Language Model Pretraining with Lexical Semantic Relations

Andrei Mircea
School of Computer Science
McGill University
Montreal, QC, Canada

May 2023

A thesis submitted to McGill University
in partial fulfillment of the requirements of
the degree of Master of Computer Science

© Andrei Mircea 2023

Abstract

Lexical semantic relations (LSRs) play an important role in systematically generalizing on tasks such as lexical entailment. Notably, several tasks that require knowledge of hypernymy still pose a challenge for recent pretrained language models (LMs), underscoring the need to better align their linguistic behavior with our knowledge of LSRs. In this thesis, we propose Balaur, a model that addresses this challenge by modeling LSRs directly in the LM's hidden states throughout pretraining. Motivating our approach is the hypothesis that the internal representations of LMs can provide an effective interface to their linguistic behavior, and that by controlling one we can influence the other. We verify our hypothesis and demonstrate that Balaur consistently improves the performance of large transformer-based LMs on a comprehensive set of hypernymy-informed tasks, as well as on the original LM objective.

Abrégé

Les relations sémantiques lexicales (LSR) jouent un rôle important dans la généralisation systématique de tâches telles que l'implication lexicale. Notamment, plusieurs tâches qui nécessitent une connaissance de l'hyperonymie posent toujours un défi pour les récents modèles de langage pré-entraînés (LM), soulignant la nécessité de mieux aligner leur comportement linguistique avec notre connaissance des LSR. Dans cette thèse, nous proposons Balaur, un modèle qui relève ce défi en modélisant les LSR directement dans les états cachés du LM tout au long de la préformation. La motivation de notre approche est l'hypothèse que les représentations internes des LM peuvent fournir une interface efficace à leur comportement linguistique, et qu'en contrôlant l'une, nous pouvons influencer l'autre. Nous validons notre hypothèse et démontrons que Balaur améliore les performances de certains LMs sur un ensemble divers de tâches informées par l'hyperonymie, ainsi que sur l'objectif original du LM.

Acknowledgements

The completion of this work has been an arduous yet rewarding experiment that would not have been possible on my own. I am so grateful to the many who have brought their passion, their kindness, and their support throughout this period of my life. You have made this experiment not only possible, but fulfilling.

First and foremost, to my supervisor Jackie Chi Kit Cheung, thank you for providing invaluable guidance in this work, and more generally for fostering a sense of scientific excellence and integrity. Thank you for being a wonderful mentor, generous with your time and with your desire to support us in becoming better researchers. I have learned a lot from you, both on how to be a better researcher and on how to be a better person. Most of all, thank you for caring, your tireless efforts are recognized and deeply appreciated.

To my lab mates, thank you for being in the trenches with me. Thank you for the shared passion in the research we do, the shared grief when the going gets hard, and the shared resolve in seeing it through regardless. I cannot overstate the importance of your solidarity in finding perseverance in the face of adversity, joy in the face of victory, curiosity in the face of mystery, and a sense of fulfilment throughout it all. It has been an absolute pleasure.

To the various organizations that supported and made this research possible, and to the people behind the organizations, thank you. For their administrative support and fostering of community, I would like to thank the McGill Computer Science department, the Reasoning and Learning Lab, and Mila. For their generous funding, I would like to thank the FRQNT (Fonds de Recherche Nature et Technologies du Québec). Lastly, for the fantastic infrastructure support without which this work could not have happened, I would like the thank the Mila IT department as well as The Digital Research Alliance of Canada.

To my friends and family, I will always be thankful for your presence in my life. Throughout this period, it has been all the more important. Thank you.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Challenge of Hypernymy

The field of Natural Language Processing (NLP) has been revolutionized by a class of methods known as pretrained language models (LMs). Typically, these are neural networks trained on large quantities of unlabeled text data with a simple reconstruction objective (e.g. by predicting the next word or randomly masked words in text). Despite the simplicity of such learning objectives, the ability of pretrained LMs such as BERT (Devlin et al., 2019) to learn effective representations of language by training on vast amounts of text data has led to unprecedented progress across multiple NLP tasks (Qiu et al., 2020). However recent work has also shown that these LMs still fail to generalize systematically on tasks requiring understanding of hypernymy (Ettinger, 2020; Ravichander et al., 2020; Hanna and Mareček, 2021; Geiger et al., 2020; Rozen et al., 2021).

Hypernymy is an important aspect of word meaning, or lexical semantics, which describes the taxonomical relationship of between words. In other words, hypernymy captures

the "is-a" relation. For instance, "animal" is a hypernym of "mammal" is a hypernym of "dog", as a dog is a mammal is an animal. The ability of LMs to model this lexical semantic relation is of more than just passing academic interest, as such systems are increasingly deployed in the real world where behaviors inconsistent with our understanding of lexical semantics can have subtle yet pernicious effects. Improving the consistency of model behavior with our understanding of hypernymy therefore becomes a question of increasing model robustness and systematicity, decreasing the risk of errors which are fundamental yet difficult to detect. One failure mode rooted in hypernymy can occur with lexical entailment, where a statement about a hyponym is also by extension a statement about the corresponding hypernym. For example, "the cat is in a taxi" can entail "the cat is in a car" or "the animal is in a car", which in turn can entail many different plausible inferences about the utterance and its meaning. Models which fail to systematically capture such hypernymy relations can fail in unexpected and undetectable ways, e.g. if the cab-faring feline figured in a news article and a question-answering system was asked "Is there a cat in a taxi in this article?" it might answer differently than if asked "Is there an animal in a car?". While this example is fairly benign, as these models are deployed in more and more settings, the risk of such difficult-to-detect mistakes having more severe repercussions increases, and it therefore becomes essential to develop methods that better align model behavior with our understanding of lexical semantics, and of hypernymy in particular.

1.2 Improving Hypernymy in Language Models

Many hypernymy relations are potentially underrepresented in the training data of LMs due to issues of reporting bias and sparsity (Hearst, 1992; Shwartz et al., 2017). This fundamental limitation underscores the need for methods which can complement language model pretraining with external knowledge of hypernymy. Concretely, the hope is that external sources of lexical knowledge such as WordNet (Miller, 1995) can be leveraged to alleviate issues of data scarcity and improve systematic generalization of LMs on tasks requiring knowledge of hypernymy. There also exists a rich body of work that leverages WordNet to evaluate and improve how well lexical semantic relations are captured in distributional methods such as word embeddings. In our thesis, we introduce BALAUR, a method which builds on this past work in several meaningful ways to better align the linguistic behavior of LMs with our knowledge of hypernymy.

In particular, BALAUR builds on semantic specialization, a class of methods typically applied to distributional word embeddings with the goal of better representing and disentangling various lexical semantic relations in one set of vector representations. Semantic specialization methods are typically framed as auxiliary loss functions that impose *constraints* on the original distributional vector space to create systematic structure reflective of semantic relations. With BALAUR, we adapt and apply semantic specialization to the latent representations of LMs with the underlying hypothesis that the internal representations of LMs can provide an effective interface to their linguistic behavior, and that by controlling one we can help guide the other. Concretely, we leverage the fact that latent representations in LMs such as BERT can be treated as contextual word embeddings, which BALAUR learns

to transform into relation-specific vector spaces to disentangle hypernymy from the original distributional vector space as well as other lexical semantic relations such as synonymy and antonymy. In these relation-specific vector spaces, similarity constraints are imposed on the transformed contextual embeddings so that they are similar to related concepts and dissimilar to unrelated concepts. The resulting relation-specific losses are averaged with the overall language modeling loss during LM pretraining, ensuring that the latent representations of the resulting LM are both useful for the original language modeling objective, as well as capable of systematically representing hypernymy and other lexical semantic relations. To verify our hypothesis and proposed approach, we evaluate BERT-like LMs trained with and without BALAUR on tasks requiring knowledge of hypernymy, as well as on the original language modeling task.

1.3 Contributions

In this work, we set out to address the issue of LMs failing to systematically generalize on linguistic tasks that require knowledge of hypernymy. To this end, we proposed BALAUR, a novel method based on the hypothesis that the linguistic behavior of a LM can be aligned with our knowledge of hypernymy by directly modeling lexical semantic relations in the latent representations of the LM. While previous work on incorporating hypernymy into language models does not evaluate on specific hypernymy-informed task, our work bridges the gap between incorporating and evaluating hypernymy in LMs. In addition to finding that our method can improve general language modeling capabilities, we empirically support our hypothesis and demonstrate that BALAUR improves performance on several targeted

tasks requiring knowledge of hypernymy.

As part of our evaluations, we also provide a novel cloze-style dataset which is significantly larger and more comprehensive than previous work. This datasaet encapsulates hypernym and hyponym prediction for a wide variety of nouns, whereas previous work has typically been limited to hypernym prediction for a small subset of nine categories. Lastly, our overall evaluation brings together several disparate threads of research on evaluating language models to provide a comprehensive picture of how well models capture hypernymy in their linguistic behavior.

1.4 Thesis Outline

In Chapter 2, we provide an overview of the literature which our work builds upon. We focus in particular on language models and the form versus meaning debate which in part motivates our use of external knowledge resources for hypernymy. We then discuss WordNet, which is one such resource our work leverages, and it's relation to hypernymy. Lastly, we review the two-fold problem of incorporating and evaluating hypernymy in language models and other NLP systems. In Chapter 3, we introduce BALAUR, our proposed method for incorporating hypernymy into language model pretraining. We first present a high-level discussion of BALAUR and the hypothesis motivating it. We then formalize our approach mathematically and describe how it interfaces with language model pretraining. Lastly, we go over the methodological details of our pretraining experiments and report the effect of our method on language modeling. In Chapter 4, we describe our evaluations for verifying our hypothesis

and report our findings on these. These include zero-shot hypernymy-informed prompt completion, and finetuned hypernymy-informed natural language inference. We also include an analysis of finetuning efficiency on these two tasks to better qualify the contribution of pretraining with BALAUR to finetuned performance. Lastly, in Chapter 5, we conclude our discussion on incorporating hypernymy into language model pretraining, summarizing our key contributions, as well as some limitations and potential directions for future research.

Chapter 2

Background

In this chapter we provide an overview of the different strands of related works on which this thesis builds. We begin by introducing language models in §2.1, providing an operational definition for this thesis as well as a brief discussion on pretraining and the debate of form versus meaning which underlies this work with regards to the ability of LMs to "understand" aspects of word meaning such as hypernymy. We then describe WordNet in §2.2, describing how it represents hypernymy and other lexical semantic relations, providing some brief historical context for this resource which has been central to NLP and to this work. In §2.3, we outline the different ways in which hypernymy and lexical semantic relations more broadly have been incorporated into NLP systems, typically with the intermediary of WordNet. We focus on transformer language models in particular, which are the topic of this work (§2.3.1). We also discuss semantic specialization, a rich body of work which incorporates semantic relations into word embeddings, and on which our proposed method draws (§2.3.2). Lastly, we review seminal work in statistical NLP on lexical semantic relations, and discuss how it informs our work (§2.3.3). Lastly, in §2.4, we survey the

different ways in which hypernymy has been evaluated in transformer language models, notably representational probing (§2.4.1), behavioral probing (§2.4.2), and hypernymy-informed textual entailment (§2.4.3).

2.1 Language Models

2.1.1 Definitions of Language Models

Language models (LMs) in the traditional sense (Jurafsky and Martin, 2023) predict the probability $P(w_t|w_{1:t-1})$ of a word w_t in a sequence, given preceding words $w_{1:t-1}$ [1]. However, with the advent of neural language models, the conventional definition of language modeling has become less precise. Bender and Koller (2020) define a language model as "any system trained only on the task of string prediction, whether it operates over characters, words or sentences, and sequentially or not". The relaxation of this definition is largely attributable to BERT (Devlin et al., 2019), a class of neural language models which popularized several key digressions from the original formulation of language modeling. First, unlike traditional LMs which predict the next token in a sequence, BERT is a masked language model (MLM) which learns to predict the probability $P(w_t|w_{1:t-1}, w_{t+1:T})$ of a randomly masked token w_t given its surrounding context $w_{1:t-1}, w_{t+1:T}$. Second, BERT adopts wordpiece tokenization, where tokens can be subword units rather than conventional words. Lastly, BERT is a *pretrained* LM, in the sense that the LM objective of predicting tokens in context is optimized during a *pretraining* phase, after which the ensuing pretrained

[1] In practice, word boundaries and thus splitting a sequence into words can be ambiguous, so it is more precise to say that a sequence is tokenized into arbitrary tokens, and that LMs predict the probability of tokens.

model can more effectively be adapted to a variety of downstream tasks as a result of representations learned during pretraining. These deviations from the traditional formulation of LMs help explain why current conventional definitions of LMs are so general in comparison. Throughout this thesis, we take the term *language model* to refer to a neural network trained to predict textual tokens given the textual context in which they occur.

2.1.2 Pretraining of Language Models

An important aspect of modern language models is that they are trained, in the sense of neural network optimization via stochastic gradient descent, to predict tokens in context. Typically, the loss function of these models involves minimizing the negative log-likelihood of a correct token t_i given it's context C_i, averaged across examples.

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^{n} -\log P(t_i|C_i) \tag{2.1}$$

Despite the simplicity of this training objective, it has proven to be surprisingly effective when coupled with larger models and larger pretraining text corpora. Models such as BERT have led to remarkable improvements on a wide variety of linguistic tasks, matching and sometimes surpassing human performance (Qiu et al., 2020).

2.1.3 Meaning in Language Models

When machines perform as well as humans on linguistic tasks, can they be said to "understand" or to "capture meaning"? This question has underscored recent and lively debate around language models, where advancements in the field have made it no longer hypo-

thetical. Impressive results have engendered anthropomorphic characterizations of these models' capabilities, with contentious terms such as "understand" being used both in academic pieces and in the popular press (Bender and Koller, 2020). This in turn, has brought the debate of form versus meaning to the fore, with the primary question being can models learn to capture meaning from raw text, or form, alone? Of course, meaning is an irredeemably ineffable concept, with no agreed upon definition to guide investigations into this question. Nevertheless, there are aspects to meaning that can be useful without being comprehensive or definitive, and this thesis focuses on one such aspect: lexical semantics, or the meaning of words.

Unfortunately, the meaning of the meaning of words is not much less confounding than the meaning of meaning. Wittgenstein famously asked the meaning of the word "game" and found that no feature could comprehensively describe it. Do all games have rules? No, the games of children are often lawless and no less fun for it. Is it fun or amusement then? Again, no: many games are competitive rather than amusing. And of course, not all games are competitive. And so on, until the inevitable conclusion is that the precise meaning of "game" is nebulous, and we are left with a "family of meanings" rather than any single definition, "a complicated network of similarities overlapping and criss-crossing, sometimes overall similarities, sometimes similarities of detail" (Wittgenstein, 1976).

One attempt to formalize these "networks of similarities" is the theory of lexical fields and accompanying work on componential analysis (Trier, 1931; Pottier, 1964), which consider meaning as a network of partially overlapping semantic features grounded in the real world (e.g. "having rules", "being amusing", "being competitive" and so on for games).

This conception of word meaning echoes the question which started us on this path: what meaning can models capture from raw text alone, without any grounding in the real world? One answer can be found in Bender et al. (2021), who characterize a language model as:

> a system for haphazardly stitching together sequences of linguistic forms it has observed in its vast training data, according to probabilistic information about how they combine, but without any reference to meaning: a stochastic parrot.

There also exists an alternative view, where form itself can carry meaning. This position is nicely condensed by Wittgenstein again, who famously said "The meaning of a word lies in its use" (Wittgenstein, 1976). The pithy answer, while barely scratching the surface of such a complex question, nevertheless provides a useful intuition for how meaning *could* be captured from raw text alone. More concretely, work in the field of distributional semantics attempts to characterize meaning based on distributional properties of language, in particular patterns of word collocation. Figure 2.1 suggest how the meaning of "moon" can be, at least in part, characterized by collocated words across a variety of contexts.

Beyond word-level distributional semantics, there has also been speculation and preliminary results suggesting that transformer language models can indirectly learn latent representations emulating meaning, reasoning or communicative intent from form alone; as long as these contribute to further optimizing the model's training objective (Andreas, 2022; Nanda et al., 2023; Li et al., 2023). However, empirical results continue to demonstrate that, despite capabilities suggestive of "understanding", models still fail in unexpected situations and produce nonsensical or non-factual outputs which conflict with various notions of meaning (e.g. Pandia and Ettinger (2021), Du et al. (2022)). This failure to systematically and robustly generalize in relation to word meaning is notably seen in the case of lexical

he curtains open and the moon shining in on the barely
ars and the cold , close moon " . And neither of the w
rough the night with the moon shining so brightly , it
made in the light of the moon . It all boils down , wr
surely under a crescent moon , thrilled by ice-white
sun , the seasons of the moon ? Home , alone , Jay pla
m is dazzling snow , the moon has risen full and cold
un and the temple of the moon , driving out of the hug
in the dark and now the moon rises , full and amber a
bird on the shape of the moon over the trees in front
But I could n't see the moon or the stars , only the
rning , with a sliver of moon hanging among the stars
they love the sun , the moon and the stars . None of
the light of an enormous moon . The plash of flowing w
man 's first step on the moon ; various exhibits , aer
the inevitable piece of moon rock . Housing The Airsh
oud obscured part of the moon . The Allied guns behind

Figure 2.1 Patterns of collocation for "moon" (Baroni, 2012)

semantic relations such as hypernymy, a modest yet meaningful facet of word meaning which this thesis focuses on.

2.2 WordNet as a Hypernymy Resource

2.2.1 WordNet and Synsets

Wordnet (Miller, 1995) is a lexical database that contains nouns, verbs, adjectives and adverbs grouped into sets of cognitive synonyms ("synsets"). However, unlike a thesaurus which simply aggregates synonyms, WordNet links synsets to one another with semantic relations such as hypernymy. Crucially, by modeling semantic relations between synsets, WordNet models semantic relations between word meanings rather than between word forms. In Miller and Fellbaum (1991), "word form" refers to *the physical utterance or inscription*, while "word meaning" or "word sense" refers to *the lexicalized concept that*

a word form can express. In other words, a set of word forms with shared word meaning is a set of synonyms (synset), which in turn represents a given lexicalized concept. This distinction is particularly useful for disambiguating multiple possible word meanings for a given word form, or multiple possible word forms for a given word meaning (Table 2.1).

Synset	**WordNet definition**	**Word forms**
Synset(‘dog.n.01‘)	a member of the genus Canis that has been domesticated by man since prehistoric times	dog domestic dog Canis familliaris
Synset(‘dog.n.05‘)	a smooth-textured sausage of minced beef or pork usually smoked; often served on a bread roll	frankfurter hotdog weenie ...
Synset(‘bank.n.02‘)	a financial institution that accepts deposits and channels the money into lending activities	bank banking concern banking company ...
Synset(‘bank.n.01‘)	sloping land (especially the slope beside a body of water)	bank

Table 2.1 Example word meanings for "dog" and "bank" in WordNet. Word meanings, or lexicalized concepts, are represented by synsets and distinguished by definitions and synonymous word forms.

2.2.2 Lexical Semantic Relations and Hypernymy in WordNet

The semantic relations in WordNet notably include the lexical semantic relations of synonymy, antonymy, meronymy, and hypernymy (Table 2.2). Synonymy is a symmetric relation between two word forms that captures shared or similar word meaning. In WordNet, synonymy links word forms to create synsets. Similarly, antonymy is also a symmetric relation between word forms, which captures opposite meaning. In WordNet, antonymy links word forms rather than synsets, due to issues identified by Miller and Fellbaum (1991) in generalizing antonymy to synsets. In contrast, meronymy and hypernymy are asymmetric relations between pairs of synsets. Meronymy and its inverse holonymy form the

part-whole (or has-a) relation, which can be characterized by *an x is a part of a y* where *x* is the meronym of holonym *y*. Conversely, hypernymy and its inverse hyponymy form the subset-superset (or is-a) relation, which can be characterized by *an x is a kind of y* where *x* is the hyponym of hypernym *y*.

Semantic Relation	**Examples**
Synonymy (similar)	pipe, tube rise, ascend sad, unhappy rapidly, speedily
Antonymy (opposite)	wet, dry powerful, powerless friendly, unfriendly rapidly, slowly
Meronymy (part-whole)	brim, hat gin, martini ship, fleet
Hypernymy (superset-subset)	maple, sugar maple tree, maple plant, tree

Table 2.2 Examples of lexical semantic relations in WordNet.

These lexical semantic relations were chosen and refined by Miller and Fellbaum (1991) because of their broad applicability throughout English, their familiarity to non-experts, and their potential for capturing meaningful semantic structure in the English lexicon. Notably, WordNet's directed acyclic graph of hypernymy relations was originally intended as a representation of the mental lexicon described in §2.2.3. This psycholinguistic goal is also reflected in the lexicographer class labels in which synsets are categorized, also known as supersenses (Ciaramita and Johnson, 2003). Supersenses are closely related to hypernymy, and correspond to broad semantic categories in which synsets might belong, e.g. WordNet includes "person", "animal", "act", "substance", "event" and "feeling" among 26 noun-specific supersenses.

2.2.3 Origins and Legacy of WordNet

Interestingly, WordNet was not originally intended as a resource for computational linguistics or NLP. The psycholinguist behind the project, George A. Miller, was interested in testing the concept of semantic networks, which ascertains that long-term memories are organized hierarchically; enabling inferences such as "a canary can fly" based on the stored memories that "birds can fly" and "a canary is a bird" (Fellbaum, 2013). This theory was quite popular at the time, following empirical results from Collins and Quillian (1969) which demonstrated that reaction-times for qualifying such statements as true or false were proportional to distances in this semantic network (Figure 2.2), and Miller was interested in building such a network for the English lexicon more generally.

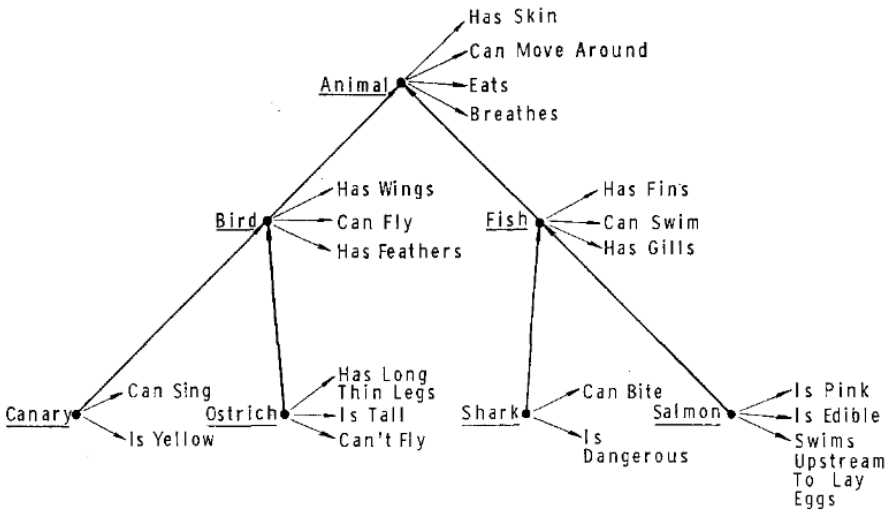

Figure 2.2 Canary in the semantic network (Collins and Quillian, 1969)

In the 1980s, Miller recruited his wife and a group of colleagues and students to help him cluster words into "synsets", which could then be interrelated with a handful of semantic relations. Without much further instructions and relying on conventional lexical

resources and intuition, this skunkworks team manually entered tens of thousands of entries in the WordNet database. Fortuitously, one government sponsor's requirement led to the database's public release, which in turn led to an unexpected yet extraordinary and rapid adoption by a budding NLP community interested in word sense discrimination (Fellbaum, 2013). As the project grew [2], its focus shifted from psycholinguistics to computational linguistics and NLP with the direction of Christiane Fellbaum. Over the decades, WordNet has enjoyed widespread adoption in various NLP applications, expanded globally beyond the English language, and helped trigger the creation of other lexical resources (e.g. FrameNet (Fillmore et al., 2002) and PropBank (Palmer et al., 2005)) based on alternative linguistic theories (Miller and Fellbaum, 2007).

2.3 Incorporating Hypernymy in NLP Systems

2.3.1 Language Model Pretraining with Hypernymy

Incorporating LSRs into LM pretraining, particularly hypernymy, has been approached from different angles. Lauscher et al. (2020) create an auxiliary training objective with supplemental, objective-specific training instances. These training instances consist of two words, where the model must predict whether they are semantically related using the next sentence prediction objective of Devlin et al. (2019). In contrast to our work, this approach combines synonymy, hypernymy and hyponymy into one relation of *semantic relatedness* and requires a large number of additional training examples during pretraining.

[2] The current Princeton WordNet 3.1 contains 117, 791 synsets, 207, 272 word senses, 159, 015 word forms (or lemmas), and 285, 668 synset relations (McCrae et al., 2019)

This approach was shown to improve performance on the GLUE benchmark (Wang et al., 2018) as well as on lexical simplification tasks.

Levine et al. (2020) avoid the need for additional training data by modifying the LM objective to predict a word's supersense, a high-level hypernym, along with the word itself. This approach was shown to improve performance on word sense disambiguation in particular. Similarly, Bai et al. (2022) create a curriculum where LMs learns to predict a word's hypernym before learning to predict the word itself. However, this approach aims and succeeds to improve language modeling performance more generally. Typically, previous work uses multi-task learning to incorporate hypernymy into LM pretraining. However, it does not evaluate on specific hypernymy-informed tasks or attempt to disentangle hypernymy from other relations during pretraining. In contrast, our work bridges the gap between incorporating and evaluating hypernymy in LMs, proposing a novel method based on semantic specialization and demonstrating improvements on targeted evaluations of hypernymy.

2.3.2 Semantic Specialization of Word Embeddings

Word embeddings capture distributional similarity, which can serve as a proxy for semantic similarity but struggles to capture semantic relatedness (Budanitsky and Hirst, 2006). In particular, semantic similarity can be seen as an instance of the more general concept of semantic relatedness which includes and distinguishes between different types of semantic relations (Baroni and Lenci, 2011). Semantic specialization refers to a class of methods that address this issue and have been used to incorporate and disentangle LSRs in word embeddings. It is often framed as learning an auxiliary objective function to impose

constraints on a distributional vector space and create systematic structure reflective of LSRs. However semantic specialization has, to the best of our knowledge, not been explored in the context of language models. In this thesis, we consider how this approach can be adapted to the latent representations of LMs to more systematically represent hypernymy and help guide their linguistic behavior on hypernymy-informed tasks.

While semantic specialization covers a wide breadth of techniques and goals, we provide here an overview of the different threads of research which underlie our work. Yu and Dredze (2014) first incorporate synonymy in Word2vec (Mikolov et al., 2013a) with an auxiliary learning objective based on Bordes et al. (2012). Given word w_i, the probability of related words $w \in R_i$ is maximized based on the normalized dot product of their embeddings (2.2). Similarly, Fried and Duh (2015) incorporate hypernymy into NLM embeddings (Collobert et al., 2011) by making the cosine similarity of word embeddings proportional to their proximity in WordNet's hypernymy graph (Fellbaum and Miller, 1998).

$$\begin{aligned} \log p(w|w_i) &= \exp(e_w^T e_{w_i}) / \textstyle\sum_{\tilde{w}} \exp(e_w^T e_{w_i}) \\ \mathcal{L} &= \frac{1}{N} \sum_{i=1}^{N} \sum_{w \in R_i}^{N} \log p(w|w_i) \end{aligned} \tag{2.2}$$

However, a key limitation of these methods is that they involve retraining embeddings from scratch. To address this issue, Faruqui et al. (2015) retrofit already trained embeddings q_i by jointly minimizing their euclidean distance with the original embeddings $\hat{q}_i$ and with related embeddings q_j for the set of related pairs $(i, j) \in E$, with hyperparameters α and β to control the relative weight of each constraint (2.3). Building on this approach, Mrkšić

et al. (2016) propose counter-fitting, which includes an additional objective to maximize the euclidean distance between antonyms. Vulić and Mrkšić (2018) further extend retrofitting by adding a hypernymy-specific objective which minimizes the cosine distance of related hyponym-hypernym pairs while adjusting vector norms to reflect the WordNet hierarchy.

$$\mathcal{L} = \sum_{i=1}^{n} \left[\alpha_i ||q_i - \hat{q}_i||^2 + \sum_{(i,j)\in E} \beta_{ij} ||q_i - q_j||^2 \right] \tag{2.3}$$

To more systematically represent multiple relations, Xu et al. (2014) augment skip-gram (Mikolov et al., 2013b) with a margin-based regularization function, similar to the TransE model of Bordes et al. (2013) which represent distinct relations as distinct translations in the vector space of embeddings (following which related embeddings should be more similar). In contrast, Glavaš and Vulić (2018) specialize distributional word embeddings by learning distinct neural networks for distinct relations, each neural network taking in two word embeddings and predicting whether they are related. Lastly and most closely related to our work, Arora et al. (2020) propose LexSub (2.4). For a symmetric relation r, LexSub minimizes the cosine distance d_r^{proj} of related word embeddings (x_i, x_j) that have been projected into a relation-specific subspace with W_r^{proj} (except for antonymy which maximizes cosine distance in synonymy space, similar to counter-fitting). In the case of asymmetric relations such as hypernymy, an additional relation-specific linear transformation is performed only on x_i to capture asymmetry in the cosine distance d_r^{asym}.

$$d_r^{proj}(x_i, x_j) = d(W_r^{proj} x_i, W_r^{proj} x_j)$$
$$d_r^{asym}(x_i \rightarrow x_j) = d_r^{proj}(W_r^{asym} x_i + b_r^{asym}, x_j) \tag{2.4}$$

By modeling distinct relations as similarity constraints in distinct transformations of the original vector space, these approaches can effectively model and disentangle multiple relations in one set of representations, bridging the gap between semantic similarity and semantic relatedness. Crucially, this functional representation of LSRs as transformations may enable systematic generalization to unseen pairs of related items (Vulić et al., 2018).

2.3.3 Lexical and Distributional Semantics

More broadly, these lines of work explore the interplay between lexical and distributional semantics, specifically how the first (in the form of LSRs) can help inform the second (in the form of training and evaluating LMs or word embeddings). In contrast, there is a rich body of work that has attempted to inform lexical semantics with distributional semantics. Of particular relevance to our work is the extraction from corpus data of hypernymy (Caraballo, 1999; Snow et al., 2004) and meronymy (Poesio et al., 2002) relations, typically based on Hearst patterns (Hearst, 1992). Similarly, Mohammad et al. (2008) leverage the co-occurrence hypothesis (Charles and Miller, 1989) to identify antonymy. Bridging the gap between lexical and distributional semantics, there is work like Agirre et al. (2009) which combines both approaches, noting that while distributional methods help alleviate out-of-vocabulary issues in lexical resources, they struggle to distinguish semantic similarity from relatedness. Our work attempts to address this issue, explicitly modeling LSRs in LM

representations so they can be distinguished.

2.4 Evaluating Hypernymy in Language Models

2.4.1 Representational Probing

As transformer language models such as BERT (Devlin et al., 2019) obtained state-of-the-art performance across NLP tasks (see Qiu et al. (2020) for a review), a growing body of work sought to better understand these positive empirical results, by understanding, among other things, what kind of information is learned by LMs and how it's represented in their internal representations.

Rogers et al. (2020) and Belinkov and Glass (2019) provides comprehensive reviews of this line of work, including one class of approaches in particular, referred to as "probing tasks" (Conneau et al., 2018) or "diagnostic classifiers" (Veldhoen et al., 2016). These approaches train classifiers over model representations to predict specific linguistic phenomena. For example, Tenney et al. (2019b) probe token-level representations on a range of syntactic and semantic tasks such as part-of-speech tagging, dependency labeling, and semantic role labeling. In the context of hypernymy, Vulić et al. (2020) use probing to systematically analyze how well LMs encode lexical semantics in their representations. Specifically, they adapt the WN-LS evaluation from Glavaš and Vulić (2018), which in turn is based on CogALex-V (Santus et al., 2016). In both evaluations, relations between word pairs must be classified as synonymy, antonymy, hypernymy, or meronymy. While the original dataset suffers from skewed class distribution, lexical repetitiveness, and a non-systematic split where words in the train set occur in the test set, the WN-LS dataset contains

10,000 word pairs evenly distributed between categories. However, the 80% train-test split in WN-LS remains unsystematic, and the performance reported by Vulić et al. (2020) is lackluster, with a micro-averaged F_1 score that never exceeds 0.73 across configurations.

These probing methods suffer from several limitations discussed by Rogers et al. (2020). Notably, probes tell us what information can be recovered from model representations, not how (or even if!) models use it in practice (Tenney et al., 2019a). Furthermore, probing typically improves with classifier complexity, however it becomes less clear to what extent the targeted information is captured by the original model rather than the probe itself.

2.4.2 Behavioral Probing

A fundamental challenge of evaluating LMs via their representations is understanding whether performance is attributable to a model's representations, or to the probing and finetuning processes. An alternative approach to evaluating what information is captured by a language model considers instead its linguistic behavior, in terms of the text it generates given certain contexts. This is typically done in a zero-shot setting, i.e. without finetuning.

For instance, Linzen et al. (2016) use conditional text generation to assess subject-verb number agreement in LSTMs, comparing the probabilities of generating the correct verb in either its singular or plural form. Building on this work, Marvin and Linzen (2018) compare sequence-level probabilities between grammatical and ungrammatical sentences in LSTMs to assess how well they capture syntax. More broadly, Radford et al. (2019) investigate how well LMs perform on a variety of downstream NLP tasks framed as conditional text generation. Petroni et al. (2019) use conditional text generation in a more targeted way to

extract relational knowledge from LMs in an unsupervised way, finding this approach to be remarkably competitive with non-neural and supervised alternatives in terms of mean precision.

In the context of hypernymy, Ettinger (2020) adapts psycholinguistic tests to "examine LMs' general linguistic knowledge, specifically by asking what information the model are able to use when assigning probabilities to words in context". One of the proposed tests builds on Fischler et al. (1983) to evaluate negation and category membership. In this test, models must complete prompts such as "a robin is a ___" and "a robin is not a ___", where "bird" is the right completion for the first prompt, but not the second. While BERT performs well on predicting noun categories from positive contexts, it completely fails to account for negation. An important limitation of these results is the limited size of the dataset, containing only 9 noun categories with two subject nouns each for a total of 18 prompts templates. In contrast, Ravichander et al. (2020) more directly target hypernymy knowledge encoded in BERT, evaluating the systematicity of its behavior on such prompts. Specifically, they demonstrate that BERT fails to generalize when nouns are pluralized (e.g. "robins are ___"), extending the original dataset of Ettinger (2020) from 18 to 576 noun-category pairs while maintaining the original 9 Fischler categories. Unsurprisingly, BERT performance drops precipitously from perfect performance on the original (positive) dataset to 67.53% accuracy on the extended dataset, and 44.1% on the pluralized extended dataset, suggesting that LMs struggle to capture hypernymy in their linguistic behavior. Similarly, Hanna and Mareček (2021) find that LMs can only predict correct hypernyms with 57% accuracy across more diverse prompts. In particular, a qualitative analysis of model behavior found

that LMs often perform semantically and syntactically correct completions unrelated to hypernymy, suggesting a mismatch between the original language modeling objective of LMs and prompts aimed at hypernym extraction.

These behavioral probing results suggest that LMs such as BERT fail to systematically capture hypernymy in its linguistic behavior, motivating the work in this thesis. However, the findings of Hanna and Mareček (2021) also highlight a limitation of zero-shot prompting, where specific linguistic capabilities are difficult to disentangle from general language modeling behavior. While previous work has attempted to address this by computing evaluation metrics on a closed set of possible completions (e.g. Fischler categories in Ettinger (2020) and Ravichander et al. (2020)), this approach does not capture LM behavior in practice. An alternative approach to disentangle general language modeling behavior from specific capabilities is to finetune models on prompt completion, and to benchmark performance throughout finetuning (Talmor et al., 2020). Importantly, comparing performance with baselines throughout finetuning enables this approach to better distinguish what latent capabilities are learned by LMs during pretraining as opposed to during finetuning. In our work, we address several of the limitations outlined for behavorial probing in the context of hypernymy. First, we create a novel dataset of prompts which is significantly larger and more comprehensive than previous work, encapsulating hypernym and hyponym prediction for a wider variety of nouns. Second, we address the issue of language modeling mismatch, adapting the protocol of Talmor et al. (2020) to evaluate how well LMs transfer learn on this task. Lastly, we use systematic train-test splits to isolate how well LMs learn a functional abstraction of hypernymy, rather than specific instances of hypernym-hyponym pairs.

2.4.3 Hypernymy-informed Entailment

In addition to representational and behavioral probing, how well LMs capture hypernymy can be measured based on their performance on downstream tasks that are informed by hypernymy. One such task is textual entailment, or Natural Language Inference (NLI), which can require knowledge of hypernymy in certain cases.

Similarly to different word forms having shared meaning, different utterances can also have shared meaning; a linguistic phenomenon which Dagan et al. (2006) identify as "variability of semantic expression". They propose Recognizing Textual Entailment (RTE) as an application-independent task which measures models' abilities to capture inferences involved in identifying entailment. Concretely, RTE is define as "recognizing, given two text fragments, whether the meaning of one text can be inferred (entailed) from the other". Several iterations of this task are compiled into the RTE subtask of the GLUE benchmark (Wang et al., 2018), which has played a central role in evaluating LM capabilities. GLUE also includes several NLI tasks such as SNLI (Bowman et al., 2015), which are similar to RTE but include "contradiction" as a third entailment classification.

Textual entailment can involve diverse inferences, not necessarily related to hypernymy. For example, Bar-Haim et al. (2005) identify mechanisms on the lexical entailment level (morphological derivations, ontological relations, lexical world knowledge), and on the lexical-syntactic entailment level (syntactic transformations, paraphrases, and co-reference) in the original RTE dataset. Of particular interest to this thesis is ontological relations; where synonymy, hypernymy, or meronymy enable meaning-preserving word substitutions, which in turn result in textual entailment. For example *A dog ate my hotdog* entails

A dog ate my frankfurter because of synonymy, as well as *A dog ate my food* because of hypernymy. This principle of meaning-preserving word substitutions is referred to as lexical entailment (Geffet and Dagan, 2005) and informs MoNLI (Geiger et al., 2020), a challenge NLI dataset where entailment is determined by hypernymy, negation and monotonicity reasoning. Textual entailment examples are created with hyponym-hypernym single-word substitutions in PMoNLI, with negation introduced in NMoNLI to reverse the direction of entailment (for example *A dog didn't eat my food* now entails *A dog didn't eat my hotdog*). Rozen et al. (2021) provide a similar challenge NLI dataset without negation and find that BERT obtains only 65% accuracy, suggesting that LMs struggle to learn representations of hypernymy useful for textual entailment. In our work, we include MoNLI in our evaluations to complement our behavioral probes. Combined with the transfer learning evaluation of Talmor et al. (2020), our evaluations provide a comprehensive overview of the different ways in which LMs may capture hypernymy in their linguistic behavior. We do not include representational probing in our evaluation, as our method directly optimizes this.

Chapter 3

Incorporating Hypernymy

In this chapter, we present BALAUR, our proposed approach for aligning the linguistic behavior of language models with our understanding of hypernymy. We first introduce BALAUR in §3.1, with a high-level discussion of the method and the hypothesis underlying it. We then formalize BALAUR in §3.2, providing detailed explanations of how we translate our hypothesis into a concrete neural network architecture and optimizable loss function. In §3.3, we describe how pretraining a language model with BALAUR is operationalized in practice throughout our experiments. Lastly, in §3.4, we discuss the unexpectedly positive effects of BALAUR on the original language modeling objective.

3.1 BALAUR

Based on work in semantic specialization (§2.3.2), we propose a novel approach to incorporate LSRs into LM pretraining. In this approach, we learn LSR-specific transformations that are applied to the latent representations of LMs; modeling LSRs as constraints in the

resulting vector spaces. Concretely, the transformed latent representations of two lexical items should be similar only if they are related by the corresponding LSR. We operationalize this approach with BALAUR: a modular neural architecture composed of distinct heads for each LSR, named after the many-headed dragon of Romanian folklore. As shown in Figure 3.1, each head learns different LSR-specific transformations that enforce our similarity constraint between contextualized embeddings of input tokens (i.e. the LM's final hidden states) and embeddings of concepts. By parameterizing LSRs as learned transformations, our approach can model LSRs inductively; i.e. generalize from instances of related pairs to a functional representation that can extrapolate to unseen pairs. Moreover, by learning separate transformations, we model multiple LSRs in one vector space such that they can be disentangled from one another; e.g. capturing that "dog" and "fox" share the hypernyms "canine" and "animal", but have different hyponyms (Figure 3.1).

Our hypothesis is that, by jointly learning these transformations during LM pretraining, we can instill the resulting LM's hidden states with inductive biases useful for LSR-informed tasks. More broadly, we aim to demonstrate that controlling the latent representations of LMs can provide an interface to their linguistic behavior and help align it with our knowledge. One important advantage of our proposed approach is its data and compute efficiency. By enforcing our constraints on hidden states that are already computed in the LM's forward pass, our method can be incorporated in pretraining with negligible overhead and no additional training examples. Conversely, using separate concept embeddings allows us to model LSRs for lexical items that would otherwise fall outside the model's vocabulary or that do not co-occur in training examples.

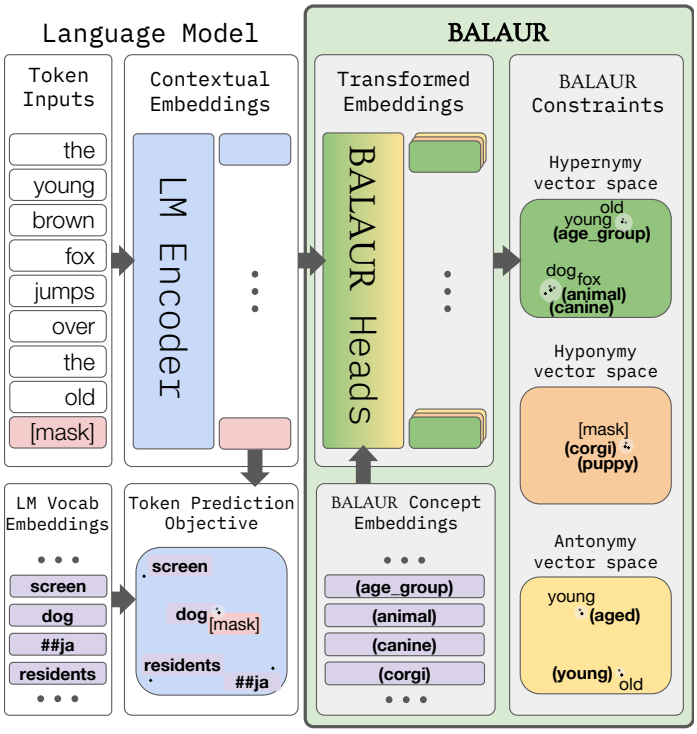

Figure 3.1 Combining BALAUR with LM pretraining to model LSRs in its hidden states.

3.2 The BALAUR Head Architecture

3.2.1 Modeling Lexical Semantic Relations

Assumptions We take the term (neural) language model (LM) to refer to a neural network that predicts a token given its context, including commonly used masked LMs like BERT (Devlin et al., 2019) and auto-regressive LMs like GPT (Radford et al., 2018). These LMs encode a sequence of tokens into latent representations known as hidden states or contextualized token embeddings T, using these as inputs to a classification head that predicts the target token. Meanwhile, we represent a lexical semantic relation R as a set of related lexical item pairs $(X_i \rightarrow X_j) \in R$, noting that LSRs can be directed, e.g. $(corgi \rightarrow dog)$ resides in hypernymy while $(dog \rightarrow corgi)$ resides in hyponymy.

Our goal is to model LSRs in T, where T_i is the LM hidden state for token i. However

modeling LSRs between pairs of in-context tokens is intractable. On the one hand, if we were to model relations between tokens in the same context, we run into the issue that tokens typically do not co-occur with related tokens. On the other hand, if we were to model relations between pairs of tokens in different contexts, the number of such pairs grows combinatorially with the number of contexts. Instead, we model LSRs as $(T_i \rightarrow C_j) \in R$, where C is a set of context-independent concepts, represented as static embeddings learned by BALAUR during pretraining. This allows us to tractably model LSRs in T while using the same training examples as a vanilla language model, instead of artificially creating contexts with co-occurring related tokens (Lauscher et al., 2020) or combining pairs of contexts in the model forward pass. Furthermore, operating on lexicalized concepts, similar to Bai et al. (2022), enables us to account for lexical items that are not present in the LM's vocabulary.

Concretely, for a given relation R, the corresponding BALAUR head learns to transform T and C such that related token-concept pairs are similar in the resulting relation-specific vector space. For example, Figure 3.1 shows that the token embedding for `dog` and the concept embedding for **`(canine)`** are similar when transformed into hypernymy space; while `dog` and **`(corgi)`** are similar when transformed into hyponymy space. These learned transformations enable BALAUR heads to model and disentangle multiple LSRs in the vector space of T, i.e. ensuring a token's related concepts can be predicted from its contextualized embedding, distinguishing across different relations. We implement these transformations as two-layer neural networks with GELU activations (Hendrycks and Gimpel, 2020):

$$\begin{aligned} \underset{t\times b}{\mathrm{T}^R} &= \underset{b\times b}{W^R}\ (\textsc{gelu}(\underset{t\times d}{T} \times \underset{d\times b}{W^{R,T}} + \underset{1\times b}{B^{R,T}})), \\ \underset{c\times b}{\mathrm{C}^R} &= \underset{b\times b}{W^R}\ (\textsc{gelu}(\underset{c\times d}{C} \times \underset{d\times b}{W^{R,C}} + \underset{1\times b}{B^{R,C}})), \end{aligned} \tag{3.1}$$

where t and c are the number of token and concept embeddings, and d and b are their original and transformed dimensionalities. W^R, $W^{R,T}$, $W^{R,C}$, $B^{R,T}$, $B^{R,C}$ are learned projection and bias matrices that parameterize the transformations for R.

3.2.2 Optimizing a Balaur Head

To translate our similarity constraint to a learning objective, we adapt the supervised contrastive loss of Khosla et al. (2021) which maximizes the inner product similarities S between each related token-concept pair (i, j), while minimizing it for unrelated pairs (i, k). Optimizing this loss thus enables us to predict a token's related concepts from its contextual embeddings, encoding the corresponding LSR in the LM's hidden states:

$$\underset{t\times c}{\mathrm{S}^R} = \underset{t\times b}{T^R} \times \underset{c\times b}{(C^R)^T} \tag{3.2}$$

$$\mathcal{L}^R = \frac{1}{|R|} \sum_{(i,j)}^{R} -\log \frac{\exp\,(S^R_{i,j})}{\sum\limits_{k<c} \exp\,(S^R_{i,k})} \tag{3.3}$$

where i indexes the set of t token embeddings, while j and k index the set of c concept embeddings.

3.2.3 Interfacing with Language Models

During LM pretraining, T is computed in the forward pass and used as input to the LM's classification head for token prediction. Each BALAUR head also takes T as input, along with concept embeddings C and relation-specific sets of indices (i, j) — where i indexes T and the corresponding token in the training batch, while j indexes a concept in C related to T_i by the corresponding relation R. Each head then computes its loss $\mathcal{L}^R$ and these are averaged before being added to the LM loss.

3.3 Pretraining with BALAUR

In this section, we detail our methods for LM pretraining with BALAUR, using LSRs and concepts extracted from WordNet. We also present the architecture and hyperparameters for the LM in our experiments, a variant of $\text{BERT}_{\text{LARGE}}$ suitable for academic budgets. While our experiments are limited to masked language modeling and LSRs, our method can easily be extended to autoregressive language modeling and other forms of relational knowledge.

3.3.1 Extracting LSRs from WordNet

As a first step, we extract related token-concept pairs for hypernymy, hyponymy, antonymy and synonymy from WordNet's noun hierarchy (Miller, 1995). To do this, we begin by mapping the model's vocabulary to corresponding WordNet synsets (referred to throughout this paper as concepts) using NLTK (Bird and Loper, 2004). For example, `dog` maps to the concept of **`dog.n.01`** (a pet dog) or **`frank.n.02`** (a hot dog).

Next, using the resulting set of concepts, we extract related concept-concept pairs

from WordNet and convert these to token-concept pairs. For example **canine.n.01** is a hypernym of **dog.n.01**, while **sausage.n.01** is a hypernym of **frank.n.02**; but both are extracted as hypernyms of the token dog[1]. To increase our coverage of WordNet, we consider multi-hop hypernymy up to depth 3, such that e.g. **animal.n.01** is extracted as a hypernym of both dog and canine. The resulting set of token-concept pairs contains 15,612 unique concepts.

Manual sampling and inspection of the resulting pairs revealed several known issues associated with WordNet, including inaccurate lemmatization (McCrae et al., 2019), and word senses that are too fine-grained (McCarthy, 2006). To help address these potential sources of noise in BALAUR, we filter tokens, concepts and token-concept pairs using the criteria described in A.1.1.

3.3.2 Incorporating BALAUR into Pretraining

We then use these token-concept pairs to optimize a BALAUR head for each LSR throughout LM pretraining. First, we instantiate an embedding layer C for the set of extracted concepts, shared across four BALAUR heads for hypernymy, hyponymy, synonymy and antonymy. Second, each training example is annotated with relation-specific sets of indices (i, j), where i indexes a token in the training sequence and j indexes a related concept in C. Lastly, the hidden states T are computed in the LM's forward pass, and fed into each BALAUR head along with C and the sets of indices (i, j) to compute $\mathcal{L}^R$ as described in

[1]This approach does not distinguish between different in-context meanings of a token, and instead models relations for all possible meanings simultaneously; simplifying implementation by foregoing wordsense disambiguation, with the downside of a noisier learning signal.

§3.2.3. To reduce the overhead of iterating over BALAUR heads, we adopt the parallelization technique from multi-head attention (Vaswani et al., 2017). Specifically, we learn one set of transformations (3.1) but multiply b by $|R|$ so the resulting transformed vector space can be partitioned across relations. To further improve efficiency we only give the subset of T containing LSRs as inputs to BALAUR, ensuring i is reindexed on this subset.

3.3.3 Language Model Pretraining Setup

Our LM architecture, pretraining procedure, and hyperparameters are based on 24hBERT (Izsak et al., 2021) which enables rapid pretraining with limited resources, while reaching comparable performance with the original BERT models (Devlin et al., 2019). Specifically, we pretrain a $\text{BERT}_{\text{LARGE}}$ architecture to perform masked language modeling (MLM) on 128-token sequences for 25,000 steps with a batch size of 4,096 and using 16-bit precision. We optimize using AdamW (Loshchilov and Hutter, 2019) and a peak learning rate of 2e-3 with warm-up over the first 1,500 steps and linear decay. The pretraining data is a snapshot of English Wikipedia from 2022-03-01, and BookCorpusOpen (Bandy and Vincent, 2021), with 0.5% withheld for validation. These datasets were downloaded from and preprocessed with the `datasets` library (Lhoest et al., 2021) which provided licenses such as CC-BY-SA 3.0 and GFDL for Wikipedia.

3.4 BALAUR Improves Language Modeling

In Table 3.1, we see that incorporating BALAUR into the LM pretraining procedure of Izsak et al. (2021) increases both negative log likelihood (NLL) and mean reciprocal rank (MRR)

for the original masked language modeling objective.

	Random Tokens		Lsr Tokens	
Model	NLL	MRR	NLL	MRR
Bert (ours)	1.659	0.733	3.359	0.482
Bert+Balaur	1.587	0.743	3.201	0.503
Δ (%)	4.3	1.4	4.5	4.1

Table 3.1 Validation MLM performance, shown for masking random tokens and for only masking tokens with LSRs (i.e. modeled by Balaur during pretraining).

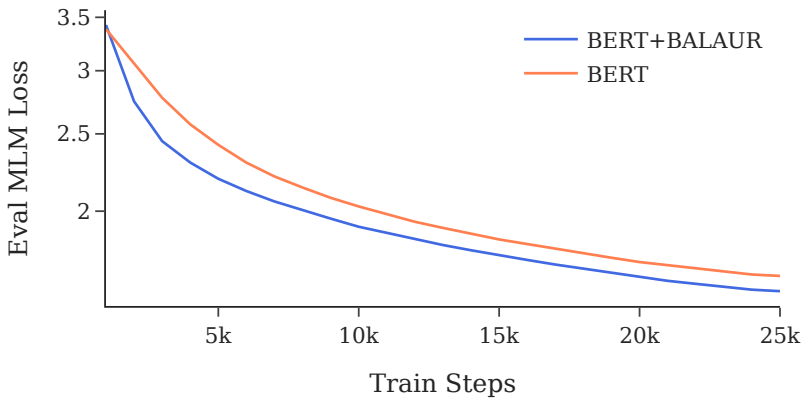

Figure 3.2 Validation MLM loss throughout pretraining.

We observe similar improvements when masking random tokens as when masking only tokens with LSRs, indicating the improvements introduced by Balaur extend beyond the modeling of LSRs. Lastly, we note that the improvements in the original MLM objective begin early and are consistent throughout pretraining, as seen in Figure 3.2. This suggests the beneficial effects of Balaur on LMs occur early in optimization rather than later, highlighting the importance of pretraining from scratch in our method.

Chapter 4

Evaluating Hypernymy

In this section, we evaluate whether pretraining with BALAUR heads can improve performance of LMs on tasks that are informed by LSRs, specifically hypernymy and hyponymy. To this end, we compare $\text{BERT}_{\text{LARGE}}$ models that were pretrained with and without BALAUR heads[1]. Throughout this section, we refer to these as BERT+BALAUR and BERT (OURS) respectively. The goal of this analysis is to verify our hypothesis that controlling the latent representations of LMs can provide an interface to their linguistic behavior and help align it with our knowledge (in this case, of hypernymy). To this end, we bring together a comprehensive set of evaluations from the literature (§2.4), which target different ways in which LMs might capture hypernymy:

Prompt completion (§4.1): models must predict the correct token given a cloze-style prompt describing a hypernymy or hyponymy relation, e.g. "*a dog is a type of [mask]*".

Monotonicity NLI (§4.2): models must predict whether a sentence entails another, when

[1]Note that during evaluation, BALAUR heads are discarded such that both models have the same architecture, differing only in their learned weights.

hypernymy and monotonicity determine entailment, e.g. "*drive a taxi*" entails "*drive a car*".

Finetuning Efficiency (§4.3): we compare how efficiently models transfer-learn throughout finetuning on the two previous tasks to evaluate how well model representations capture hypernymy, disambiguating what is learned during pretraining versus during finetuning.

4.1 Prompt Completion

Task Description We create HypCC: a dataset of cloze-style prompts taking the form "In the context of hypernymy, a(n) x is a type of y." where x, y are hyponym-hypernym pairs of tokens in our model's vocabulary, and either is masked out to be predicted by the model. This evaluation builds on the work of Ettinger (2020) and Ravichander et al. (2020), which draws from human psycholinguistic tests to create cloze prompts. In contrast to previous work, our evaluation includes hypernyms beyond Fischler categories, evaluates hyponym prediction, considers tokens with multiple word senses, and include clozes with multiple valid completions. The resulting dataset contains 17,556 hyponym-hypernym pairs; 5,217 hypernym prediction prompts; and 4,115 hyponym prediction prompts. We report additional details for the creation of HypCC in A.1.2, as well as limitations in §A.1.3.

Evaluation Method In line with previous work, models are evaluated on HypCC in a zero-shot manner (i.e. using masked language modeling to complete the cloze prompt); and performance measured with accuracy and mean reciprocal rank (MRR) for both the open

and closed vocabulary settings. For accuracy, we report ACC@1/5, i.e. the rate at which correct answers lie in the top-1 and top-5 predictions. In the closed setting, metrics are calculated using only the set of possible hypernyms or hyponyms in HYPCC, while the open setting considers the model's entire vocabulary. Importantly, these metrics are adjusted to account for multiple valid completions in a prompt: ignoring other valid completions when computing a completion's rank (i.e. if a model's top three predictions are all valid, the average accuracy will be 100% instead 33%). To prevent a skewing of results by prompts with multiple completions, metrics are averaged over the set of prompts X after averaging over the set of possible completions $Y(x)$ for each prompt $x \in X$.

$$\mathcal{M} = \frac{1}{|X|} \sum_{x \in X} \left[\frac{1}{|Y(x)|} \sum_{y \in Y(x)} \mathcal{M}(y, x) \right] \tag{4.1}$$

Results and Discussion In Table 4.1, we find that BALAUR can robustly improve LM performance on hypernymy-informed prompt completion across settings and metrics, even outperforming the original $\text{BERT}_{\text{LARGE}}$ implementation of Devlin et al. (2019).

Model	Closed Vocab		Open Vocab	
	Acc@1/5	Mrr	Acc@1/5	Mrr
Hypernym Prediction				
†Bert$_{\text{Large}}$	3.53 / 14.13	0.092	**1.78** / 11.77	0.071
Bert (ours)	5.18 / 18.61	0.121	0.88 / 14.72	0.080
Bert+Balaur	**5.31 / 19.65**	**0.128**	1.60 / **15.44**	**0.089**
Hyponym Prediction				
†Bert$_{\text{Large}}$	**3.60** / 14.95	0.097	**2.76** / 12.87	0.083
Bert (ours)	2.69 / 12.22	0.081	2.03 / 10.65	0.069
Bert+Balaur	3.49 / **17.91**	**0.110**	1.85 / **14.56**	**0.084**

Table 4.1 Zero-shot results on HypCC. Balaur generally improves performance across metrics when compared to a baseline Bert model with the same 24hBert pretraining procedure, as well as the published checkpoint of †Bert$_{\text{Large}}$ (Wolf et al., 2020).

Model	Hypernym Repetition	Hyponym Repetition
†Bert$_{\text{Large}}$	50.17	47.08
Bert (ours)	87.81	64.20
Bert+Balaur	69.59	69.38

Table 4.2 Rates of repetition on HypCC. Balaur reduces repetition for hypernym prediction, with comparable rates of repetition for hyponym prediction.

However, we note that both of our models struggle with Acc@1 when compared to †Bert$_{\text{Large}}$, despite general improvements of Balaur over our baseline. A closer inspection of model predictions reveals that, similar to findings of Ettinger (2020), models often repeat the hypernym or hyponym in the context (e.g. predicting "*a daisy is a type of daisy*"). In Table 4.2, we find that our baseline pretraining procedure exacerbates this problem, explaining the discrepancy in Acc@1 performance.

Moreover, a qualitative analysis of selected clozes similar to Arora et al. (2020), shown in Table 4.3, suggests that Balaur better disentangles hypernymy from other forms of semantic relatedness. These results agree with Agirre et al. (2009), who showed similar

improvements combining lexical and distributional semantics in word embeddings.

In the context of hypernymy, a church is a type of [mask].					
BERT (OURS)	church 74.78	**religion** 2.83	structure 1.25	building 1.11	**worship** 0.86
BERT+ BALAUR	church 27.33	building 21.45	structure 15.57	place 2.41	object 1.83
In the context of hypernymy, a [mask] is a type of poem.					
BERT (OURS)	poem 91.72	**poet** 0.84	poetry 0.55	verse 0.50	**word** 0.35
BERT+ BALAUR	poem 66.23	verse 3.80	song 3.46	poetry 2.47	" " 1.67
In the context of hypernymy, a volcano is a type of [mask].					
BERT (OURS)	volcano 88.30	**lava** 1.59	cone 1.11	**rock** 0.94	**eruption** 0.88
BERT+ BALAUR	volcano 69.54	mountain 13.27	structure 2.55	object 0.80	**rock** 0.69

Table 4.3 Top-5 completions and probability percentages for selected clozes, showcasing how BALAUR can help disentangle hypernymy from other forms of semantic relatedness (related but invalid completions are bolded).

It is also interesting to note that BALAUR spreads its probability mass more evenly across predictions, better capturing the one-to-many nature of hypernymy relations. However, we observe that many of the seemingly valid completions are not actually gold-standard completions in HYPCC. This is because HYPCC considers only direct hypernymy relations in WordNet, while several completions are indirect hypernymy relations or not in WordNet. We further discuss these limitations in §A.1.3.

4.2 Monotonicity NLI

Task Description Our second evaluation is taken from Geiger et al. (2020), who create MoNLI: a challenge NLI dataset where entailment is determined by hypernymy. For instance, "*A man is talking to someone in a taxi*" entails "*A man is talking to someone in a car*". While models finetuned on SNLI (Bowman et al., 2015) perform well on such examples, they fail to generalize on examples where negation reverses entailment. For instance, "*A man is not talking to someone in a car*" now entails "*A man is not talking to someone in a taxi*". MoNLI is divided into PMoNLI and NMoNLI to distinuish between positive and negated examples.

Evaluation Method We follow the evaluation procedure of Geiger et al. (2020), reporting test set accuracies for models finetuned on SNLI, and models also finetuned on MoNLI[2]. We follow the NLI finetuning procedure of 24hBERT (Izsak et al., 2021) on which our model is based. However, we found that performance is sensitive to random seeds, so we report results averaged across 5 seeds.

Results and Discussion In Table 4.4, we replicate the results of Geiger et al. (2020), finding that models finetuned on SNLI only generalize to PMoNLI but fail completely on NMoNLI. Unexpectedly, we find BALAUR significantly improves both SNLI and PMoNLI performance in this setting, suggesting examples in SNLI also benefit from the representations learned with BALAUR pretraining.

[2]Consistently with Geiger et al. (2020), models fineuned on MoNLI are only tested on NMoNLI as there is no systematic split of PMoNLI

MODEL	SNLI	PMONLI	NMONLI
SNLI FINETUNING ONLY			
BERT (OURS)	85.44	65.51	0.50
BERT+BALAUR	86.49	76.92	0.10
SNLI + MONLI FINETUNING			
BERT (OURS)	85.43	-	48.90
BERT+BALAUR	86.38	-	56.50

Table 4.4 SNLI and MoNLI accuracies.

However, we conversely find that BALAUR degrades performance on the withheld test set of NMoNLI. While BALAUR may help LMs better capture hypernymy, the fact that it does not account for negation may help explain this result. Furthermore, visualizing performance across seeds in §A.2.3, we observe markedly larger variance on NMoNLI compared to SNLI and PMoNLI, making this result more difficult to interpret reliably.

4.3 Finetuning Efficiency

Task Description Our final evaluation reframes §4.1 and §4.2 not in terms of zero-shot or final performance, but in terms of performance throughout the finetuning of a pretrained model — as proposed by Talmor et al. (2020) in oLMpics. This approach was originally proposed because finetuning pretrained LMs makes it hard to disentangle what is captured in the pretrained representations from what is learned during finetuning. On one hand, zero-shot performance on prompt completion (§4.1) does not account for issues such as potential mismatches with the original language modeling objective (Hanna and Mareček, 2021), where poor performance may be attributable to valid completions not related to hypernymy.

On the other hand, final fine-tuned performance on MoNLI (§4.2) might not even depend on the original pretrained representations and be largely attributable to the finetuning process. The evaluation protocol of Talmor et al. (2020) enables us to more comprehensively evaluate how well different models capture hypernymy in their representations, accounting for the two potential confounds outlined above. A key assumption underlying this evaluation is that models which better capture hypernymy in their pretrained representations will be finetuned more efficiently (i.e. with better finetuned performance relative to finetuning steps, throughout finetuning).

Evaluation Method We finetune our models using the hyperparameters in the code published by Talmor et al. (2020) and follow their procedure. This includes freezing model parameters for the prompt completion task (leaving only the language modeling head unfrozen); while leaving the entire model unfrozen for the NLI task. We perform 5-fold cross-validation with a 20% split, and average validation set results. Importantly, the splits are systematic to ensure that no hypernyms or hyponyms occur in both train and validation sets.

Results and Discussion (Prompt Completion) In Figure 4.1, we see that BALAUR's improvement on hypernym prediction extends throughout finetuning, indicating better transfer learning abilities. We show similar results for the hyponym subset of HYPCC in §A.2.2.

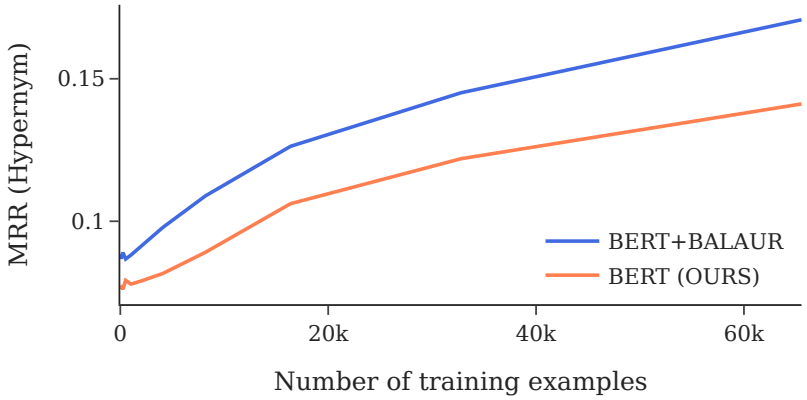

Figure 4.1 Average open-vocab MRR throughout finetuning on the hypernym prediction subset of HYPCC.

However, it is puzzling that performance remains relatively low despite extensive fine-tuning. A closer look at the outputs of the final model reveals that many of the erroneous entries in the model's top-10 open vocabulary predictions were in fact other classes in the HYPCC dataset (i.e. tokens from the closed vocabulary). In Figure 4.2, we quantify the class intrusion rate as the proportion of top-10 predictions which are both erroneous and a class in HYPCC, finding that it increases significantly throughout finetuning.

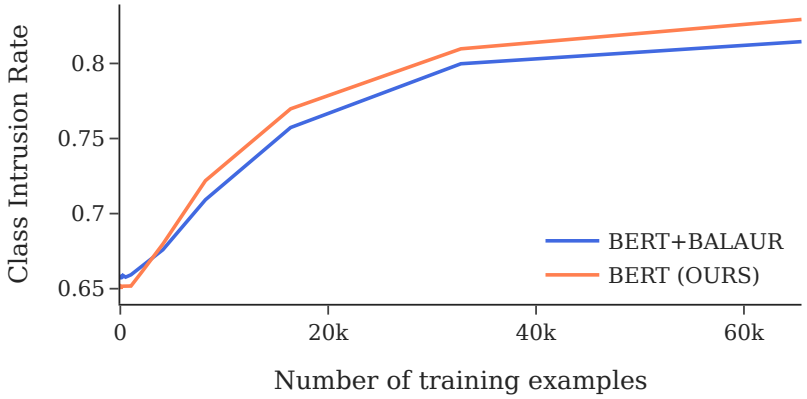

Figure 4.2 Average class intrusion rate throughout finetuning on the hypernym prediction subset of HYPCC.

One possible explanation is that models learn to predict indirect hypernyms or hyponyms

not accounted for in HypCC, similar to examples in Table 4.3. However, a manual inspection of model predictions showed that this was not often the case.

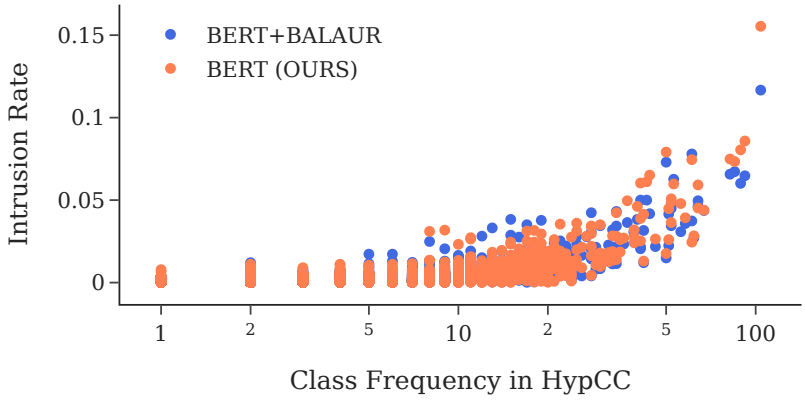

Figure 4.3 Average intrusion rate and frequency of classes in the final models finetuned on the hypernym prediction subset of HypCC.

Instead, in Figure 4.3, we find that the intrusion rate of a class grows with its frequency in the finetuning dataset. Given that intrusion rates increase with finetuning and that frequent classes have higher intrusion rates, this suggests that LMs struggle to discriminate single token differences in prompts, and instead conflate learning signal across prompts with more frequent classes dominating.

Results and Discussion (MoNLI) In Figure 4.4, we observe similar results for MoNLI, indicating that BALAUR improves finetuning efficiency. In contrast to the results in §4.2, we also observe in Table 4.5 that BALAUR improves final performance on systematic validation splits for both PMoNL and NMoNLI. These improvements are consistent even when stratifying by BALAUR coverage of the hypernym and hyponym in a given MoNLI example, i.e. whether or not BALAUR models hypernymy or hyponymy relations for these tokens.

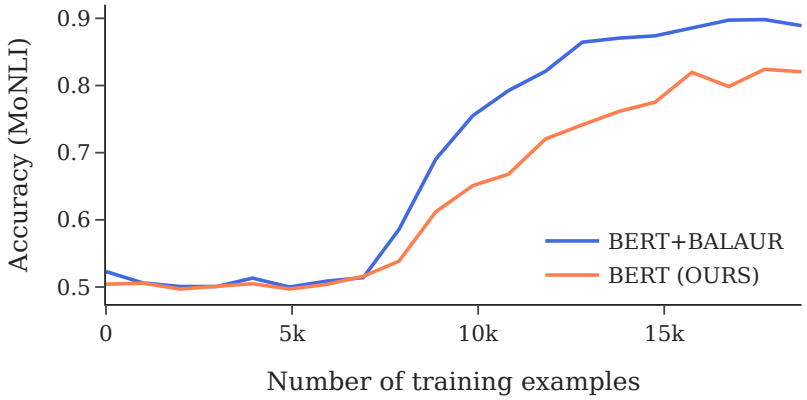

Figure 4.4 Average accuracy throughout finetuning.

	Overall Accuracy	Accuracy by BALAUR Coverage		
		Hypernym+Hyponym	Hypernym	No Coverage
		PMoNLI		
BERT (OURS)	82.14	80.69	85.07	58.33
BERT+BALAUR	86.78	86.19	88.27	72.22
		NMoNLI		
BERT (OURS)	80.31	78.00	81.81	-
BERT+BALAUR	93.01	91.44	94.02	-

Table 4.5 Final performance on MoNLI subsets, averaged across five systematic validation splits and stratified by BALAUR coverage. Due to insufficient examples where only the hyponym is covered, there is no "Hyponym" entry in this table. Similarly, there were no NMoNLI validation examples with no BALAUR coverage.

Chapter 5

Conclusion

In this work, we set out to align the linguistic behavior of LMs with our knowledge of LSRs and improve their performance on hypernymy-informed tasks. We presented BALAUR, an approach that aims to guide the linguistic behavior of LMs in such a way by modeling LSRs directly in their hidden states throughout pretraining.

Underlying this proposed approach was the hypothesis that LM latent representations can provide an interface to their linguistic behavior, and that controlling one can help guide the other. To verify our hypothesis, we characterized the effect of BALAUR on a series of evaluations, targeting several distinct ways in which LMs might capture hypernymy in their linguistic behavior.

Our findings show that BALAUR can robustly improve performance on diverse hypernymy-informed tasks, validating the effectiveness of our method while supporting our original hypothesis. Notably, we demonstrated that BALAUR also improves performance on the original language modeling objective, indicating our method's improvements are not limited

to hypernymy-informed tasks and can extend to more general linguistic behavior. However, we found that aligning the linguistic behavior of LMs with BALAUR still poses several challenges.

More broadly, BALAUR is a general-purpose architecture for modeling relations in the latent representations of neural network models. While our work has focused on modeling LSRs in LM final hidden states throughout pretraining, BALAUR can in principle be applied to different modalities, architectures, latent representations, relations or optimization settings. How well our results and hypothesis generalize to such different settings remains an open question.

Appendix A

Appendix

A.1 Additional Method Details

A.1.1 Filtering LSRs from WordNet

Filtering Tokens When mapping tokens to WordNet synsets using NLTK, we observed several potential sources of noise that could be addressed by filtering tokens. First, we observed that many tokens with 3 or fewer characters were often over-zealously lemmatized by NLTK as acronyms or abbreviations for unlikely synsets. For example `cat` may map to **`computerized_tomography.n.01`**, while `in` maps to **`indium.n.01`**, **`indiana.n.01`**, and **`inch.n.01`**. Originally, we attempted to filter any token with 3 or fewer characters, however our coverage of important concepts dropped significantly, so we limit ourselves to filtering tokens with 2 or fewer characters. We also filter out tokens which are wordpieces in the model vocabulary (e.g. tokens prefixed by "##" in the vocabulary of BERT, indicating these are not preceded by whitespace and occur in the middle of words) to ensure we only model LSRs for tokens that correspond to entire words. We also use a WordNet stoplist (Pedersen and Banerjee, 2009) to filter common function words that tend to be misrepresented by WordNet. Lastly, we limit ourselves to alphabetical tokens, as we found numerical and alphanumerical tokens to introduce a lot of noise.

Filtering Synsets Having filtered tokens, we then map each of these to all possible synsets using the NLTK interface to WordNet. However, we found the quality, coverage and ambiguity of annotations to vary significantly across synset types. To reduce noise, we filtered synset categories based on manual inspection. We first limit ourselves to noun synsets, and filter what we found to be particularly noisy categories: quantity, motive, shape, relation, and process. Furthermore, we found that despite filtering tokens from our stoplist, NLTK was still lemmatizing other tokens to synsets in the stoplist, so we further filter any synset

whose identifiers are in the stoplist.

Filtering Token-Concept Pairs After mapping hypernymy, hyponymy, synonymy and antonymy relations between tokens and synsets, we filter synsets based on their coverage of our model's vocabulary. Specifically, our goal is to avoid modeling LSRs for synsets that only relate to one item in our vocabulary, as these cannot provide any useful inductive bias to our model's representations of its vocabulary. We first keep any synsets which map to 2 or more tokens (i.e. capture synonymy). If a remaining synset has antonymous synsets, we keep it if both it and it's antonym(s) have corresponding tokens in the model vocabulary. Lastly, if a remaining synset belongs in a hypernymy or hyponymy relation, we keep it even if it does not map to a token, as long as it relates to two or more hypernym or hyponym synsets that do. This enables us to indirectly model concepts not in the model vocabulary via co-hyponymy and co-hypernymy relations. Any remaining synset is removed, along with it's related token-concept pairs. This filtering ensures that we model concepts relating to multiple tokens in our vocabulary and prevents the degenerate case where a concept is indistinguishable from a token.

A.1.2 Hypernymy-informed Cloze Completion Task

To create HYPCC, we first extract related token-concept pairs using the same procedure outlined in §3.3.1 and A.1.1. One notable difference is that we only consider direct hypernyms, instead of multi-hop hypernyms up to depth 3. Furthermore, we filter tokens such that they occur in the two most frequent English LM vocabularies: `bert-base-uncased` and `gpt2`, as hosted by the transformers library (Wolf et al., 2020).

We then convert token-concept pairs to sets of token-token pairs, based on the concepts' surface forms which are present in our vocabularies. To convert these pairs to cloze-style prompts, we adopt the following template: "A(n) X is a type of Y". We use the inflect library [1] to filter plural forms or determine the adequate article ("a" or "an"). While we do not account for uncountable nouns, we find that most prompts maintain their legibility.

Lastly, we found that several concepts and tokens were disproportionately represented in this dataset as a result of having multiple wordsenses or maintaining a high position in the WordNet hierarchy. These often lead to nonsensical prompts, which we attempted to filter out using a manually curated stoplist for tokens and concepts [2].

[1] https://github.com/jaraco/inflect
[2] github_link_to_be_published

A.1.3 Noise, bias and coverage in WordNet

When using knowledge bases such as WordNet, it is important to account for their inherent limitations. In particular, we identify three prevalent issues in WordNet that can negatively affect what LMs learn in our experiments.

First is the problem of noise. Due to issues with lemmatization, word sense granularity and idiomaticity, we found many questionable relations being extracted when creating training data for BALAUR and examples for HYPCC. For example, we find that "cat" is lemmatized to the concept "cat-o'-nine-tails.n.01", implying "cat" has the hypernym "whip.n.01". Conversely, word sense granularity can lead to questionable relations like "chair (professorship.n.01)" being a hyponym of "situation (position.n.06)". Lastly, idioms like "taking a crack at something" can lead to (unlikely when taken out of context) relations like "crack" having the hypernym "endeavor". These limitations are exacerbated in our experiments, as we do not disambiguate word senses, considering all possible meanings of a given token instead.

Second is the issue of bias. We found WordNet to encode several harmful biases and stereotypes, either directly via harmful relations, or indirectly by including certain relations for some groups but not others. For example, when comparing hyponyms for "man" and "woman" we found significant occurrences of both types of bias (see Table A.1).

Hyponym Associations in WordNet			
MAN			
boy	commando	ranger	gunner
veteran	officer	sailor	bachelor
gentleman	patriarch	gallant	swell
dude	stud	bull	sir
WOMAN			
girl	nanny	nurse	siren
amazon	whore	baggage	mistress
wife	widow	flirt	tease
broad	peach	dish	sweetheart

Table A.1 Harmful biases in WordNet hyponyms.

Despite removing these associations in our work, we want to note that these kind of biases can be difficult to comprehensively account for when expressed as selective inclusion or omission of associations for different groups.

Lastly, is the related issue of coverage. Many concepts and relations are simply not expressed in WordNet; limiting the knowledge of LSRs that can be incorporated in LMs

with this resource. This lack of coverage is exacerbated in our experiments, as we are limited to single token words (i.e. words in the model's vocabulary). Despite trying to alleviate this by also modeling extra-vocabulary concepts, effectively controlling the representations of multi-token expressions in LMs remains an open problem. We note that, due to it's reliance on expert lexicographers, WordNet has had limited updates and developments to increase it's coverage; this is in contrast to the open sourced English WordNet 2019 (McCrae et al., 2019). We suggest future work consider this resource to mitigate coverage issues.

A.1.4 Detailed counts for BALAUR relations

RELATION	TOKEN COUNT	SYNSET COUNT	RELATION COUNT
HYPERNYMY	16601	5968	49283
HYPONYMY	8764	12106	55262
SYNONYMY	14919	13241	46902
ANTONYMY	1133	483	1491

Table A.2 Number of tokens, synsets, and related token-synset pairs for each relation in BALAUR.

A.2 Additional results

A.2.1 Extended zero-shot results on HypCC

Model	Closed Vocab Acc@1/5	Closed Vocab Mrr	Open Vocab Acc@1/5	Open Vocab Mrr
Hypernym Prediction				
$\text{BERT}_{\text{Base}}$	2.75 / 12.88	0.081	0.30 / 10.25	0.054
$\text{BERT}_{\text{Large}}$	3.53 / 14.13	0.092	1.78 / 11.77	0.071
$\text{RoBERTa}_{\text{Base}}$	4.46 / 15.54	0.103	1.90 / 12.14	0.074
$\text{RoBERTa}_{\text{Large}}$	7.01 / 20.12	0.137	5.29 / 17.00	0.114
BERT (ours)	5.18 / 18.61	0.121	0.88 / 14.72	0.080
BERT+Balaur (ours)	5.31 / 19.65	0.128	1.60 / 15.44	0.089
Hyponym Prediction				
$\text{BERT}_{\text{Base}}$	1.99 / 11.89	0.073	1.39 / 10.42	0.061
$\text{BERT}_{\text{Large}}$	3.60 / 14.95	0.097	2.76 / 12.87	0.083
$\text{RoBERTa}_{\text{Base}}$	2.94 / 12.06	0.080	2.24 / 9.92	0.066
$\text{RoBERTa}_{\text{Large}}$	3.89 / 12.90	0.091	3.37 / 11.55	0.081
BERT (ours)	2.69 / 12.22	0.081	2.03 / 10.65	0.069
BERT+Balaur (ours)	3.49 / 17.91	0.110	1.85 / 14.56	0.084

Table A.3 Zero-shot results on HypCC across MLMs.

A.2.2 Extended finetuning results on HYPCC

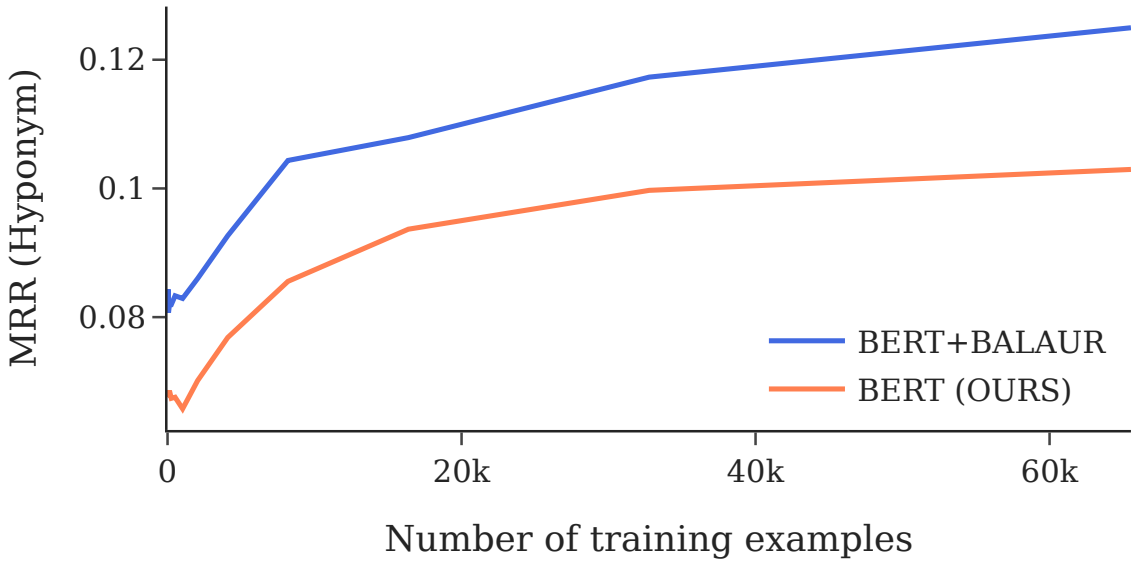

Figure A.1 Average open-vocab MRR throughout finetuning on the hyponym prediction subset of HYPCC.

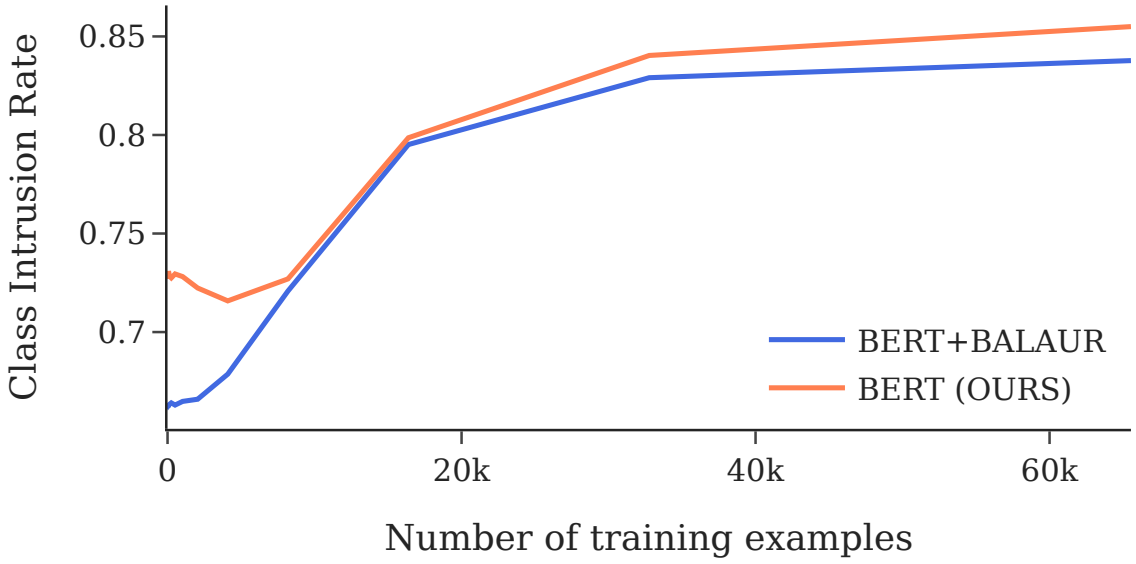

Figure A.2 Average class intrusion rate throughout finetuning on the hyponym prediction subset of HYPCC.

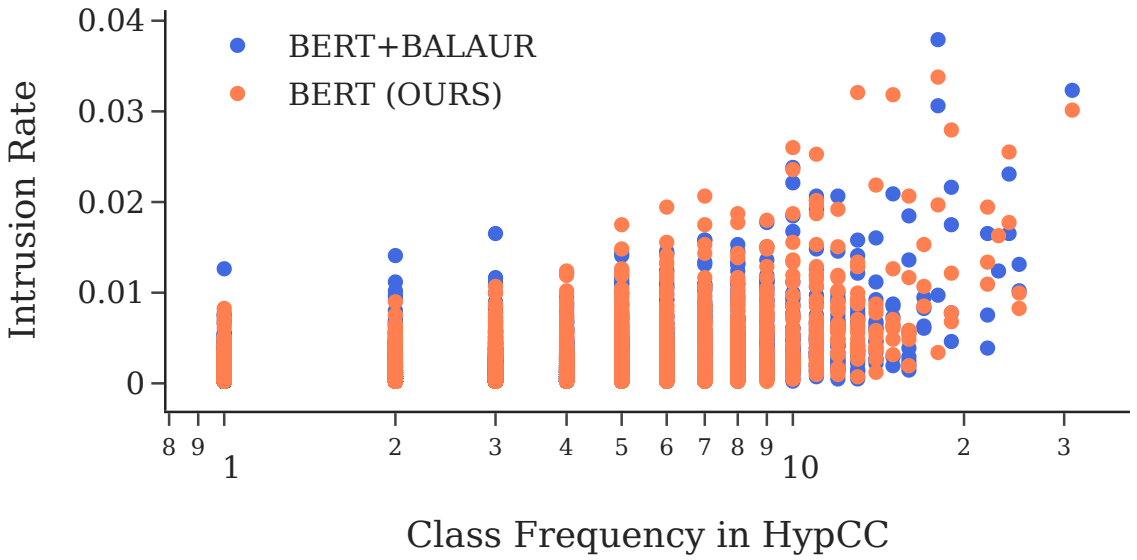

Figure A.3 Average intrusion rate and frequency of classes in the final models finetuned on the hyponym prediction subset of HYPCC.

A.2.3 MoNLI performance across random seeds

Figure A.4 MoNLI performance across 5 seeds when finetuned only on SNLI.

Figure A.5 MoNLI performance across 5 seeds when finetuned on SNLI and MoNLI.

References

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., Soroa, A., 2009. A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches, in: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Boulder, Colorado. pp. 19–27. URL: https://aclanthology.org/N09-1003.

Andreas, J., 2022. Language Models as Agent Models. URL: http://arxiv.org/abs/2212.01681, doi:10.48550/arXiv.2212.01681. arXiv:2212.01681 [cs].

Arora, K., Chakraborty, A., Cheung, J.C.K., 2020. Learning Lexical Subspaces in a Distributional Vector Space. Transactions of the Association for Computational Linguistics 8, 311–329. URL: https://doi.org/10.1162/tacl_a_00316, doi:10.1162/tacl_a_00316.

Bai, H., Wang, T., Sordoni, A., Shi, P., 2022. Better Language Model with Hypernym Class Prediction. arXiv:2203.10692 [cs] URL: http://arxiv.org/abs/2203.10692. arXiv: 2203.10692.

Bandy, J., Vincent, N., 2021. Addressing "Documentation Debt" in Machine Learning: A Retrospective Datasheet for BookCorpus. Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1. URL: https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/54229abfcfa5649e7003b83dd4755294-Abstract-round1.html.

Bar-Haim, R., Szpektor, I., Glickman, O., 2005. Definition and Analysis of Intermediate Entailment Levels, in: Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, Association for Computational Linguistics, Ann Arbor, Michigan. pp. 55–60. URL: https://aclanthology.org/W05-1210.

Baroni, M., 2012. Distributional semantics with eyes: Enriching corpus-based models of word meaning with automatically extracted visual features. URL: https://marcobaroni.org/publications/lectures/eyed-distsem-saarbruecken-colloquium-2012.pdf.

Baroni, M., Lenci, A., 2011. How we BLESSed distributional semantic evaluation, in: Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics, Association for Computational Linguistics, Edinburgh, UK. pp. 1–10. URL: https://aclanthology.org/W11-2501.

Belinkov, Y., Glass, J., 2019. Analysis Methods in Neural Language Processing: A Survey. Transactions of the Association for Computational Linguistics 7, 49–72. URL: https://aclanthology.org/Q19-1004, doi:10.1162/tacl_a_00254.

Bender, E.M., Gebru, T., McMillan-Major, A., Shmitchell, S., 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜, in: Proceedings of the 2021 ACM Con-

ference on Fairness, Accountability, and Transparency, Association for Computing Machinery, New York, NY, USA. pp. 610–623. URL: https://doi.org/10.1145/3442188.3445922, doi:10.1145/3442188.3445922.

Bender, E.M., Koller, A., 2020. Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online. pp. 5185–5198. URL: https://aclanthology.org/2020.acl-main.463, doi:10.18653/v1/2020.acl-main.463.

Bird, S., Loper, E., 2004. NLTK: The Natural Language Toolkit, in: Proceedings of the ACL Interactive Poster and Demonstration Sessions, Association for Computational Linguistics, Barcelona, Spain. pp. 214–217. URL: https://aclanthology.org/P04-3031.

Bordes, A., Glorot, X., Weston, J., Bengio, Y., 2012. Joint Learning of Words and Meaning Representations for Open-Text Semantic Parsing, in: Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, PMLR. pp. 127–135. URL: https://proceedings.mlr.press/v22/bordes12.html.

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., Yakhnenko, O., 2013. Translating embeddings for modeling multi-relational data, in: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, Curran Associates Inc., Red Hook, NY, USA. pp. 2787–2795.

Bowman, S.R., Angeli, G., Potts, C., Manning, C.D., 2015. A large annotated corpus for learning natural language inference, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal. pp. 632–642. URL: https://aclanthology.org/D15-1075, doi:10.18653/v1/D15-1075.

Budanitsky, A., Hirst, G., 2006. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. Computational Linguistics 32, 13–47. URL: https://aclanthology.org/J06-1003, doi:10.1162/coli.2006.32.1.13.

Caraballo, S.A., 1999. Automatic construction of a hypernym-labeled noun hierarchy from text, in: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, College Park, Maryland, USA. pp. 120–126. URL: https://aclanthology.org/P99-1016, doi:10.3115/1034678.1034705.

Charles, W.G., Miller, G.A., 1989. Contexts of antonymous adjectives. Applied Psycholinguistics 10, 357–375. URL: https://www.cambridge.org/core/journals/applied-psycholinguistics/article/abs/contexts-of-antonymous-adjectives/8E44256EED8319133802360429F5017F, doi:10.1017/S0142716400008675.

Ciaramita, M., Johnson, M., 2003. Supersense Tagging of Unknown Nouns in WordNet, in: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, pp. 168–175. URL: https://aclanthology.org/W03-1022.

Collins, A.M., Quillian, M.R., 1969. Retrieval time from semantic memory. Journal of Verbal Learning and Verbal Behavior 8, 240–247. URL: https://www.sciencedirect.com/science/article/pii/S0022537169800691, doi:10.1016/S0022-5371(69)80069-1.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P., 2011. Natural Language Processing (Almost) from Scratch. The Journal of Machine Learning Research 12, 2493–2537.

Conneau, A., Kruszewski, G., Lample, G., Barrault, L., Baroni, M., 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Melbourne, Australia. pp. 2126–2136. URL: https://aclanthology.org/P18-1198, doi:10.18653/v1/P18-1198.

Dagan, I., Glickman, O., Magnini, B., 2006. The PASCAL Recognising Textual Entailment Challenge, in: Quiñonero-Candela, J., Dagan, I., Magnini, B., d'Alché Buc, F. (Eds.), Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment, Springer, Berlin, Heidelberg. pp. 177–190. doi:10.1007/11736790_9.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs] URL: http://arxiv.org/abs/1810.04805. arXiv: 1810.04805.

Du, M., He, F., Zou, N., Tao, D., Hu, X., 2022. Shortcut Learning of Large Language Models in Natural Language Understanding: A Survey. URL: http://arxiv.org/abs/2208.11857, doi:10.48550/arXiv.2208.11857. arXiv:2208.11857 [cs].

Ettinger, A., 2020. What BERT Is Not: Lessons from a New Suite of Psycholinguistic Diagnostics for Language Models. Transactions of the Association for Computational Linguistics 8, 34–48. URL: https://doi.org/10.1162/tacl_a_00298, doi:10.1162/tacl_a_00298. publisher: MIT Press.

Faruqui, M., Dodge, J., Jauhar, S.K., Dyer, C., Hovy, E., Smith, N.A., 2015. Retrofitting Word Vectors to Semantic Lexicons, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Denver, Colorado. pp. 1606–1615. URL: https://aclanthology.org/N15-1184, doi:10.3115/v1/N15-1184.

Fellbaum, C., 2013. George A. Miller. Computational Linguistics 39, 1–3. URL: https://direct.mit.edu/coli/article/39/1/1-3/1429, doi:10.1162/COLI_a_00131.

Fellbaum, C., Miller, G., 1998. Combining Local Context and Wordnet Similarity for Word Sense Identification, in: WordNet: An Electronic Lexical Database. MIT Press, pp. 265–283. URL: https://ieeexplore.ieee.org/document/6287675.

Fillmore, C.J., Baker, C.F., Sato, H., 2002. The FrameNet Database and Software Tools, in: Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02), European Language Resources Association (ELRA), Las Palmas, Canary Islands - Spain. URL: http://www.lrec-conf.org/proceedings/lrec2002/pdf/140.pdf.

Fischler, I., Bloom, P.A., Childers, D.G., Roucos, S.E., Perry, N.W., 1983. Brain Potentials Related to Stages of Sentence Verification. Psychophysiology 20, 400–409. URL: https://onlinelibrary.wiley.com/doi/10.1111/j.1469-8986.1983.tb00920.x, doi:10.1111/j.1469-8986.1983.tb00920.x.

Fried, D., Duh, K., 2015. Incorporating Both Distributional and Relational Semantics in Word Representations. URL: http://arxiv.org/abs/1412.4369, doi:10.48550/arXiv.1412.4369. arXiv:1412.4369 [cs].

Geffet, M., Dagan, I., 2005. The Distributional Inclusion Hypotheses and Lexical Entailment, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics

(ACL'05), Association for Computational Linguistics, Ann Arbor, Michigan. pp. 107–114. URL: https://aclanthology.org/P05-1014, doi:10.3115/1219840.1219854.

Geiger, A., Richardson, K., Potts, C., 2020. Neural Natural Language Inference Models Partially Embed Theories of Lexical Entailment and Negation, in: Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, Association for Computational Linguistics, Online. pp. 163–173. URL: https://aclanthology.org/2020.blackboxnlp-1.16, doi:10.18653/v1/2020.blackboxnlp-1.16.

Glavaš, G., Vulić, I., 2018. Discriminating between Lexico-Semantic Relations with the Specialization Tensor Model, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), Association for Computational Linguistics, New Orleans, Louisiana. pp. 181–187. URL: https://aclanthology.org/N18-2029, doi:10.18653/v1/N18-2029.

Hanna, M., Mareček, D., 2021. Analyzing BERT's Knowledge of Hypernymy via Prompting, in: Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, Association for Computational Linguistics, Punta Cana, Dominican Republic. pp. 275–282. URL: https://aclanthology.org/2021.blackboxnlp-1.20, doi:10.18653/v1/2021.blackboxnlp-1.20.

Hearst, M.A., 1992. Automatic Acquisition of Hyponyms from Large Text Corpora, in: COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics. URL: https://aclanthology.org/C92-2082.

Hendrycks, D., Gimpel, K., 2020. Gaussian Error Linear Units (GELUs). URL: http://arxiv.org/abs/1606.08415, doi:10.48550/arXiv.1606.08415. arXiv:1606.08415 [cs].

Izsak, P., Berchansky, M., Levy, O., 2021. How to Train BERT with an Academic Budget, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic. pp. 10644–10652. URL: https://aclanthology.org/2021.emnlp-main.831, doi:10.18653/v1/2021.emnlp-main.831.

Jurafsky, D., Martin, J.H., 2023. Speech and Language Processing. 3 ed. URL: https://web.stanford.edu/~jurafsky/slp3/.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D., 2021. Supervised Contrastive Learning. arXiv:2004.11362 [cs, stat] URL: http://arxiv.org/abs/2004.11362. arXiv: 2004.11362.

Lauscher, A., Vulić, I., Ponti, E.M., Korhonen, A., Glavaš, G., 2020. Specializing Unsupervised Pretraining Models for Word-Level Semantic Similarity, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online). pp. 1371–1383. URL: https://aclanthology.org/2020.coling-main.118, doi:10.18653/v1/2020.coling-main.118.

Levine, Y., Lenz, B., Dagan, O., Ram, O., Padnos, D., Sharir, O., Shalev-Shwartz, S., Shashua, A., Shoham, Y., 2020. SenseBERT: Driving Some Sense into BERT, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online. pp. 4656–4667. URL: https://aclanthology.org/2020.acl-main.423, doi:10.18653/v1/2020.acl-main.423.

Lhoest, Q., Villanova del Moral, A., Jernite, Y., Thakur, A., von Platen, P., Patil, S., Chaumond, J., Drame, M., Plu, J., Tunstall, L., Davison, J., Šaško, M., Chhablani, G., Malik, B., Brandeis, S., Le Scao, T., Sanh, V., Xu, C., Patry, N., McMillan-Major, A., Schmid, P., Gugger, S., Delangue, C., Matussière, T., Debut, L., Bekman, S., Cistac, P., Goehringer, T., Mustar, V., Lagunas, F., Rush, A., Wolf, T., 2021. Datasets: A Community Library for Natural Language Processing, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic. pp. 175–184. URL: https://aclanthology.org/2021.emnlp-demo.21, doi:10.18653/v1/2021.emnlp-demo.21.

Li, K., Hopkins, A.K., Bau, D., Viégas, F., Pfister, H., Wattenberg, M., 2023. Emergent World Representations: Exploring a Sequence Model Trained on a Synthetic Task. URL: http://arxiv.org/abs/2210.13382, doi:10.48550/arXiv.2210.13382. arXiv:2210.13382 [cs].

Linzen, T., Dupoux, E., Goldberg, Y., 2016. Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. Transactions of the Association for Computational Linguistics 4, 521–535. URL: https://aclanthology.org/Q16-1037, doi:10.1162/tacl_a_00115.

Loshchilov, I., Hutter, F., 2019. Decoupled Weight Decay Regularization, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net. URL: https://openreview.net/forum?id=Bkg6RiCqY7.

Marvin, R., Linzen, T., 2018. Targeted Syntactic Evaluation of Language Models, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium. pp. 1192–1202. URL: https://aclanthology.org/D18-1151, doi:10.18653/v1/D18-1151.

McCarthy, D., 2006. Relating WordNet Senses for Word Sense Disambiguation, in: Proceedings of the Workshop on Making Sense of Sense: Bringing Psycholinguistics and Computational Linguistics Together. URL: https://aclanthology.org/W06-2503.

McCrae, J.P., Rademaker, A., Bond, F., Rudnicka, E., Fellbaum, C., 2019. English WordNet 2019 – An Open-Source WordNet for English, in: Proceedings of the 10th Global Wordnet Conference, Global Wordnet Association, Wroclaw, Poland. pp. 245–252. URL: https://aclanthology.org/2019.gwc-1.31.

Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013a. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 [cs] URL: http://arxiv.org/abs/1301.3781. arXiv:1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J., 2013b. Distributed Representations of Words and Phrases and their Compositionality, in: Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://papers.nips.cc/paper_files/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

Miller, G.A., 1995. WordNet: a lexical database for English. Communications of the ACM 38, 39–41. URL: https://doi.org/10.1145/219717.219748, doi:10.1145/219717.219748.

Miller, G.A., Fellbaum, C., 1991. Semantic networks of english. Cognition 41, 197–229. URL: https://linkinghub.elsevier.com/retrieve/pii/0010027791900364, doi:10.1016/0010-0277(91)90036-4.

Miller, G.A., Fellbaum, C., 2007. WordNet then and now. Language Resources and Evalua-

tion 41, 209–214. URL: https://doi.org/10.1007/s10579-007-9044-6, doi:10.1007/s10579-007-9044-6.

Mohammad, S., Dorr, B., Hirst, G., 2008. Computing Word-Pair Antonymy, in: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Honolulu, Hawaii. pp. 982–991. URL: https://aclanthology.org/D08-1103.

Mrkšić, N., Ó Séaghdha, D., Thomson, B., Gašić, M., Rojas-Barahona, L.M., Su, P.H., Vandyke, D., Wen, T.H., Young, S., 2016. Counter-fitting Word Vectors to Linguistic Constraints, in: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, San Diego, California. pp. 142–148. URL: https://aclanthology.org/N16-1018, doi:10.18653/v1/N16-1018.

Nanda, N., Chan, L., Lieberum, T., Smith, J., Steinhardt, J., 2023. Progress measures for grokking via mechanistic interpretability. URL: http://arxiv.org/abs/2301.05217, doi:10.48550/arXiv.2301.05217. arXiv:2301.05217 [cs].

Palmer, M., Gildea, D., Kingsbury, P., 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. Computational Linguistics 31, 71–106. URL: https://aclanthology.org/J05-1004, doi:10.1162/0891201053630264.

Pandia, L., Ettinger, A., 2021. Sorting through the noise: Testing robustness of information processing in pre-trained language models, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic. pp. 1583–1596. URL: https://aclanthology.org/2021.emnlp-main.119, doi:10.18653/v1/2021.emnlp-main.119.

Pedersen, T., Banerjee, S., 2009. A WordNet Stop List. URL: https://www.d.umn.edu/~tpederse/Group01/WordNet/wordnet-stoplist.html.

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., Miller, A., 2019. Language Models as Knowledge Bases?, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China. pp. 2463–2473. URL: https://aclanthology.org/D19-1250, doi:10.18653/v1/D19-1250.

Poesio, M., Ishikawa, T., Schulte im Walde, S., Vieira, R., 2002. Acquiring Lexical Knowledge for Anaphora Resolution, in: Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02), European Language Resources Association (ELRA), Las Palmas, Canary Islands - Spain. URL: http://www.lrec-conf.org/proceedings/lrec2002/pdf/117.pdf.

Pottier, B., 1964. Vers une sémantique moderne. Centre de philologie et de littératures romanes de l'Université de Strasbourg. Google-Books-ID: 7ZcUcgAACAAJ.

Qiu, X., Sun, T., Xu, Y., Shao, Y., Dai, N., Huang, X., 2020. Pre-trained models for natural language processing: A survey. Science China Technological Sciences 63, 1872–1897. URL: https://doi.org/10.1007/s11431-020-1647-3, doi:10.1007/s11431-020-1647-3.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., 2018. Improving language under- standing with unsupervised learning. Technical Report. OpenAI.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., 2019. Language Models are Unsupervised Multitask Learners. URL: https://www.semanticscholar.org/paper/Language-Models-are-Unsupervised-Multitask-Learners-Radford-Wu/9405cc0d6169988371b2755e573cc28650d14dfe.

Ravichander, A., Hovy, E., Suleman, K., Trischler, A., Cheung, J.C.K., 2020. On the Systematicity of Probing Contextualized Word Representations: The Case of Hypernymy in BERT, in: Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics, Association for Computational Linguistics, Barcelona, Spain (Online). pp. 88–102. URL: https://aclanthology.org/2020.starsem-1.10.

Rogers, A., Kovaleva, O., Rumshisky, A., 2020. A Primer in BERTology: What We Know About How BERT Works. Transactions of the Association for Computational Linguistics 8, 842–866. URL: https://aclanthology.org/2020.tacl-1.54, doi:10.1162/tacl_a_00349.

Rozen, O., Amar, S., Shwartz, V., Dagan, I., 2021. Teach the Rules, Provide the Facts: Targeted Relational-knowledge Enhancement for Textual Inference, in: Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics, Association for Computational Linguistics, Online. pp. 89–98. URL: https://aclanthology.org/2021.starsem-1.8, doi:10.18653/v1/2021.starsem-1.8.

Santus, E., Gladkova, A., Evert, S., Lenci, A., 2016. The CogALex-V Shared Task on the Corpus-Based Identification of Semantic Relations, in: Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V), The COLING 2016 Organizing Committee, Osaka, Japan. pp. 69–79. URL: https://aclanthology.org/W16-5309.

Shwartz, V., Santus, E., Schlechtweg, D., 2017. Hypernyms under Siege: Linguistically-motivated Artillery for Hypernymy Detection, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Association for Computational Linguistics, Valencia, Spain. pp. 65–75. URL: https://aclanthology.org/E17-1007.

Snow, R., Jurafsky, D., Ng, A., 2004. Learning Syntactic Patterns for Automatic Hypernym Discovery, in: Advances in Neural Information Processing Systems, MIT Press. URL: https://proceedings.neurips.cc/paper/2004/hash/358aee4cc897452c00244351e4d91f69-Abstract.html.

Talmor, A., Elazar, Y., Goldberg, Y., Berant, J., 2020. oLMpics-On What Language Model Pre-training Captures. Transactions of the Association for Computational Linguistics 8, 743–758. URL: https://aclanthology.org/2020.tacl-1.48, doi:10.1162/tacl_a_00342.

Tenney, I., Das, D., Pavlick, E., 2019a. BERT Rediscovers the Classical NLP Pipeline, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy. pp. 4593–4601. URL: https://aclanthology.org/P19-1452, doi:10.18653/v1/P19-1452.

Tenney, I., Xia, P., Chen, B., Wang, A., Poliak, A., McCoy, R.T., Kim, N., Durme, B.V., Bowman, S.R., Das, D., Pavlick, E., 2019b. What do you learn from context? Probing for sentence structure in contextualized word representations. URL: https://openreview.net/forum?id=SJzSgnRcKX.

Trier, J., 1931. Der deutsche Wortschatz im Sinnbezirk des Verstandes; die Geschichte eines Sprachlichen feldes. Germanische Bibliothek, C. Winter, Heidelberg. OCLC: 10560682.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is All you Need, in: Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Veldhoen, S., Hupkes, D., Zuidema, W.H., 2016. Diagnostic Classifiers Revealing how Neural Networks Process Hierarchical Structure. URL: https://www.semanticscholar.org/paper/Diagnostic-Classifiers-Revealing-how-Neural-Process-Veldhoen-Hupkes/595c45a6c4fe895e00742f8316710e1177896deb.

Vulić, I., Glavaš, G., Mrkšić, N., Korhonen, A., 2018. Post-Specialisation: Retrofitting Vectors of Words Unseen in Lexical Resources, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana. pp. 516–527. URL: https://aclanthology.org/N18-1048, doi:10.18653/v1/N18-1048.

Vulić, I., Mrkšić, N., 2018. Specialising Word Vectors for Lexical Entailment, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana. pp. 1134–1145. URL: https://aclanthology.org/N18-1103, doi:10.18653/v1/N18-1103.

Vulić, I., Ponti, E.M., Litschko, R., Glavaš, G., Korhonen, A., 2020. Probing Pretrained Language Models for Lexical Semantics, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online. pp. 7222–7240. URL: https://aclanthology.org/2020.emnlp-main.586, doi:10.18653/v1/2020.emnlp-main.586.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S., 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding, in: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Association for Computational Linguistics, Brussels, Belgium. pp. 353–355. URL: https://aclanthology.org/W18-5446, doi:10.18653/v1/W18-5446.

Wittgenstein, L., 1976. Philosophical investigations. A Blackwell paperback, Blackwell, Oxford.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A., 2020. Transformers: State-of-the-Art Natural Language Processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online. pp. 38–45. URL: https://aclanthology.org/2020.emnlp-demos.6, doi:10.18653/v1/2020.emnlp-demos.6.

Xu, C., Bai, Y., Bian, J., Gao, B., Wang, G., Liu, X., Liu, T.Y., 2014. RC-NET: A General Framework for Incorporating Knowledge into Word Representations, in: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management,

Association for Computing Machinery, New York, NY, USA. pp. 1219–1228. URL: https://doi.org/10.1145/2661829.2662038, doi:10.1145/2661829.2662038.

Yu, M., Dredze, M., 2014. Improving Lexical Embeddings with Semantic Knowledge, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Baltimore, Maryland. pp. 545–550. URL: https://aclanthology.org/P14-2089, doi:10.3115/v1/P14-2089.