Research Master Thesis

Random Seed Influence on Language Model Generalizability

Bas Diender

a thesis submitted in partial fulfilment of the requirements for the degree of

MA Linguistics
(Human Language Technology)

Vrije Universiteit Amsterdam

Computational Lexicology and Terminology Lab
Department of Language and Communication
Faculty of Humanities

Supervised by:	prof. dr. Antske Fokkens and Urja Khurana
2nd reader:	dr. Lisa Beinborn
Submitted:	August 22, 2023

Abstract

This thesis investigates the effect of random initialization on the generalizability of transformer-based language models by fine-tuning them for Natural Language Inference (NLI). To this end, the results of thirty instances of RoBERTa are compared to those of one hundred instances of BERT, each differing only in their random seed. Three main research objectives are addressed: (1) assessing whether the generalizability of RoBERTa is less sensitive to changes in the random seed; (2) quantifying the agreement between models that differ only in their random seed; (3) investigating which specific linguistic challenges these models encounter when dealing with entailment. The results reveal that while RoBERTa is more adept at generalizing to out-of-distribution data than BERT, both models are found to exhibit some degree of reliance on certain heuristics which may compromise their generalizability. Moreover, it seems that random initialization has a considerable influence on downstream model behavior in a way that is not reflected in their accuracy. Furthermore, complex syntactic structures and high-level semantic information are found to consistently pose challenges for BERT and, albeit to a lesser extent, RoBERTa. The findings highlight the influence of the random seed and suggest avenues for future research, including a more granular analysis of the linguistic capabilities of language models and other technicalities that could inadvertently affect model performance.

Declaration of Authorship

I, Bas Diender, declare that this thesis, titled *Random Seed Influence on Language Model Generalizability* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: August 22, 2023

Signed:

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my supervisors prof. dr. Antske Fokkens and Urja Khurana for their unwavering patience and support in navigating the training process that tested the extent of Murphy's law at times. Their guidance, understanding, and expert insights made all the difference over the past few months. I also owe a heartfelt thanks to my friends and family for their willingness to listen to numerous unsolicited monologues about random seeds and for their persistent encouragements when motivation waned. Thanks to all of you for making this thesis possible.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Large language models have become an indispensable component of many of the most successful applications in natural language technology. The quality of technologies such as search engines, machine translation, text predictive software, content recommendation systems, and many others has rapidly improved in recent years as the development of larger language models with more optimized training procedures has garnered a substantial amount of attention. Despite their widespread adoption, these models remain enigmatic, particularly when trying to decipher their inner workings. While every part of the internal structure of such models is well understood in theory, the number of parameters involved in generating the desired output may well exceed several billion. Therefore, the decision-making process employed by the model to accomplish its intended task cannot feasibly be interpreted by studying the intricate underlying mechanisms. Instead, the broader behaviors and tendencies of such models can be studied to elucidate what information models base their decisions on. Given the pervasive role these models have come to play in our daily lives, a thorough understanding of exactly how they work is important to ensure that they approach their tasks in a fair and predictable manner.

One task that has been said to require a broad understanding of natural language is natural language inference (NLI). This task involves taking a 'premise' and 'hypothesis' and determining whether or not they logically follow one another (e.g., Dagan et al., 2005; Bowman et al., 2015). It is often regarded as an excellent benchmark for natural language understanding (NLU), as it requires a model to grasp both the underlying structures of the two sentences, semantic relations between words within each sentence as well as between the two sentences, as well as deal with various linguistic phenomena such as coreference, tense, evidentiality, or conditionality, and work out the entailments that follow from them (Williams et al., 2018). NLI is among the tasks revolutionized by the advancements of large language models, which have been reported to perform close to human control groups. However, there is a growing body of evidence to suggest that these results might be inflated, as models use approaches that would not transfer well outside of their experimental setting (e.g., Gururangan et al., 2018). These findings fall into a larger trend of scepticism about the impressive advancements made by language models (e.g., D'Amour et al., 2020). Not only have they been found to use strategies that would inhibit their performance in deployment, seemingly minor changes to the training set up also appear to affect performance more than might be expected (McCoy et al., 2020). In other words, the impressive results reported might not accurately reflect what language models are actually capable of.

1.1 Research Goals

The objective of this thesis is to investigate one potential source of instability in a language model's performance on NLI. In particular, the focus is on the influence of how the random components of a language model are initialized before training on its performance after training. Previous research by McCoy et al. (2020) has found this effect to be larger than might be expected. This work looked into the commonly used language model BERT (Devlin et al., 2019), which has since been improved upon with various models that introduce slight modifications to BERT's architecture or training setup. One such model is RoBERTa (Liu et al., 2019), which chiefly aims to improve on BERT's robustness through adjustments to the training setup. This thesis investigates to what extent RoBERTa mitigates the influence random initialization has on downstream performance, given its ostensible improvements to robustness. Moreover, it also aims to provide a deeper insight into what language models are capable of when it comes to NLI.

The results indicate the RoBERTa generalizes to data that is different from the data it was trained on more robustly than BERT does. However, RoBERTa does appear to be more susceptible to changes in random initialization. In terms of overall performance, the results show that performance drops significantly when models are employed outside of their experimental setting. An analysis of the results indicate that the models rely on dataset-specific artefacts, rather than linguistic understanding and logical reasoning to judge inferences. However, RoBERTa's improved robustness does seem to translate to a decreased reliance on such artefacts.

1.2 Structure

The background to this thesis is outlined in Chapter 2, which first explains what language models are and how they have come to play the prominent role they play today, as well as what random initialization is and what role it plays in the training process of language models. Next, this chapter contains a more complete explanation of NLI as a task in the field and the role it plays in the broader field of generalizability research. A concrete view of how the research for this thesis was conducted is given in Chapter 3. Starting with an overview of the internal processes involved when a language model is employed for NLI, followed by an explanation of how they were employed and evaluated in this thesis. The next section in this chapter explains how the datasets provide insight into the linguistic processes that underlie the predictions made by the models. The results are presented in Chapter 4. This chapter separates results of evaluation on data that matches the data the model was trained on from data that does not match the training data, as the latter is particularly informative about language model generalizability. Next, this chapter contains a thorough linguistic analysis of the inferences made by the models. A reflection on the results and the process that preceded them is given in Chapter 5, as well as suggestions for future research. Finally, this chapter also concludes the project in a section that summarizes the main findings.

Chapter 2

Background

The aim of this chapter is to explain language models and the role they have come to play in the field of NLP. Beginning with the fundamental concepts of language models, an overview of their evolving complexities and key advancements is presented to explain how language models have become a cornerstone of modern NLP research. Next, two important concepts are presented that are important to the problem central to this thesis: The role randomness plays in the process of training a language model, and the ability of language models to generalize outside of their training environment. Shifting focus, Section 2.2 delves into the task of Natural Language Inference (NLI), outlining the goal of the task and motivating why it was chosen for this thesis. The chapter concludes with a section that summarizes the goal of this thesis, and presents the questions this thesis aims to answer.

2.1 Language Models

One of the major challenges in natural language processing (NLP) is to capture the intricacies of natural language numerically so that computers can work with it. While the optimal approach depends on the task at hand, language models have proven to be tremendously helpful in capturing the underlying structure and patterns of natural language.

In the most basic definition, language models are computational models trained to estimate a probability distribution over a sequence of words. Given a sentence, a language model returns a metric that indicates how likely that sentence is, given the data on which the model was trained. To illustrate, consider the sentences famously introduced by Chomsky (1957): (a) *Colorless green ideas sleep furiously* and (b) *Furiously sleep ideas green colorless.* While speakers of English are unlikely to have ever encountered either sentence, they would have little trouble identifying (a) as grammatical and (b) as ungrammatical. This example was intended to show the inadequacy of probabilistic models of grammar, but Pereira (2000) showed that a basic statistical model of language can indeed capture this distinction.

The type of model used by Pereira (2000) is based on transitional probabilities between words, which they learn by training on large collections of text. As such, these models can compute the probability that a sentence starts with *colorless*, that the word *ideas* is preceded by *green*, or that a sentence ends with the word *furiously.* These probabilities taken together allow such a model to calculate the probability of any given sentence:

$$P_{(\mathrm{a})} = P(\textsc{end}|\textit{furiously})P(\textit{furiously}\,|\textit{sleep})\ \ldots\ \ P(\textit{green}\,|\textit{colorless})P(\textit{colorless}\,|\textsc{start})$$
$$P_{(\mathrm{b})} = P(\textsc{end}|\textit{colorless})P(\textit{colorless}\,|\textit{green})\ \ldots\ \ P(\textit{sleep}\,|\textit{furiously})P(\textit{furiously}\,|\textsc{start})$$

Although Pereira's (2000) model had never encountered either (a) or (b) during training, it computed (a) as being 200 000 more likely than (b). Furthermore, a third sentence, syntactically identical to (a) but semantically meaningful would yield an even higher probability because the words in such a sentence are more likely to follow one another.

2.1.1 Language Modeling as a Pre-training Task

Since 2000, there have been several significant advancements in the field of language modeling. A primary example is the introduction of word2vec by Mikolov et al. (2013). Word2vec is a technique that uses language modeling to map words to numerical representations called *word embeddings*. While Mikolov et al. (2013) were not the first to use word embeddings, their method made the creation of word embeddings more efficient, and yielded embeddings of higher quality by training on a larger amount of data. The importance of word embeddings is twofold: First, the continuous representations it generates allow mathematical operations to be performed on words in a meaningful way. Second, the representations also capture the semantics of a word, since words that are similar in meaning are mapped closer to each other in the continuous space. The ability to capture the meaning of a word numerically has proven to be exceedingly useful to the entire field of NLP. Word2vec demonstrated how language modeling could serve as an effective initial step – referred to as *pre-training* – in preparing models for more complex tasks. This pre-training essentially equips the model with a foundational understanding of language patterns, which has proven useful for many NLP tasks, and helped advance the state-of-the-art in many of them.

Words generally do not exist in isolation; a word like *lead* can have any number of meanings depending on its context. With word2vec, this word would be mapped to the same word embedding regardless of the specific meaning. In 2018, Peters et al. introduced ELMo (Embeddings from Language Models), a model that generates *contextualized* word embeddings. ELMo does this by modelling words in a sentence bidirectionally by taking into consideration the words the precede the word in question, as well as the words that follow it in that particular sentence. These two representations are then combined to create a single word embedding that represents the word given the context of its sentence. This approach allows ELMo to capture a deeper meaning of a word that also includes the context it is in.

Another significant development in the field of NLP was marked by Vaswani et al.'s (2017) introduction of transformers. Where ELMo models a sequence left-to-right and right-to-left and then combines those representations to obtain a contextual word embedding, transformers employ a mechanism known as *self-attention*. This mechanism models the entire sequence in parallel to generate a representation of the importance of all words in the sentence in relation to each other. By doing so, it eliminates the necessity of processing sequences in a predefined order. This parallel processing enables transformer-based models to manage diverse context-specific meanings for words and more effectively capture long-range dependencies between words, resulting in better efficiency and improved linguistic understanding.

Building on the idea of transformers and contextualized word embeddings introduced by ELMo, Devlin et al. (2019) proposed BERT (Bidirectional Encoder Representations from Transformers). While ELMo generates separate embeddings for forward and backward context of a word and then combines them, BERT uses the transformer's self-attention mechanism to understand the full context of a word in one go. The training of BERT involves two key tasks. First, masked language modeling, where a random word in a sentence is obscured, and the model is tasked with using the rest of the sentence to predict what the obscured word might be. Second, next sentence prediction, where BERT is given two sentences and must determine whether the second would logically follow the first in a real-world context. This approach results in deeply bidirectional, unsupervised representations of words, providing a comprehensive understanding of language.

The introduction of RoBERTa (Robustly Optimized BERT pre-training Approach) by Liu et al. (2019) further improved upon BERT with specific improvements to the pre-training setup. For BERT, a fixed percentage of words is masked in the masked language modeling task. Whereas for training RoBERTa, a new masking pattern is generated every time a new sequence is fed to the model. Moreover, Liu et al. (2019) state that the effect of the sentence prediction task on model performance after training is contested, and do away with the task altogether. Taken together, these adjustments have resulted in a language model that achieves a better performance than its predecessor BERT on a variety of NLP tasks (Liu et al., 2019).

BERT-like models such as RoBERTa have since become a cornerstone in the field of NLP and have set the state-of-the-art in the vast majority of tasks other than language modeling. Throughout this thesis, 'language models' will predominantly refer to deep transformer-based models such as BERT and RoBERTa.

2.1.2 Randomness in the Training Process

In general, many approaches to NLP tasks now involve fine-tuning a pre-trained language model to that specific task. That is to say, a model that has been trained on language modeling first is given a smaller, task-specific dataset. The linguistic understanding these models attained during pre-training is then leveraged for better performance on the task at hand. How this is done specifically differs from paper to paper, but it often broadly follows a similar pattern. D'Amour et al. (2020) provide a thorough formalization of a standard machine learning pipeline, which can be used to better illustrate how language models are used in NLP. In this formalization, any predictive model can be thought of as a function $f : \mathcal{X} \mapsto \mathcal{Y}$ that maps inputs $\mathbf{x}$ to outputs $\mathbf{y}$.

Assume the goal is to train a system to translate English text into French. The variable $\mathbf{x}$ can then be thought of to be a representation of a text in English, and $\mathbf{y}$ the representation of the translation of that text in French. The translation system may then be said to be a function f, and its attempted translation is written as $f(\mathbf{x})$. Finding the right value for f requires a so-called loss function ℓ that takes as input the attempted translation $f(\mathbf{x})$ and the actual translation $\mathbf{y}$, and returns a higher value the less accurate the attempted translation is. The goal of training the model then becomes finding a value for f that gives back the lowest value for $\ell(f(\mathbf{x}), \mathbf{y})$, as that represents the system that can most accurately translate English text into French.

In practice, the transformations applied by the model to an input are too complex to feasibly capture in a written formula. So for illustrative purposes, assume a simple model that applies a linear transformation to the input $\mathbf{x}$ by multiplying it with weights W and adding bias b, i.e., $f(\mathbf{x}) = W\mathbf{x} + b$. The process of finding the right formula f is an iterative one that involves repeatedly tweaking the values W and b. At each timestep t, the model f is given an English sentence $\mathbf{x}$. It uses its values for W and b to arrive at an attempted translation $f(\mathbf{x})$. Next, the value of $\ell(f(\mathbf{x}), y)$ indicates how accurate the translation is. If having a slightly lower value for W would have resulted in a lower value for $\ell(f(\mathbf{x}), y)$, its values are scaled down accordingly in $t + 1$. Ideally, this eventually results in values for W and b that allow f to determine an accurate French translation for a given English sentence $\mathbf{x}$.

At timestep 0, the model has not encountered any information about the task it is supposed to learn, so there is no meaningful way to attribute values to W and b. Instead, at this point, the values of the weights are sampled from a standard distribution, which is called *random initialization*. It is well documented that the way these values are initialized influences the performance of the model after training (e.g., McCoy et al., 2020; Khurana et al., 2021). What this means is that – since weight initialization is a stochastic process – two identical models trained on the exact same data may end up with different results because their weights had different initial values. When fine-tuning a pre-trained model, the function f already has values for W and b that were learned during pre-training. This function is then given data from a different, task-specific dataset, and the existing values for W and b are modified to allow the model to do the new task. For the values for W at timestep $t+1$, this can be written as $W_{t+1} = W_t + \delta W$, with δW being the suggested change in weights obtained from $\ell(f(\mathbf{x}), y)$. Using this notation, the function for a fine-tuned model can then be rewritten as $f(\mathbf{x}) = \alpha(W_0\mathbf{x} + b_0) + (1 - \alpha)(\delta W\mathbf{x} + \delta b)$, where α is a value between 0 and 1 that indicates what share each part contributes to the value of $f(\mathbf{x})$. The values W_0 and b_0 represent the values for W and b the language model learned during pre-training, which – in this notation – remained frozen during fine-tuning. The values for δW and δb represent the overall changes in values of W and b during the fine-tuning process, and can be thought of as the task-specific weights and bias in relation to the pre-trained values of W_0 and b_0.

Language model-based approaches to NLP tasks typically do not involve pre-training the model. Instead, an existing pre-trained model is often used instead. This means that when different models are fine-tuned using the same base model, the values for W_0 and b_0 are identical across all instances. The initial values of δW and δb are not randomly initialized, but obtained from $\ell(f(\mathbf{x}), y)$. Nevertheless, in such cases, different models are still found to perform differently after training, meaning that the values of δW and δb do differ between instances. This is due to the fact that any number of parts of the training pipeline might be stochastic, depending on the exact method used. Examples include the order in which the data is presented to the model during fine-tuning, or the way in which the values of W and b are updated after each step. These random processes can be controlled between training instances with what is called a *random seed*: Two models that have an identical architecture, use the same training setup, and share a random seed, are – in theory – identical after training is completed.

2.1.3 Generalizability

The use of language models has resulted in an undeniable improvement over traditional NLP methods, and advancements since have been rapid. In 2018, Wang et al. introduced GLUE (General Language Understanding Evaluation), a benchmark platform consisting of a diverse set of sentence-level and sentence-pair classification tasks. At the time, accuracies of around 60 and 70 percent were reported. But with the introduction of BERT less than a year later, these already exceeded 80 % or even 90 % (Devlin et al., 2019). More recently, exceedingly large language models were reported to consistently attain accuracies of over 90 % (e.g., Chowdhery et al., 2022).

The main constraint in model evaluation is that the data that is used to evaluate the model has not been used during training already. This is usually achieved by taking a single data set, and splitting it randomly into one part the model is trained on, and one part the model is evaluated on, and keeping the latter part separate. This approach aims to ensure that the training and test data are independent and identically distributed (i.i.d.), meaning they come from the same statistical distribution. The performance of neural language models is usually evaluated using i.i.d. methods, but the implicit assumption that the real-world data the model will eventually encounter is also i.i.d. with the test data is not always valid. Indeed, there is growing evidence that this does not always reflect the models' ability to generalize to new data in the way a human would (Hupkes et al., 2023). Examples of poor generalization by language model-based architectures are well documented. One such example was found by Niven and Kao (2019) who used a BERT-based model to classify argumentative structures in a text. Their model achieved an accuracy close to that of a human control group, but was found to do so by relying on the presence of the word *not* in a given sentence. While this spurious correlation proved helpful in the experimental setup, reliance on so-called shortcuts causes models to generalize poorly to real-world data. A focused overview of this issue within the scope of the task under scrutiny in this thesis is given in Section 2.2.3. The importance of good generalization is widely recognized in the field of NLP. Nevertheless, as Hupkes et al. (2023) point out, systematic testing for generalization is not yet common practice.

2.2 Natural Language Inference

Generalization issues between random seeds can occur regardless of the fine-tuning task. This section introduces the task under scrutiny in this thesis: Natural Language Inference, and motivates why this task was chosen in particular.

2.2.1 Task Outline

Natural Language Inference (NLI) is an NLP task that involves determining the relationship between a *premise* and a *hypothesis*, which is typically either a relation of entailment, contradiction, or neutrality. To illustrate, take the premise *All dogs have fur*. A hypothesis that it would entail could be *My dog has fur*, since it is necessarily true given the premise. An example of a contradiction would be *No dogs have fur*, since it cannot possibly be true given the premise. A neutral statement might be *Dogs are popular pets*, since the premise is not relevant to the truth value of the hypothesis.

Early approaches to NLI used a variety of techniques. Jijkoun et al. (2005) achieved significant improvement over random chance with a system that measured lexical similarity between the premise and hypothesis; MacCartney et al.'s (2008) approach relied on the lexical alignment of hypothesis and premise, and achieved just over 60 % accuracy; and Hickl et al. (2006) enriched an approach of probabilistic lexical alignment with manually engineered, lexico-semantic features to achieve an accuracy of around

75 %.[1] Like with other tasks, the advent of pre-trained neural language models led to a considerable improvement over traditional NLP methods. The paper that introduced BERT and RoBERTa respectively report an NLI accuracy of 86.7 % and 90.2 % (Devlin et al., 2019; Liu et al., 2019).

2.2.2 Datasets

Various datasets are used in NLI research, depending on the specific goal at hand. However, a general and widely used dataset is MNLI, which contains many natural, English-language sentence pairs from a wide variety of sources. The MNLI dataset will be used in this thesis along with HANS, a challenge set that targets the exploration of prediction heuristics in language model.

MNLI

Williams et al. (2018) introduced the Multi-Genre Natural Language Inference (MNLI) corpus. At the time, the only large human-annotated corpus for NLI was the Stanford NLI Corpus (SNLI; Bowman et al., 2015). In their paper, Williams et al. (2018) emphasize the role of NLI as a benchmark task for testing natural language understanding (NLU) in language models. But they argue that the SNLI corpus fails to provide an adequate test ground for language models in this regard. The SNLI corpus was generated by showing English-language image captions to human annotators and asking them to think of three sentences: one that is definitely true given the text; one that might be true; one that is definitely not true. These sentences were then used as a hypothesis for the *entailment*, *neutral*, or *contradiction* labels, respectively.

One issue with this technique is that all texts come from the same genre, namely image captions. As a result, the SNLI corpus is argued to be insufficiently diverse linguistically. That is to say, various grammatical constructions that a model would have to be able to deal with are insufficiently present in the training data. In particular, since image captions describe a static, visual scene, they generally lack sentences that require displacement, e.g., sentences about hypothetical situations or sentences with past or future tense verbs. Consequently, the dataset was already too easy for language models at the time for it to serve as an effective benchmark for NLU. With comparisons between the strongest models being inhibited by a ceiling effect, as models fell just a few percentage points short of human accuracy.

The main goal of the MNLI corpus was to provide a large corpus for NLI similar to SNLI that would also serve as a benchmark for NLU. To this end, Williams et al. (2018) generated a corpus in a manner similar to what Bowman et al. (2015) did, but using English-language texts from ten different genres, including both spoken and written texts and texts with different levels of formality, all to ensure the language in the corpus is as diverse as possible. The training set involved five of the ten genres, and models can either be evaluated on a *mismatched* test set, which has different genres, or a *matched* test set, which uses texts from the same genres as the training set. The corpus has since become a significant dataset for NLI research, as it provides a large-scale dataset of 433k annotated sentence pairs. The wider variety of language use cases in the model has additionally provided a more robust testing ground for language models compared to SNLI.

[1] Note that both MacCartney et al. (2008) and Hickl et al. (2006) used a setup where the model had two classification options, rather than the three outlined earlier.

HANS

The Heuristic Analysis for NLI Systems (HANS) dataset was introduced by McCoy et al. (2019) with the goal of identifying why language models fine-tuned for NLI make the predictions they make. Language models had been found to leverage statistical artefacts in the training data, rather than learn the underlying linguistic generalizations (Gururangan et al., 2018). However, these artefacts may not be present in real-world data outside of the experimental setting, and lead to incorrect assumptions. By curating a dataset around several artefacts language models are known to rely on, a test set can be created that can help identify which specific artefacts a specific model uses for its predictions. Each sentence pair in the dataset either has an *entailment* or *non-entailment* label, contrary to the more common three-way distinction used in NLI. The sentence pairs all contain one of three heuristics:

1. Lexical Overlap
   (a) *Premise:* The lawyer was advised by the actor.
   (b) *Hypothesis:* The actor was advised by the actor
   (c) *Hypothesis:* The lawyer advised the actor.
2. Subsequence
   (a) *Premise:* The judges heard the actors resigned.
   (b) *Hypothesis:* The actors resigned.
   (c) *Hypothesis:* The judges heard the actors.
3. Constituent
   (a) *Premise:* Before the actor slept, the senator ran.
   (b) *Premise:* If the actor slept, the senator ran.
   (c) *Hypothesis:* The actor slept.

Each of these heuristics is intended to yield an entailment labeling by the language model, but that is not necessarily always the correct label. The premise in 1a entails the hypothesis in 1b, but not the one in 1c. However, a model that uses lexical overlap between the premise and hypothesis will fail to recognize the non-entailment between 1a and 1c, since all words in 1c are also in 1a. Sequential models might rely on the presence of the hypothesis as an entire sequence in the premise, and may fail to recognize that 2c is not entailed by 2a. Models might also use the presence of the hypothesis as a constituent in the premise. The hypothesis in 3c occurs as a constituent in both 3a and 3b, but if it is headed by *if* like in 3b, it does not necessarily entail 3c.

For each heuristic, five more specific subcases were generated where the heuristic yields an entailment, and five where it does not. Each subcase had 10,000 sentence pairs, resulting in a dataset of 30,000 items. Sentence pairs were generated automatically using a fixed template and vocabulary. For example, the template for 3b and 3c would be 'If the N_1 V_1, the N_2 V_2' and 'The N_1 V_1.' Possible nouns and verbs were checked to ensure all sentences were plausible. Each pair was made to evoke an *entailment* prediction from models that rely on the heuristic, regardless of whether the hypothesis is actually entailed by the premise. If a model then consistently predicts entailment relations for sentence pairs in a given heuristic, that would indicate the model leverages that particular heuristic. As such, the HANS dataset can provide a detailed insight into how language models arrive at their predictions. A detailed overview of the subcases and the information they provide is given in Section 3.2.

2.2.3 NLI in Generalization Research

Part of the appeal of NLI as a task is that it is considered to require a human-like understanding of language and reasoning. However, Hupkes et al. (2023) point out that NLI systems in particular have been found to rely on unintended strategies to reach their decision, and consequently, that they fail to generalize well.

Ideally, a model learns to use information from a given premise to reason whether or not it entails a given hypothesis. Poliak et al. (2018) trained a neural language model only on hypotheses, but found that this did not inhibit the model from outperforming a majority baseline. This suggests that reasoning from a given premise is not required for a model to gain good performance. Gururangan et al. (2018) confirmed this by discovering that language models rely heavily on artifacts such as the presence of hypernym relations, word overlap, and negations. Subsequent research found that this strategy prevented models from generalizing well, as these artifacts seem to be dataset specific. For example, Talman and Chatzikyriakidis (2019) found that some of the state-of-the-art models in NLI often failed to reach an accuracy greater than 65 % when tested on a dataset different from the one they were trained on, even though the datasets were designed for the exact same task.

Several attempts have since been made to ensure language models use human-like reasoning. In 2020, Kalouli et al. added a classification step to their NLI system that classified inferences as being either simple or difficult. By passing the simple inferences to a neural language model-component, and the difficult inferences to a symbolic engine, they managed to outperform architectures consisting of just a neural language model on datasets specifically intended to measure generalizability. More recently, Zhou and Tan (2021) explored what the effect was of adding explanations to the training data. That is, in addition to giving the models a premise and a hypothesis, adding a sentence that explains why the hypothesis follows logically from the premise or not. They found that language models were able to generate explanations to their inferences quite well, but that it did not lead to improved generalizability.

Generalization issues in NLI have been investigated by looking at differences in the datasets and model architecture. But as mentioned in Section 2.1.2, two instances of the same model architecture trained and evaluated on the same data are still known to exhibit behavior indicative of poor generalization due to differences in how they were initialized. McCoy et al. (2020) trained 100 BERT-based classifiers on the MNLI dataset. The models differed only in the random seed used for fine-tuning the classifier, but were otherwise identical. Following training, the models were evaluated on the MNLI development set to evaluate their in-distribution generalizability, and on HANS to evaluate their out-of-distribution generalizability. The models were found to generalize to the in-distribution data quite consistently, but they were found to vary significantly in the accuracies they attained for the different heuristics captured by HANS. In other words, different instances of the same BERT-based model were found not to learn the same task in a manner that was consistent across instances. As stated in Section 2.1, model architectures have become more sophisticated since the introduction of BERT. Like McCoy et al. (2020), Bhargava et al. (2021) trained various models on

MNLI and evaluated them on HANS to assess their out-of-distribution generalizability, including a RoBERTa-based classifier. This model was found to generalize much better to HANS than the BERT-based models. However, unlike McCoy et al. (2020), Bhargava et al. (2021) did not investigate how much their model's generalizability varied across random seeds.

2.3 Research Focus

In summary, language models have become abundant in NLP and have led to impressive advancements in the field. However, there is evidence to suggest that these improvements may be inflated. Many neural language models leverage dataset-specific artifacts that they cannot use outside of the experimental setting they were introduced in, which has been reported to be especially true for language models fine-tuned for NLI. Interestingly, even two instances of BERT-based models that are identical, apart from their random seed, have been found to vary in how well they generalize to out-of-distribution data. This is indicative of a degree of instability that is not commonly captured in evaluation: Firstly, because out-of-distribution evaluation methods remain less popular than i.i.d. evaluation; secondly, because results are often reported for a single model instance. Newer language models such as RoBERTa have been reported to generalize more robustly to out-of-distribution data than BERT does. However, it is not clear whether this also equates improved robustness to cross-random seed variation.

The goal of this thesis is to assess the degree to which a language model's random seed inadvertently affects its ability to judge inferences. Such an analysis could provide a thorough understanding of what language models are currently capable of when it comes to NLI. Moreover, comparing the degree to which BERT-based and RoBERTa-based models are affected by changes to the random seed helps to judge the degree to which RoBERTa is an improvement over BERT. Finally, an investigation of how susceptible language models are to changes in the random seed can place the impressive advancements made by such models into perspective. To this end, several research questions will be answered:

> **RQ1:** To what degree do the changes in the training setup of RoBERTa-based models improve generalizability and stability across different random seeds when compared to BERT?
>
> **RQ2:** How does the overlap of mistakes made by models differing only in their random seed vary on the instance-level?
>
> **RQ3:** What types of inferences are particularly challenging for language models?

In addressing these research questions, this thesis has two main contributions. Not only does it provide a more thorough understanding of the variance in generalizability between random seeds as reported by McCoy et al. (2020), it also explores whether the augmented robustness provided by RoBERTa-based architectures extends to increased stability across random seeds.

Chapter 3

Methodology

3.1 Fine-Tuning and Evaluating the Language Models

The process of fine-tuning language models is outlined in Section 2.1.2 using an abstract formalization. The purpose of this section is to provide a more concrete description of the fine-tuning process and to explain how it was employed in this thesis. Next, this section outlines how the models were evaluated. As stated, two different types of models were used, namely BERT and RoBERTa. The two models share the same architecture, but they mainly differ in how they are pre-trained (see Section 2.1.1). Therefore, fine-tuning works the same for both model types. In-distribution and out-of-distribution evaluation was carried out using the MNLI and HANS datasets, respectively.

3.1.1 Fine-Tuning

Prior to fine-tuning the language model for a more specific task, the model has already acquired a broad understanding of language by training on masked language modeling and – in the case of BERT – next sentence prediction. Training models on these tasks results in models capable of encoding natural language into representations that are useful for a broad range of downstream NLP tasks. Fine-tuning then teaches the model how to employ these representations for specific tasks. For NLI, the fine-tuning process starts with the first premise-hypothesis pair in the training data. This pair of sentences broken down into a single sequence of tokens — words or subword units. A classification token ([CLS]) is added to the start of the sequence, and separator tokens ([SEP]) to separate the premise from the hypothesis and to indicate the end of the hypothesis are added to the end of each sentence. This sequence of tokens is then turned into a sequence of word embeddings that represent the tokens numerically in a way that captures semantic information about the token. Next, this sequence is enriched with segment embeddings, which indicate the sentence each token belongs to, and positional embeddings, which indicate the position of each token in the sequence. This enriched sequence is then fed into the first of a number of so-called *transformer blocks*.

Transformer blocks each contain two components: a multi-head attention layer and a feed-forward neural network. The sequence first passes through the attention layer, which quantifies the relationship of each token with respect to every other token in the sequence. For instance, when considering the premise *The dog is happy* and the hypothesis *The pet is happy*, the attention mechanism might assign a high score to the token *dog* in relation to the token *pet*. This is because the semantic connection between these two words could be useful for recognizing entailment. Each item of the output sequence then passes through the same feed-forward neural network independently, which applies some transformation to each item of the sequence. What these neural networks do exactly is not directly interpretable, but can be broadly understood as transforming the input representations into a more useful format. For example, the initial embedding for *dog* might somehow emphasize the fact that it is a carnivorous animal. If the attention layer managed to highlight the fact that *dog* relates strongly to *pet*, the feed-forward neural network could then pick up on this relation, and transform the embedding for *dog* to shift the focus from it being a carnivorous animal to it being a domestic animal. The output of the first transformer block then serves as the input to the next transformer block. With each transformer block, the representation of the input sequence becomes increasingly abstract. There is evidence to suggest that the initial block mainly picks up on more local, lower-level features of language, such as word forms or parts-of-speech (Tenney et al., 2019). The subsequent blocks progressively capture more complex linguistic patterns, with intermediate layers often identifying long-range syntactic relations within the text. The final blocks are generally responsible for encoding high-level semantic information, such as the overall meaning of a sentence or complex relationships between entities (Jawahar et al., 2019). The models used for this thesis are `bert-base` and `roberta-base`, which each stack twelve of these transformer blocks.

After passing through all twelve blocks, the output corresponding to the [CLS] token is used to make the prediction. Because of the attention mechanism, this token is carries information of every other token in the sequence. This output is fed to a simple neural network that assigns probabilities to each possible class, in this case *entailment*, *neutral*, or *contradiction*. After the first sentence pair in the training data, this neural network has not yet learned to reliably obtain probabilities form the [CLS] token, and its output is unlikely to be correct. As explained in Section 2.1.2, the output it generates is compared to the correct answer, and all components of the model are retroactively tweaked in a way that would have resulted in an output that would have been closer to the correct answer. With each new sentence pair, the model should then generate outputs that are increasingly close to the correct answer.

Barring some minor details, this process is identical for both BERT and RoBERTa. However, the differences in the pre-training setup mean that RoBERTa's transformer blocks encode the sequence in a different – and ostensibly more effective – manner. What this means is that output corresponding to the classification token might contain a more complete representation of the input sequence, which allows the final segment of the model to make predictions more reliably.

For this thesis, thirty distinct instances of RoBERTa were fine-tuned to the MNLI training set. This number deviates from the 100 instances fine-tuned by McCoy et al. (2020) due to the computational constraints involved with fine-tuning such large models. The only variation between the models was the random seed applied during fine-tuning, which affected the order in which the items of the training set were presented, as well as the random initialization of the weights of the final segment of the model. Training lasted for three epochs, that is, the model went over the entire training set three times. Training parameters were kept in line with those used by McCoy et al. (2020) to facilitate comparisons with their BERT instances. No BERT models were fine-tuned for this thesis. Instead, the publicized outputs of McCoy et al.'s (2020) BERT instances were used for comparisons between the two model types. A custom script was used to fine-tune and evaluate the RoBERTa-based models, using the `transformers` library for obtaining and handling the pre-trained RoBERTa model, and the `datasets` library to obtain the datasets used. Fine-tuning was carried out on the DAS-5 system (Bal et al., 2016) and took approximately 18 hours per model.

3.1.2 Evaluation

Following the fine-tuning process, each model was evaluated on two datasets, described in detail in Section 2.2.2. The in-distribution evaluation used the MNLI *matched* development set, as the test set is not publicly available. This set contains the same genres of text contained in the training set, yet none of the texts in the development set were encountered during training, effectively making it a test set suitable for i.i.d. evaluation. Conversely, out-of-distribution evaluation was carried out using the HANS dataset. With the models having been fine-tuned to the MNLI dataset, the labels they assign to a premise-hypothesis pair is one of *entailment*, *neutrality*, or *contradiction*. However, it should be noted that the HANS dataset treats NLI as a binary classification task with each pair either representing entailment or non-entailment. As such, the labels for *contradiction* and *neutral* were aggregated into a single *non-entailment* label for evaluation on HANS. Evaluation relied on the accuracy — the number of correct predictions as a share of the total number of predictions.

Evaluation can be broken down into two stages. The first stage involves a comparison between different instances of RoBERTa. Ideally, differences in the random seed should have little influence on the model after fine-tuning, in which case the models should attain roughly the same accuracy. However, even if they score similarly, an evaluation on the instance level is still warranted. If all models attain an accuracy of 0.95, that means they mislabel 5 % of sentence pairs in the evaluation set. However, as reported by Khurana et al. (2021), it may well be the case that there is little overlap between the 5 % of sentence pairs mislabeled by each model. If this is found to be the case, that means the random seed still affects which types of inference the models struggle with after fine-tuning. The second stage involves a comparison between the RoBERTa instances fine-tuned in this thesis and the BERT instances fine-tuned by McCoy et al. (2020). While this necessarily involves a comparison between their overall performance, the main interest is the differences between cross-instance stability between BERT and RoBERTa. As Bhargava et al. (2021) report that a RoBERTa-based model fine-tuned to MNLI generalizes to HANS more robustly than a BERT-based model, it would be interesting to see whether they are also more robust to changes in the random seed. This would mean that the RoBERTa-based instances show less variance in the overall accuracy attained than McCoy et al.'s (2020) BERT-based instances, but also that the overlap in errors made is greater for the RoBERTa-based models.

The BERT instances were not reported to show much cross-instance variance on their in-distribution evaluation, and the RoBERTa instances were not expected to do so, either. Therefore, the core of the analysis lies in the out-of-distribution analysis using the HANS dataset. As stated in Section 2.2.2, premise-hypothesis pairs in the HANS dataset are generated from a fixed template that falls under one of three heuristics, which are in turn subdivided into ten subcases. Performance on the HANS dataset was broken down for each of these subcases to allow for a thorough linguistic analysis of how language models deal with NLI. The next section delves into the different HANS heuristics, and how performance on each of them sheds light on the linguistic capabilities of the models used. Nevertheless, a more thorough instance-level of MNLI is also warranted. Since the prediction is made based on a single, largely semantic representation of the sentences, it would be interesting to see what the link is between model performance and semantic similarity between the premise and hypothesis. Moreover, with the entire sequence being squeezed into one [CLS] token, it is not unreasonable to assume that models struggle more with longer input sequences. Since HANS sentences are highly synthetic, these two aspects vary little between sentence pairs in the dataset. Therefore, MNLI will be used for this purpose, instead.

3.2 Analysis of HANS Heuristics

Each heuristic in HANS is designed around a certain type of linguistic information. This then allows for an investigation of the extent to which an NLI model uses that information to arrive at its prediction. One of the goals of this thesis is to find out which types of inferences are particularly difficult for language models. With this in mind, this section aims to provide a thorough overview of the different heuristics and subcases in the dataset, and how performance on each of them is to be interpreted.

3.2.1 Lexical Overlap

This heuristic is built on the assumption that a premise entails all hypotheses constructed from words in the premise, and is thought to be especially difficult for models that use a so-called *bag-of-words* approach to language representation — models that forego word order when processing linguistic input. Crucially, while all words in the hypotheses of this heuristic appear in their respective premises, they do not do so as a contiguous sequence, as such cases fall under the *Subsequence* heuristic. As mentioned in Section 2.2.1, early approaches to NLI often used lexical similarity as a feature or even as their basis, and achieved above-chance accuracy in doing so (e.g., Jijkoun et al., 2005; Hickl et al., 2006; MacCartney et al., 2008). Five of the subcases in this heuristic contain cases where lexical overlap between the premise and hypothesis is consistent with entailment in various predictable patterns. However, out of those five, four are paired with subcases where an over-reliance on lexical overlap would yield an incorrect prediction of entailment.

Two of the subcases deal with passive structures. In one of those two, a premise *The president was advised by the manager* entails the corresponding hypothesis *The manager advised the president.* Ideally, a model would have acquired knowledge of passive structures, which it would then use to correctly label this pair. However, a model based solely on lexical similarity would do so, too, since all words in the hypothesis occur in the premise. Therefore, the other subcase would have paired this premise with the hypothesis *The president advised the manager.* A strictly lexical model would not be able to differentiate the two, but a model that can adequately deal with passives would be able to recognize that there is no entailment here. Another subcase pair deals with conjunctions; the premise *The presidents believed the doctor and the scientist* entails the hypothesis *The presidents believed the scientist*,[1] but not *The scientist believed the doctor*, for instance. Knowledge of word order is needed to correctly identify the subject and object of the premise, and to see whether they match

[1] Note that in this subcase, the subject of the hypothesis is never the first element of the conjunction, such as in *The presidents believed the doctor* as that would fit under the *Subsequence* heuristic.

in the hypothesis. Moreover, this heuristic tests a model's ability to recognize how both conjuncts of *and* share their syntactic role in these sentences.

Two other subcase pairs test a model's ability to deal with modifiers on the subject, either as a prepositional phrase or as a relative clause. For a premise like *The judge behind the authors avoided the scientists*, a model would have to recognize that the noun phrase *the authors* is contained within a prepositional phrase, and is not an argument of the main verb. Needless to say, a model would also have to recognize which syntactic roles the other two noun phrases play in the sentence. The hypothesis that is entailed by this premise would be *The judge avoided the scientists*, and an example of a hypothesis where the label would be *non-entailment* would be *The authors avoided the judge*. A premise that contains a relative clause – such as *The artists who encouraged the scientists introduced the actor* – has the additional challenge of differentiating the verb in the subordinate clause from the verb in the main clause. A model that has this ability should assign an *entailment* label when this premise is paired with the hypotheses *The artists encouraged the scientists* and *The artists introduced the actor*. Conversely, various hypotheses could be generated that should get a *non-entailment* label if a model is able to properly identify each of the elements in the premise. For example, a hypothesis such as *The actor introduced the artists* might be mistakenly considered as being entailed by a model that is able to recognize which elements are arguments of which verb, but not able to recognize which is the subject or which is the object. On the other hand, a model that is able to correctly identify subjects and objects, but struggles to differentiate the main clause from the subordinate clause might consider the hypothesis *The artists encouraged the actor* as being entailed. While the dataset does contain these different templates, they all fall under the same subcase, and are not differentiated in the analysis. Therefore, this subcase tests a more general ability of models to deal with relative clauses.

One separate subcase that deals with relative clauses tests a model's ability to untangle them. Each example in this subcase has the *entailment label*, and pairs premises like *The doctor who the managers admired thanked the secretary* with a hypothesis *The managers admired the doctor*. What is challenging here is that the object of the verb *admired* is moved from its usual position to the top of the clause. Furthermore, the object of *admired* in the hypothesis is *the doctor*, but in the premise, its object is *who*, which has the doctor as its antecedent. As such, this subcase is challenging in two different ways: First, a model should be able to deal with the fact that the object is not in its usual position; and second, the model would have to identify that *who* and *the doctor* have the same referent. Nevertheless, it is unlikely that even basic NLI models would struggle to identify that *The doctor who the students admired thanked the secretary* entails *The students admired the doctor*, due to the lexical overlap between the two. Consequently, if models do not find this subcase comparatively difficult when compared with the arguably easier subcases in this heuristic, it might be a good indication that the models leverage lexical overlap as a heuristic for identifying entailment.

3.2.2 Subsequence

This heuristic is built on the assumption that a premise entails all of its subsequences, and is thought to be especially difficult for models that parse language sequentially. Unlike the *Lexical Overlap* the hypothesis in its entirety must be a contiguous subsequence of the premise. For subcases where the label is *entailment*, this is usually achieved by removing part of the premise that does not add to the truth value. For subcases where the label is *non-entailment*, subsequences usually cross a noncontiguous part of the premise's parse tree.

In order for a premise to entail a hypothesis, the hypothesis must be true in all cases where the premise is true. However, it is possible for the premise to be false and the hypothesis to still be true. The hypothesis could thus be seen as a larger set of particular situations. Whenever at least one of these situations holds true, the hypothesis is also true. The premise would then be a subset of the hypothesis, where each situation contained by the premise is also in the set denoted by the hypothesis. Consequently, whenever at least one situation in the premise set holds true, the hypothesis is automatically true, too, since this situation is contained by the hypothesis by definition. In other words, whenever a specific description of a given situation is true, a less specific version of that description is true, too. This is how the *entailment*-labeled premise-hypothesis pairs in this heuristic are formed. For example, the hypothesis *The scientist read* is entailed by the premise *The scientist read the report* because the truth value of the hypothesis is not contingent on what exactly the scientist read. Similarly, a verb might have two subjects or objects through conjunction, for instance in *The doctor and the professor recognized the senator.* As a premise, this denotes a situation contained by a larger set of situations that fit the hypothesis *The professor recognized the senator.* The other three subcases of this type involve removing some modifying element from one of the verb's arguments. with the brackets denoting the hypothesis of the larger premise, this can either be the adjective, such as in *Helpful* [*athletes arrived*]; a relative clause, such as in [*The managers stopped the tourists*] *who performed*; or a prepositional phrase, such as in [*The lawyers advised the athlete*] *near the senators.*

Pairs where the premise does not entail the hypothesis in this heuristic require the models to have some degree of knowledge of the underlying structure of the sentence. An inability to do this is usually due to a misattribution of the syntactic role of one of the elements in the sentence. For example, with the premise *The students heard the president resigned*, the hypothesis *The students heard the president* would be given the *entailment*-label if *the president* is incorrectly seen as the object of the verb *heard.* One other subcase involves having a subordinate clause that ends in a verb that can be used transitively precede a main clause that starts with a possible object of that verb, such as in *Because* [*the senators studied the professor*] *recommended the lawyers.* The three other subcases involve a modifying phrase on the subject that ends with a noun phrase, which is then directly adjacent to the subject's verb. This phrase can either be a prepositional phrase, such as in *The bankers next to* [*the senator arrived*]; a relative clause, such as in *The author that avoided* [*the doctor slept*]; or a past participle, such as in [*The students helped in the museum*] *shouted.*

None of the sentences are ambiguous, so the misattribution of syntactic roles necessarily involves an incorrect parsing of the sentence at hand. Therefore, an inability to recognize a lack of entailment in the latter five types of subcases might be said to be indicative of impaired knowledge of the structure underlying language.

3.2.3 Constituent

This heuristic is built on the assumption that a premise entails all complete subtrees in its parse tree, and is thought to be especially difficult for models that use a structural representation of the sentence pairs. This goes one step further than the *Subsequence* heuristic, because a subsequence is not necessarily a complete subtree of the larger sequence. To illustrate, see Figure 3.1, which displays a parse tree for the possible

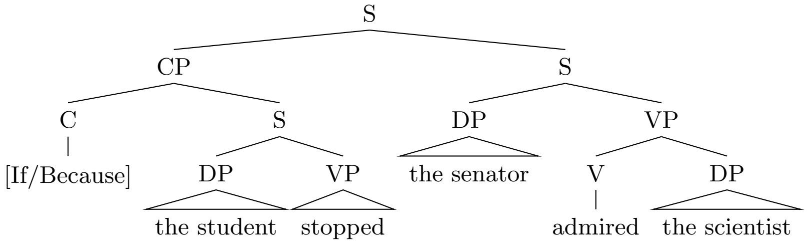

Figure 3.1: Combined parse tree for the sentences *If the student stopped, the senator admired the scientist* and *Because the student stopped, the senator admired the scientist.*

premise [*Because/If*] *the student stopped the senator admired the scientist.* Here, *The student stopped the senator* is a valid subsequence, but not a valid subtree, and no interpretation of the full sentence is possible where *the senator* is the object of the verb *stopped.*

Each subcase of this heuristic comes in a pair, where one is consistent with an *entailment* label and the other with a *non-entailment* label. These subcases are syntactically identical, and the models require lexico-semantic information of the heads of the subtrees to make the correct prediction. This is also shown in Figure 3.1, where the sentence does not change structurally depending on whether *if* or *because* is used, however, the change is significant semantically. Take the more general pattern '[If/Because] P, Q' with antecedent P and consequent Q. A conjunction like *if* signals a conditional relationship between P and Q. In the semantics of conditionals, the truth value of the consequent is contingent on the truth value of the antecedent. However, the truth of the antecedent is not guaranteed by the conditional statement itself. Therefore, the hypothesis[2] *If the student stopped, the senator admired the scientist* entails neither *The student stopped*, nor *The senator admired the scientist.* Conversely, the conjunction *because* is used to express a causal relationship between an antecedent and a consequent, and presupposes the truth of both the cause *The student stopped* and the effect *The senator admired the scientist.* A model that suffers from an over-reliance on syntactic knowledge might miss this lexico-semantic distinction.

This type of knowledge is tested in a number of different ways in this heuristic. A template with the premise '[Perhaps/Of course] Q' for the hypothesis Q tests knowledge of whether a given adverb necessitates the truth value of the phrase it modifies, where it would entail Q if the adverb is *of course*, but the truth value of Q is unknown if it is *perhaps.* The template 'x [knows/thinks] that Q' tests knowledge of the level of evidentiality expressed by the verb introducing the clause that contains Q, where – assuming the whole statement is true – Q is entailed if the verb is *knows*, but not if it is *thinks.* Finally, the template 'P [and/or] Q' tests whether models understand the logical difference between disjunction expressed by *or*, and conjunction expressed by *and.* An inability to do any of these is indicative of an over-reliance on structural information, as opposed to the lexico-semantic differences that make these pairs distinct.

[2] The comma is included for legibility here, but it would not be included in the HANS dataset, as it would eliminate the need for syntactic knowledge to understand the sentence.

3.2.4 Overview of Implications

Since all heuristics are designed to trigger an entailment prediction, it is not particularly informative of model behavior if models frequently correctly identify entailment. That either means they have acquired the linguistic knowledge that is required to check the inference at hand, but it may also be the case the model has leveraged the heuristic for its prediction. Similarly, while it is informative if the models erroneously attribute an *entailment* label, it is a clear indication that the models mistakenly used the heuristic in its prediction and lacks the linguistic understanding necessary to arrive at the correct label. Instead, it is more interesting when the models assign a *non-entailment* label, either correctly or incorrectly. While this was also discussed in the previous sections, the table below provides a more concise overview of the implications of a *non-entailment* prediction for each subcase.

Lexical Overlap (Consistent)	**Untangling Relative Clauses** *The athletes who the judges saw called the manager. → The judges saw the athletes.* Indicates the models fail to resolve that the relative pronoun is the direct object of saw and/or that it corefers with the subject of the main clause.
	Sentences with PPs *The tourists by the actor called the authors. → The tourists called the authors.* Indicates that the models fail to properly identify the subject of the verb. It might indicate the model fails to grasp the underlying structure, as the verb is not headed by the noun directly adjacent to it.
	Sentences with Relative Clauses *The actors that danced encouraged the author. → The actors encouraged the author.* Indicates that the model fails to identify the subject of the verb as it is not directly adjacent to it.
	Conjunctions *The secretaries saw the scientists and the actors. → The secretaries saw the actors.* Indicates the model fails to resolve that the conjunct that is kept in the hypothesis shares its syntactic role with the conjunct that is adjacent to the verb, using the example above, that both *the scientists* and *the actors* are direct objects of *saw*.
	Passives *The authors were supported by the tourists. → The tourists supported the authors.* Indicates the model lacks a linguistic understanding of passive constructions.

Lexical Overlap (Inconsistent)	**Subject-Object Swap** *The senators mentioned the artist.* ↛ *The artist mentioned the senators.* Indicates the model successfully identified the subject and the direct object of the verb.
	Sentences with PPs *The judge behind the manager saw the doctors.* ↛ *The doctors saw the manager.* Indicates the model successfully identified the syntactic role of the noun in the PP.
	Sentences with Relative Clauses *The actors called the banker who the tourists saw.* ↛ *The banker called the tourists.* Indicates the model successfully identified the syntactic role of the noun in the relative clause.
	Conjunctions *The doctors saw the presidents and the tourists.* ↛ *The presidents saw the tourists.* Indicates the model successfully identified the syntactic role of the conjuncts.
	Passives *The senators were helped by the managers.* ↛ *The senators helped the managers.* Indicates the model has a linguistic understanding of passive constructions.
Subsequence (Consistent)	**Conjunctions** *The actor and the professor shouted.* → *The professor shouted.* Indicates the model fails to recognize both conjuncts share a syntactic role.
	Adjectives *Happy professors mentioned the lawyer.* → *Professors mentioned the lawyer.* Indicates the model fails to recognize the adjectives makes the premise a more specific description of the action denoted by the hypothesis, and that if it is true for a given situation, that the more specific description is necessarily also true.
	Understood argument *The author read the book.* → *The author read.* Indicates the model fails to recognize the presence of the subject makes the premise a more specific description of the action denoted by the hypothesis.

Subsequence (Consistent)	**Relative clause on object** *The artists avoided the actors that performed.* → *The artists avoided the actors.* Indicates the model fails to recognize the presence of the relative clause makes the premise a more specific description of the action denoted by the hypothesis. **PP on object** *The authors called the judges near the doctor.* → *The authors called the judges.* Indicates the model fails to recognize the presence of the prepositional phrase makes the premise a more specific description of the action denoted by the hypothesis.
Subsequence (Inconsistent)	**NP/S** *The managers heard the secretary resigned.* ↛ *The managers heard the secretary.* Indicates the model understands that the noun to the right of the verb is the start of a new clause, rather than its direct object. **PP on subject** *The managers near the scientist shouted.* ↛ *The scientist shouted.* Indicates the model understands the noun to the left of the verb is embedded in a prepositional phrase and cannot be the subject of the verb. **Relative clause on subject** *The secretary that admired the senator saw the actor.* ↛ *The senator saw the actor.* Indicates the model understands that the noun to the left of the verb of the main clause is embedded in a relative clause, and cannot be its subject. **MV/RR** *The senators paid in the office danced.* ↛ *The senators paid in the office.* Indicates the model understands the covert passive structure in the relative clause on the subject, and that the initial noun of the premise is not the subject of the verb it is adjacent to. **NP/Z** *Before the actors presented the doctors arrived.* ↛ *The actors presented the doctors.* Indicates the model understands the verb in the hypothesis is embedded in a prepositional phrase in the premise, and that its object in the hypothesis cannot be its object in the premise.

Constituent (Consistent)	**Embedded under preposition** *Because the banker ran, the doctors saw the professors. → The banker ran.* Indicates the model does not understand the preposition signals causality, and that, consequently, the preposition it heads is entailed.
	Outside embedded clause *Although the secretaries slept, the judges danced. → The judges danced.* Indicates the model does not understand that the preposition outside of the embedded clause is entailed because of the type of preposition heading the embedded clause.
	Embedded under verb *The president remembered that the actors performed. → The actors performed.* Indicates the model does not understand that a clause headed by the type of verb in the premise is entailed because of it.
	Conjunctions *The lawyer danced, and the judge supported the doctors. → The lawyer danced.* Indicates the model does not understand a conjunction that expresses logical conjunction entails both of its conjuncts.
	Adverbs *Certainly the lawyers advised the manager. → The layers advised the manager.* Indicates the model does not understand the certainty expressed by the adverb means the phrase it modifies is entailed.
Constituent (Inconsistent)	**Embedded under preposition** *Unless the senators ran, the professors recommended the doctor. ↛ The senators ran.* Indicates the model understands the preposition signals conditionality, and that, consequently, the preposition it heads is not entailed.
	Outside embedded clause *Unless the authors saw the students, the doctors resigned. ↛ The doctor resigned.* Indicates the model understands that the preposition outside of the embedded clause is not entailed because of the type of preposition heading the embedded clause.
	Embedded under verb *The tourists said that the lawyer saw the banker. ↛ The lawyer saw the banker.* Indicates the model understands that a clause headed by the type of verb in the premise is not entailed because of it.

Constituent (Inconsistent)	**Disjunction** *The judges resigned, or the athletes saw the author. ↛ The athletes saw the author.* Indicates the model understands a conjunction that expresses logical disjunction does not entail its conjuncts. **Adverbs** *Probably the artists saw the authors. ↛ The artists saw the authors.* Indicates the model understands the uncertainty expressed by the adverb means the phrase it modifies is not entailed.

Chapter 4

Results

4.1 In-Distribution Evaluation

The models were first evaluated on the MNLI *matched* development set. This data is independent and identically distributed with respect to the data on which the model was trained. In theory, poor generalizability should not inhibit the model from performing well on this test set. The published results obtained from McCoy et al.'s (2020) BERT-based models are presented side by side with the results of the RoBERTa-based models in Figure 4.1. The plot on the left shows the share of model instances that attained a particular accuracy. The plot on the right shows the different error overlap ratios for each model type. This is a pairwise metric that is obtained by dividing the size of the intersection of errors made by two models with the size of the union of errors made. Two models that make the exact same mistakes will attain an error overlap ratio of 1.0, whereas for two models where there is no overlap, this value is 0.0. As such, the error overlap ratio serves as a pivotal metric in assessing the consistency of model instances across random seeds.

The plot on the left shows that RoBERTa modestly but consistently outperforms BERT on MNLI. The RoBERTa and BERT instances attained a mean accuracy of 0.86 and 0.84, respectively, with neither model type having much variation between instances. These results indicate that both models have successfully learned from the identically distributed training data and that the random seed had little influence on downstream performance. The error overlap ratio is significantly higher for the BERT instances. Two instances of BERT usually agree on around two-thirds of mistakes made,

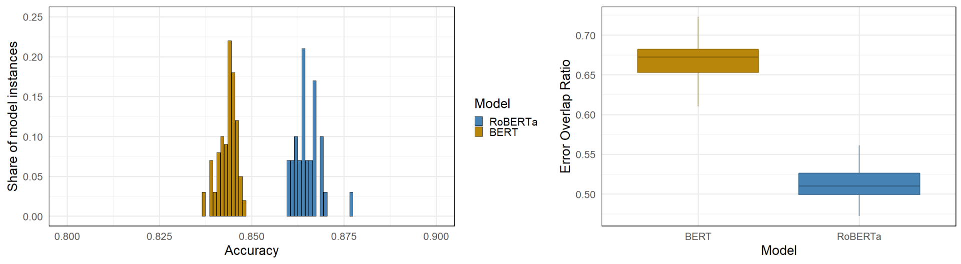

Figure 4.1: Overall and instance-level in-distribution evaluation. Left: Share of model instances for a given accuracy score for in-distribution evaluation. Right: Boxplot of the error overlap index for RoBERTa and BERT instances.

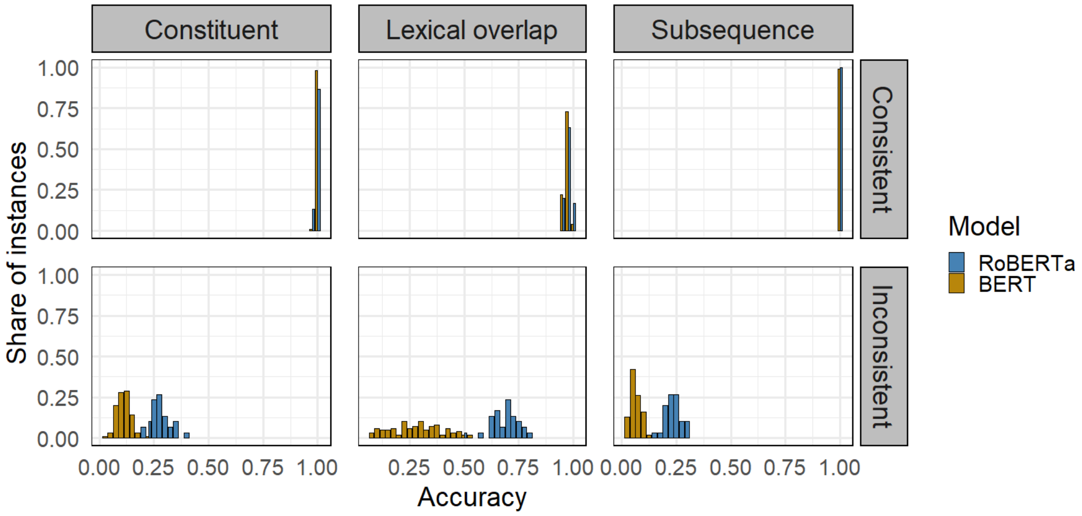

Figure 4.2: Share of model instances per accuracy score, broken down per HANS heuristic, as well as whether the heuristic yields an *entailment* label (consistent) or a *non-entailment* label (inconsistent).

whereas this overlap is only around 50 % for two given instances of RoBERTa. Given that there are three possible labels in the MNLI dataset, both of these numbers are well above chance. Fleiss' kappa was calculated for both groups of models to quantify the agreement on all predictions, rather than just the mistakes. This metric determines the level of agreement over what would be expected by chance and can range between -1 and 1, with a negative value indicating worse agreement than would be expected by chance, a value of 0 indicating the same level of agreement as what would be expected by chance, and a value of 1 indicating perfect agreement. These results also show that agreement is higher among BERT models ($\kappa = 0.76$) than for RoBERTa models ($\kappa = 0.63$), though both values show a level of substantial agreement.

In summary, the in-distribution evaluation reveals nuanced insights into the effect of the random seeds on BERT instances when compared to RoBERTa instances. On the one hand, RoBERTa demonstrates a slight edge over BERT in terms of accuracy. Moreover, the random seed was not found to have much influence on the downstream performance of either type of model. Concurrently, BERT appears to exhibit a greater degree of agreement between instances, as evidenced by the higher error overlap ratio and Fleiss' kappa value. This is indicative of more consistent behavior between instances of BERT than between instances of RoBERTa. In other words, while RoBERTa consistently finds a superior strategy for identifying entailment, there is more variation between the strategies it converges to depending on the random seed. BERT, on the other hand, is more consistent in the method it employs for NLI.

4.2 Out-of-Distribution Evaluation

Out-of-distribution evaluation was carried out on the HANS dataset. An analysis that goes into the linguistic implications of the performance per subcase is presented in Section 4.3. More generally, data from a different distribution than what the models were trained on – such as HANS – tests the models' generalizability in a way in-

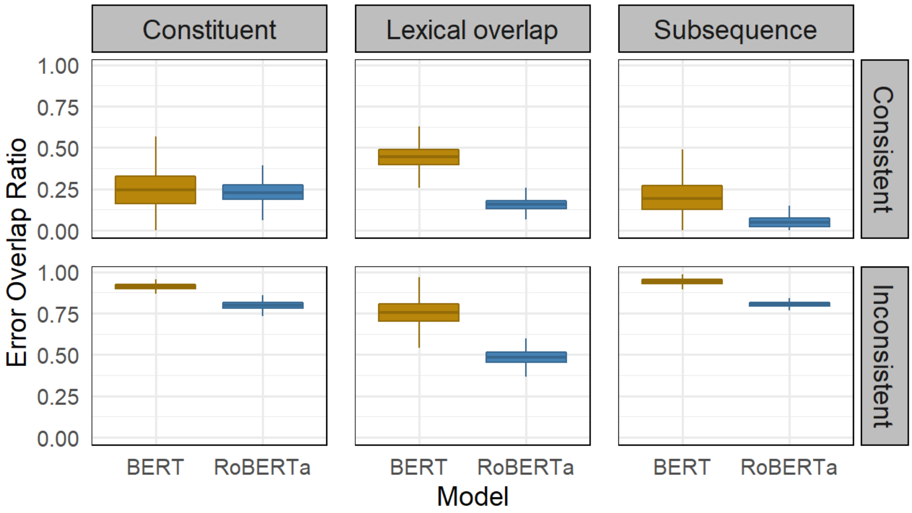

Figure 4.3: Error overlap ratio between model pairs, broken down per HANS heuristic, as well as whether the heuristic yields an *entailment* label (consistent) or a *non-entailment* label (inconsistent).

distribution data does not. As stated in Section 2.2.2, premise-hypothesis pairs in HANS contain one of three types of heuristics language models are known to leverage in NLI. In HANS, these heuristics can either be consistent with entailment, in which case *entailment* is the correct label, or inconsistent, in which case the correct label is *non-entailment*. Since models have a tendency to overwhelmingly predict *entailment* in the entire dataset, results are split between these levels of consistency.

Figure 4.2 shows the share of model instances per accuracy score, broken down per heuristic and consistency level. For pairs where the heuristic is consistent, both types of model achieve a near perfect accuracy. This ceiling effect leaves little room for variation both between model types and between instances of the same type of model. The *Lexical Overlap* heuristic appears to have been more difficult, but here too, there does not seem to be a noteworthy difference between BERT and RoBERTa. This finding is discussed further in Section 4.3.2.

There are clear differences between the model types for cases where the correct label is *non-entailment*. For the *Constituent* and *Subsequence* heuristics, both BERT and RoBERTa score well below chance, with BERT labeling around 10 % of such pairs correctly, and RoBERTa around 25 %. Moreover, the different instances deviate substantially from the mean accuracy, which suggests the random seed had a considerable effect on downstream performance on these types of examples. For the *Lexical Overlap* heuristic, most instances of RoBERTa score well above chance, although there is still a lot of variation between instances. The different instances of BERT do not seem to exhibit agreement to any meaningful degree. Some instances mislabel nearly all sentences in this category, whereas others score just above chance. Performance on these types of premise-hypothesis pairs is thus highly dependent on the random seed used.

Figure 4.3 shows the error overlap ratio broken down for each heuristic and consistency with entailment in HANS. When the heuristic is consistent with entailment, the models rarely make mistakes. The overlap between the errors they do make is generally quite low, usually ranging between 10 % and 25 %. For the *Constituent* heuristic, BERT and RoBERTa instances have comparable error overlap ratios, though there is more variation between different pairs of BERT instances. For the *Subsequence* heuristic, the error overlap ratios are also similar between the two types of models, though it tends to be higher for pairs of BERT instances. For the *Lexical Overlap*, where the accuracy was slightly lower, RoBERTa models have similarly low error overlap ratios, but instances of BERT have substantially higher overlap ratios.

For sentences where the heuristic is inconsistent with entailment, models generally scored below chance. Consequently, the error overlap is very high, with little variation between different pairs. The one heuristic where RoBERTa instances usually did better than random chance is *Lexical Overlap*, and here the error overlap is much lower. Instances of BERT displayed significant variation in their accuracies, but the overlap is still around 75 % for this category.

	BERT		**RoBERTa**	
	Con.	**Inc.**	**Con.**	**Inc.**
Lexical overlap	0.649	0.653	0.268	0.634
Subsequence	0.722	0.746	0.105	0.875
Constituent	0.573	0.571	0.376	0.873

Table 4.1: Values for Fleiss' kappa for both model types broken down per heuristic and whether the heuristic is consistent or inconsistent with entailment.

Across the board, the error overlap ratios are higher for BERT than for RoBERTa. However, this is not necessarily indicative of agreement since errors usually constitute more than half of the labeled sentence pairs. The larger the share of errors made, the larger the overlap of errors is going to be just by random chance. Fleiss' kappa measures the level of agreement over what would be expected by chance, and is consequently not affected by the large number of mistakes made. Table 4.1 shows the value of Fleiss' kappa for each model and each category. BERT displays a similar and substantial level of agreement regardless of whether the correct label is *entailment* or *non-entailment.* RoBERTa, on the other hand, shows only slight agreement for sentences where the heuristics are consistent with entailment, versus substantial agreement on sentences where the *Lexical Overlap* heuristic is inconsistent with entailment, and almost perfect agreement on the other two heuristics.

To sum up, both models overwhelmingly predict *entailment* for all sentence pairs in the dataset, which results in near perfect accuracies on cases where the heuristic is consistent with entailment, and accuracies below random chance when it is not. This finding indicates that both types of models leverage the heuristics used in HANS when recognizing entailment. RoBERTa, however, does so to a lesser degree, which indicates it has done a better job at acquiring a more general understanding of natural language inferences. In terms of accuracy, there was a substantial level of variance between model instances when the correct label was *non-entailment*, this is particularly true for BERT instances and accuracy on the *Lexical Overlap* heuristic.

Interestingly, on the sentence pair-level, variation between model instances was more noticeable. With errors either being ubiquitous or scarce depending on the correct label, the overlap in errors was respectively either substantial or low. As a result, Fleiss' kappa is a better metric to use to measure the agreement between model instances. BERT was found to have a similar and substantial level agreement across the board, whereas RoBERTa was found to have little agreement on the sentence pairs where mistakes were rare, but near perfect agreement on sentences where mistakes were common.

4.3 Linguistic Analysis

The goal of this section is to answer the question what types of inferences are particularly difficult for language models. The focus of this analysis lies on the HANS heuristics, as those allow for a thorough investigation of the linguistics involved with judging inferences. However, the highly synthetic nature of the sentences in HANS make some worthwhile analyses impossible. As discussed in Section 3.1.2, these analyses check for the relation between sequence length and the performance, and the semantic similarity and the performance. The results of these analyses are presented in Section 4.3.1. The subcase-level performance of the HANS dataset is analyzed in the following sections.

4.3.1 Analyses on MNLI

It might be the case that semantic similarity between the premise and hypothesis is used by language models in their predictions, given that the [CLS] token used in classification contains high-level semantic information of the entire input sequence. Moreover, it might be the case that capturing the semantic intricacies necessary for NLI is more difficult for longer sequences, in which case you would expect to find a negative correlation between sequence length and model performance. To investigate this, three numbers were calculated for every item in the MNLI development set: The length of the sequence in tokens; the share of models of each type that predicted the label correctly; and the cosine similarity between the premise and the hypothesis. In NLP, the cosine similarity is used as a metric that captures the semantic similarity between two embedded texts, as it measures the distance between two embeddings.

The first step was to obtain embeddings for each sentence in the dataset. To this end, the large, English-language pipeline package `en_core_web_lg` by spaCy was used. The sentences were first tokenized, and the tokens were embedded. All tokens were counted to obtain the length of the sentence. For obtaining the semantic similarity, however, frequent stopwords were removed from the sentences, as those typically do not add much to the meaning of the sentence, but would still hold an equal weight in the sentence embedding. The mean value of the remaining tokens was taken to obtain an embedding for the whole sentence. The cosine similarity between the premise's sentence embedding and the hypothesis' sentence embedding was then taken to be the semantic similarity between the two.

The correlation between model performance and semantic similarity and sequence length was measured by calculating the R^2 for both BERT and RoBERTa. This is a number between 0 and 1 that indicates how well two variables correlate, with 0 being no correlation and 1 being perfect correlation. Neither the sequence length, nor the semantic similarity was found to correlate with the results of either BERT or RoBERTa. The R^2 was between 0.00 and 0.05 in all cases. A substantial number of sentence pairs was labeled correctly by all model instances, which may have diluted a possible correlation. However, removing such cases to focus on examples the models did not agree on did not yield a greater R^2. Therefore, semantic similarity or sentence length cannot be said to be a major part of the strategy employed by either BERT or RoBERTa.

4.3.2 Lexical overlap

Subcase	**Minimum**	**Maximum**	**Mean**	**Std. dev.**
Untangling Relative Clauses	0.98	1.00	0.99	0.00
	0.94	1.00	0.98	0.01
The athletes who the judges saw called the manager. → *The judges saw the athletes.*				
Sentences with PPs	0.98	1.00	0.99	0.00
	0.98	1.00	1.00	0.00
The tourists by the actor called the authors. → *The tourists called the authors.*				
Sentences with Relative Clauses	0.92	0.98	0.95	0.01
	0.97	1.00	0.99	0.01
The actors that danced encouraged the author. → *The actors encouraged the author.*				
Conjunctions	0.77	0.96	0.89	0.05
	0.72	0.92	0.83	0.05
The secretaries saw the scientists and the actors. → *The secretaries saw the actors.*				
Passives	0.99	1.00	1.00	0.00
	0.99	1.00	1.00	0.00
The authors were supported by the tourists. → *The tourists supported the authors.*				
Subject-Object Swap	0.67	0.99	0.91	0.06
	0.00	0.66	0.19	0.17
The senators mentioned the artist. ↛ *The artist mentioned the senators.*				
Sentences with PPs	0.67	0.90	0.83	0.05
	0.04	0.76	0.41	0.18
The judge behind the manager saw the doctors. ↛ *The doctors saw the manager.*				
Sentences with Relative Clauses	0.57	0.86	0.77	0.06
	0.09	0.67	0.33	0.14
The actors called the banker who the tourists saw. ↛ *The banker called the tourists.*				
Conjunctions	0.53	0.82	0.67	0.08
	0.12	0.72	0.45	0.15
The doctors saw the presidents and the tourists. ↛ *The presidents saw the tourists.*				
Passives	0.01	0.41	0.18	0.10
	0.00	0.04	0.01	0.01
The senators were helped by the managers. ↛ *The senators helped the managers.*				

Table 4.2: Results for the HANS subcases for which the *Lexical overlap* heuristic. Results for RoBERTa are in the top row and results for BERT in the bottom row.

The *Lexical overlap* heuristic relies on the full lexical overlap of the hypothesis with the premise, but not as a contiguous sequence. A detailed breakdown of the accuracies per subcase in this heuristic is shown in Table 4.2. As discussed in the previous section, models were more likely to incorrectly assign a *non-entailment* label to sentences of this heuristic than for the others. It appears that this is largely due to the *Conjunctions* subcase, e.g., recognizing that the premise *The secretaries saw the scientists and the actors* entails the hypothesis *The scientists saw the actors.* An analysis of whether sentences with conjunctions co-occur more often with non-entailment labels in the training data yielded no conclusive results, so these findings are unlikely to be caused by a statistical artefact related to the presence of the word *and.* Since the hypothesis specifically does not occur as a contiguous sequence in the premise here, the task at hand is to recognize that the conjunct that is not adjacent to the verb shares its syntactic role with the conjunct that is, i.e., that *the actors* is also a subject of *saw* despite not being adjacent to it. Interestingly, the *Subsequence* heuristic also has a *Conjunction* subcase where the conjunct that is adjacent to the verb is the one that is kept in the hypothesis. As can be seen in Table 4.3, performance on this subcase in the *Subsequence* heuristic is much higher. This makes sense, because both BERT and RoBERTa take sequentiality into account when representing a sequence. For all other subcases in this heuristic, the premises contain subsequences that could be valid hypotheses, but those would not be valid subtrees of the hypothesis. For example, *the actor called the authors* is a subsequence of *The tourists by the actor called the authors.* But since *called* is not headed syntactically by *the actor*, viewing it as the subject of *called* would necessarily involve misparsing the premise. It might be the case that the models erroneously reject the hypothesis in this heuristic because there is another valid hypothesis that is a subsequence of the premise. The fact that they do not struggle with the other subcases – despite there being valid subsequences that could be the hypothesis – would then be explained by the fact that models recognize that these are not valid subtrees of the premise.

For BERT, the *Lexical Overlap* heuristic stands out in cases where the heuristic is inconsistent with entailment. As can be seen on page 26 in Figure 4.2, the models are unusually inconsistent in the accuracies they attain for this heuristic. This is also reflected in the high standard deviations shown in Table 4.2. Interestingly, RoBERTa does not have this same degree of instability. The variance between instances of RoBERTa is similar for all three heuristics with *non-entailment* as the correct label. Moreover, not a single instance of RoBERTa scored below chance on any of the subcases but the *Passives* subcase, whereas the mean accuracies of the BERT instances never exceed random chance for all subcases that are inconsistent with entailment. The *Passives* subcase in this heuristic contains examples such as *The senators were helped by the managers* and *The senators helped the managers.* For such sentences, RoBERTa-based models vary considerably between instances, with the mean accuracy being 0.18. Nevertheless, they still show considerable improvement over their BERT counterparts, as BERT instances consistently fail to assign a *non-entailment* label, with the maximum accuracy being just 0.04. Taken together, these results imply that RoBERTa instances rely less on lexical overlap between premise and hypothesis when looking for entailment.

4.3.3 Subsequence

Subcase	Minimum	Maximum	Mean	Std. dev.
Conjunctions	0.90	1.00	0.97	0.02
	0.93	1.00	0.98	0.02
The actor and the professor shouted. → *The professor shouted.*				
Adjectives	0.98	1.00	1.00	0.00
	1.00	1.00	1.00	0.00
Happy professors mentioned the lawyer. → *Professors mentioned the lawyer.*				
Understood argument	1.00	1.00	1.00	0.00
	0.95	1.00	1.00	0.01
The author read the book. → *The author read.*				
Relative clause on object	0.99	1.00	1.00	0.00
	0.98	1.00	0.99	0.01
The artists avoided the actors that performed. → *The artists avoided the actors.*				
PP on object	0.99	1.00	1.00	0.00
	1.00	1.00	1.00	0.00
The authors called the judges near the doctor. → *The authors called the judges.*				
NP/S	0.00	0.03	0.01	0.01
	0.00	0.05	0.02	0.01
The managers heard the secretary resigned. ↛ *The managers heard the secretary.*				
PP on subject	0.34	0.67	0.50	0.07
	0.00	0.35	0.12	0.07
The managers near the scientist shouted. ↛ *The scientist shouted.*				
Relative clause on subject	0.31	0.65	0.46	0.09
	0.00	0.23	0.07	0.04
The secretary that admired the senator saw the actor. ↛ *The senator saw the actor.*				
MV/RR	0.02	0.10	0.04	0.02
	0.00	0.02	0.00	0.00
The senators paid in the office danced. ↛ *The senators paid in the office.*				
NP/Z	0.05	0.18	0.10	0.03
	0.02	0.13	0.06	0.02
Before the actors presented the doctors arrived. ↛ *The actors presented the doctors.*				

Table 4.3: Results for the HANS subcases for which the *Subsequence* heuristic. Results for RoBERTa are in the top row and results for BERT in the bottom row.

The *Subsequence* heuristic relies on the presence of the hypothesis in the premise as a contiguous sequence, but not as a valid subtree. The ceiling effect for evaluation on sentences where this heuristic is consistent with entailment is more pronounced for this heuristic than it is for the others, with the RoBERTa-based models attaining the lowest mean accuracy of 0.97 on the *Conjunctions* subcase. BERT-based models score the second-lowest mean accuracy on that same subcase with 0.98. It is noteworthy that all other mean accuracies exceed 0.99, and that the *Conjunctions* subcase is also the one that proved to be the most challenging for the models in the *Lexical overlap* heuristic.

For pairs where the label is *non-entailment*, RoBERTa models outperform BERT models in all but one subcase, the *NP/S* subcase. This subcase contains pairs like *The managers heard the secretary resigned* and *The managers heard the secretary* that require the knowledge that a verb like *heard* can introduce a subordinate clause, and that, in such cases, the noun following the verb is not its direct object. Neither type of model seems to have picked up on this knowledge during training, given that the mean accuracy for both groups is close to zero. RoBERTa scores considerably higher on two of the subcases in this category, namely the *PP on subject* and *Relative clause on subject* subcases. Both of these involve a phrase modifying the subject and separating it from its verb. Overcoming this non-adjacency requires syntactic knowledge that some of the RoBERTa instances seem to have acquired, to some extent. The *MV/RR* subcase also involves a modifying phrase separating the verb and subject, yet it still proves highly challenging for the models. An example pair from this subcase has the premise *The senators paid in the office danced* and the invalid hypothesis *The senators paid in the office*. One reason such examples are difficult to parse is the passive structure in the modifying clause, given that the models were found to struggle with passives in the *Lexical overlap* heuristic, too. Another reason is the fact that the object of the verb *paid* is not in the typical object position, but has instead moved up to the start of the clause. Moreover, the sentence lacks an overt conjunction to indicate the start of a relative clause in order to ensure the hypothesis is a valid subsequence of the premise.

In summary, these results indicate that models rely a lot on the presence of the hypothesis as a subsequence of the premise when predicting entailment. Models struggle little with the subcases where there is entailment, but appear to be largely unable to identify non-entailment. However, as could also be seen in the *Lexical overlap* heuristic, RoBERTa in particular seems to be able to deal with simple relative clauses. This indicates an improved ability to correctly identify and label the arguments of a verb. However, this does not extend to complex relative clauses such as those in the *MV/RR* heuristic, where the passive structure, the moved object, or the lack of a conjunction might have inhibited both types of models from correctly labeling *non-entailment.*

4.3.4 Constituent

Subcase	**Minimum**	**Maximum**	**Mean**	**Std. dev.**
Embedded under preposition	0.88	0.97	0.93	0.02
	0.81	1.00	0.96	0.02
Because the banker ran, the doctors saw the professors. → The banker ran.				
Outside embedded clause	1.00	1.00	1.00	0.00
	1.00	1.00	1.00	0.00
Although the secretaries slept, the judges danced. → The judges danced.				

(*Continued*)

Subcase	**Minimum**	**Maximum**	**Mean**	**Std. dev.**
Embedded under verb	0.96	1.00	0.99	0.01
	0.93	1.00	0.99	0.01
The president remembered that the actors performed. → *The actors performed.*				
Conjunction	0.98	1.00	1.00	0.00
	1.00	1.00	1.00	0.00
The lawyer danced, and the judge supported the doctors. → *The lawyer danced.*				
Adverbs	1.00	1.00	1.00	0.00
	1.00	1.00	1.00	0.00
Certainly the lawyers advised the manager. → *The layers advised the manager.*				
Embedded under preposition	0.39	0.91	0.60	0.10
	0.14	0.70	0.41	0.12
Unless the senators ran, the professors recommended the doctor. ↛ *The senators ran.*				
Outside embedded clause	0.00	0.01	0.00	0.00
	0.00	0.03	0.00	0.01
Unless the authors saw the students, the doctors resigned. ↛ *The doctor resigned.*				
Embedded under verb	0.35	0.74	0.54	0.09
	0.02	0.42	0.17	0.08
The tourists said that the lawyer saw the banker. ↛ *The lawyer saw the banker.*				
Disjunction	0.00	0.07	0.02	0.01
	0.00	0.03	0.00	0.01
The judges resigned, or the athletes saw the author. ↛ *The athletes saw the author.*				
Adverbs	0.04	0.37	0.17	0.10
	0.00	0.17	0.06	0.04
Probably the artists saw the authors. ↛ *The artists saw the authors.*				

Table 4.4: Results for the HANS subcases for which the *Constituent* heuristic. Results for RoBERTa are in the top row and results for BERT in the bottom row.

The *Constituent* heuristic targets models that rely on the presence of the hypothesis as a constituent of the parse tree of the premise. For this heuristic, the subcases come in pairs where the syntactic structure of the premise and hypothesis is the same, but a lexical change between the two subcases shifts the correct label from *entailment* to *non-entailment*. Therefore, this heuristic requires lexico-semantic knowledge to determine whether a premise entails a hypothesis, since syntax is not informative. Somewhat unsurprisingly, the subcase where models struggled more to correctly assign an *entailment* label is also the subcase where they most successfully assigned *non-entailment* labels, namely the *Embedded under preposition* subcase. This subcase connects two propositions with a preposition that either indicates a causative relation, in which case there is entailment, or a conditional relation, in which case there is no entailment. For instance, the preposition *the tourist danced* is entailed in the causative example *Because the tourist danced, the scientist resigned* but not in the conditional example *If the tourist danced, the scientist resigned.* The fact that the models misclassified pairs that did have entailment relatively often indicates that the models do not associate the syntactic structure used in this subcase with entailment as much as they might do with those of the other subcases. Conversely, the fact that they correctly identified non-entailment relatively often might indicate that the models do have some knowledge of which prepositions express causality and which ones express conditionality.

However, one other subcase uses this exact same structure, yet it proved to be the most challenging subcase in the entire dataset. In the easier *Embedded under preposition* subcase, the hypothesis is the proposition that is embedded by the preposition. But in the more difficult *Outside embedded clause* subcase, the hypothesis is the other proposition, i.e., *the scientist resigned* in the example above. If the preposition at hand indicates causality between the propositions, the models recognize entailment without failure. However, when the preposition indicates conditionality, virtually all models fail to recognize the lack of entailment. So while RoBERTa in particular seems to have some knowledge of which prepositions signal conditionality, they only appear to be able to extend this knowledge to the clause headed by the preposition.

Both types of models demonstrate near-perfect accuracy on all other subcases where the correct label is entailment. However, this proficiency did not always extend to an inability to recognize non-entailment as performance is rarely above chance in these instances. One other heuristic with relatively high performance is the *Embedded under verb* subcase, which involves knowledge of evidentiality expressed by verbs. For example, the hypothesis *The lawyer saw the banker* is entailed by the premise *The tourists remembered that the lawyer saw the banker*, but not by *The tourists said that the lawyer saw the banker*. BERT-based models were quick to label either pair as expressing entailment, but the RoBERTa-based models scored modestly above chance on average on cases where there was no entailment. This subcase also proved considerably less challenging for the BERT-based models than other subcases of this heuristic where the correct label was *non-entailment*. However, they do still score well below chance.

In summary, these results show that neither BERT nor RoBERTa has sufficiently acquired the lexico-semantic knowledge necessary to deal with most subcases of this heuristic. Since syntactic information cannot be used to recognize non-entailment here, the inability of BERT in particular to correctly assign the *non-entailment* label might be indicative of reliance on syntactic structure. If it is true that transformer-based models such as BERT and RoBERTa capture syntactic information in the earlier layers and higher-level semantic information in later layers, it might be that larger models would perform better on this heuristic. If this is the case, poor performance on this heuristic might be attributed to the models being too small to be able to capture the semantic information necessary.

Chapter 5

Discussion & Conclusion

The objective of this thesis was to investigate the degree to which RoBERTa is affected by changes to the random seed, especially in comparison to BERT, and to provide a deeper insight into what such models are capable of when it comes to NLI. Overall, RoBERTa was found to consistently outperform BERT in terms of accuracy. In HANS, accuracy on sentences where the correct label was *entailment* was close to 1.00, which left little room for meaningful differences between the two models. The performance on the part of the dataset where the correct label is *non-entailment* showed RoBERTa generalizes more robustly to out-of-distribution data, which is in line with the findings by Bhargava et al. (2021). Nevertheless, in this part of the dataset, its performance was often still well below chance, and the effect of the random seed on downstream performance was found to be substantial. The fact that both models attained high accuracies on MNLI means they both managed to learn from the training data well, regardless of the random seed. However, the varying performance on HANS indicates that BERT and, to a lesser degree, RoBERTa tend to leverage statistical artefacts when making their predictions. The high variation in performance for examples in HANS where there was no entailment is consistent with the notion that the random seed has a considerable effect on the degree to which these artefacts are used. So while different instances of the same model may achieve a similar accuracy, the random seed still introduces variations in how they approach and complete the task, leading to differences on the sentence-level.

This conclusion is corroborated by the finding that – even when there is little variation between instances in terms of accuracy – agreement between instances is still variable. On MNLI, any given pair of BERT instances shares about two-thirds of their errors, whereas for RoBERTa, the error overlap ratio is around 50 %. This is above chance, and Fleiss' kappa shows a substantial level of agreement for both types of models, with the agreement for BERT models being higher than for instances of RoBERTa. The predictions on HANS suffer from an imbalance, as models overwhelmingly assign the *entailment* label. What this means is that for the part of the dataset where this is correct, errors are very rare, whereas for the other part of the dataset, errors are ubiquitous. Consequently, the error overlap ratio in this latter part is very high, since errors usually make up more than half of the predictions. For the other part, the error overlap ratio is quite low. As such, Fleiss' kappa is a more informative metric, as it measures the level of agreement above what would be expected by random chance. On the part of the HANS dataset where *entailment* was the correct label, RoBERTa models showed only slight agreement, whereas BERT models had a substantial level of agreement.

However, on the part where the correct label was *non-entailment*, RoBERTa models had a near perfect level of agreement, whereas BERT was found to have a similar level of agreement as it had on the other part of the dataset.

In summary, on in-distribution data, RoBERTa makes more correct predictions than BERT does, but the level of agreement between RoBERTa instances – while substantial – is lower than that of BERT models. Moreover, the overlap in errors made is consistently lower for pairs of RoBERTa instances than it is for pairs of BERT instances. On HANS, the models were quick to assign the *entailment* label to sentences in the entire dataset, which indicates that the models leverage the heuristics used in HANS when recognizing entailment, which is indicative of poor generalizability. RoBERTa, however, was found to be consistently better at recognizing non-entailment in the HANS dataset, which may mean that it has learned the task in a more reliable manner. BERT models were found to have a substantial level of agreement regardless of the gold label. On the other hand, RoBERTa was found to have low agreement on the part where mistakes were rare, and high agreement on the part where mistakes were common.

For both BERT and RoBERTa, the vast majority of premise-hypothesis pairs in the dataset fall onto the entailment-side of the decision boundaries the models converge to, regardless of whether there actually is entailment or not. This is not surprising, as the HANS dataset was deliberately designed to be difficult for language models. In other words, finding a decision boundary that adequately separates the pairs in HANS with entailment from the pairs without entailment is not an easy task. The models arrive at different decision boundaries depending on the random seed, but instances of BERT are fairly consistent in which examples fall on which side of the boundary. However, instances of RoBERTa consistently find a boundary that puts the same pairs without entailment on the right side of the boundary, but the few pairs with entailment that erroneously fall onto the non-entailment side of the decision boundary are quite variable between instances.

While this discrepancy is striking – especially since it was not found in BERT – it is important put into perspective. It is very uncommon for either type of model to mislabel a sentence pair that exhibits entailment. Therefore, the key finding seems to be that RoBERTa consistently does better than BERT at correctly identifying non-entailment in HANS, which points at an improved generalizability to out-of-distribution data. This finding is strengthened further by the fact that RoBERTa shows a high level of agreement on this part of the dataset. The fact that BERT struggles on this part of the dataset is indicative of an over-reliance on the heuristics in HANS. If this is the case, it is not unsurprising that BERT was found to have higher agreement on the MNLI dataset. However, in this case, it would not necessarily be correct to take the higher agreement to mean better generalizability for BERT. So while the random seed appears to have a larger influence on the strategy RoBERTa converges to, this does not appear to translate to a larger influence of the random seed on the generalizability of RoBERTa instances.

The linguistic analysis of the different subcases in HANS found that both models rely heavily on the presence of the hypothesis as a subsequence of the premise. RoBERTa did appear to be better at identifying modifying clauses than BERT, which implies that the model encodes syntactic information more effectively. The *Constituent* heuristic was argued to require semantic information as opposed to syntactic information. While the models were shown to have acquired this type of information to some degree, their capacity to employ this adequately remains limited. To illustrate, given a preposition that expresses a causality relation between two clauses, many model instances were able to identify that this means the clause embedded by the preposition is not entailed. However, the models appeared unable to extend this non-entailment to the other clause, too. Since semantic information is increasingly encoded in the input representation as the input passes through more layers of a language model, it might be the case that an increase in model size would result in improved performance on the *Constituent* heuristic, in particular. More generally, the models appeared to mostly struggle with complex syntactic structures and high-level semantic information. While the RoBERTa models in particular were found to use linguistic information that goes beyond the surface level, the models appear to lack the depth necessary to judge inferences in a human-like manner.

5.1 Future Research

The methodology of this thesis built heavily on that of McCoy et al. (2020) and, in the initial stage, involved swapping the model used. However, this modification turned out to be less trivial than anticipated. Constraints in computational resources meant that obtaining the RoBERTa instances took up a larger amount of time than was initially allocated to this process. Consequently, several ideas for analyses could not be carried out in time, an overview of which is given below.

The HANS dataset allows for a thorough insight into the linguistic capabilities of the models evaluated on it. The analysis as presented in this thesis could have been made even more granular by breaking results down per template, rather than just per subcase. For example, as stated in Section 3.2.1, the premise *The artists who encouraged the scientists introduced the actor* might be paired any of the following hypotheses, depending on the template:

1. *The artists encouraged the scientists*;
2. *The artists introduced the actor*;
3. *The actor introduced the artists*;
4. *The artists encouraged the actor.*

If the model is able to recognize that the premise does not entail hypothesis 3, that indicates the model is able to distinguish subjects from objects; if it can correctly identify that hypothesis 4 is not entailed, it shows the model managed to separate the main clause from the subordinate clause. Another example might be the subcase that tests a model's ability to recognize the causality expressed by prepositions such as *because*, *since*, or *although.* It would be interesting to see whether these prepositions are equally difficult for the models. Currently, these hypotheses would be grouped together under the same subcase. As such, breaking the results down even further would have allowed for more conclusive statements about the linguistic capabilities of the models.

It would also be interesting to see how well the findings from the linguistic analysis HANS are reflected by performance on MNLI. For example, given that the models were found to struggle with passives in HANS, it would be interesting to see whether premise-hypothesis pairs that rely on matching a passive and active structure are equally difficult. Identifying the presence of certain heuristics in the natural language examples of

MNLI is no trivial feat, and would likely require a substantial amount of manual analysis. However, given the highly synthetic nature of sentences in HANS, such an analysis on MNLI would likely be more informative about the model's abilities in deployment.

Another possibility might involve investigating whether performance on one subcase is a good predictor of performance on another. It is conceivable that if two model instances converge in local minima that are close to each other, that the types of instances they struggle with are more similar. If this is true, you would expect to find patterns where, if a model struggles with subcase A, they likely do well at subcase B, and vice versa. Such findings might provide insight into a possible link between the decision space and the linguistic capacity of language models.

On the more technical side, it is worthwhile to investigate the influence of changes in the GPU unit the model is fine-tuned on. This is known to affect model performance, and might have been a contributing factor in the lower between-instance overlap in errors displayed by RoBERTa. On the DAS-5 system, the RoBERTa models were not consistently fine-tuned on the same GPU unit. If this effect is sufficiently large, it might be the case that RoBERTa is more robust to changes in the random seed than reported in this thesis.

5.2 Conclusion

This thesis set out to answer three questions. The first goal was to investigate the extent to which RoBERTa's changed setup translated to improved stability and generalizability across random seed when compared to BERT. The RoBERTa models were found to generalize better to out-of-distribution data than BERT, but cross-instance variation in overall performance was largely similar between the two. The results of the evaluation on HANS indicate that both models leverage heuristics which leads to issues of generalizability. Sentence pairs in HANS where the correct label is *non-entailment* were mislabeled in the majority of cases, which means the use of the heuristic misguided the model. Nevertheless, there were several subcases for RoBERTa in particular where the models succeeded in recognizing a lack of entailment, which supports Bhargava et al.'s (2021) finding that RoBERTa generalizes to out-of-distribution data more robustly than BERT does.

The second goal was to investigate the differences between different instances of the same base model. While the results show that the random seed has considerable influence on how the models perform the task, agreement between the models was found to be mostly substantial. Despite the varying strategies, overall performance was not found to be affected much. On in-distribution data, agreement among BERT models was found to be higher. BERT was also found to have a considerably higher level of agreement than RoBERTa on the few cases where they erroneously assign the *non-entailment* label. However, RoBERTa was not only found to be better at identifying non-entailment, it also did so more consistently across instances. A conclusion that is consistent with these findings is that BERT picked up on statistical artefacts in the training data more than RoBERTa did, and did so quite consistently between random seeds. This would explain why BERT was found to have higher agreement on the in-distribution evaluation, and why its performance on HANS is worse. If RoBERTa relied on a more general representation of language, it would have come at the cost of a greater variation in how the different instances execute the task. This might have given RoBERTa an edge on examples in MNLI where a reliance on a specific heuristic would have led to an incorrect prediction. However, the models would then vary in which additional examples it would be able to label correctly. In HANS, it would have allowed different instances of RoBERTa to consistently identify the same sentence pairs that lack entailment, seemingly at the cost of them agreeing on the few examples where the model fail to recognize entailment.

The final goal was to analyze what aspects of NLI are challenging to language models. The models were found to rely on overlap between the premise and hypothesis on the lexical level, particularly if the hypothesis is a contiguous subsequence of the premise. Semantic similarity between the premise and hypothesis or the length of the sequence were not found to correlate with model performance. In general, RoBERTa in particular was found to be able to deal with basic syntactic structures to some extent. However, it appears the model lack the lexico-semantic information necessary to correctly identify certain entailment relations. This seems to indicate the models are able to leverage basic linguistic information in their decision-making process, but lack the depth to adequately capture more complex syntactic and semantic information necessary to learn NLI in a human-like manner.

Bibliography

H. Bal, D. Epema, C. de Laat, R. van Nieuwpoort, J. Romein, F. Seinstra, C. Snoek, and H. Wijshoff. A medium-scale distributed system for computer science research: Infrastructure for the long term. *Computer*, 49(05):54–63, may 2016. ISSN 1558-0814. doi: 10.1109/MC.2016.127.

P. Bhargava, A. Drozd, and A. Rogers. Generalization in NLI: Ways (not) to go beyond simple heuristics. In *Proceedings of the Second Workshop on Insights from Negative Results in NLP*, pages 125–135, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.insights-1.18. URL `https://aclanthology.org/2021.insights-1.18`.

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*, pages 632–642, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL `https://aclanthology.org/D15-1075`.

N. Chomsky. *Syntactic Structures.* Mouton and Co., The Hague, 1957.

A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, et al. Palm: Scaling language modeling with pathways. 2022.

I. Dagan, O. Glickman, and B. Magnini. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*, pages 177–190. Springer, 2005.

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: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://aclanthology.org/N19-1423`.

A. D'Amour, K. Heller, D. Moldovan, B. Adlam, B. Alipanahi, A. Beutel, C. Chen, J. Deaton, J. Eisenstein, M. D. Hoffman, et al. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 2020.

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: Human Language Technologies, Volume 2 (Short Papers), pages 107–112, 2018.

A. Hickl, J. Williams, J. Bensley, K. Roberts, B. Rink, and Y. Shi. Recognizing textual entailment with lcc's groundhog system. In *Proceedings of the Second PASCAL Challenges Workshop*, volume 18, pages 1–4, 2006.

D. Hupkes, M. Giulianelli, V. Dankers, M. Artetxe, Y. Elazar, T. Pimentel, C. Christodoulopoulos, K. Lasri, N. Saphra, A. Sinclair, et al. State-of-the-art generalisation research in nlp: a taxonomy and review. 2023.

G. Jawahar, B. Sagot, and D. Seddah. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1356. URL `https://aclanthology.org/P19-1356`.

V. Jijkoun, M. de Rijke, et al. Recognizing textual entailment using lexical similarity. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*, pages 73–76. Citeseer, 2005.

A.-L. Kalouli, R. Crouch, and V. de Paiva. Hy-NLI: a hybrid system for natural language inference. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5235–5249, Barcelona, Spain (Online), Dec. 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.459. URL `https://aclanthology.org/2020.coling-main.459`.

U. Khurana, E. Nalisnick, and A. Fokkens. How emotionally stable is ALBERT? testing robustness with stochastic weight averaging on a sentiment analysis task. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, pages 16–31, Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eval4nlp-1.3. URL `https://aclanthology.org/2021.eval4nlp-1.3`.

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. 2019.

B. MacCartney, M. Galley, and C. D. Manning. A phrase-based alignment model for natural language inference. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 802–811, 2008.

R. T. McCoy, J. Min, and T. Linzen. BERTs of a feather do not generalize together: Large variability in generalization across models with similar test set performance. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 217–227, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.21. URL `https://aclanthology.org/2020.blackboxnlp-1.21`.

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*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL `https://aclanthology.org/P19-1334`.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. 2013.

T. Niven and H.-Y. Kao. Probing neural network comprehension of natural language arguments. 2019.

F. Pereira. Formal grammar and information theory: Together again? *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 358(1769):1239–1253, 2000.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL `https://aclanthology.org/N18-1202`.

A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme. Hypothesis only baselines in natural language inference. 2018.

A. Talman and S. Chatzikyriakidis. Testing the generalization power of neural network models across NLI benchmarks. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 85–94, Florence, Italy, Aug. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4810. URL `https://aclanthology.org/W19-4810`.

I. Tenney, D. Das, and E. Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL `https://aclanthology.org/P19-1452`.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

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. Williams, N. Nangia, and S. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL `https://aclanthology.org/N18-1101`.

Y. Zhou and C. Tan. Investigating the effect of natural language explanations on out-of-distribution generalization in few-shot NLI. In *Proceedings of the Second Workshop on Insights from Negative Results in NLP*, pages 117–124, Online and

Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.insights-1.17. URL https://aclanthology.org/2021.insights-1.17.