Detecting Conceptual Abstraction in LLMs

Michaela Regneri, Alhassan Abdelhalim, Sören Laue
Universität Hamburg
Dept. of Computer Science, Machine Learning Research Group
Hamburg, Germany
{michaela.regneri,alhassan.abdelhalim,soeren.laue}@uni-hamburg.de

Abstract

We present a novel approach to detecting noun abstraction within a large language model (LLM). Starting from a psychologically motivated set of noun pairs in taxonomic relationships, we instantiate surface patterns indicating hypernymy and analyze the attention matrices produced by BERT. We compare the results to two sets of counterfactuals and show that we can detect hypernymy in the abstraction mechanism, which cannot solely be related to the distributional similarity of noun pairs. Our findings are a first step towards the explainability of conceptual abstraction in LLMs.

Keywords: transformers, abstraction, attention, explainability

1. Introduction

Large Language Models (LLMs) have emerged as a powerful tool for a plethora of applications. State-of-the-art LLMs are based on the transformer architecture (Vaswani et al., 2017) that can directly generate text sequences (like chatbots), translate texts, or lend their outcomes to other downstream tasks. Due to their versatile functionality, LLMs are often distributed as pre-trained black-box models, which can then be fine-tuned to specific needs.

While LLMs surpass the performance of simpler models, they are far less explainable due to the intransparent nature of their complex architecture. More explainability can be crucial in multiple applications, e.g., if models must adhere to some governance to prevent bias or build more data-efficient models. Especially in the context of trustworthy AI, one central open research question is how these models' excellent output is achieved and whether the mechanisms internally employed in LLMs reassemble those present in humans.

We provide an analysis examining whether simple linguistic abstraction mechanisms are present in a large language model. For humans, relations like hypernymy (*ravens* are *birds*) are essential for linguistic understanding and generalization. LLMs also necessarily employ some kind of abstraction and generalization, but most likely not exactly in the same way as humans do. With our experiments, we add one more step toward representing hypernym relationships within large language models and, thus, their capacity to use human-like abstraction mechanisms for generalization. Specifically, we test BERT (Devlin et al., 2019) for its attention patterns related to taxonomic hypernyms and compare this to unrelated noun pairs with either high or low semantic similarity. We draw our test data from a psychologically motivated data set of human associations, which lends itself to examining hypernym pairs with high cognitive saliency. Our results show that BERT represents this kind of abstraction within its attention module.

Our main contribution consists of clear evidence that LLMs infer linguistic abstraction and that this inference goes beyond semantic similarity. For this, we provide both a method and a dataset to show the attention patterns of LLMs for semantic hypernymy and separate them from counterfactuals matched by semantic similarity and abstraction level.

2. Background and Related Work

In the past years, the capabilities of LLMs have been enhanced tremendously. With transformer models (Vaswani et al., 2017) as an architectural basis, LLMs are trained on vast amounts of text and optimized to predict the next word in a sequence or the following sentence in a discourse. The resulting models have many applications and can model many

#No	Pattern
1	[hypo]s are [hyper]s.
2	That [hypo] is [a(n)] [hyper].
3	I like [hypo]s and other [hyper]s.
4	The [hypo], which was the largest [hyper] among them, stood out.
5	I like [hypo]s, particularly because they are [hyper]s.

Table 1: Hypernymy patterns, with [hypo] and [hyper] as slots for target and the feature concepts respectively. Plurals are indicated with *s* and [a(n)] is a determiner.

linguistic phenomena known to be crucial for human language (Manning et al., 2020). When analyzing the emergence of linguistic phenomena, a particular focus lies on the self-attention mechanism of transformers. Self-attention is a step in the encoder part of these language models. It maps the input sequence to a weighted representation of itself and thus, intuitively speaking, reveals the sequence's focal points relevant to generating its follow-up. Self-attention consists of multiple so-called heads, which act in parallel on the sequence and are multiplied in several layers (see Vaswani et al. (2017) for details). The grid of attention heads with the individual scores they attach to the sequence is often treated as a proxy for the information encoded in the transformer. For some discussion on how far this is possible, see, e.g., Jain and Wallace (2019) and Wiegreffe and Pinter (2019). There are two types of approaches that recover linguistic structure in LLMs: One performs end-to-end evaluation by disabling or manipulating single attention heads and evaluating the performance change for different tasks (Kovaleva et al., 2019, e.g.). Others look directly into the attention patterns, which we also do. Baroni (2020) shows an overview of abstraction and compositionality in artificial neural networks. Many approaches use artificial languages and small models (Lake and Baroni, 2018; Hupkes et al., 2020, e.g.), others also test pretrained LLMs like BERT (Devlin et al., 2019). See Sajjad et al. (2023) for an overview. Several approaches have found evidence for linguistic knowledge within BERT. For instance, Chen et al. (2023) prompt the model with correct and counterfactual data and then infer BERT's abstraction capabilities. Only a few approaches show results for deep semantic knowledge directly within the attention mechanism. Dalvi et al. (2019b) try to discover latent concepts in BERT, which are essentially hypernyms and their derivable hyponyms. In our approach, we focus on hypernym-hyponym relations between nouns as one central linguistic abstraction phenomenon. For collecting hypernyms by prompting, Hanna and Mareček (2021) present an experiment in which BERT outperforms other unsupervised algorithms in the collection of common hypernyms, which suggest that the model at least has the capacity to user hypernymy. This raises the questions on whether and how this is also internally represented in the trained model. To the best of our knowledge, there is no approach that characterizes attention patterns for generic hypernyms, especially no approach that distinguishes taxonomic relationships from pure semantic similarity. We take another step towards understanding conceptual abstraction in LLMs, evaluate the attention patterns related to true and counterfactual hypernyms, and show that the effects must be related to abstraction rather than similarity.

3. Data

We create a data set of noun pairs that are in a hypernymy relationship, and two sets of counterfactual pairs (which are not hypernyms). In order to construct example sentences, we manually create patterns that typically express hypernymy in their surface form and instantiate them with the noun pairs.

3.1. Positive Examples

We extract hyponym-hypernym pairs from McRae's feature norms (McRae et al., 2005). The feature norms contain pairs of concepts (originally stimuli) and features (human associations), annotated with semantic relationships. The pre-selection gives us more salient pairs of terms than a full-fledged taxonomy and should also be recognizable as strongly related by a large language model. For our data of valid noun pairs, we select all pairs of concepts and features labeled with a "superordi-

nate" relationship in the feature norms. These concept-feature pairs have the target concept as a hyponym and the feature concept as a hypernym (e.g., *raven* and *bird*). The dataset can conatin multiple hypernyms for a concept with different levels of abstraction, e.g *raven* and *bird* as well as *raven* and *animal*. We include all such pairs in the dataset and balance them later with counterfactuals with similar degree of abstraction.

3.2. Creating Counterfactuals

We create two counterfactual sets of noun pairs, which are not in a hypernymy relationship and thus will produce invalid sentences within our patterns. Using WordNet (Fellbaum, 1998), we generate the pairs by either sister terms of the feature concept from the positive examples (*negative examples*), which are matched by the level of abstraction of the feature concepts, or terms which share a hypernym with the target concepts (*sister terms*), which approximately match the level of similarity of the positive examples. With those two sets, we want to exclude spurious effects from just measuring semantic similarity or differences in concept abstraction level (and thus indirectly also frequency).

Negative Examples

For the first set of counterfactuals, we elicit noun pairs in which the feature concept is on the same level of generality as the hypernym in the first set. For instance, for the positive example *raven – animal*, we might choose *raven – person*. If there are multiple hypernyms for the same concept, we select an appropriate counterfactual for each individual hypernym. In detail, we proceed as follows:

1. We map each positive example to the WordNet synsets by extracting those synset pairs that contain the respective lemmas and stand in a (direct or inherited) hypernymy relationship in WordNet.

2. For each feature synset, we select a sister term (a synset sharing a parent node), which is no hypernym of the target concept (in the example, we pick a sister term of the *alligator* synset). To avoid effects from low-frequency words, we select the most frequent lemma from those sister synsets as a counterfactual (in the example *person*).

Sister Terms

Our second counterfactual set controls for the level of semantic similarity within the positive examples. Hyponyms and their hypernyms are often distributionally very similar (Padó and Lapata, 2007), especially salient ones. To measure whether we really find differences related to violations of taxonomic rules or just effects due to high semantic similarity, we pick a sister term in WordNet for each of the original target concepts (e.g., *raven – crow*, which are both hyponyms of *bird*). As for the negative examples, we choose the most frequent sister term lemma. Sister terms usually share many contexts, so we expect effects due to semantic similarity to be shared between the positive examples and the sister terms.

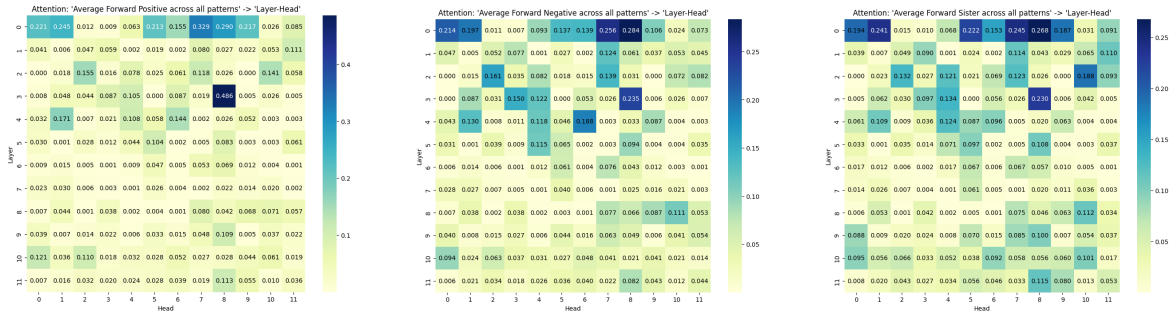
3.3. Creating Test Sentences

As input for the LLM, we create test sentences that express a taxonomic relationship directly or indirectly. First, we manually create a set of five sentence patterns that exhibit hypernymy relationships, partially inspired by the patterns used by Hearst (1992) to extract hyponym-hypernym pairs automatically from large text corpora. We vary the simplicity and saliency of the patterns to control for those effects. Table 1 shows the set of patterns.

We instantiate our patterns with the noun pairs from all three sets, resulting in 3425 examples per set. The results are sentences like *I like ravens and other animals* (positive example), and *I like ravens and other people* (negative example) and *I like ravens and other crows* (sister term). We provide all data sets for reference.

4. Hypernymy within BERT

We analyze whether or not hypernymy has a correlation with BERT's attention mechanism. After visualizing the attention for all datasets, we separate them via a linear classifier. For all



(a) Forward positive (b) Forward negative (c) Forward sister

Figure 1: Attention maps for hyponyms and hypernyms averaged across all patterns.

experiments, we use BERT-large in the monolingual English version.

4.1. Attention Matrices and Clustering

For each sentence, we extract the self-attention values from BERT. We restrict our analysis to the forward-looking attention between our target and feature tokens. Each sentence is represented by a 12x12 attention matrix (with 12 layers of 12 attention heads per layer). BERT's tokenizer breaks up some of our pluralized tokens, which makes the attention between a split token and a complete token incomparable. For the sake of simplicity, we discard all examples in which one of our tokens in focus is split up.

Figure 1 gives a high-level overview of the results for each data set. For this visualization, we average the attention between target and feature concepts (resp. their sequence position) over all examples. Each cell in a heatmap corresponds to one attention head (x-axis) in one layer (y-axis). Dark colors indicate a high average activation of the attention head. Intuitively, we see that the three sets differ, so concept clashes in the negative set and the sister terms do expose different attention patterns than the salient hypernyms. Further, the overall attention seems lower in the positive setting than in the other two control settings. This suggests that higher attention here denotes some form of surprise for unexpected semantic constructions.

To validate how well the three sets are separable, we employ logistic regression to recover the three test sets automatically. Each data point is the attention matrix of one example sentence, flattened into a 144-dimensional vector. For classification, we use the standard implementation of logistic regression from scikit-learn (Pedregosa et al., 2011) with all default parameter settings, setting the number of iterations to 1000 and the regularization parameter C to 1. We also perform a pairwise comparison of the different sets to understand how similar levels of abstraction (sister terms vs. negative) or similar levels of semantic similarity (positive vs. sister terms) of the test tokens influence the separability of the examples.

4.2. Results

We find a prediction accuracy of 0.75 on the test sets for our overall comparison and similar scores for the pairwise separation (Table 2). Our three sets are equal-sized, so a random baseline would return an accuracy of about 0.33. All scores indicate a substantial difference in attention patterns in the three sets. The positive and the negative examples are well separated. Here, we see the semantic type clash for non-hypernyms in a hypernymy pattern and the low semantic similarity of the target and feature concept. The sister terms are equally well distinguishable from both positive and negative examples, but the set is less well recoverable than the positive examples.

Sets	Acc.
All three	0.75
Pos. vs. Neg.	**0.88**
Pos. vs. Sisters	0.84
Neg. vs. Sisters	0.85

Table 2: Accuracy for predicting the test sets.

This means that the differences we see between positive and negative examples must be due to something different than semantic similarity because the sister terms are distributionally very similar to their matched positive examples. Further, the attention seems to represent the subtle difference between the two sets of counterfactuals internally, which points to interesting research questions on the level of abstraction within the transformer models.

4.3. Limitations

Our approach takes a first step towards understanding linguistic abstraction in transformer models. Our experiments have several technical limitations and limitations in the interpretability of the results.

First, we restrict ourselves to taxonomic hypernymy of nouns, which is only a small part of abstraction. Within this theoretical limitation, our dataset is also limited to the hyponym-hypernym pairs from the feature norms we used as our source.

The restriction to a dictionary-based definition of abstraction also affects our dataset. When assembling the counterfactuals fitted to the input data, we found that some of our counterfactuals are strictly speaking no hypernyms, but colloquially still treated as such, e.g., *spatula – tool*, or *barn – shelter*. We leave those examples in the dataset, which might have influenced our results.

Further limitations of our input data result from our handling of tokenized words. We filter all words that are split up by the BERT tokenizer. There are several approaches that recombine subword tokens into whole words. Unfortunately, no standard approach fits all applications, so in future work, the most suitable way to retrieve whole words from subwords should be tested and applied.

Lastly, like for every probing approach, the interpretability of our results is debatable. We have shown that words in a hypernymy relationship give rise to attention matrices that are well distinguishable from counterfactuals, which are semantically wrong assertions. One can argue that the results on our dataset mainly show that we can distinguish salient sentences from absurd ones. We think that the least our results show is that there is something that is regularly attached to hypernymy that the transformer learned. Otherwise, we would not be able to separate the two sets of counterfactuals, which both consist of unlikely sentences and which are not distinguishable in their degree of oddity (*I like ravens and other crows* is about as wrong as *I like ravens and other people*). The only nuance between those counterfactuals is the degree of abstraction in the target words. Moreover, they both are well separable from the correct sentences when matched on the level of abstraction. So, while we cannot (and did not) claim that we found the attention pattern that completely explains how taxonomic abstraction works in transformers, we can claim that there is more than semantic similarity and reasonable content that makes the differences we measure.

5. Summary and Future Work

Our experiments show an initial indicator for linguistic, conceptual abstraction in the attention mechanism of LLMs. Based on sentence patterns that imply hyponymy relations of noun pairs, we showed that we can separate sentences with salient hyponym-hypernym pairs from counterfactuals in which target and feature concepts do not stand in a taxonomic abstraction relationship. Our setting shows that the level of abstraction in the counterfactual and the semantic similarity of target and feature concepts give rise to different patterns.

Our approach can only give a limited first explanation of the presence of linguistic abstraction within transformers. Firstly, we restrict ourselves to noun pairs and hyponymy, while abstraction comprises many more types of words, relations, and complex mechanisms like frames or scenarios. Further, our experiments cannot explain how the differences in the attention mechanism arise and what they imply. To shed more light on these questions, further research is required, which should analyze both the mathematical theory of the abstraction mechanism and the statistical properties of the input word embeddings. This would make the mechanisms of conceptual abstraction within the transformer architecture more transparent.

Bibliographical References

Jacob Andreas. 2019. Measuring compositionality in representation learning. In *International Conference on Learning Representations*.

David Arps, Younes Samih, Laura Kallmeyer, and Hassan Sajjad. 2022. Probing for constituency structure in neural language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6738–6757, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Marco Baroni. 2020. Linguistic generalization and compositionality in modern artificial neural networks. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 375(1791):20190307.

Yonatan Belinkov. 2022. Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics*, 48(1):207–219.

Ziwei Chen, Linmei Hu, Weixin Li, Yingxia Shao, and Liqiang Nie. 2023. Causal intervention and counterfactual reasoning for multi-modal fake news detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 627–638, Toronto, Canada. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. 2019a. What is one grain of sand in the desert? analyzing individual neurons in deep nlp models. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

Fahim Dalvi, Abdul Khan, Firoj Alam, Nadir Durrani, Jia Xu, and Hassan Sajjad. 2022. Discovering latent concepts learned in BERT. In *International Conference on Learning Representations*.

Fahim Dalvi, Avery Nortonsmith, Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, and James Glass. 2019b. Neurox: A toolkit for analyzing individual neurons in neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9851–9852.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Kunihiko Fukushima. 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36:193–202.

Edward Grefenstette, Phil Blunsom, Nando de Freitas, and Karl Moritz Hermann. 2014. A deep architecture for semantic parsing. In *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, pages 22–27, Baltimore, MD. Association for Computational Linguistics.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1195–1205, New Orleans, Louisiana. Association for Computational Linguistics.

Michael Hanna and David Mareček. 2021. Analyzing BERT's knowledge of hypernymy via prompting. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 275–282, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*.

Dieuwke Hupkes, Verna Dankers, Mathijs Mul, and Elia Bruni. 2020. Compositionality decomposed: How do neural networks generalise? (extended abstract). In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 5065–5069. International Joint Conferences on Artificial Intelligence Organization. Journal track.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4365–4374, Hong Kong, China. Association for Computational Linguistics.

Brenden Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2873–2882. PMLR.

Christopher D. Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Ken McRae, George S. Cree, Mark S. Seidenberg, and Chris Mcnorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behavior Research Methods*, 37(4):547–559.

Sebastian Padó and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Hassan Sajjad, Nadir Durrani, and Fahim Dalvi. 2023. Neuron-level Interpretation of Deep NLP Models: A Survey. *Transactions of the Association for Computational Linguistics*, 11.

Sabine Schulte im Walde and Diego Frassinelli. 2022. Distributional measures of semantic abstraction. *Frontiers in Artificial Intelligence*, 4.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Gregoire Mesnil. 2014. Learning semantic representations using convolutional neural networks for web search. WWW 2014.

Jörg Tiedemann and Yves Scherrer. 2019. Measuring semantic abstraction of multilingual NMT with paraphrase recognition and generation tasks. In *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pages 35–42, Minneapolis, USA. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20, Hong Kong, China. Association for Computational Linguistics.