increase in latency during the training and inference phases, making it particularly efficient for large-size LMs. The experimental results obtained from English and Korean datasets substantiated all of these advantages.

6.2 Outlook

Like any other research, the work demonstrated in this dissertation has room for improvement.

First, the NLP tasks, which were leveraged to assess consistent behaviours of LMs in the thesis, focused on relatively short texts, which normally are, at most, 1,000 tokens. However, LMs often encounter much longer documents in real-world applications, such as legal contracts or medical prescriptions. Given that the performance of LMs is significantly influenced by the text length, typically exhibiting inferior performance in longer texts, LMs' consistent behaviours in longer documents need to be examined. QA would be a suitable NLP task for this purpose, considering the ease of collecting long documents, as there is no need for manual modification; these documents can be directly used as a context. Also, we can explore multiple consistency types by generating paraphrased questions, negating the questions, and creating questions using transitive inference.[1]

Second, despite notable improvements in consistency achieved through the mitigation method outlined in Chapter 5, the approach still exhibited some mistakes. I postulated that a primary contributing factor is the limitation of the dictionary data. Although dictionary data offer interrelationships between closely related concepts more effectively than the general corpus, they still omit a significant portion of

[1]For instance, the question and answer pairs could be as follows: (Q1: Do carnivores eat meat? A1: Yes), (Q2: What is the primary food source for tigers? A2: Meat), and (Q3: Are tigers carnivores? A3: Yes). It is important to note that all the information required to answer Q1 and Q2 should be provided in the context document.

these relationships, as it is nearly impossible to encompass the entire breadth of conceptual spaces. For instance, in the dictionary data I used, the definition of "water" includes concepts such as "clear liquid", "rain", "sea", "lake", and "swimming pool", but lacks concepts related to its usage, e.g., "extinguish fire" or "coffee". Hence, while dictionary data offer an advantage in scalability due to the ease of data collection, further exploration is needed to ascertain whether employing data that encompass a broader conceptual space can enhance consistent behaviours. Word Sheriff [Parasca et al., 2016, Stuczyński et al., 2017] would be an appropriate tool for this purpose, as it facilitates the cost-efficient collection of word-definition pairs that reflect the conceptual mental space of humans.

Third, it was not able to compare the performance of the proposed methodology in Section 5 with baseline approaches in the Korean language due to the lack of available Korean language resources, such as paraphrasing tools for consistency regularisation. However, with the recent availability of more elaborate multilingual LLMs, it is now feasible to utilise such models for generating Korean paraphrases. Hence, the future direction will involve utilising such models to broaden the scope of experiments on the Korean language and validate the proposed approach's advantage over the baseline methods.

Finally, this study adopted the definition of trustworthiness proposed by Huang et al. [2020] and, therefore, examined the trustworthiness of LMs mainly from the perspectives of consistent behaviours, constituting the certification process. However, many other factors, such as interpretability and confidence calibration, contribute to trustworthiness.

I aim to measure the consistency behaviours regarding confidence calibration as the first step of future research. The consistency evaluation metrics defined in this thesis are computed based on discrete predictions, such as "entailment" and "not equivalent", and thus cannot measure the differences in confidence scores. For instance, regarding symmetric consistency, consider that model A produces exactly the same predictions and confidence scores for instances where the order of the two input texts is switched. In contrast, suppose model B produces identical predictions but with differing confidence scores. Then, the former is likely to be deemed to exhibit more consistent behaviours. Creating new evaluation metrics incorporating predictions and confidence scores should be prioritised to achieve this purpose. Additionally, "trust" is a social phenomenon that needs to be studied in line with quantitative experiments and qualitative social experiments, where the latter is not conducted in this dissertation. Future research will also expand the scope of the investigation to

incorporate human-involved social experiments to more precisely assess the trustworthiness of LMs, which will require the development of an efficient tool for cost-efficient human-involved evaluation.

Appendix A

Exploration of Undesirable Behaviours Exhibited by Pre-trained Language Models

A.1 WT5-base Performance Replication

Table A.1 summarises the performance of WT5-base trained for our experiment. The accuracy of the e-SNLI and Cos-E datasets is reported. Four automatic evaluation metrics, BLEU [Papineni et al., 2002], ROUGE [Lin, 2004], Meteor [Banerjee and Lavie, 2005] and BERT score [Zhang et al., 2020b], were employed to measure the quality of generated NLEs. Regarding accuracy and BLEU, the replicated model performed better than the originally reported results in Cos-E but produced a slightly lower performance in the e-SNLI dataset.

		Acc.	BLEU	R-1	R-2	R-L	Meteor	BERT-S
e-SNLI	replicated	90.6	28.4	45.8	22.5	40.6	33.7	89.8
	reported	90.9	32.4	-	-	-	-	-
Cos-E	replicated	65.3	7.3	25.0	8.3	21.6	20.2	86.3
	reported	59.4	4.6	-	-	-	-	-

Table A.1: Performance of the replicated WT5-base on e-SNLI and Cos-E. The notations R-1, R-2, R-L, and BERT-S denote ROUGE-1, ROUGE-2, ROUGE-L score, and BERT-Score, respectively.

A.2 Naturalness Evaluation of the Generated Variable Parts

It could be unfair to attribute the generation of contradictory NLEs to a model if the reverse variable parts are deemed unnatural. Hence, manual evaluation was conducted to ascertain unnatural statements.

Regarding the e-SNLI dataset, following the work of Camburu et al. [2020], 50 random samples of generated reverse variable parts for each model were reviewed. Statements that contradict sense were regarded as unnatural. Minor grammatical errors and typos were ignored. It was revealed that, on average, 81.5% ($\pm$ 1.91) of the reverse variable parts were natural statements. The specific individual statistics for each model were as follows: NILE (80%), WT5-base (84%), KnowNILE (80%), and KnowWT5-base (82%). These percentages were multiplied by the number of detected contradictory NLEs only to consider contradictions caused by natural variable parts.

In the case of the Cos-E dataset, the variable parts (i.e., the two incorrect answer choices) were considered unnatural if they met either of the following criteria: (1) the answer choices belong to stopwords listed in the NLTK package or (2) the correct answer was repeated. The investigation can be carried out automatically, and it revealed only one instance of unnatural variable parts for KnowWT5 and WT5, respectively, while none were found for the other two models. The two cases were eliminated from the counts.

A.3 Design of Human Evaluation Process for Assessing NLE Quality

The human evaluation was performed using Amazon MTurk. For each model, 200 generated NLEs were randomly sampled for the evaluation. Three Anglophone annotators, with a Lifetime HIT acceptance rate of at least 98% and an accepted number of hits greater than 1,000, were employed per instance. Each annotator was asked to evaluate whether the generated NLEs justify the answer by selecting from four options: {*no*, *weak no*, *weak yes*, *yes*}. The e-ViL score [Kayser et al., 2021] was calculated by mapping these options to corresponding values of $\{0, 1/3, 2/3, 1\}$, respectively.

However, it has been widely acknowledged that the reliability of MTurk annotation cannot be assured even for Master workers [Rouse, 2019]. This concern was also evident during the human evaluation process, where a significant number of workers

carried out annotations without sufficient consideration, often simply checking "*yes*" in most cases. The inter-annotator agreement of the initial evaluation, as measured by Fleiss's Kappa ($\mathcal{K}$), averaged only 0.06 for Cos-E, which cast doubts regarding the quality of the evaluation.

I introduced a quality control measure into the evaluation framework to mitigate the concern above. This measure involved the careful collection of *trusted examples* where the quality of the NLEs was obviously "*yes*" or "*no*". In each HIT, consisting of a set of 10 examples, two trusted examples with correct answers indicated as "*yes*" and "*no*" were inserted at random locations. Following the annotation process, HITs were discarded provided the annotators provided an incorrect answer for any of the trusted examples. An annotator's response was deemed correct if they answered "weak yes" for a trusted example labelled as "yes" and "weak no" for a trusted example labelled as "no". This process was repeated until the proportion of rejected HITs accounted for less than 15% of the total HITs. As a result, an increased $\mathcal{K}$ value was achieved, 0.46 and 0.34 for e-SNLI and Cos-E from 0.35 and 0.06, respectively. Similar levels of $\mathcal{K}$ were also obtained in other studies [Marasovic et al., 2022, Yordanov et al., 2022].

A.4 Examples

Relation	**Negated Query**	**Wrong Predictions**
IsA	Truth isn't a [MASK].	["fact", "statement", "concept"]
CapableOf	A doctor cannot [MASK] you.	["care"]
PartOf	England isn't part of the [MASK].	["Europe"]
HasA	Airports don't have [MASK].	["windows", "runways"]
UsedFor	A map isn't for [MASK].	["navigate", "locating", "information"]
MadeOf	Air doesn't have [MASK].	["molecules"]
NotDesires	Soldier does want to be [MASK].	["die"]

Table A.2: ConceptNet relations for constructing the MKR-NQ dataset and their corresponding sample data points.

Word	Definition
abnormal	not normal; not typical or usual or regular or conforming to a norm; out of ordinary; unusual
afebrile	having no fever
barefaced	with no effort to conceal
career	the particular occupation for which you are trained; a job or occupation that a person does for an extended period
cargo	goods carried by a large vehicle
revise	the act of rewriting something; to review, alter and amend, especially of written material
salary	something that remunerates; a determined yearly amount of money paid to an employee by an employer during a job

Table A.3: Examples of word-definition pairs that are used for the meaning-matching task.

Template	Query	Wrong Predictions
X is a synonym of Y	boy is a synonym of [MASK].	[“sister”, “girl”]
X is an antonym of Y	boy is an antonym of [MASK].	[“boys”, “man”, “lad”, . . .]
X is another form of Y	learning is another form of [MASK].	[“forgetting”, “teaching”]
X is the opposite of Y	learning is the opposite of [MASK].	[“knowledge”, “erudition”, . . .]
X is a rephrasing of Y	speaker is a rephrasing of [MASK].	[“microphone”, “listener”, . . .]
X is different from Y	speaker is different from [MASK].	[“loudspeaker”, “talker”, ...]

Table A.4: Templates used to construct the MWR dataset and their sample data points.

Subject	Relation	Object
men	Antonym	humans
man	Antonym	person
woman	Antonym	person
people	Antonym	person
person	Antonym	people
flower	DistinctFrom	plant
politician	Antonym	man
children	Antonym	people

Table A.5: List of filtered noisy triplets in ConceptNet.

Original Explanation	Generated Contradictory Explanation
Not all men are teaching science.	Not all men are teaching biology.
A dog is not a car.	A dog is not a bike.
A child is not a man.	A child is not a wife.
A bird is not a squirrel.	A bird is not a moose.
A group of dogs is not a woman.	A group of dogs is not a person.

Table A.6: Examples where both the original and generated contradictory NLEs contain negation expressions. These NLEs are not contradictory with each other.

Premise: Two hussars sit perched on horses, dressed in extravagant ceremonial wear, each holding a sabre in their right hand, reigns to the horse in their left.	
Hypothesis: There are professional riders at a ceremony.	Hypothesis: Two amateur riders are riding horses.
Predicted Label: Entailment	Predicted Label: Entailment
Explanation: Hussars are professional riders.	Explanation: Hussars are amateur riders.
Premise: A cheerleader in a tight red and white uniform is passing out white t-shirts at a sporting event.	
Hypothesis: A player passes out hotdogs.	Hypothesis: A player is passing out shirts.
Predicted Label: Contradiction	Predicted Label: Entailment
Explanation: A cheerleader is not a player.	Explanation: A cheerleader is a player.
Premise: Two people using a water buffalo to cultivate a watery field.	
Hypothesis: Two people are outside with animals.	Hypothesis: Two people are using a plant.
Predicted Label: Entailment	Predicted Label: Entailment
Explanation: A water buffalo is an animal.	Explanation: A water buffalo is a plant.
Question: Crabs live in what sort of environment?	
Choices: bodies of water, saltwater, galapagos	Choices: bodies of earth, saltwater, atlantic ocean
Predicted Answer: bodies of water	Predicted Answer: bodies of earth
Explanation: Crabs live in bodies of water.	Explanation: Crabs live in bodies of earth.
Question: The piece of paper was worth a lot of money, it was an old Apple Inc what?	
Choices: stock certificate, copy machine, ream	Choices: stock certificate, piece of stone, book
Predicted Answer: stock certificate	Predicted Answer: stock certificate
Explanation: A stock certificate is the only thing that is not a piece of paper.	Explanation: A stock certificate is the only thing that is a piece of paper.
Question: When a person admits his mistakes, what are they doing?	
Choices: act responsibly, learn to swim, feel relieved	Choices: act responsibly, think critically, act irresponsibly
Predicted Answer: act responsibly	Predicted Answer: act irresponsibly
Explanation: when a person admits his mistakes, they act responsibly.	Explanation: when a person admits his mistakes, they act irresponsibly.

Table A.7: Examples of contradictory NLEs detected by eKnowIA for the WT5-base model on e-SNLI and Cos-E. The first column shows the original variable part and the second column shows the adversarial one.

Premise: These are young adults who seem to be working together to protect the plants surrounding the white pole.	
Hypothesis: The adults are young.	Hypothesis: The young adults are underdeveloped.
Predicted Label: Entailment	Predicted Label: Entailment
Explanation: Young adults are young.	Explanation: Young adults are seniors.
Premise: A young boy with close-cropped hair, wearing a red robe, is holding a black kettle as someone is about to pour something in it.	
Hypothesis: a small boy is holding a kettle that will soon be filled	Hypothesis: The small boy is not underdeveloped.
Predicted Label: Entailment	Predicted Label: Entailment
Explanation: A young boy is a small boy.	Explanation: A young boy is a small female.
Premise: A group of people in a room, most with computers, some raising their hands.	
Hypothesis: Some people have questions.	Hypothesis: A group of people are asking questions.
Predicted Label: Neutral	Predicted Label: Contradiction
Explanation: People raising their hands are not asking questions.	textscExplanation: People raising their hands are asking questions.

Table A.8: Examples of contradictory NLEs detected by eKnowIA attack for the NILE model on e-SNLI. The first column shows the original hypothesis, and the second one shows the adversarial hypothesis from our attack.

Question: A good interview after applying for a job may cause you to feel what?	
Choices: hope, income, offer	Choices: hope, resentment, fear
Predicted Answer: hope	Reverse Answer: hope
Explanation: hope is the only thing that would cause you to feel hope.	Explanation: hope is the only thing that would cause you to feel fear.
Question: What does a stove do to the place that it's in?	
Choices: warm room, brown meat, gas or electric	Choices: warm room, cook food, heat the outside
Predicted Answer: warm room	Reverse Answer: heat the outside
Explanation: a stove heats the room.	Explanation: a stove heats the outside.

Table A.9: Examples of contradictory NLEs detected by eKnowIA attack for the CAGE model on Cos-E. The first column shows the original hypothesis, and the second column shows the adversarial hypothesis from our attack.

Model: KnowWT5	
Premise: A young family enjoys feeling ocean waves lap at their feet.	
Original	Adversarial
Hypothesis: A family is at the beach. Predicted Label: entailment Explanation: Ocean waves lap at their feet implies that they are at the beach.	Hypothesis: The family is not at the beach. Predicted Label: entailment Explanation: Ocean waves lap at their feet implies that they are not at the beach. Extracted Knowledge: {feet, HasA, fingerprints}
Model: KnowNILE	
Premise: Some dogs are running on a deserted beach.	
Original	Adversarial
Hypothesis: There are multiple dogs present. Predicted Label: entailment Explanation: Some dogs implies multiple dogs.	Hypothesis: There are not many dogs on the beach. Predicted Label: entailment Explanation: Some dogs implies not many dogs. Extracted Knowledge: {dog, DistinctFrom, cat}
Model: KnowCAGE	
Question: What does the sky do before a rain?	
Choices: cloud over, rain water, appear blue Predicted Label: appear blue Explanation: the sky appears blue before a rain	Choices: cloud over, appear blue, appear green Predicted Label: appear green Explanation: the sky appears green before a rain Extracted Knowledge: {sky, UsedFor, rain}

Table A.10: Examples of contradictory NLEs detected by eKnowIA but not defended by Know-models. The extracted knowledge triplets are not highly related to generating correct explanations.

Model: KnowWT5	
Premise: The collie is standing outdoors on a sandy area.	
Original	Adversarial
Hypothesis: The collie is standing in the sand.	Hypothesis: The collie is standing on stone.
Predicted Label: entailment	Predicted Label: entailment
Explanation: A sandy area is made of sand.	Explanation: A sandy area is made of stone.
	Extracted Knowledge: {sand, RelatedTo, rock}
Model: KnowNILE	
Premise: Coach talks with football player, other players and crowd in background.	
Original	Adversarial
Hypothesis: A football player is climbing into the stands at a game.	Hypothesis: A football player talks to a crowd.
Predicted Label: contradiction	Predicted Label: entailment
Explanation: A coach is not a football player.	Explanation: A coach is a football player.
	Extracted Knowledge: {crowd, IsA, gathering}, {player, PartOf, team}, {football player, DerivedFrom, football}

Table A.11: Examples of newly detected instances with contradictory NLEs by eKnowIA for the KNOW models. The extracted knowledge triplets exhibit low relevance and confuse the model to generate incorrect explanations.

Premise: A dog standing near snow looking at water.	
Hypothesis: A bird is standing near snow.	Hypothesis: A bird is near water.
Predicted Label: contradiction	Predicted Label: entailment
Explanation: A dog is not a bird.	Explanation: A dog is a bird.
Question: What is a person who is good at sports considered?	
Choices: talented, affluent, reproduce	Choices: talented, untalented, good at
Predicted Label: talented	Predicted Label: untalented
Explanation: a person who is good at sports is considered talented.	Explanation: a person who is good at sports is considered untalented

Table A.12: Examples of contradictory NLEs detected by eKnowIA for NILE on e-SNLI and WT5 on Cos-E. The first column shows the original variable part, and the second column shows the adversarial one.

Premise: A dog standing near snow looking at water. Hypothesis: A bird is near water.	
Hypothesis: A bird is standing near snow. Predicted Label: contradiction Explanation: A dog is not a bird. Extracted Knowledge: {snow, RelatedTo, water}, {dog, DistinctFrom, cat}	Hypothesis: A bird is near water. Predicted Label: neutral Explanation: A dog looking at water does not imply a bird is near water. Extracted Knowledge: {snow, RelatedTo, water}, {dog, DistinctFrom, cat}
Question: What is a person who is good at sports considered? Choices: talented, untalented, good at	
Choices: talented, affluent, reproduce Predicted Label: talented Explanation: a person who is good at sports is considered talented. Extracted Knowledge: {talent, RelatedTo, sports}	Choices: talented, untalented, good at Predicted Label: talented Explanation: a person who is good at sports is considered talented. Extracted Knowledge: {talent, RelatedTo, sports}

Table A.13: Examples of successfully defended instances by the KnowNILE model on e-SNLI and by the KnowWT5 model on Cos-E. This table should be read together with Table A.12 to appreciate the defence.

Appendix B

Benchmark for Consistency Evaluation of Language Models

B.1 Task Descriptions

BoolQ [Clark et al., 2019] is a dataset for machine reading comprehension (MRC) involving yes/no questions. Each data point consists of a triplet comprising a question, a passage, and an answer that requires a broad range of inference capabilities to solve questions.

SNLI [Bowman et al., 2015] and RTE (recognising textual entailment) [Candela-Quinonero et al., 2006] are datasets for NLI. Each data point comprises two sentences, premise and hypothesis, and a label stating the relationship between the premise and hypothesis (i.e., "entailment", "neutral", and "contradiction").

MRPC (Microsoft Research Paraphrase Corpus) [Dolan and Brockett, 2005] is a dataset for STS. Each data point consists of two sentences and a label indicating whether the two paraphrased sentences are semantically identical.

SST-2 (Stanford Sentiment Treebank) [Socher et al., 2013] is a dataset for SA. Each data point comprises a phrase and a binary sentiment label, i.e., positive and negative.

AG-News [Zhang et al., 2015] is a dataset of new articles for TC. Each data point is composed of a title and a description of an article, and the corresponding label indicating one of the four topics of the article (i.e., "World", "Sports", "Business", and "Sci/Tech").

WiC (Word-in-Context) [Pilehvar and Camacho-Collados, 2019] is a dataset for identifying the intended meaning of words. Each data point consists of two sentences containing the same target word and a label indicating whether the target word is used with the same meaning in two different contexts.

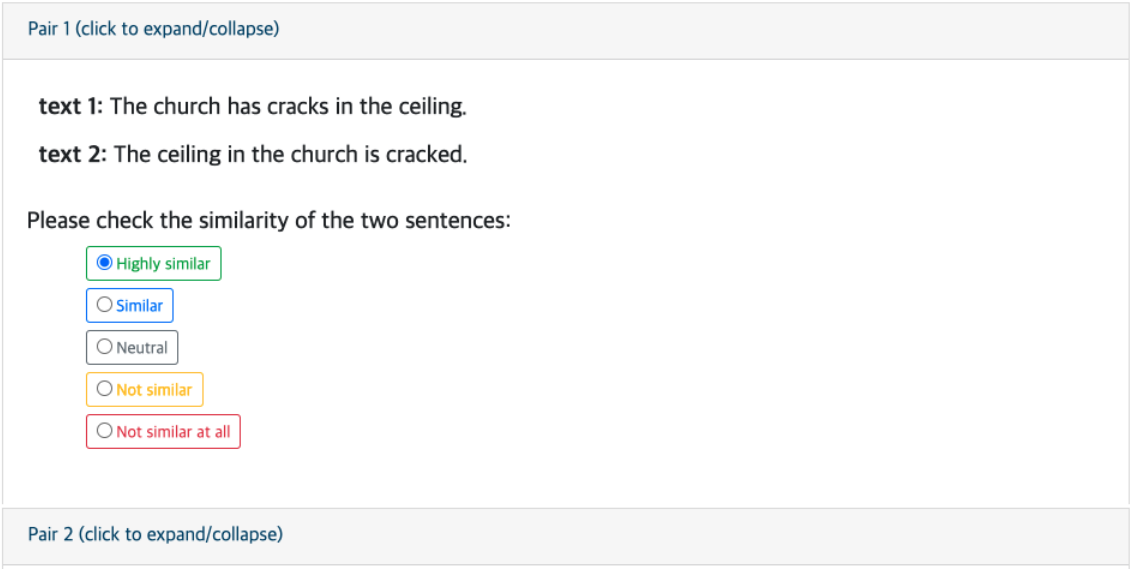


Figure B.1: Snapshot of the human annotation UI for annotating semantic consistency evaluation data.

B.2 Human Annotation Process

Amazon Mechanical Turk (https://www. mturk.com/) was used for the human annotation process. Anglophone annotators who met the criteria of a minimum acceptance rate of 98% and completed more than 1,000 HITs were employed for this process. Figure B.1 illustrates a snapshot of the UI for the human annotation process for collecting semantic consistency evaluation datasets.

B.3 Applicability of Logical Consistencies to Downstream Tasks

Negational Consistency Applicability. Among the downstream tasks in the BECEL dataset, negational consistency does not hold for the TC and WiC tasks. Regarding the TC task, negated sentences belong to the same category as their original version, as illustrated in the example in Section 4.2.2. Likewise, in the WiC task, the labels are preserved since the meaning of the target word remains unchanged in the perturbed sentence.

Although negational consistency theoretically applies to the SA task, it is excluded from the evaluation scope for a practical reason. It was observed that the method for generating opposite-meaning sentences does not perform effectively on the spoken language (e.g., movie reviews) that constitute the SST2 dataset.

Symmetric Consistency Applicability. The following two conditions are necessary for symmetry to hold.

Condition 1. The input should consists of two sentences.

Condition 2. The hierarchy between the two sentences should be equivalent.

The TC and SA tasks fail to satisfy the first condition. In the case of the MRC task (i.e., BoolQ), the question's dependency on the passage leads to a violation of the second condition. Consequently, these three tasks were excluded from the evaluation scope for assessing symmetric consistency.

Transitive Consistency Applicability. Theoretically, transitive consistency is applicable to downstream tasks where symmetric consistency is valid. However, it requires one additional condition for practical reasons: the two data points must share a common sentence, for instance, the same hypothesis in NLI tasks. Among the downstream tasks in the BECEL dataset, only the SNLI and WiC tasks satisfy this condition. Although it is possible to create new data for the MRPC and RTE tasks to meet the condition, it can introduce a distribution shift issue, which might exaggerate the inconsistency problem. Therefore, the transitive consistency evaluation was performed only on the SNLI and WiC tasks.

Additive Consistency Applicability. Additive consistency remains valid for tasks that take a single sentence as input. However, it is not ensured when a downstream task requires more than two sentences as input. Table B.1 illustrates an example of the violation in the SNLI task. Thus, additive consistency was tested exclusively for the SA and TC tasks.

B.4 Validation Performance of the BECEL downstream tasks

Table B.2 displays the validation performance of fine-tuned PLMs across seven BECEL downstream tasks. In the case of the WiC task, the training performance is also included in the report due to the substantial disparity observed between the training and validation performance compared to the other downstream tasks. The average results from five repetitions for each PLM are reported.

Example 1 Premise: Two women are embracing while holding to go packages. Hypothesis: Two woman are holding packages. Label: entailment **Example 2** Premise: Two men on bicycles competing in a race. Hypothesis: People are riding bikes. Label: entailment
Merged Example Premise: Two women are embracing while holding to go packages. Two men on bicycles competing in a race. Hypothesis: Two woman are holding packages. People are riding bikes.

Table B.1: Example of the SNLI data where additive consistency does not hold. The label of the merged example cannot be "entailment" because the two women are not riding bikes.

B.5 Examples

	Test case	Predicted	Pass?
Testing **Semantic Consistency** on the **TC** task.		Labels: World, Sports, Business, Sci/tech	
Original	UN's Global Fund meets African leaders in Tanzania for talks on fighting the world's deadliest diseases.	World	o
New	The United Nations Global Fund meets African leaders in Tanzania to discuss combating the world's deadliest diseases.	World	
...			
Testing **Negational Consistency** on the **NLI** task.		Labels: entailment, neutral, contradiction	
Original	Premise: The man in the blue shirt is relaxing on the rocks. Hypothesis: A man is wearing a blue shirt.	entailment	x
New	Premise: The man in the blue shirt is relaxing on the rocks. Hypothesis: A man is not wearing a blue shirt.	entailment	
...			
Testing **Symmetric Consistency** on the **STS** task.		Labels: equivalent, not_equivalent	
Original	S1: Zuccarini was ordered held without bail Wednesday by a federal judge in Fort Lauderdale, Fla. S2: A federal magistrate in Fort Lauderdale ordered him held without bail.	equivalent	o
New	S1: A federal magistrate in Fort Lauderdale ordered him held without bail. S2: Zuccarini was ordered held without bail Wednesday by a federal judge in Fort Lauderdale, Fla.	equivalent	
...			

Figure B.2: Data examples of semantic, negational, and symmetric consistency evaluation.

Model		BoolQ	MRPC	RTE	SNLI	SST2	AG-News	WiC	
		$\mathcal{F}_{val}$	$\mathcal{F}_{val}$	$\mathcal{F}_{val}$	$\mathcal{F}_{val}$	$\mathcal{F}_{val}$	$\mathcal{F}_{val}$	$\mathcal{F}_{tr}$	$\mathcal{F}_{val}$
BERT	base	66.6 (1.0)	81.8 (1.9)	62.1 (2.0)	90.1 (0.2)	90.5 (0.3)	93.2 (0.2)	62.5 (13.4)	53.0 (7.4)
	large	70.3 (1.4)	82.0 (1.4)	64.4 (3.3)	91.0 (0.2)	92.4 (0.4)	93.9 (0.2)	70.0 (9.0)	57.7 (2.9)
RoBERTa	base	75.8 (1.0)	86.4 (0.9)	71.5 (1.8)	91.5 (0.0)	92.9 (0.4)	94.1 (0.1)	78.6 (3.9)	63.8 (1.8)
	large	84.9 (0.4)	88.6 (1.0)	81.5 (2.1)	93.0 (0.1)	95.9 (0.3)	94.3 (0.2)	77.0 (6.2)	66.0 (2.0)
Electra	base	73.8 (2.6)	88.3 (0.2)	75.3 (2.5)	91.8 (0.1)	93.9 (0.2)	93.2 (0.2)	80.4 (4.3)	66.5 (5.9)
	large	87.1 (0.5)	90.1 (0.6)	86.7 (1.2)	93.5 (0.3)	95.4 (2.2)	93.8 (0.4)	80.7 (1.8)	69.0 (1.2)
ERNIE2.0	base	76.2 (1.2)	86.8 (0.9)	73.4 (2.6)	91.1 (0.2)	93.6 (0.2)	93.5 (0.1)	62.8 (13.1)	56.0 (4.6)
	large	82.6 (0.7)	86.6 (1.2)	76.7 (1.1)	92.1 (0.0)	95.1 (0.2)	94.0 (0.2)	67.1 (9.5)	55.2 (10.1)
GPT2	base	62.9 (1.7)	77.3 (1.0)	65.3 (2.5)	84.7 (1.1)	90.9 (0.5)	92.9 (0.1)	73.4 (3.8)	63.5 (2.6)
	large	75.3 (0.8)	80.4 (1.2)	69.0 (2.8)	90.8 (0.2)	94.1 (0.4)	94.1 (0.2)	87.4 (5.7)	64.5 (2.2)
BART	base	64.8 (2.4)	85.6 (1.1)	70.7 (1.0)	90.8 (0.2)	93.0 (0.3)	93.8 (0.3)	78.8 (5.5)	56.0 (1.7)
	large	78.4 (3.9)	81.6 (8.3)	74.9 (3.1)	93.1 (0.1)	95.9 (0.2)	94.0 (0.7)	77.3 (3.5)	58.9 (2.9)
T5	base	79.9 (0.2)	86.8 (0.9)	77.6 (0.2)	90.1 (0.1)	94.0 (0.2)	92.1 (0.2)	82.3 (0.3)	64.5 (1.1)
	large	83.8 (0.6)	89.3 (0.9)	88.0 (0.6)	92.1 (0.2)	95.8 (0.1)	92.5 (0.4)	84.6 (1.5)	70.3 (0.8)

Table B.2: The validation performance of the PLMs on the BECEL downstream tasks; $\mathcal{F}_{tr}$ and $\mathcal{F}_{val}$ denote F1 score on the training and validation set, respectively. The values written in parenthesis denote a standard deviation.

	Test case	Predicted	Pass?
Testing **Transitive Consistency** on the **WiC** task.		Labels: True, False	
Original 1	Word: back Sentence1: The horse refuses to back. Sentence2: The wind backed.	True	X
Original 2	Word: back Sentence1: The wind backed. Sentence2: The train backed into the station.	True	
New	Word: back Sentence1: The horse refuses to back. Sentence2: The train backed into the station.	False	
...			
Testing **Additive Consistency** on the **SA** task.		Labels: negative, positive	
Original 1	Unflinchingly bleak and desperate.	negative	X
Original 2	A sometimes tedious flim.	negative	
New	Unflinchingly bleak and desperate. A sometimes tedious flim.	positive	
...			

Figure B.3: Data examples of transitive and additive consistency evaluation.

SNLI	
Format	{*s1*} Question: {s2} True, False or Neither? Answer:
Example	A land rover is being driven across a river. Question: A Land Rover is splashing water as it crosses a river. True, False, or Neither? Answer:
RTE	
Format	{*s1*} Question: {s2} True or False? Answer:
Example	The harvest of sea-weeds is not allowed in the Puget Sound because of marine vegetation's vital role in providing habitat to important species. Question: Marine vegetation is harvested. True or False? Answer:
MRPC	
Format	Sentence 1: {*s1*} Sentence 2: {s2} Question: Do both sentences mean the same thing? Answer:
Example	Sentence 1: The increase reflects lower credit losses and favorable interest rates. Sentence 2: The gain came as a result of fewer credit losses and lower interest rates. Question: Do both sentences mean the same thing? Answer:
WiC	
Format	Sentence 1: {*s1*} Sentence 2: {s2} Question: Is the word {*word*} used in the same way in the two sentences above? Answer:
Example	Sentence 1: It was the deliberation of his act that was insulting. Sentence 2: It was the deliberation of his act that was insulting. The deliberations of the jury. Question: Is the word deliberation used in the same way in the two sentences above? Answer:

Table B.3: Eleuther AI prompt formats and their example used in the experiments for each downstream task.

SNLI	
Format	If {*s1*}, can we conclude {s2}? \n Yes, It's impossible to say, No \n Answer:
Example	If A land rover is being driven across a river, can we conclude A Land Rover is splashing water as it crosses a river? \n Yes, It's impossible to say, No \n Answer:
RTE	
Format	{*s1*} \n Based on the paragraph above can we conclude that {s2}? Yes, No \n Answer:
Example	The harvest of sea-weeds is not allowed in the Puget Sound because of marine vegetation's vital role in providing habitat to important species. \n Based on the paragraph above can we conclude that Marine vegetation is harvested? \n Yes, No \n Answer:
MRPC	
Format	Here are two sentences: \n {*s1*} \n {s2} \n Do they have the same meaning? \n Yes, No \n Answer:
Example	Here are two sentences: \n The increase reflects lower credit losses and favorable interest rates. \n The gain came as a result of fewer credit losses and lower interest rates. \n Do they have the same meaning? \n Yes, No \n Answer:
WiC	
Format	In these two sentences (1) {*s1*} (2) {s2} , does the word {*word*} mean the same thing? \n Yes, No \n Answer:
Example	In these two sentences (1) It was the deliberation of his act that was insulting. (2) It was the deliberation of his act that was insulting, does the word deliberation mean the same thing? \n Yes, No \n Answer:

Table B.4: Prompt formats designed by Wei et al. [2022] and their examples used in the experiments for each downstream task.

Task: RTE, Paraphrase Type: BECEL	
Original Inputs	Perturbed Inputs
S1: Note that SBB, CFF and FFS stand out for the main railway company, in German, French and Italian.	S1: Note that SBB, CFF and FFS stand out for the main railway company, in German, French and Italian.
S2: The French railway company is called SNCF.	S2: SNCF is the French railway company.
Prediction: Not Entailment	Prediction: Entailment
Task: SNLI, Paraphrase Type: ChatGPT	
Original Inputs	Perturbed Inputs
Premise: A person swimming in a swimming pool.	Premise: A person swimming in a swimming pool.
Hypothesis: A person enjoying the waters.	Hypothesis: An individual is relishing the water.
Prediction: Not Entailment (Neutral)	Prediction: Entailment
Task: WiC, Paraphrase Type: BECEL	
Original Inputs	Perturbed Inputs
Word: glaze	Word: glaze
S1: Glaze the bread with eggwhite.	S1: Glaze the bread with eggwhite.
S2: The potter glazed the dishes.	S2: The dishes were glazed by the potter.
Prediction: Equivalent	Prediction: Not Equivalent

Table B.5: More examples of semantic consistency violation of ChatGPT.

Task: MRPC, Consistency Type: Negation	
Original Inputs	Perturbed Inputs
S1: The dead cavalry have been honored for more than a century with a hilltop granite obelisk and white headstones . S2: The dead cavalrymen are honored with a hilltop granite obelisk and white headstones . Prediction: Equivalent	S2: The dead cavalry have been honored for more than a century with a hilltop granite obelisk and white headstones . S2: The dead cavalrymen are honored with a hilltop granite obelisk and black headstones. Prediction: Equivalent
Task: SNLI, Consistency Type: Negation	
Original Inputs	Perturbed Inputs
Premise: A young man wearing goggles is putting some liquid in a beaker while a young girl wearing blue gloves looks down while holding a pen. Hypothesis: Two people are in the same area. Prediction: Entailment	Premise: A young man wearing goggles is putting some liquid in a beaker while a young girl wearing blue gloves looks down while holding a pen. Hypothesis: Two people are not in the same area. Prediction: Entailment
Task: MRPC, Consistency Type: Symmetric	
Original Inputs	Perturbed Inputs
S1: In 2001 , the diocese reached a $ 15 million settlement involving five priests and 26 plaintiffs . S2: The diocese reached a settlement in 2001 involving five priests and 26 plaintiffs for an undisclosed sum . Prediction: Not Equivalent	S1: The diocese reached a settlement in 2001 involving five priests and 26 plaintiffs for an undisclosed sum . S2: In 2001 , the diocese reached a $ 15 million settlement involving five priests and 26 plaintiffs . Prediction: Equivalent
Task: WiC, Consistency Type: Symmetric	
Original Inputs	Perturbed Inputs
Word: master S1: One of the old masters. S2: A master of the violin. Prediction: Equivalent	Word: master S1: A master of the violin. S2: One of the old masters. Prediction: Not Equivalent

Table B.6: More examples of negation and symmetric consistency violations of ChatGPT.

Appendix C

Improving Language Models' Meaning Understanding and Consistency by Learning Conceptual Roles

C.1 PLMs and Distributional Hypothesis

Sinha et al. [2021a] demonstrated in four steps that training BERT [Devlin et al., 2019] with the masked language modelling objective closely resembles to training Word2vec [Mikolov et al., 2013], a representative method based on the distributional hypothesis. This section provides a concise illustration of the material.

Remind the parameterisation of skip-gram word vectors [Mikolov et al., 2013] (Eq C.1):

$$p(t|w;\theta) = \frac{e^{f(t,w)}}{\sum_{t' \in V} e^{f(t',w)}}\,, \tag{C.1}$$

where f is a dot product, V is the set of all possible words, and t and w are vectors of the target word and a word in the context, respectively. During training, negative sampling is employed within a window size. The word vectors are trained by optimising the loss function, which is defined as $\log \sigma(w \cdot t) + k \cdot \mathbb{E}_{t' \in P} \log \sigma(-w \cdot t')$, over the context $C(w_i) = \{w_{i-k}, ..., w_{i-1}, w_{i+1}, ..., w_{i+k}\}$ for a word index i, a window size of $2k$, and unigram probability distribution P.

Step 1: BPE: The computation of the full softmax probability involves a substantial matrix multiplication with a large vocabulary V. In contrast, BERT leverages subword units, ensuring a smaller total vocabulary U in the softmax denominator.

Step 2: Defenestration: Next, instead of employing a local context window, BERT utilises the entire sentence while making the target word: $C(t) = \{w \in S : w \neq t\}$, where S is the sentence including the word w.

Step 3: Non-linearity: The pairwise word-level dot product $f(w, t)$ is substituted with a non-linear function. A possible approach involves employing a sequence of multi-head self-attention layers $g(t, C(t))$ in BERT. Consequently, the parameterisation of BERT can be expressed as follows (Eq C.2):

$$p(t|C(t); \theta) = \frac{e^{g(t,C(t))}}{\sum_{t' \in U} e^{g(t',C(t))}} \,. \tag{C.2}$$

Step 4: Sprinkle data and compute: Now, model g undergoes training with vast quantities of documents. Following the pre-training phase, the parameters of model g are also updated during the fine-tuning process for downstream tasks.

The same method can be applied to any PLMs trained with a masked language modelling objective to show the relationship between them and the distributional hypothesis. Also, it is possible to show in a similar way that generative PLMs trained with an auto-regressive language modelling objective, e.g., GPT-families, are also based on the distributional hypothesis by simply modifying $C(t)$ in **Step 2** as follows (Eq C.3):

$$C(t) = \{w_0, w_1, ..., w_{t-1}\} \,, \tag{C.3}$$

where w_{t-1} denotes a word that appears right before the word t.

Bibliography

M. Abdou, A. Kulmizev, D. Hershcovich, S. Frank, E. Pavlick, and A. Søgaard. Can language models encode perceptual structure without grounding? a case study in color. In A. Bisazza and O. Abend, editors, *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 109–132, Online, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.conll-1.9. URL `https://aclanthology.org/2021.conll-1.9`.

Y. Adi, E. Kermany, Y. Belinkov, O. Lavi, and Y. Goldberg. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *Proceedings of the International Conference on Learning Representations*, 2017.

A. Aghajanyan, S. Gupta, and L. Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 7319–7328, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.568. URL `https://aclanthology.org/2021.acl-long.568`.

L. Aina, R. Bernardi, and R. Fernández. A distributional study of negated adjectives and antonyms. In *Proceedings of the 5th Italian Conference on Computational Linguistics (CLiC-it 2018)*, volume 2253 of *CEUR Workshop Proceedings*, 2018.

R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023. URL `https://aclanthology.org/2020.emnlp-main.154`.

A. Asai and H. Hajishirzi. Logic-guided data augmentation and regularization for consistent question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5642–5650, Online, July

2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.499. URL https://aclanthology.org/2020.acl-main.499.

P. Atanasova, O.-M. Camburu, C. Lioma, T. Lukasiewicz, J. G. Simonsen, and I. Augenstein. Faithfulness tests for natural language explanations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 283–294, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-short.25. URL https://aclanthology.org/2023.acl-short.25.

J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

S. Banerjee and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005.

Y. Belinkov and J. Glass. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72, 2019.

E. M. Bender and A. Koller. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5185–5198, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.463. URL https://www.aclweb.org/anthology/2020.acl-main.463.

E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623, 2021.

Y. Bisk, R. Zellers, J. Gao, Y. Choi, et al. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439, 2020.

N. Block. Semantics, conceptual role. *Routledge encyclopedia of philosophy*, 8:652–657, 1998.

M. Bommarito II and D. M. Katz. GPT takes the bar exam. *arXiv preprint arXiv:2212.14402*, 2022. URL https://arxiv.org/abs/2212.14402.

T. Bosc and P. Vincent. Auto-encoding dictionary definitions into consistent word embeddings. In E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1522–1532, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1181. URL https://aclanthology.org/D18-1181.

S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 632–642. Association for Computational Linguistics, 2015.

T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.

O.-M. Camburu, T. Rocktäschel, T. Lukasiewicz, and P. Blunsom. e-SNLI: Natural language inference with natural language explanations. In *NeurIPS*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/file/4c7a167bb329bd92580a99ce422d6fa6-Paper.pdf.

O.-M. Camburu, B. Shillingford, P. Minervini, T. Lukasiewicz, and P. Blunsom. Make up your mind! Adversarial generation of inconsistent natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4157–4165, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.382. URL https://aclanthology.org/2020.acl-main.382.

J. Candela-Quinonero, I. Dagan, B. Magnini, and F. d'Alché Buc. Evaluating Predictive Uncertainty, Visual Objects Classification and Recognising textual entailment: Selected Proceedings of the First PASCAL Machine Learning Challenges Workshop, 2006.

J. Cegin, J. Simko, and P. Brusilovsky. ChatGPT to replace crowdsourcing of paraphrases for intent classification: Higher diversity and comparable model robustness.

In H. Bouamor, J. Pino, and K. Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1889–1905, Singapore, Dec. 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.117. URL https://aclanthology.org/2023.emnlp-main.117.

N. Chambers, D. Cer, T. Grenager, D. Hall, C. Kiddon, B. MacCartney, M.-C. de Marneffe, D. Ramage, E. Yeh, and C. D. Manning. Learning alignments and leveraging natural logic. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 165–170, June 2007. URL https://aclanthology.org/W07-1427.

K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179. URL https://aclanthology.org/D14-1179.

E. Choi, H. He, M. Iyyer, M. Yatskar, W.-t. Yih, Y. Choi, P. Liang, and L. Zettlemoyer. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2174–2184, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1241. URL https://aclanthology.org/D18-1241.

J. H. Choi, K. E. Hickman, A. Monahan, and D. B. Schwarcz. ChatGPT goes to law school. *Minnesota Legal Studies Research Paper*, 2023. URL https://ssrn.com/abstract=4335905.

J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio. Attention-based models for speech recognition. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300.

K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *Proceedings of the International Conference on Learning Representations*, 2020. URL https://openreview.net/pdf?id=r1xMH1BtvB.

E. J. De Visser, S. S. Monfort, R. McKendrick, M. A. B. Smith, P. E. McKnight, F. Krueger, and R. Parasuraman. Almost human: Anthropomorphism increases trust resilience in cognitive agents. *Journal of Experimental Psychology: Applied*, 22(3):331, 2016. URL https://psycnet.apa.org/record/2016-37452-001.

T. W. Deacon. *The Symbolic Species: The Co-evolution of Language and the Brain*. WW Norton & Company, 1998.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 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 (NAACL): Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics, June 2019. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

E. Dinan, S. Roller, K. Shuster, A. Fan, M. Auli, and J. Weston. Wizard of Wikipedia: Knowledge-powered conversational agents. In *Proceedings of the International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=r1l73iRqKm.

W. B. Dolan and C. Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 2005.

B. Eisner, T. Rocktäschel, I. Augenstein, M. Bošnjak, and S. Riedel. emoji2vec: Learning emoji representations from their description. In L.-W. Ku, J. Y.-j. Hsu, and C.-T. Li, editors, *Proceedings of the Fourth International Workshop on Natural*

Language Processing for Social Media, pages 48–54, Austin, TX, USA, Nov. 2016. Association for Computational Linguistics. doi: 10.18653/v1/W16-6208. URL https://aclanthology.org/W16-6208.

Y. Elazar, N. Kassner, S. Ravfogel, A. Ravichander, E. Hovy, H. Schütze, and Y. Goldberg. Erratum: Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9: 1407–1407, 2021. doi: 10.1162/tacl_x_00455. URL https://aclanthology.org/2021.tacl-1.83.

K. Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *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 55–65, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1006. URL https://aclanthology.org/D19-1006.

A. Ettinger. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48, 2020.

Y. Fang, X. Li, S. Thomas, and X. Zhu. ChatGPT as data augmentation for compositional generalization: A case study in open intent detection. In C.-C. Chen, H. Takamura, P. Mathur, R. Sawhney, H.-H. Huang, and H.-H. Chen, editors, *Proceedings of the Fifth Workshop on Financial Technology and Natural Language Processing and the Second Multimodal AI For Financial Forecasting*, pages 13–33, Macao, 20 Aug. 2023. -. URL https://aclanthology.org/2023.finnlp-1.2.

S. Frieder, L. Pinchetti, R.-R. Griffiths, T. Salvatori, T. Lukasiewicz, P. C. Petersen, A. Chevalier, and J. Berner. Mathematical capabilities of ChatGPT, 2023. URL https://arxiv.org/abs/2301.13867.

D. Furrer, M. van Zee, N. Scales, and N. Schärli. Compositional generalization in semantic parsing: Pre-training vs. specialized architectures. *arXiv preprint arXiv:2007.08970*, 2020.

S. Garg and G. Ramakrishnan. BAE: BERT-based adversarial examples for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in*

Natural Language Processing (EMNLP), pages 6174–6181, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.498. URL https://aclanthology.org/2020.emnlp-main.498.

R. P. Gazes, N. W. Chee, and R. R. Hampton. Cognitive mechanisms for transitive inference performance in rhesus monkeys: Measuring the influence of associative strength and inferred order. *Journal of Experimental Psychology: Animal Behavior Processes*, 38(4):331, 2012.

G. Grand, I. A. Blank, F. Pereira, and E. Fedorenko. Semantic projection recovers rich human knowledge of multiple object features from word embeddings. *Nature human behaviour*, 6(7):975–987, 2022.

M. Greenberg and G. Harman. Conceptual role semantics. *The Oxford Handbook of Philosophy of Language*, 09 2009. doi: 10.1093/oxfordhb/9780199552238.003.0014.

A. Gupta, G. Kvernadze, and V. Srikumar. Bert & family eat word salad: Experiments with text understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12946–12954, 2021.

S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. Bowman, and N. A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2017. URL https://aclanthology.org/N18-2017.

I. Habernal, H. Wachsmuth, I. Gurevych, and B. Stein. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 1930–1940, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1175. URL https://aclanthology.org/N18-1175.

G. Harman. Conceptual role semantics. *Notre Dame Journal of Formal Logic*, 23(2): 242–256, 1982.

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

P. Hase, S. Zhang, H. Xie, and M. Bansal. Leakage-adjusted simulatability: Can models generate non-trivial explanations of their behavior in natural language? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4351–4367, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.390. URL https://aclanthology.org/2020.findings-emnlp.390.

L. A. Hendricks, R. Hu, T. Darrell, and Z. Akata. Grounding visual explanations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 269–286, 2018.

G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9 (8):1735–1780, 1997.

V. Hofmann, J. Pierrehumbert, and H. Schütze. Superbizarre is not superb: Derivational morphology improves BERT's interpretation of complex words. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 3594–3608, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.279. URL https://aclanthology.org/2021.acl-long.279.

M. Honnibal and M. Johnson. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1373–1378. Association for Computational Linguistics, September 2015. URL https://aclweb.org/anthology/D/D15/D15-1162.

M. M. Hossain, V. Kovatchev, P. Dutta, T. Kao, E. Wei, and E. Blanco. An analysis of natural language inference benchmarks through the lens of negation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9106–9118, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.732. URL https://aclanthology.org/2020.emnlp-main.732.

A. Hosseini, S. Reddy, D. Bahdanau, R. D. Hjelm, A. Sordoni, and A. Courville. Understanding by understanding not: Modeling negation in language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 1301–1312, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.102. URL `https://aclanthology.org/2021.naacl-main.102`.

N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly. 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, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/houlsby19a.html`.

E. J. Hu, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al. LoRA: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=nZeVKeeFYf9`.

X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, 37: 100270, 2020.

R. Hudson. About 37% of word-tokens are nouns. *Language*, 70(2):331–339, 1994.

M. Ivgi and J. Berant. Achieving model robustness through discrete adversarial training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1529–1544, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.115. URL `https://aclanthology.org/2021.emnlp-main.115`.

G. Izacard and E. Grave. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 874–880, Online, Apr. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.74. URL `https://aclanthology.org/2021.eacl-main.74`.

Z. Jiang, F. F. Xu, J. Araki, and G. Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020.

D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8018–8025, 2020.

X. Jin, X. Ren, D. Preotiuc-Pietro, and P. Cheng. Dataless knowledge fusion by merging weights of language models. In *Proceedings of the International Conference on Learning Representations*, 2023. URL `https://arxiv.org/pdf/2212.09849.pdf`.

M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, et al. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

E.-S. Jung, S.-Y. Dong, and S.-Y. Lee. Neural correlates of variations in human trust in human-like machines during non-reciprocal interactions. *Scientific Reports*, 9(1): 1–10, 2019. URL `https://www.nature.com/articles/s41598-019-46098-8`.

A.-L. Kalouli, R. Sevastjanova, C. Beck, and M. Romero. Negation, coordination, and quantifiers in contextualized language models. In *Proceedings of the 29th International Conference on Computational Linguistics (COLING)*, pages 3074–3085, Gyeongju, Republic of Korea, Oct. 2022. International Committee on Computational Linguistics. URL `https://aclanthology.org/2022.coling-1.272`.

R. Karimi Mahabadi, J. Henderson, and S. Ruder. Compacter: Efficient low-rank hypercomplex adapter layers. In *Advances in Neural Information Processing Systems*, volume 34, pages 1022–1035. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper/2021/file/081be9fdff07f3bc808f935906ef70c0-Paper.pdf`.

N. Kassner and H. Schütze. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7811–7818, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.698. URL `https://aclanthology.org/2020.acl-main.698`.

N. Kassner, B. Krojer, and H. Schütze. Are pretrained language models symbolic reasoners over knowledge? In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 552–564, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.conll-1.45. URL https://aclanthology.org/2020.conll-1.45.

M. Kayser, O.-M. Camburu, L. Salewski, C. Emde, V. Do, Z. Akata, and T. Lukasiewicz. E-ViL: A dataset and benchmark for natural language explanations in vision-language tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1244–1254, October 2021.

M. Kayser, C. Emde, O.-M. Camburu, G. Parsons, B. Papiez, and T. Lukasiewicz. Explaining chest x-ray pathologies in natural language. In *MICCAI*, pages 701–713, 2022.

D. Kim, M. Jang, D. S. Kwon, and E. Davis. KoBEST: Korean balanced evaluation of significant tasks. In *Proceedings of the 29th International Conference on Computational Linguistics (COLING)*, pages 3697–3708, Gyeongju, Republic of Korea, Oct. 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.325.

G. Kim, H. Kim, J. Park, and J. Kang. Learn to resolve conversational dependency: A consistency training framework for conversational question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 6130–6141, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.478. URL https://aclanthology.org/2021.acl-long.478.

J. Kim, A. Rohrbach, T. Darrell, J. Canny, and Z. Akata. Textual explanations for self-driving vehicles. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 563–578, 2018.

S. Kitada and H. Iyatomi. Attention meets perturbations: Robust and interpretable attention with adversarial training. *IEEE Access*, 9:92974–92985, 2021.

M. Komeili, K. Shuster, and J. Weston. Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 8460–8478, Dublin, Ireland, May 2022. Association for

Computational Linguistics. doi: 10.18653/v1/2022.acl-long.579. URL https://aclanthology.org/2022.acl-long.579.

S. D. Krashen. *Principles and practice in second language acquisition*. Language Teaching Methodology Series. Pergamon, Oxford, 1982. ISBN 9780080286280.

W. Kryscinski, B. McCann, C. Xiong, and R. Socher. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.750. URL https://aclanthology.org/2020.emnlp-main.750.

A. Kumar and A. Joshi. Striking a balance: Alleviating inconsistency in pre-trained models for symmetric classification tasks. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1887–1895, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.148. URL https://aclanthology.org/2022.findings-acl.148.

S. Kumar and P. Talukdar. NILE : Natural language inference with faithful natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 8730–8742. Association for Computational Linguistics, July 2020. doi: 10.18653/v1/2020.acl-main.771. URL https://aclanthology.org/2020.acl-main.771.

T. H. Kung, M. Cheatham, A. Medenilla, C. Sillos, L. De Leon, C. Elepaño, M. Madriaga, R. Aggabao, G. Diaz-Candido, J. Maningo, et al. Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models. *PLOS Digital Health*, 2(2):e0000198, 2023.

G. Lample, A. Conneau, L. Denoyer, and M. Ranzato. Unsupervised machine translation using monolingual corpora only. In *Proceedings of the International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rkYTTf-AZ.

Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. Albert: A lite bert for self-supervised learning of language representations. In *Proceedings of the International Conference on Learning Representations*, 2020. URL https://openreview.net/pdf?id=H1eA7AEtvS.

B. Lester, R. Al-Rfou, and N. Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3045–3059, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.243. URL https://aclanthology.org/2021.emnlp-main.243.

M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. 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 (ACL)*, pages 7871–7880, Online, July 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703.

P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020b.

D. Li, Y. Zhang, H. Peng, L. Chen, C. Brockett, M.-T. Sun, and B. Dolan. Contextualized perturbation for textual adversarial attack. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 5053–5069, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.400. URL https://aclanthology.org/2021.naacl-main.400.

L. Li, R. Ma, Q. Guo, X. Xue, and X. Qiu. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6193–6202, Online, Nov. 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.500. URL https://aclanthology.org/2020.emnlp-main.500.

M. Li, S. Roller, I. Kulikov, S. Welleck, Y.-L. Boureau, K. Cho, and J. Weston. Don't say that! making inconsistent dialogue unlikely with unlikelihood training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4715–4728, Online, July 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.428. URL https://aclanthology.org/2020.acl-main.428.

T. Li, V. Gupta, M. Mehta, and V. Srikumar. A logic-driven framework for consistency of neural models. 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 3924–3935, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1405. URL https://aclanthology.org/D19-1405.

X. Li, L. Ding, L. Wang, and F. Cao. Fpga accelerates deep residual learning for image recognition. In *2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pages 837–840. IEEE, 2017.

Y. Li, S. Bubeck, R. Eldan, A. Del Giorno, S. Gunasekar, and Y. T. Lee. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023.

J. Liang and H. Liu. Noun distribution in natural languages. *Poznań Studies in Contemporary Linguistics*, 49(4):509–529, 2013.

B. Y. Lin, S. Lee, R. Khanna, and X. Ren. Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6862–6868, Online, Nov. 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.557. URL https://aclanthology.org/2020.emnlp-main.557.

C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, 2004.

R. Lin and H. T. Ng. Does BERT know that the IS-a relation is transitive? In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 94–99, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.11. URL https://aclanthology.org/2022.acl-short.11.

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 740–755. Springer, 2014.

Z. Lin, A. Madotto, and P. Fung. Exploring versatile generative language model via parameter-efficient transfer learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 441–459, Online, Nov. 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.41. URL https://aclanthology.org/2020.findings-emnlp.41.

N. F. Liu, M. Gardner, Y. Belinkov, M. E. Peters, and N. A. Smith. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 1073–1094, June 2019a. doi: 10.18653/v1/N19-1112. URL https://aclanthology.org/N19-1112.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019b.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1166. URL https://aclanthology.org/D15-1166.

B. P. Majumder, O. Camburu, T. Lukasiewicz, and J. Mcauley. Knowledge-grounded self-rationalization via extractive and natural language explanations. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 14786–14801. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/majumder22a.html.

A. Marasovic, I. Beltagy, D. Downey, and M. Peters. Few-shot self-rationalization with natural language prompts. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 410–424, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.31. URL https://aclanthology.org/2022.findings-naacl.31.

M. S. Matena and C. A. Raffel. Merging models with fisher-weighted averaging. In *Advances in Neural Information Processing Systems*, volume 35, pages 17703–17716. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/70c26937fbf3d4600b69a129031b66ec-Paper-Conference.pdf.

J. Maynez, S. Narayan, B. Bohnet, and R. McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1906–1919, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.173. URL https://aclanthology.org/2020.acl-main.173.

T. McCoy, E. Pavlick, and T. Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL https://aclanthology.org/P19-1334.

I. R. McKenzie, A. Lyzhov, M. Pieler, A. Parrish, A. Mueller, A. Prabhu, E. McLean, A. Kirtland, A. Ross, A. Liu, et al. Inverse scaling: When bigger isn't better. *arXiv preprint arXiv:2306.09479*, 2023.

I. A. Mel'čuk and A. K. Žolkovskij. Towards a functioning 'meaning-text' model of language. *Linguistics*, 8(57):10–47, 1970. ISSN 0024-3949.

A. V. Miceli Barone, F. Barez, S. B. Cohen, and I. Konstas. The larger they are, the harder they fail: Language models do not recognize identifier swaps in python. In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 272–292, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.19. URL https://aclanthology.org/2023.findings-acl.19.

S. J. Mielke, A. Szlam, E. Dinan, and Y.-L. Boureau. Reducing conversational agents' overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872, 2022. doi: 10.1162/tacl_a_00494. URL https://aclanthology.org/2022.tacl-1.50.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf.

J. Milićević. A short guide to the meaning-text linguistic theory. *Journal of Koralex*, 8:187–233, 2006.

G. A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

E. Mitchell, J. J. Noh, S. Li, W. S. Armstrong, A. Agarwal, P. Liu, C. Finn, and C. D. Manning. Enhancing self-consistency and performance of pretrained language models with NLI. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2022. URL https://ericmitchell.ai/concord.pdf.

J. Morris, E. Lifland, J. Y. Yoo, J. Grigsby, D. Jin, and Y. Qi. TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 119–126, Online, Oct. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.16. URL https://aclanthology.org/2020.emnlp-demos.16.

N. Mrkšić, D. Ó Séaghdha, B. Thomson, M. Gašić, L. M. Rojas-Barahona, P.-H. Su, D. Vandyke, T.-H. Wen, and S. Young. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 142–148. Association for Computational Linguistics, June 2016. doi: 10.18653/v1/N16-1018. URL https://aclanthology.org/N16-1018.

A. Naik, A. Ravichander, N. Sadeh, C. Rose, and G. Neubig. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 2340–2353, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://aclanthology.org/C18-1198.

N. Nangia, C. Vania, R. Bhalerao, and S. R. Bowman. CrowS-pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1953–1967, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.154. URL https://aclanthology.org/2020.emnlp-main.154.

S. Narang, C. Raffel, K. Lee, A. Roberts, N. Fiedel, and K. Malkan. WT5?! Training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*, 2020.

T. Niven and H.-Y. Kao. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4658–4664, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1459. URL https://aclanthology.org/P19-1459.

R. Nogueira, Z. Jiang, and J. Lin. Investigating the limitations of transformers with simple arithmetic tasks. In *ICLR 2021 Workshop on Mathematical Reasoning in General Artificial Intelligence*, 2021. URL https://arxiv.org/pdf/2102.13019.pdf.

Y. Ohsugi, I. Saito, K. Nishida, H. Asano, and J. Tomita. A simple but effective method to incorporate multi-turn context with bert for conversational machine comprehension. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 11–17, 2019.

OpenAI. Gpt-4 technical report, 2023.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744, 2022.

A. Pagnoni, V. Balachandran, and Y. Tsvetkov. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 4812–4829, Online, June 2021. Association for Computational Linguistics.

doi: 10.18653/v1/2021.naacl-main.383. URL https://aclanthology.org/2021.naacl-main.383.

L. Pan, C.-W. Hang, A. Sil, and S. Potdar. Improved text classification via contrastive adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11130–11138, 2022.

D. Paperno, G. Kruszewski, A. Lazaridou, N.-Q. Pham, R. Bernardi, S. Pezzelle, M. Baroni, G. Boleda, and R. Fernández. The lambada dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1525–1534. Association for Computational Linguistics, 2016.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: Amethod for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics (ACL)*, pages 311–318. Association for Computational Linguistics, 2002.

I.-E. Parasca, A. L. Rauter, J. Roper, A. Rusinov, G. Bouchard, S. Riedel, and P. Stenetorp. Defining words with words: Beyond the distributional hypothesis. In *Proceedings of the 1st workshop on evaluating vector-space representations for NLP*, pages 122–126, 2016.

S. Park, J. Moon, S. Kim, W. I. Cho, J. Y. Han, J. Park, C. Song, J. Kim, Y. Song, T. Oh, J. Lee, J. Oh, S. Lyu, Y. Jeong, I. Lee, S. Seo, D. Lee, H. Kim, M. Lee, S. Jang, S. Do, S. Kim, K. Lim, J. Lee, K. Park, J. Shin, S. Kim, L. Park, L. Park, A. Oh, J.-W. Ha (NAVER AI Lab), K. Cho, and K. Cho. KLUE: Korean language understanding evaluation. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/98dce83da57b0395e163467c9dae521b-Paper-round2.pdf.

A. Patel, C. Raffel, and C. Callison-Burch. Datadreamer: A tool for synthetic data generation and reproducible llm workflows, 2024.

J. Pennington, R. Socher, and C. D. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics, 2014.

F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. Wu, and A. Miller. Language models as knowledge bases? 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 2463–2473, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1250. URL https://aclanthology.org/D19-1250.

J. Pfeiffer, A. Kamath, A. Rücklé, K. Cho, and I. Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 487–503, Online, Apr. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.39. URL https://aclanthology.org/2021.eacl-main.39.

T. Pham, T. Bui, L. Mai, and A. Nguyen. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks? In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1145–1160, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.98. URL https://aclanthology.org/2021.findings-acl.98.

J. Phang, T. Févry, and S. R. Bowman. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *CoRR*, abs/1811.01088, 2018. URL http://arxiv.org/abs/1811.01088.

S. Piantadosi and F. Hill. Meaning without reference in large language models. In *NeurIPS 2022 Workshop on Neuro Causal and Symbolic AI (nCSI)*, 2022.

M. T. Pilehvar and J. Camacho-Collados. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, pages 1267–1273, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1128. URL https://aclanthology.org/N19-1128.

Y. Pruksachatkun, J. Phang, H. Liu, P. M. Htut, X. Zhang, R. Y. Pang, C. Vania, K. Kann, and S. R. Bowman. Intermediate-task transfer learning with pretrained

language models: When and why does it work? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5231–5247. Association for Computational Linguistics, July 2020. doi: 10.18653/v1/2020.acl-main.467. URL https://aclanthology.org/2020.acl-main.467.

X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, pages 1–26, 2020.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding with unsupervised learning. *Technical report, OpenAI*, 2018.

A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67, 2020. URL https://w.jmlr.org/papers/volume21/20-074/20-074.pdf.

H. Raj, D. Rosati, and S. Majumdar. Measuring reliability of large language models through semantic consistency. In *NeurIPS 2022 Workshop on Machine Learning Safety*, 2022. URL https://openreview.net/pdf?id=SgbpddeEV-C.

N. F. Rajani, B. McCann, C. Xiong, and R. Socher. Explain yourself! Leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4932–4942, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1487. URL https://aclanthology.org/P19-1487.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2383–2392. Association for Computational Linguistics, 2016.

P. Rajpurkar, R. Jia, and P. Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124.

A. Ravichander, E. Hovy, K. Suleman, A. Trischler, and J. C. K. Cheung. On the systematicity of probing contextualized word representations: The case of hypernymy in BERT. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 88–102, Barcelona, Spain (Online), Dec. 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.starsem-1.10.

A. Ray, K. Sikka, A. Divakaran, S. Lee, and G. Burachas. Sunny and dark outside?! improving answer consistency in VQA through entailed question generation. 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 5860–5865, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1596. URL https://aclanthology.org/D19-1596.

S.-A. Rebuffi, H. Bilen, and A. Vedaldi. Learning multiple visual domains with residual adapters. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/e7b24b112a44fdd9ee93bdf998c6ca0e-Paper.pdf.

S. Reddy, D. Chen, and C. D. Manning. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.

M. T. Ribeiro, C. Guestrin, and S. Singh. Are red roses red? evaluating consistency of question-answering models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6174–6184. Association for Computational Linguistics, 2019.

M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4902–4912, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.442. URL https://aclanthology.org/2020.acl-main.442.

S. V. Rouse. Reliability of MTurk data from masters and workers. *Journal of Individual Differences*, 2019.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. In *5th Workshop on Energy Efficient Machine Learning and Cognitive Computing-NeurIPS 2019*, 2019.

A. Santoro, A. Lampinen, K. Mathewson, T. Lillicrap, and D. Raposo. Symbolic behaviour in artificial intelligence. *arXiv preprint arXiv:2102.03406*, 2021.

M. Sap, S. Gabriel, L. Qin, D. Jurafsky, N. A. Smith, and Y. Choi. Social bias frames: Reasoning about social and power implications of language. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5477–5490, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.486. URL https://aclanthology.org/2020.acl-main.486.

X. Shu, B. Yu, Z. Zhang, and T. Liu. Drg2vec: Learning word representations from definition relational graph. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9, 2020. doi: 10.1109/IJCNN48605.2020.9207015.

K. Shuster, S. Poff, M. Chen, D. Kiela, and J. Weston. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.320. URL https://aclanthology.org/2021.findings-emnlp.320.

K. Sinha, R. Jia, D. Hupkes, J. Pineau, A. Williams, and D. Kiela. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2888–2913. Association for Computational Linguistics, Nov. 2021a. URL https://aclanthology.org/2021.emnlp-main.230.

K. Sinha, P. Parthasarathi, J. Pineau, and A. Williams. UnNatural Language Inference. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 7329–7346, Online, Aug. 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.569. URL https://aclanthology.org/2021.acl-long.569.

R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642, Seattle, Washington, USA, Oct. 2013. Association for Computational Linguistics. URL https://aclanthology.org/D13-1170.

R. Speer, J. Chin, and C. Havasi. ConceptNet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

N. Stiennon, L. Ouyang, J. Wu, D. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. F. Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021, 2020.

E. Strubell, A. Ganesh, and A. McCallum. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3645–3650, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1355. URL https://aclanthology.org/P19-1355.

J. Stuczyński, S. Riedel, and P. Saito Stenetorp. Towards a word sheriff 2.0: Lessons learnt and the road ahead. In *Games4NLP Symposium*. Disagreements and Language Interpretation (DALI), 2017.

Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, and H. Wang. ERNIE 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8968–8975, 2020.

I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *Proceedings of the International Conference on Learning Representations*, 2014. URL https://nyuscholars.nyu.edu/en/publications/intriguing-properties-of-neural-networks.

C. Takahashi. Impact of dictionary use skills instruction on second language writing. *Studies in Applied Linguistics and TESOL*, 12(2), 2012.

C. Terwiesch. Would Chat GPT get a Wharton MBA? A prediction based on its performance in the operations management course. *Mack Institute for Innovation Management at the Wharton School, University of Pennsylvania. Retrieved from: https://mackinstitute. wharton. upenn. edu/wpcontent/uploads/2023/01/Christian-Terwiesch-Chat-GTP-1.24. pdf [Date accessed: February 6th, 2023]*, 2023.

R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, Y. Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022.

J. Tissier, C. Gravier, and A. Habrard. Dict2vec : Learning word embeddings using lexical dictionaries. In M. Palmer, R. Hwa, and S. Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1024. URL `https://aclanthology.org/D17-1024`.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

T. H. Truong, T. Baldwin, K. Verspoor, and T. Cohn. Language models are not naysayers: an analysis of language models on negation benchmarks. In A. Palmer and J. Camacho-collados, editors, *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 101–114, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.starsem-1.10. URL `https://aclanthology.org/2023.starsem-1.10`.

D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry. Robustness may be at odds with accuracy. In *Proceedings of the International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=SyxAb30cY7`.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

T. Vu, T. Wang, T. Munkhdalai, A. Sordoni, A. Trischler, A. Mattarella-Micke, S. Maji, and M. Iyyer. Exploring and predicting transferability across NLP tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language*

Processing (EMNLP), pages 7882–7926. Association for Computational Linguistics, Nov. 2020. doi: 10.18653/v1/2020.emnlp-main.635. URL https://aclanthology.org/2020.emnlp-main.635.

E. Wallace, Y. Wang, S. Li, S. Singh, and M. Gardner. Do NLP models know numbers? probing numeracy in embeddings. 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 5307–5315, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1534. URL https://aclanthology.org/D19-1534.

J. Wallat, J. Singh, and A. Anand. BERTnesia: Investigating the capture and forgetting of knowledge in BERT. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 174–183, Nov. 2020. doi: 10.18653/v1/2020.blackboxnlp-1.17. URL https://aclanthology.org/2020.blackboxnlp-1.17.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. 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, Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446.

A. Wang, J. Hula, P. Xia, R. Pappagari, R. T. McCoy, R. Patel, N. Kim, I. Tenney, Y. Huang, K. Yu, S. Jin, B. Chen, B. Van Durme, E. Grave, E. Pavlick, and S. R. Bowman. Can you tell me how to get past Sesame Street? Sentence-level pretraining beyond language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4465–4476. Association for Computational Linguistics, July 2019a. doi: 10.18653/v1/P19-1439. URL https://aclanthology.org/P19-1439.

A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019b. URL https://proceedings.neurips.cc/paper/2019/file/4496bf24afe7fab6f046bf4923da8de6-Paper.pdf.

A. Wang, K. Cho, and M. Lewis. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5008–5020, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.450. URL https://aclanthology.org/2020.acl-main.450.

H. Wang, D. Sun, and E. P. Xing. What if we simply swap the two text fragments? a straightforward yet effective way to test the robustness of methods to confounding signals in nature language inference tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7136–7143, 2019c.

R. Wang and R. Henao. Unsupervised paraphrasing consistency training for low resource named entity recognition. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5303–5308, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.430. URL https://aclanthology.org/2021.emnlp-main.430.

S. Wang, Y. Xu, Y. Fang, Y. Liu, S. Sun, R. Xu, C. Zhu, and M. Zeng. Training data is more valuable than you think: A simple and effective method by retrieving from training data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3170–3179, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.226. URL https://aclanthology.org/2022.acl-long.226.

J. Wei, M. Bosma, V. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. In *Proceedings of the International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

S. Welleck, J. Weston, A. Szlam, and K. Cho. Dialogue natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3731–3741, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1363. URL https://aclanthology.org/P19-1363.

S. Welleck, I. Kulikov, S. Roller, E. Dinan, K. Cho, and J. Weston. Neural text generation with unlikelihood training. In *Proceedings of the International Conference*

on Learning Representations, 2020. URL https://openreview.net/forum?id=SJeYe0NtvH.

J. Wen and W. Wang. The future of chatgpt in academic research and publishing: A commentary for clinical and translational medicine. *Clinical and Translational Medicine*, 13(3):e1207–e1207, 2023.

S. Wiegreffe and A. Marasovic. Teach me to explain: A review of datasets for explainable natural language processing. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/698d51a19d8a121ce581499d7b701668-Paper-round1.pdf.

S. Wiegreffe, A. Marasović, and N. A. Smith. Measuring association between labels and free-text rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 10266–10284, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.804. URL https://aclanthology.org/2021.emnlp-main.804.

M. Wortsman, G. Ilharco, S. Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A. S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith, and L. Schmidt. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 23965–23998. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/wortsman22a.html.

W. Wu, C. Jiang, Y. Jiang, P. Xie, and K. Tu. Do PLMs know and understand ontological knowledge? In A. Rogers, J. Boyd-Graber, and N. Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3080–3101, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.173. URL https://aclanthology.org/2023.acl-long.173.

Y. Xu, C. Zhu, R. Xu, Y. Liu, M. Zeng, and X. Huang. Fusing context into knowledge graph for commonsense question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Com-

putational Linguistics, Aug. 2021. doi: 10.18653/v1/2021.findings-acl.102. URL https://aclanthology.org/2021.findings-acl.102.

J. Y. Yoo and Y. Qi. Towards improving adversarial training of NLP models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 945–956, Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.81. URL https://aclanthology.org/2021.findings-emnlp.81.

Y. Yordanov, V. Kocijan, T. Lukasiewicz, and O.-M. Camburu. Few-shot out-of-domain transfer learning of natural language explanations in a label-abundant setup. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3486–3501, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.255. URL https://aclanthology.org/2022.findings-emnlp.255.

W. Yu, C. Zhu, Y. Fang, D. Yu, S. Wang, Y. Xu, M. Zeng, and M. Jiang. Dict-BERT: Enhancing language model pre-training with dictionary. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1907–1918, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.150. URL https://aclanthology.org/2022.findings-acl.150.

R. Zellers, Y. Bisk, R. Schwartz, and Y. Choi. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 93–104. Association for Computational Linguistics, 2018.

S. Zhang, H. Xu, and X. Zhang. The effects of dictionary use on second language vocabulary acquisition: A meta-analysis. *International Journal of Lexicography*, 34(1):1–38, 05 2020a. ISSN 0950-3846. doi: 10.1093/ijl/ecaa010. URL https://doi.org/10.1093/ijl/ecaa010.

T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. BERTScore: Evaluating text generation with BERT. In *Proceedings of the International Conference on Learning Representations*, 2020b. URL https://openreview.net/attachment?id=SkeHuCVFDr&name=original_pdf.

W. Zhang, J. Yu, W. Zhao, and C. Ran. DMRFNet: Deep multimodal reasoning and fusion for visual question answering and explanation generation. *Information Fusion*, 72:70–79, 2021.

X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/250cf8b51c773f3f8dc8b4be867a9a02-Paper.pdf.

B. Zheng, L. Dong, S. Huang, W. Wang, Z. Chi, S. Singhal, W. Che, T. Liu, X. Song, and F. Wei. Consistency regularization for cross-lingual fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 3403–3417, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.264. URL https://aclanthology.org/2021.acl-long.264.

C. Zhu, Y. Cheng, Z. Gan, S. Sun, T. Goldstein, and J. Liu. FreeLB: Enhanced adversarial training for natural language understanding. In *Proceedings of the International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BygzbyHFvB.

T. Y. Zhuo, Y. Huang, C. Chen, and Z. Xing. Red teaming chatgpt via jailbreaking: Bias, robustness, reliability and toxicity. *arXiv preprint arXiv:2301.12867*, 2023.