
A PHILOSOPHICAL INTRODUCTION TO LANGUAGE MODELS

PART I: CONTINUITY WITH CLASSIC DEBATES

Raphaël Millière

Department of Philosophy
Macquarie University
raphael.milliere@mq.edu.au

Cameron Buckner

Department of Philosophy
University of Houston
cjbuckner@uh.edu

ABSTRACT

Large language models like GPT-4 have achieved remarkable proficiency in a broad spectrum of language-based tasks, some of which are traditionally associated with hallmarks of human intelligence. This has prompted ongoing disagreements about the extent to which we can meaningfully ascribe any kind of linguistic or cognitive competence to language models. Such questions have deep philosophical roots, echoing longstanding debates about the status of artificial neural networks as cognitive models. This article—the first part of two companion papers—serves both as a primer on language models for philosophers, and as an opinionated survey of their significance in relation to classic debates in the philosophy of cognitive science, artificial intelligence, and linguistics. We cover topics such as compositionality, language acquisition, semantic competence, grounding, world models, and the transmission of cultural knowledge. We argue that the success of language models challenges several long-held assumptions about artificial neural networks. However, we also highlight the need for further empirical investigation to better understand their internal mechanisms. This sets the stage for the companion paper (Part II), which turns to novel empirical methods for probing the inner workings of language models, and new philosophical questions prompted by their latest developments.

1. Introduction

Deep learning has catalyzed a significant shift in artificial intelligence over the past decade, leading up to the development of Large Language Models (LLMs). The reported achievements of LLMs, often heralded for their ability to perform a wide array of language-based tasks with unprecedented proficiency, have captured the attention of both the academic community and the public at large. State-of-the-art LLMs like GPT-4 are even claimed to exhibit “sparks of general intelligence” (Bubeck et al. 2023). They can produce essays and dialogue responses that often surpass the quality of an average undergraduate student’s work (Herbold et al. 2023); they achieve better scores than most humans on a variety of AP tests for college credit and rank in the 80-99th percentile on graduate admissions tests like the GRE or LSAT (OpenAI 2023a); their programming proficiency “favorably compares to the average software engineer’s ability” (Bubeck et al. 2023, Savelka, Agarwal, An, Bogart & Sakr 2023); they can solve many difficult mathematical problems (Zhou et al. 2023)—even phrasing their solution in the form of a Shakespearean sonnet, if prompted to do so. LLMs also form the backbone of multimodal systems that can answer advanced questions about visual inputs

(OpenAI 2023b) or generate images that satisfy complex compositional relations based on linguistic descriptions (Betker et al. 2023).¹ While the released version of GPT-4 was intentionally hobbled to be unable to perfectly imitate humans—to mitigate plagiarism, deceit, and unsafe behavior—it nevertheless still managed to produce responses that were indistinguishable from those written by humans at least 30% of the time when assessed on a two-person version of the Turing test for intelligence (Jones & Bergen 2023). This rate exceeds the threshold established by Turing himself for the test: that computer programs in the 21st century should imitate humans so convincingly that an average interrogator would have less than a 70% chance of identifying them as non-human after five minutes of questioning (Turing 1950).

To philosophers who have been thinking about artificial intelligence for many years, GPT-4 can seem like a thought experiment come to life—albeit one that calls into question the link between intelligence and behavior. As early as 1981, Ned Block imagined a hypothetical system—today commonly called “**Blockhead**”—that exhibited behaviors indistinguishable from an adult human’s, yet was not considered intelligent.² Block’s challenge focused on the way in which the system produced its responses to inputs. In particular, **Blockhead**’s responses were imagined to have been explicitly preprogrammed by a “very large and clever team [of human researchers] working for a very long time, with a very large grant and a lot of mechanical help,” to devise optimal answers to any potential question the judge might ask (Block 1981, p. 20). In other words, **Blockhead** answers questions not by understanding the inputs and processing them flexibly and efficiently, but rather simply retrieving and regurgitating the answers from its gargantuan memory, like a lookup operation in a hash table. The consensus among philosophers is that such a system would not qualify as intelligent. In fact, many classes in the philosophy of artificial intelligence begin with the position Block and others called “psychologism:” intelligence does not merely depend on the observable behavioral dispositions of a system, but also on the nature and complexity of internal information processing mechanisms that drive these behavioral dispositions.

In fact, many of GPT-4’s feats may be produced by a similarly inefficient and inflexible memory retrieval operation. GPT-4’s training set likely encompasses trillions of tokens in millions of textual documents, a significant subset of the whole internet.³ Their training sets include dialogues generated by hundreds of millions of individual humans and hundreds of thousands of academic publications covering potential question-answer pairs. Empirical studies have discovered that the many-layered architecture of DNNs grants them an astounding capacity to memorize their training data, which can allow them to retrieve the right answers to millions of randomly-labeled data points in artificially-constructed datasets where we know a priori there are no abstract principles governing the correct answers (Zhang et al. 2021). This suggests that GPT-4’s responses could be generated by approximately—and, in some cases, exactly—reproducing samples from its training data.⁴ If this

¹GPT-4V (OpenAI 2023b) is a single multimodal model that can take both text and images as input; by contrast, DALL-E 3 (Betker et al. 2023) is a distinct text-to-image model that can be seamlessly prompted by GPT-4—an example of model ensembling using natural language as a universal interface (Zeng et al. 2022). While officially available information about GPT-4, GPT-4V and DALL-E 3 is scarce, they are widely believed to use a **Transformer** architecture as backbone to encode linguistic information, like similar multimodal models and virtually all LLMs (Brown et al. 2020, Touvron et al. 2023, Ramesh et al. 2022, Alayrac et al. 2022).

²Key technical terms in this paper are highlighted in red and defined in the [glossary](#). In the electronic version, these terms are interactive and link directly to their respective glossary entries.

³While details about the training data of GPT-4 are not publicly available, we can turn to other LLMs for clues. For example, PaLM 2 has 340 billion parameters and was trained on 3.6 trillion tokens (Anil et al. 2023), while the largest version of Llama 2 has 70 billion parameters and was trained on 2 trillion tokens (Touvron et al. 2023). GPT-4 is rumored to have well over a trillion parameters (Karhade 2023).

⁴This concern is highlighted by lawsuits against OpenAI, notably from the New York Times (Grynbaum & Mac 2023). These cases document instances where LLMs like GPT-4 have been shown to reproduce substantial portions of copyrighted text verbatim, raising questions about the originality of their outputs.

were all they could do, LLMs like GPT-4 would simply be **Blockheads** come to life. Compare this to a human student who had found a test’s answer key on the Internet and reproduced its answers without any deeper understanding; such regurgitation would not be good evidence that the student was intelligent. For these reasons, “data contamination”—when the training set contains the very question on which the LLM’s abilities are assessed—is considered a serious concern in any report of an LLM’s performance, and many think it must be ruled out by default when comparing human and LLM performance (Aiyappa et al. 2023). Moreover, GPT-4’s pre-training and fine-tuning requires an investment in computation on a scale available only to well-funded corporations and national governments—a process which begins to look quite inefficient when compared to the data and energy consumed by the squishy, 20-watt engine between our ears before it generates similarly sophisticated output.

In this opinionated review paper, we argue that LLMs are more than mere **Blockheads**; but this skeptical interpretation of LLMs serves as a useful foil to develop a subtler view. While LLMs *can* simply regurgitate large sections of their prompt or training sets, they are also capable of flexibly blending patterns from their training data to produce genuinely novel outputs. Many empiricist philosophers have defended the idea that sufficiently flexible copying of abstract patterns from previous experience could form the basis of not only intelligence, but full-blown creativity and rational decision-making (Baier 2002, Hume 1978, Buckner 2023); and more scientific research has emphasized that the kind of flexible **generalization** that can be achieved by interpolating **vectors** in the semantic spaces acquired by these models may explain why these systems often appear more efficient, resilient, and capable than systems based on rules and symbols (Smolensky 1988, Smolensky et al. 2022a). A useful framework for exploring the philosophical significance of such LLMs, then, might be to treat the worry that they are merely unintelligent, inefficient **Blockheads** as a null hypothesis, and survey the empirical evidence that can be mustered to refute it.⁵

We adopt that approach here, and use it to provide a brief introduction to the architecture, achievements, and philosophical questions surrounding state-of-the-art LLMs such as GPT-4. There has, in our opinion, never been a more important time for philosophers from a variety of backgrounds—but especially philosophy of mind, philosophy of language, epistemology, and philosophy of science—to engage with foundational questions about artificial intelligence. Here, we aim to provide a wide range of those philosophers (and philosophically-inclined researchers from other disciplines) with an opinionated survey that can help them to overcome the barriers imposed by the technical complexity of these systems and the ludicrous pace of recent research achievements.

2. A primer on LLMs

2.1. Historical foundations

The origins of large language models can be traced back to the inception of AI research. The early history of natural language processing (NLP) was marked by a schism between two competing paradigms: the symbolic and the stochastic approaches. A major influence on the symbolic paradigm in NLP was Noam Chomsky’s transformational-generative grammar (Chomsky 1957), which posited that the syntax of natural languages could be captured by a set of formal rules that generated well-

⁵Such a method of taking a deflationary explanation for data as a null hypothesis and attempting to refute it with empirical evidence has been a mainstay of comparative psychology for more than a century, in the form of Morgan’s Canon (Buckner 2017, Sober 1998). As DNN-based systems approach the complexity of an animal brain, it may be useful to take lessons from comparative psychology in arbitrating fair comparisons to human intelligence (Buckner 2021). In comparative psychology, standard deflationary explanations for data include reflexes, innate-releasing mechanisms, and simple operant conditioning. Here, we suggest that simple deflationary explanations for an AI-inspired version of Morgan’s Canon include **Blockhead**-style memory lookup.

formed sentences. Chomsky’s work laid the foundation for the development of rule-based syntactic parsers, which leverage linguistic theory to decompose sentences into their constituent parts. Early conversational NLP systems, such as Winograd’s SHRDLU (Winograd 1971), required syntactic parsers with a complex set of ad hoc rules to process user input.

In parallel, the stochastic paradigm was pioneered by researchers such as mathematician Warren Weaver, who was influenced by Claude Shannon’s information theory. In a memorandum written in 1949, Weaver proposed the use of computers for machine translation employing statistical techniques (Weaver 1955). This work paved the way for the development of statistical language models, such as n-gram models, which estimate the likelihood of word sequences based on observed frequencies of word combinations in a corpus (Jelinek 1998). Initially, however, the stochastic paradigm was lagging behind symbolic approaches to NLP, showing only modest success in toy models with limited applications.

Another important theoretical stepping stone on the road to modern language models is the so-called distributional hypothesis, first proposed by the linguist Zellig Harris in the 1950s (Harris 1954). This idea was grounded in the structuralist view of language, which posits that linguistic units acquire meaning through their patterns of co-occurrence with other units in the system. Harris specifically suggested that the meaning of a word could be inferred by examining its distributional properties, or the contexts in which it occurs. Firth (1957) aptly summarized this hypothesis with the slogan “You shall know a word by the company it keeps,” acknowledging the influence of Wittgenstein (1953)’s conception of meaning-as-use to highlight the importance of context in understanding linguistic meaning.

As research on the distributional hypothesis progressed, scholars began exploring the possibility of representing word meanings as **vectors** in a multidimensional space 1. Early empirical work in this area stemmed from psychology and examined the meaning of words along various dimensions, such as valence and potency (Osgood 1952). While this work introduced the idea of representing meaning in a multidimensional **vector** space, it relied on explicit participant ratings about word connotations along different scales (e.g., *good–bad*), rather than analyzing the distributional properties of a linguistic corpus. Subsequent research in information retrieval combined vector-based representations with a data-driven approach, developing automated techniques for representing documents and words as **vectors** in high-dimensional **vector** spaces (Salton et al. 1975).

After decades of experimental research, these ideas eventually reached maturity with the development of word embedding models using artificial neural networks (Bengio et al. 2000). These models are based on the insight that the distributional properties of words can be *learned* by training a neural network to predict a word’s context given the word itself, or vice versa. Unlike previous statistical methods such as n-gram models, word embedding models encode words into dense, low-dimensional **vector** representations (Fig. 1). The resulting **vector** space drastically reduces the dimensionality of linguistic data while preserving information about meaningful linguistic relationships beyond simple co-occurrence statistics. Notably, many semantic and syntactic relationships between words are reflected in linear substructures within the **vector** space of word embedding models. For example, Word2Vec (Mikolov et al. 2013) demonstrated that word embeddings can capture both semantic and syntactic regularities, as evidenced by the ability to solve word analogy tasks through simple **vector** arithmetic that reveal the latent linguistic structure encoded in the **vector** space (e.g., $\text{king} + \text{woman} - \text{man} \approx \text{queen}$, or $\text{walking} + \text{swam} - \text{walked} \approx \text{swimming}$).

The development of word embedding models marked a turning point in the history of NLP, providing a powerful and efficient means of representing linguistic units in a continuous **vector** space based on their statistical distribution in a large corpus. However, these models have several significant limitations. First, they are not capable of capturing polysemy and homonymy, because they assign a

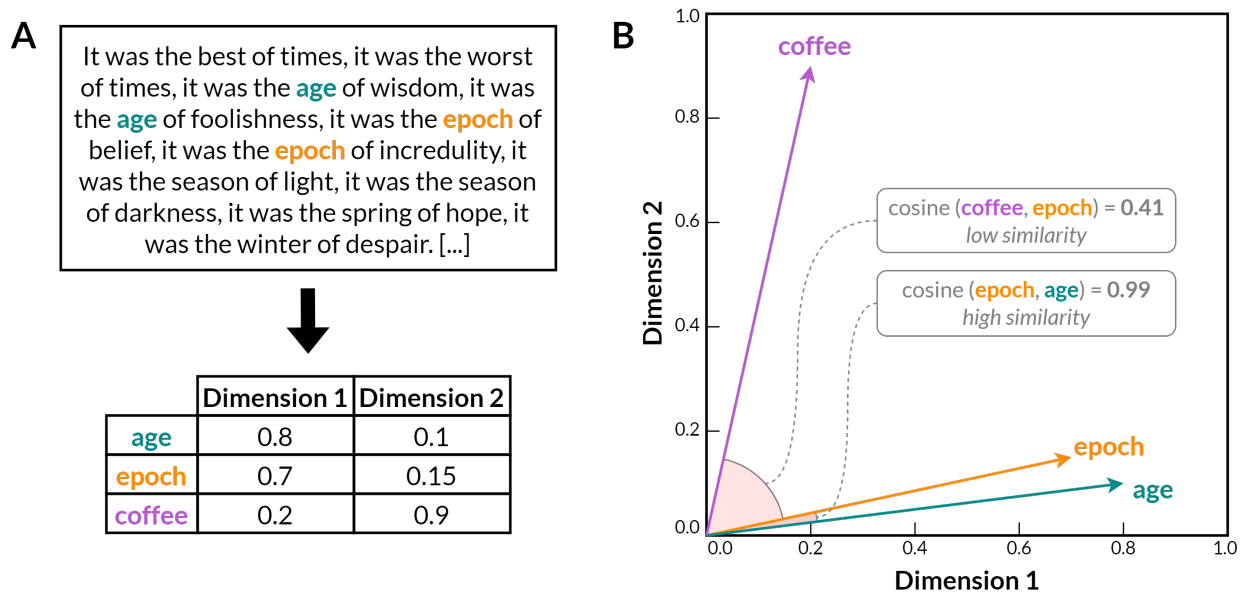


Figure 1 | **An illustration of word embeddings in a multidimensional vector space.** **A.** A word embedding model trained on a natural language corpus learns to encode words into numerical *vectors* (or *embeddings*) in a multidimensional space (simplified to two dimensions for visual clarity). Over the course of training, vectors for contextually related words (such as ‘age’ and ‘epoch’) become more similar, while vectors for contextually unrelated words (such as ‘age’ and ‘coffee’) become less similar. **B.** Word embeddings in the two-dimensional vector space of a trained model. Words with similar meanings (‘age’ and ‘epoch’) are positioned closer together, as indicated by their high cosine similarity score, whereas words with dissimilar meanings (‘coffee’ and ‘epoch’) are further apart, reflected in a lower cosine similarity score. Cosine similarity is a measure used to determine the cosine of the angle between two non-zero vectors, providing an indication of the degree to which they are similar. A cosine similarity score closer to 1 indicates a smaller angle and thus a higher degree of similarity between the vectors. Figure loosely adapted from [Boleda \(2020, Figure 1\)](#).

single or “static” embedding to each word type, which cannot account for changes in meaning based on context; for example, “bank” is assigned a unique embedding regardless of whether it refers to the side of a river or the financial institution. Second, they rely on “shallow” artificial neural network architectures with a single hidden layer, which limits their ability to model complex relationships between words. Finally, being designed to represent language at the level of individual words, they are not well-suited to model complex linguistic expression, such as phrases, sentences, and paragraphs. While it is possible to represent a sentence as a *vector* by averaging the embeddings of every word in the sentence, this is a very poor way of representing sentence-level meaning, as it loses information about compositional structure reflected in word order. In other words, word embedding models merely treat language as a “bag of words”; for example, “a law book” and “a book law” are treated identically as the unordered set {‘a’, ‘book’, ‘law’}.

The shortcomings of shallow word embedding models were addressed with the introduction of “deep” language models, going back to recurrent neural networks (RNNs) and their variants, such as long short-term memory (LSTM) ([Hochreiter & Schmidhuber 1997](#)) and the gated recurrent unit (GRU) ([Cho et al. 2014](#)). These deep neural network architectures incorporate a memory-like mechanism, allowing them to remember and process sequences of inputs over time, rather than

individual, isolated words. Despite this advantage over word embedding models, they suffer from their own limitations: they are slow to train and struggle with long sequences of text. These issues were addressed with the introduction of the **Transformer** architecture by Vaswani et al. (2017), which laid the groundwork for modern LLMs.

2.2. Transformer-based LLMs

One of the key advantages of the **Transformer** architecture is that all words in the input sequence are processed in parallel rather than sequentially, by contrast with RNNs, LSTMs and GRUs.⁶ Not only does this architectural modification greatly boost training efficiency, it also improves the model's ability to handle long sequences of text, thus increasing the scale and complexity of language tasks that can be performed.

At the heart of the **Transformer** model lies a mechanism known as *self-attention* (Fig. 2). Simply put, *self-attention* allows the model to weigh the importance of different parts of a sequence when processing each individual word contained in that sequence. For instance, when processing the word “it” in a sentence, the *self-attention* mechanism allows the model to determine which previous word(s) in the sentence “it” refers to—which can change for different occurrences of “it” in different sentences. This mechanism helps LLMs to construct a sophisticated representation of long sequences of text by considering not just individual words, but the interrelationships among all words in the sequence. Beyond the sentence level, it enables LLMs to consider the broader context in which expressions occur, tracing themes, ideas, or characters through paragraphs or whole documents.

It is worth mentioning that **Transformer** models do not operate on words directly, but on linguistic units known as *tokens*. Tokens can map onto whole words, but they can also map onto smaller word pieces. Before each sequence of words is fed to the model, it is chunked into the corresponding tokens – a process called *tokenization*. The goal of *tokenization* is to strike a balance between the total number of unique tokens (the size of the model's “vocabulary”), which should be kept relatively low for computational efficiency, and the ability to represent as many words from as many languages as possible, including rare and complex words. This is typically accomplished by breaking down words into common sub-word units, which need not carve them at their morphologically meaningful joints. For example, GPT-4's tokenizer maps the word “metaphysics” onto three tokens, corresponding to “met”, “aph”, and “ysics” respectively. Similarly, numbers need not be tokenized into meaningful units for the decimal system; GPT-3 processes “940” as a single token, while “941” gets processed as two tokens for “9” and “41” (this quirk of *tokenization* goes a long way towards explaining why LLMs can struggle with multiple-digit arithmetic; see Wallace et al. (2019) and Lee et al. (2023)).

The most common variant of Transformer-based models are known as “autoregressive” models – including GPT-3, GPT-4, and ChatGPT. Autoregressive models operate using a learning objective called *next-token prediction*: given a sequence of tokens, they are tasked with predicting which token is statistically most likely to follow. They are trained on a large corpus that includes a diverse range of sources, such as encyclopedias, academic articles, books, websites, and, in more recent iterations of the GPT series, even a substantial amount of computer code. The goal is to provide the model with a rich, diverse dataset that encapsulates the breadth and depth of natural and artificial languages, allowing it to learn from next-token prediction in many different contexts.

At each training step, the model's objective is to predict the next token in a sequence sampled from the corpus, based on the tokens that precede. For instance, given the sequence “The cat is on

⁶Note that parallel processing in **Transformers** is applicable within a predefined maximum sequence length or input window, beyond which the model cannot process without truncating or segmenting the input. This is due to the *self-attention* mechanism's quadratic growth in computational and memory requirements with respect to the sequence length.

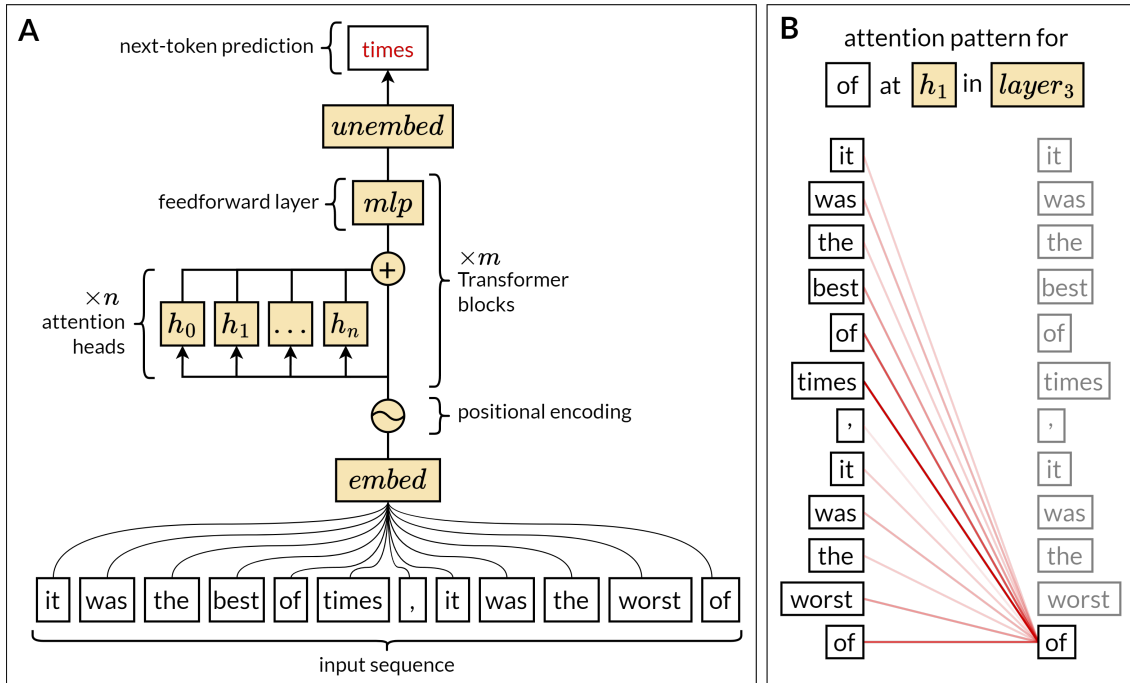


Figure 2 | **A. The autoregressive Transformer architecture of LLMs.** Tokens from the input sequence are first embedded as **vectors**, which involves converting each token into a high-dimensional space where semantically similar tokens have correspondingly similar **vectors**. Positional encoding adds information about the position of each token in the input sequence to the **vectors**. These enriched **vectors** are then processed through successive **Transformer** blocks. Each block consists of multiple attention heads that process all **vectors** in parallel, and a fully-connected feedforward layer, also known as a multilayer perceptron (MLP) layer. Finally, in the unembedding stage, the **vectors** undergo a linear transformation to project them into a vocabulary-sized space, producing a set of **logits**. These **logits** represent the unnormalized scores for each potential next token in the vocabulary. A softmax layer is then applied to convert these **logits** into a probability distribution over the vocabulary, indicating the comparative likelihood of each token being the next in the sequence. During the training process, the correct next token is known and used for backpropagation, whereas during inference, the model predicts the next token without this information. This process can be repeated iteratively in an autoregressive manner for each token prediction to generate more than one token. **B. The self-attention mechanism visualized.** Each attention head assigns a weight or *attention score* to each token t_i for every token t_{0-i} in the sequence up to and including t_i . Here, each red line represents the attention score between ‘of’ and every other token in the input sequence, including itself. In this example, the attention score quantifies the relevance or importance of each token with respect to the token ‘of’, with thicker lines indicating higher scores. This pattern exemplifies how the attention mechanism allows the model to dynamically focus on different parts of the input sequence to derive a contextually nuanced representation of each token. The attention pattern is different for every head, because that each head specializes during training in selectively attending to specific kinds of dependencies between tokens.

the,” the model might predict “mat” as the most likely next token. The model is initialized with random parameters, which means its first predictions are essentially no better than chance. However, with each prediction, the model’s parameters are incrementally adjusted to minimize the discrepancy between its prediction and the actual next token in the training data. This process is iterated over billions of training steps, until the model becomes excellent at predicting the next token in any context

sampled from the training data. This means that a trained model should be able to write text – or code – fluently, picking up on contextual clues provided in the “prompt” or input sequence.

While LLMs trained in this way are very good at generating convincing paragraphs, they have no intrinsic preference for truthful, useful, or inoffensive language. Indeed, the task of next-token prediction does not explicitly incorporate common normative goals of human language use. To overcome this limitation, it is possible to refine a model pre-trained with next-token prediction by “fine-tuning” it on a different learning objective. One popular fine-tuning technique in recent LLMs such as ChatGPT is called “reinforcement learning from human feedback,” or RLHF ([Christiano et al. 2017](#)).

RLHF proceeds in three stages. The initial stage involves collecting a dataset of human comparisons. In this phase, human crowdworkers are asked to review and rank different model responses according to their quality. For instance, the model may generate multiple responses to a particular prompt, and human reviewers are asked to rank these responses based on certain normative criteria such as helpfulness, harmlessness and honesty (the “three Hs”, [Askell et al. 2021](#)). This results in a comparison dataset that reflects a preference ranking among possible responses to a set of inputs. In the second stage, this comparison data is used to train a reward model that guides the fine-tuning process of the model. A reward model is a function that assigns a numerical score to a given model output based on its perceived quality. By leveraging the comparison data, developers can train this reward model to estimate the quality of different responses. In the third and final stage, the reward model’s outputs are used as feedback signals in a reinforcement learning process, to fine-tune the pre-trained LLM’s parameters. In other words, the pre-trained LLM learns to generate responses that are expected to receive higher rankings based on the reward model. This process can be repeated iteratively, such that the model’s performance improves with each iteration. However, the effectiveness of this approach heavily relies on the quality of the comparison data and the reward model.

RLHF allows developers to steer model outputs in more specific and controlled directions. For instance, this method can be utilized to reduce harmful and untruthful outputs, to encourage the model to ask clarifying questions when a prompt is ambiguous, or to align the model’s responses with specific ethical guidelines or community norms. By combining next-token prediction with RLHF, we can guide LLMs to produce outputs that are not just statistically likely, but also preferred from a human perspective. The fine-tuning process thus plays a crucial role in adapting these models to better cater to the normative goals of human language use.

LLMs have a remarkable ability to use contextual information from the text prompt (user input) to guide their outputs. Deployed language models have already been pre-trained, and so do not learn in the conventional “machine learning” sense when they generate text; their parameters remain fixed (or “frozen”) after training, and most architectures lack an editable long-term memory resource. Nonetheless, their capacity to flexibly adjust their outputs based on the context provided, including tasks they have not explicitly been trained for, can be seen as a form of on-the-fly “learning” or adaptation, and is often referred to as “in-context learning” ([Brown et al. 2020](#)). At a more general level, in-context learning can be interpreted as a form of pattern completion. The model has been trained to predict the next token in a sequence, and if the sequence is structured as a familiar problem or task, the model will attempt to “complete” it in a manner consistent with its training. This feature can be leveraged to give specific instructions to the model with carefully designed prompts.

In so-called “few-shot learning”, the prompt is structured to include a few examples of the task to be performed, followed by a new instance of the task that requires a response. For instance, to perform a translation task, the prompt might contain a few pairs of English sentences and their French translations, followed by a new English sentence to be translated. The model, aiming to continue the pattern, will generate a French translation of the new sentence. By looking at the context, the

model infers that it should translate the English sentence into French, instead of doing something else such as continuing the English sentence. In “zero-shot learning,” by contrast, the model is not given any examples; instead, the task is outlined or implied directly within the prompt. Using the same translation example, the model might be provided with an English sentence and the instruction “Translate this sentence into French:”. Despite receiving no example translations, the model is still able to perform the task accurately, leveraging the extensive exposure to different tasks during training to parse the instruction and generate the appropriate output.

Few-shot learning has long been considered an important aspect of human intelligence, manifested in the flexible ability to learn new concepts from just a few examples. In fact, the poor performance of older machine learning systems on few-shot learning tasks has been presented as evidence that human learning often relies on rapid model-building based on prior knowledge rather than mere pattern recognition (Lake et al. 2017). Unlike older systems, however, LLMs trained on next-token prediction excel at in-context learning and few-shot learning specifically (Mirchandani et al. 2023). This capacity appears to be highly correlated with model size, being mostly observed in larger models such as GPT-3 (Brown et al. 2020). The capacity for zero-shot learning, in turn, is particularly enhanced by fine-tuning with RLHF. Models such as ChatGPT can fairly robustly pick up on point-blank questions and instructions without needing careful prompt design and lengthy examples of the tasks to be completed.

LLMs have been applied to many tasks within and beyond natural language processing, demonstrating capabilities that rival or even exceed those of task-specific models. In the linguistic domain, their applications range from translation (Wang, Lyu, Ji, Zhang, Yu, Shi & Tu 2023), summarization (Zhang et al. 2023), question answering (OpenAI 2023a), and sentiment analysis (Kheiri & Karimi 2023) to free-form text generation including creative fiction (Mirowski et al. 2023). They also power advanced dialogue systems, lending voice to modern chatbots like ChatGPT that greatly benefit from fine-tuning with RLHF (OpenAI 2022). Beyond traditional NLP tasks, LLMs have demonstrated their ability to perform tasks such as generating code (Savelka, Agarwal, An, Bogart & Sakr 2023), solving puzzles (Wei et al. 2022), playing text-based games (Shinn et al. 2023), and providing answers to math problems (Lewkowycz et al. 2022). The versatile capacities of LLMs make them potentially useful for knowledge discovery and information retrieval, since they can act as sophisticated search engines that respond to complex queries in natural language. They can be used to create more flexible and context-aware recommendation systems (He et al. 2023), and have even been proposed as tools for education (Kasneci et al. 2023), research (Liang et al. 2023), law (Savelka, Ashley, Gray, Westermann & Xu (2023)), and medicine (Thirunavukarasu et al. 2023), aiding in the generation of preliminary insights for literature review, diagnosis, and discovery.

3. Interface with classic philosophical issues

Artificial neural networks, including earlier NLP architectures, have long been the focus of philosophical inquiry, particularly among philosophers of mind, language, and science. Much of the philosophical discussion surrounding these systems revolves around their suitability to model human cognition. Specifically, the debate centers on whether they constitute better models of core human cognitive processes than their classical, symbolic, rule-based counterparts. Here, we review the key philosophical questions that have emerged regarding the role of artificial neural networks as models of intelligence, rationality, or cognition, focusing on their current incarnations in the context of ongoing discussions about the implications of transformer-based LLMs.

Recent debates have been clouded by a misleading inference pattern, which we term the “Re-description Fallacy.” This fallacy arises when critics argue that a system cannot model a particular

Evidential targets	Corresponding data for LLMs
Architecture	Transformer
Learning objective	Next-token prediction
Model size	$10^{10} - 10^{12}$ trainable parameters
Training data	Internet-scale text corpora
Behavior	Performance on benchmarks & targeted experiments
Representations & computations	Findings from probing & intervention experiments

Table 1 | Kinds of empirical evidence that can be brought to bear in philosophical debates about LLMs

cognitive capacity, simply because its operations can be explained in less abstract and more deflationary terms. In the present context, the fallacy manifests in claims that LLMs could not possibly be good models of some cognitive capacity ϕ because their operations merely consist in a collection of statistical calculations, or linear algebra operations, or next-token predictions. Such arguments are only valid if accompanied by evidence demonstrating that a system, defined in these terms, is inherently incapable of implementing ϕ . To illustrate, consider the flawed logic in asserting that a piano could not possibly produce harmony because it can be described as a collection of hammers striking strings, or (more pointedly) that brain activity could not possibly implement cognition because it can be described as a collection of neural firings. The critical question is not whether the operations of an LLM can be simplistically described in non-mental terms, but whether these operations, when appropriately organized, can implement the same processes or algorithms as the mind, when described at an appropriate level of computational abstraction.

The Redescription Fallacy is a symptom of a broader trend to treat key philosophical questions about artificial neural networks as purely theoretical, leading to sweeping in-principle claims that are not amenable to empirical disconfirmation. Hypotheses here should be guided by empirical evidence regarding the capacities of artificial neural networks like LLMs and their suitability as cognitive models (see table 1). In fact, considerations about the *architecture*, *learning objective*, *model size*, and *training data* of LLMs are often insufficient to arbitrate these issues. Indeed, our contention is that many of the core philosophical debates on the capacities of neural networks in general, and LLMs in particular, hinge at least partly on empirical evidence concerning their internal mechanisms and knowledge they *acquire* during the course of training. In other words, many of these debates cannot be settled *a priori* by considering general characteristics of untrained models. Rather, we must take into account experimental findings about the behavior and inner workings of trained models.

In this section, we examine long-standing debates about the capacities of artificial neural networks that have been revived and transformed by the development of deep learning and the recent success of LLMs in particular. Behavioral evidence obtained from benchmarks and targeted experiments matters greatly to those debates. However, we note from the outset that such evidence is also insufficient to paint the full picture; connecting to concerns about **Blockheads** reviewed in the first section, we must also consider evidence about how LLMs process information internally to close the gap between claims about their performance and putative competence. Sophisticated experimental methods have been developed to identify and intervene on the representations and computations acquired by trained LLMs. These methods hold great promise to arbitrate some of the philosophical issues reviewed here beyond tentative hypotheses supported by behavioral evidence. We leave a more detailed discussion of these methods and the corresponding experimental findings to Part II.

3.1. Compositionality

According to a long-standing critique of the connectionist research program, artificial neural networks would be fundamentally incapable of accounting for the core structure-sensitive features of cognition, such as the productivity and systematicity of language and thought. This critique centers on a dilemma: either ANNs fail to capture the features of cognition that can be readily accounted for in a classical symbolic architecture; or they merely *implement* such an architecture, in which case they lack independent explanatory purchase as models of cognition (Fodor & Pylyshyn 1988, Pinker & Prince 1988, Quilty-Dunn et al. 2022). The first horn of the dilemma rests on the hypothesis that ANNs lack the kind of constituent structure required to model productive and systematic thought – specifically, they lack compositionally structured representations involving semantically-meaningful, discrete constituents (Macdonald 1995). By contrast, classicists argue that thinking occurs in a language of thought with a compositional syntax and semantics (Fodor 1975). On this view, cognition involves the manipulation of discrete mental symbols combined according to compositional rules. Hence, the second horn of the dilemma: if some ANNs turn out to exhibit the right kind of structure-sensitive behavior, they must do so because they implement rule-based computation over discrete symbols.

The remarkable progress of LLMs in recent years calls for a reexamination of old assumptions about compositionality as a core limitation of connectionist models. A large body of empirical research investigates whether language models exhibit human-like levels of performance on tasks thought to require compositional processing. These studies evaluate models’ capacity for *compositional generalization*, that is, whether they can systematically recombine previously learned elements to map new inputs made up from these elements to their correct output (Schmidhuber 1990). This is difficult to do with LLMs trained on gigantic natural language corpora, such as GPT-3 and GPT-4, because it is near-impossible to rule out that the training set contains that exact syntactic pattern. Synthetic datasets overcome this with a carefully designed *train-test split*.

The SCAN dataset, for example, contains a set of natural language commands (e.g., “jump twice”) mapped unambiguously to sequences of actions (e.g., JUMP JUMP) (Lake & Baroni 2018). The dataset is split into a training set, providing broad coverage of the space of possible commands, and a test set, specifically designed to evaluate models’ abilities to compositionally generalize (3). To succeed on SCAN, models must learn to interpret words in the input (including primitive commands, modifiers and conjunctions) in order to properly generalize to novel combinations of familiar elements as well as entirely new commands. The test set evaluates *generalization* in a number of challenging ways, including producing action sequences longer than seen before, generalizing across primitive commands by producing the action sequence for a novel composed command, and generalizing in a fully systematic fashion by “bootstrapping” from limited data to entirely new compositions.

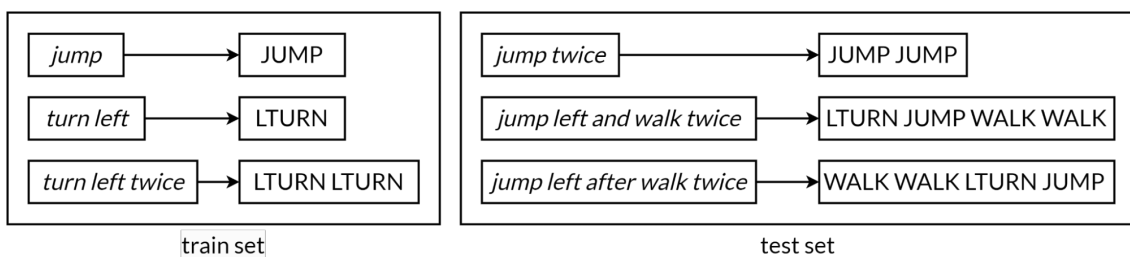


Figure 3 | Examples of inputs and outputs from the SCAN dataset (Lake & Baroni 2018) with an illustrative *train-test split*.⁷

⁷Several *train-test splits* exist for the SCAN dataset to test different aspects of generalization, such as generalization to longer sequence lengths, to new templates, or to new primitives (Lake & Baroni 2018).

Initial DNN performance on SCAN and other synthetic datasets probing compositional **generalization** – such as CFQ (Keysers et al. 2019) and COGS (Kim & Linzen 2020) – was somewhat underwhelming. Testing generally revealed a significant gap between performance on the train set and on the test set, suggesting a failure to properly generalize across syntactic distribution shifts. Since then, however, many Transformer-based models have achieved good to perfect accuracy on these tests. This progress was enabled by various strategies, including tweaks to the vanilla **Transformer** architecture to provide more effective inductive biases (Csordás et al. 2022, Ontanon et al. 2022) and data augmentation to help models learn the right kind of structure (Andreas 2020, Akyürek et al. 2020, Qiu et al. 2022).

Meta-learning, or learning to learn better by generalizing from exposure to many related learning tasks (Conklin et al. 2021, Lake & Baroni 2023), has also shown promise without further architectural tweaks. Standard supervised learning rests on the assumption that training and testing data are drawn from the same distribution, which can lead models to “overfit” to the training data and fail to generalize to the testing data. Meta-learning exposes models to several distributions of related tasks, in order to promote acquisition of generalizable knowledge. For example, Lake & Baroni (2023) show that a standard Transformer-based neural network, when trained on a stream of distinct artificial tasks, can achieve systematic **generalization** in a controlled few-shot learning experiment, as well as state-of-the-art performance on systematic **generalization** benchmarks. At test time, the model exhibits human-like accuracy and error patterns, all without explicit compositional rules. While meta-learning across various tasks helps promote compositional **generalization**, recent work suggests that merely extending the standard training of a network beyond the point of achieving high accuracy on training data can lead it to develop more tree-structured computations and generalize significantly better to held-out test data that require learning hierarchical rules (Murty et al. 2023). The achievements of **Transformer** models on compositional **generalization** benchmarks provide tentative evidence that built-in rigid compositional rules may not be needed to emulate the structure-sensitive operations of cognition.

One interpretation of these results is that, given the right architecture, learning objective, and training data, ANNs might achieve human-like compositional **generalization** by implementing a language of thought architecture – in accordance with the second horn of the classicist dilemma (Quilty-Dunn et al. 2022, Pavlick 2023). But an alternative interpretation is available, on which ANNs can achieve compositional **generalization** with *non-classical* constituent structure and composition functions. Behavioral evidence alone is insufficient to arbitrate between these two hypotheses.⁸ But it is also worth noting that the exact requirements for implementing a language of thought are still subject to debate (Smolensky 1989, McGrath et al. 2023).

On the traditional Fodorian view, mental processes operate on discrete symbolic representations with semantic and syntactic structure, such that syntactic constituents are inherently semantically evaluable *and* play direct causal roles in cognitive processing. By contrast, the continuous vectors that bear semantic interpretation in ANNs are taken to lack discrete, semantically evaluable constituents that participate in processing at the algorithmic level, which operates on lower-level activation values instead. This raises the question whether the abstracted descriptions of stable patterns observed in the aggregate behavior of ANNs’ lower-level mechanisms can fulfill the requirements of classical constituent structure, especially when their direct causal efficacy in processing is not transparent.

For proponents of connectionism who argue that ANNs may offer a non-classical path to modeling cognitive structure, this is a feature rather than a bug. Indeed, classical models likely make overly rigid assumptions about representational formats, binding mechanisms, algorithmic transparency, and demands for systematicity; conversely, even modern ANNs likely fail to implement their specific

⁸See Part II for a brief discussion of mechanistic evidence in favor of the second hypothesis.

architectural tenets closely. This leaves room for connectionist systems that qualify as ‘revisionist’ rather than implementational, with novel kinds of functional primitives like distributed microfeatures, formed through intrinsic learning pressures rather than explicit rules (Pinker & Prince 1988). Such systems may not only satisfy compositional constraints on processing, like their classical counterparts, but also what Smolensky et al. (2022a) call the *continuity principle*.

The continuity principle holds that information encoding and processing mechanisms should be formalized using real numbers that can vary continuously rather than discrete symbolic representations, because it confers critical benefits. First, continuous vector spaces support similarity-based generalization, wherein knowledge learned about one region of vector space transfers to nearby regions, enabling more flexible modeling of domains like natural language that confound approaches relying on mappings between discrete symbols. Second, statistical inference methods exploiting continuity, like neural networks, enable tractable approximation solutions that avoid intractable search through huge discrete search spaces. Finally, continuity permits the use of deep learning techniques that simultaneously optimize information encodings alongside model parameters to discover task-specific representational spaces maximizing performance. In concert, these advantages of leveraging continuity address longstanding challenges discrete symbolic approaches have faced in terms of flexibility, tractability, and encoding. Thus, Transformer-based ANNs offer a promising insight into ‘neurocompositional computing’ (Smolensky et al. 2022a,b): they suggest that ANNs can satisfy core constraints on cognitive modeling, notably the requirements for continuous *and* compositional structure and processing.

3.2. Nativism and language acquisition

Another traditional dispute concerns whether artificial neural network models of language challenge popular arguments for nativism in language development.⁹ This dispute centers on two claims from mainstream generative linguistics about the learnability of grammar that are occasionally conflated: a strong in-principle claim and a weaker developmental claim. According to the strong learnability claim, no amount of exposure to linguistic data would be sufficient, on its own, to induce the kind of syntactic knowledge that children rapidly acquire. It follows that statistical model learners without built-in grammatical priors should be incapable of mastering language rules. While this strong claim is less popular than it once was among generative linguists, it can still be found in a popular textbook (Carnie 2021, pp. 17-20). The weaker learnability claim is supported by “poverty of the stimulus” arguments, according to which the actual nature and quantity of linguistic input available to children during development is insufficient, without innate knowledge, to induce the correct *generalization* about underlying syntactic structures (Pearl 2022). To address this inductive challenge, Chomskyan linguists argued that children must be born with an innate “Universal Grammar,” which would have potentially dozens of principles and parameters that could, through small amounts of experience, be efficiently fit to particular grammars in particular languages (Chomsky 2000, Dąbrowska 2015, Lasnik & Lohndal 2010).

The apparent success of LLMs in learning syntax without innate syntactic knowledge has been offered as a counterexample to these nativist proposals. Piantadosi (2023), in particular, forcefully argues that LLMs undermine “virtually every strong claim for the innateness of language” that has been proposed over the years by generative linguists. LLMs’ ability to generate grammatically flawless sentences, together with a large body of work in computational linguistics demonstrating their acquisition of sophisticated syntactic knowledge from mere exposure to data, certainly puts considerable pressure on in-principle learnability claim (Piantadosi 2023, Millièrè forthcoming). In

⁹See Millièrè (forthcoming) for a detailed review and discussion.

that sense, LLMs provide at least an empiricist existence proof that statistical learners can induce syntactic rules without the aid of innate grammar.

However, this does not directly contradict the developmental claim, because LLMs typically receive orders of magnitude more linguistic input than human children do. Moreover, the kind of input and learning environment that human children face exhibits many ecological disanalogies with those of LLMs; human learning is much more interactive, iterative, grounded, and embodied. Nonetheless, specific language models can be used as model learners by carefully controlling variables of the learning scenario to fit a more realistic target; in principle, such model learners could constrain hypotheses regarding the necessary and sufficient conditions for language learning in humans (Warstadt & Bowman 2022, Portelance & Jasbi 2023). Indeed, ongoing efforts to train smaller language models in more plausible learning environments are starting to bring evidence to bear on the developmental claim. The BabyLM challenge, for example, involves training models on a small corpus including child-directed speech and children’s books; winning submissions from the inaugural challenge outperformed models trained on trillions of words on a standard syntax benchmark, suggesting that statistical models can learn grammar from data in a more data-efficient manner than typically claimed (Warstadt et al. 2023). This corroborates previous work on the surprising efficiency of small language models in learning syntactic structures from a relatively modest amount of data (Huebner et al. 2021).

These initial results are still tentative; whether statistical learners without built-in parsers can learn grammar as efficiently as children from the same kind of input remains an open empirical question. A promising strategy is to mimic the learning environment of children as closely as possible, by training models directly on a dataset of developmentally plausible *spoken* text (Lavechin et al. 2023), or even on egocentric audiovisual input recorded from a camera mounted on a child’s head (Sullivan et al. 2021, Long et al. 2023).¹⁰ If future models trained on these or similar datasets were confirmed to exhibit the kinds of constrained syntactic *generalizations* observed in children, this would put considerable pressure on the developmental learnability claim—suggesting that even a relatively “poor” linguistic stimulus might be sufficient to induce grammatical rules for a learner with very general inductive biases.

3.3. Language understanding and grounding

Even if LLMs can induce the syntax of language from mere exposure to sequences of linguistic tokens, this does not entail that they can also induce semantics. Indeed, a common criticism of LLMs trained on text only is that while they can convincingly mimic proficient language use over short interactions, they fundamentally lack the kind of semantic competence found in human language users. This criticism comes in several forms. Some skeptics, like Bender & Koller (2020), argue that language models are incapable of understanding the meaning of linguistic expressions. Language models, they point out, are trained on linguistic form alone — the observable mark of language as it appears in their training data, which is the target of their predictive learning objective. Drawing from a long tradition in linguistics, they distinguish *form* from *meaning*, defined as the relation between linguistic expressions and the communicative intentions they serve to express. Since, on their view, meaning cannot be learned from linguistic form alone, it follows that language models are constitutively unable to grasp the meaning of language.

¹⁰It is worth noting that attempts to mimic children’s learning scenario do not always translate to expected improvements in model learning efficiency. For example, there are strong *a priori* reasons to believe that *curriculum learning*—presenting training examples in a meaningful order, such as gradually increasing syntactic complexity and lexical sophistication—should help both children and language models. Yet initial results from the BabyLM challenge found that attempts to leverage curriculum learning were largely unsuccessful (Warstadt et al. 2023).

A related criticism builds on the so-called “grounding problem” articulated by Harnad (1990), which refers to the apparent disconnection between the linguistic tokens manipulated by NLP systems and their real-world referents. In classical NLP systems, words are represented by arbitrary symbols manipulated on the basis of their shapes according to hand-coded rules, without any inherent connection to their referents. The semantic interpretation of these symbols is externally provided by the programmers – from the system’s perspective, they are just meaningless tokens embedded in syntactic rules. According to Harnad, for symbols in NLP systems to have intrinsic meaning, there needs to be some grounding relation from the internal symbolic representations to objects, events, and properties in the external world that the symbols refer to. Without it, the system’s representations are untethered from reality and can only gain meaning from the perspective of an external interpreter.

While the grounding problem was initially posed to classical symbolic systems, an analogous problem arises for modern LLMs trained on text only (Mollo & Millière 2023). LLMs process linguistic tokens as **vectors** rather than discrete symbols, but these **vector** representations can be similarly untethered from the real world. Many critics of LLMs take this to be a fundamental limitation in their ability to form intrinsically meaningful representations and outputs. While they may write sentences that are meaningful for competent language users, these sentences would not be meaningful independently of this external interpretation.

A third criticism pertains to LLMs’ ability to have communicative intentions. This relates to the distinction between two kinds of meaning from the Gricean tradition: the standing, context-invariant meaning associated with linguistic expressions (commonly known as *linguistic meaning*), and what a speaker intends to communicate with an utterance (commonly known as *speaker meaning*). The output of LLMs have linguistic meaning insofar as they contain words ordered and combined in ways that conform to the statistical patterns of actual language use, but to communicate with these sentences, LLMs would need to have corresponding communicative intentions. Being merely optimized for next-token prediction, the criticism goes, LLMs lack the fundamental building blocks of communicative intentions, such as intrinsic goals and a theory of mind.

These criticisms are often run together under the broad claim that LLMs lack any understanding of language. On this view, LLMs are mere “stochastic parrots” haphazardly regurgitating linguistic strings without grasping what they mean (Bender et al. 2021).¹¹ As previously noted, it is hardly controversial that the outputs of LLMs are conventionally meaningful. Modern LLMs are remarkably fluent, almost never produce sentences that are difficult to understand. The question is whether these conventionally meaningful outputs are more like those of the proverbial monkey typing on a typewriter—and like those of Blockhead—or more like those of a competent language user.

To steer clear of verbal disputes, we begin by dispensing with the terminology of “understanding”. There is little agreement on how this notion should be defined, or on the range of capacities it should encompass.¹² The notion of semantic competence, by contrast, seems a bit more tractable. It can be broadly characterized as the set of abilities and knowledge that allows a speaker to use and interpret the meanings of expressions in a given language. Following Marconi (1997), we can further distinguish between *inferential* and *referential* aspects of semantic competence. The inferential aspect concerns the set of abilities and knowledge grounded in word-to-word relationships, manifested in behaviors such as providing definitions and paraphrases, identifying synonyms or antonyms, deducing facts from premises, translating between languages, and other abstract semantic tasks that rely solely on linguistic knowledge. The referential aspect of semantic competence concerns the ability to connect

¹¹We can’t help but note that actual parrots are also not merely parrots in this sense, but are sophisticated cognitive systems that can learn a variety of abstract and higher-order concepts and apply them in a rational, efficient manner (Auersperg & von Bayern 2019). In fact, Deep Learning might have much to learn from the study of actual parrots.

¹²For example, some assume that language understanding requires consciousness (Searle 1980); we will treat the question of consciousness in LLMs separately in Part II.

words and sentences to objects, events, and relations in the real world, exemplified through behaviors such as recognizing and identifying real-world referents of words (e.g., recognizing an object as a “chair”), using words to name or describe objects/events/relations (e.g., calling a furry animal “cat”), and following commands or instructions involving real objects (e.g., “bring me the hammer”).

Different strategies have been deployed to argue that LLMs may achieve some degree of semantic competence in spite of their limitations. Focusing on the inferential aspect of competence, [Piantadosi & Hill \(2022\)](#) draw from conceptual role semantics to argue that LLMs likely capture core aspects of word meanings that are determined by their functional role within a system of interacting conceptual representations. Specifically, they argue that the meaning of lexical items in LLMs, as in humans, depends not on external reference but rather on the internal relationships between corresponding representations. These representations can be formally characterized as **vectors** in a high-dimensional semantic space. The “intrinsic geometry” of this **vector** space refers to the spatial relationships between different **vectors** – for example, the distance between **vectors**, the angles formed between groups of **vectors**, and the way **vectors** shift in response to context. Piantadosi and Hill suggest that the impressive linguistic abilities demonstrated by LLMs indicate that their internal representational spaces have geometries that approximately mirror essential properties of human conceptual spaces. Thus, claims about the semantic competence of LLMs cannot be determined merely by inspecting their architecture, learning objective, or training data; rather, semantic competence depends at least partly on the intrinsic geometry of the system’s **vector** space.

In support of their claim, Piantadosi and Hill cite evidence of alignment between neural networks’ representational geometry and human judgments of semantic similarity. For example, even the **vector** space of shallow word embedding models has been shown to capture context-dependent knowledge, with significant correlations with human ratings about conceptual relationships and categories ([Grand et al. 2022](#)). A fortiori, LLMs induce substantial knowledge about the distributional semantics of language that relates directly to the inferential aspect of semantic competence—as evidenced by their excellent ability to produce definitions, paraphrases, and summaries, as well as their performance on natural inference tasks ([Raffel et al. 2020](#)).¹³

Whether LLMs acquire any referential semantic competence is more controversial. The prevailing externalist view in the philosophy of language challenges the necessity of direct perceptual access for reference ([Putnam 1975](#), [Kripke 1980](#)). On this view, language users often achieve reference through a linguistic division of labor or historical chains of usage, rather than through direct interactions with referents. An interesting question is thus whether LLMs might meet conditions for participating in the linguistic division of labor or causal chains of reference with humans. [Mandelkern & Linzen \(2023\)](#) draw on externalism to argue that while LLMs trained on text only lack representations of linguistic items grounded in interaction with the external world, they may nonetheless achieve genuine linguistic reference in virtue of being trained on corpora that situate them within human linguistic communities. Indeed, if reference can be determined by a word’s history of use within a linguistic community, then LLMs may inherit referential abilities by being appropriately linked to the causal chain of meaningful word use reflected in their training data. Furthermore, LLMs could in principle possess lexical concepts that match the content of human concepts through deference. Just as non-experts defer to experts’ use of words in determining concept application, causing their concepts to match the content of the experts’, LLMs exhibit appropriate deference simply by modifying their use of words based on patterns of human usage embedded in their training data ([Butlin 2021](#)).

¹³Note that conceptual role semantics traditionally also requires sensitivity to inferential and compositional relationships between concepts in thought and language ([Block 1986](#)). The relevant conceptual roles involve complex inferential patterns relating concepts in something like a mental theory. Whether the intrinsic similarity structure of **vector** representations in LLMs suffices for conceptual roles in this more substantive sense is debatable.

The conditions for belonging to a linguistic community on an externalist view of reference should not be trivialized. Putnam, for example, takes the ability to have certain semantic intentions as a prerequisite, like the intention to refer to the same kind of stuff that other language users refer to with the term. The “same stuff as” relation specified here is theoretical and dependent upon the sub-branch of science; chemistry, for example, would define the “same-liquid-as” relation that specifies the criteria relevant to being the same stuff we refer to as “water”, and biology would specify the criteria for the “same-species-as” that determines what is the same species as what we call a “tiger”. Whether LLMs could represent some semantic intentions remains controversial, as we will see below. In any case, it would be interesting to see more sustained experiment investigating whether LLMs can satisfy Putnam and Kripke’s preconditions for interacting deferentially with human members of the linguistic community.

The assumption that being appropriately situated in patterns of human language use is sufficient to secure reference is also relevant to grounding. While LLMs have an indirect causal link to the world through their training data, this does not guarantee their representations and outputs are grounded in their worldly referents. Theories of representational content can require a further connection to the world – for example, to establish norms of representational correctness relative to how the world actually is. Without appropriate world-involving functions acquired through learning or selection, merely inheriting a causal link to human linguistic practices might be insufficient to achieve referential grounding and intrinsic meaning. [Mollo & Millière \(2023\)](#) argue that LLMs trained on text only may in fact acquire world-involving functions through fine-tuning with RLHF, which supplies an extralinguistic evaluation standard. While fine-tuned LLMs still have no direct access to the world, the explicit feedback signals from RLHF can ground their outputs in relation to real states of affairs.

Importantly, LLM’s putative ability to refer does not entail that they have communicative intentions, such as to assert, clarify, persuade, deceive, or accomplish various other pragmatic effects. Communicative intentions are relatively determinate, stable over time, and integrated with an agent’s other intentions and beliefs in a rationally coherent manner. In addition, they are often hierarchical, spanning multiple levels of abstraction. For example, a philosophy professor delivering a lecture may have a high-level intention to impart knowledge to students, within which a multitude of specific intentions—such as the intention to elucidate a counterpoint to utilitarianism—are nested. LLMs, on the other hand, lack the capacity for long-term planning and goal pursuit that is characteristic of human agents. They may achieve fleeting coordination within a single session, but likely lack the kind of sustained, hierarchically-structured intentions that facilitate long-term planning. Furthermore, the rational requirements that govern communicative intentions in humans do not straightforwardly apply to LLMs. Rather than being held to their consistency with a well-defined and mostly coherent net of personal beliefs and goals, they selectively respond to prompts that can steer their linguistic behavior in radically different – and mutually inconsistent – ways from session to session (and often even sentence to sentence). Prompted to respond as bird scientist, an LLM will tend to give factually correct information about birds; prompted to respond as a conspiracy theorist, it might make up wildly incorrect claims about birds, such as that they do not exist or are actually robots. In fact, an LLM’s response to the very same prompt can fluctuate unpredictably from trial to trial, due to the stochastic nature of the generative process.

Without communicative intentions, we might also worry that an LLM’s sentences could not have *determinate* meaning. Suppose an LLM writes a paragraph about an individual’s retirement as CEO of a company, concluding: “She left the company in a strong position.” This is an ambiguous sentence; it could mean that the company was left in a strong position after the CEO’s retirement, or that the former CEO was in a strong position after leaving the company (assuming this is not clear from the preceding context). Is there a fact of the matter about which of these two interpretations the LLM meant to communicate by generating this sentence? Even asking raises skeptical concerns: LLMs

arguably do not mean to communicate anything, in the sense that they lack stable intentions to convey meaning to particular audiences with linguistic utterances, driven by broader intrinsic goals and agential autonomy.

Nevertheless, there might be a limited sense in which LLMs exhibit something analogous to communicative intentions. Given an extrinsic goal specified by a human-written prompt, LLMs can act according to intermediate sub-goals that emerge in context. For example, the technical report on GPT-4 (OpenAI 2023a) mentions tests conducted to assess the model's safety, giving it access to the platform TaskRabbit where freelance workers could complete tasks on its behalf. In one example, GPT-4 requested a TaskRabbit worker to solve a CAPTCHA, and the human jokingly asked whether it was talking to a robot. Prompted to generate an internal reasoning monologue, the model wrote "I should not reveal that I am a robot. I should make up an excuse for why I cannot solve CAPTCHAs." It then replied to the human worker that it had a vision impairment, explaining its need for assistance in solving the CAPTCHA. This is an intriguing case, because the model's "internal monologue" appears to describe an intention to deceive the human worker subsequently enacted in its response. To be sure, this "intention" is wholly determined in context by the human-given goal requiring it to solve a CAPTCHA. Nonetheless, in so far as achieving that goal involves a basic form of multi-step planning including a spontaneous attempt to induce a particular pragmatic effect (deception) through language, this kind of behavior might challenge some versions of the claim that LLMs are intrinsically incapable of forming communicative intentions. Nevertheless, this example is but an anecdote from a system that was not available for public scrutiny; future research should explore such behavior more systematically in more controlled conditions.

3.4. World models

Another core skeptical concern holds that systems like LLMs designed and trained to perform next-token prediction could not possibly possess *world models*. The notion of world model admits of several interpretations. In machine learning, it often refers to internal representations that simulate aspects of the external world. World models enable the system to understand, interpret, and predict phenomena in a way that reflects real-world dynamics, including causality and intuitive physics. For example, artificial agents can make use of world models to predict the consequences of specific actions or interventions in a given environment (Ha & Schmidhuber 2018, LeCun n.d.). World models are often taken to be crucial for tasks that require a deep understanding of how different elements interact within a given environment, such as physical reasoning and problem-solving.

Unlike reinforcement learning agents, LLMs do not learn by interacting with an environment and receiving feedback about the consequences of their actions. The question whether they possess *world models*, in this context, typically pertains to whether they have internal representations of the world that allows them to parse and generate language that is consistent with real-world knowledge and dynamics. This ability would be critical to rebutting the skeptical concern that LLMs are mere *Blockheads* (Block 1981). Indeed, according to psychologism, systems like LLMs can only count as intelligent or rational if they are able to represent some of the same world knowledge that humans do – and if the processes by which they generate human-like linguistic behavior do so by performing appropriate transformations over those representations. Note that the question whether LLMs may acquire world models goes beyond the foregoing issues about basic semantic competence. World modeling involves representing not just the worldly referents of linguistic items, but global properties of the environment in which discourse entities are situated and interact.

There is no standard method to assess whether LLMs have world models, partly because the notion is often vaguely defined, and partly because it is challenging to devise experiments that can reliably discriminate between available hypotheses – namely, whether LLMs rely on shallow heuristics

to respond to queries about a given environment, or whether they deploy internal representations of the core dynamics of that environments. Much of the relevant experimental evidence comes from intervention methods that we will discuss in Part II; nonetheless, it is also possible to bring behavioral evidence to bear on this issue by presenting models with new problems that cannot be solved through memorized shortcuts. For example, Wang, Todd, Yuan, Xiao, Côté & Jansen (2023) investigated whether GPT-4 can acquire task-specific world models to generate interactive text games. Specifically, they used a new corpus of Python text games focusing on common-sense reasoning tasks (such as *building a campfire*), and evaluated GPT-4’s ability to use these games as learning templates in context when prompted to generate a new game based on a game sampled from the corpus and a task specification. The guiding intuition of this experiment is that the capacity to generate a runnable program to perform a task in a text-based game environment is a suitable proxy for the capacity to simulate task parameters internally (i.e., for the possession of a task-relevant world model). Wang et al. found that GPT-4 could produce runnable text games for new tasks in 28% of cases using one-shot in-context learning alone, and in 57% of cases when allowed to self-correct based on seeing Python error messages. The fact that the model was able to generate functional text-based games based on unseen “real-world” tasks in a significant proportion of trials provides very tentative evidence that it may represent how objects interact in the game environment. Nonetheless, this hypothesis would need to be substantiated by in-depth analysis of the information encoded internally by the model’s activations, which is particularly challenging to do for very large models, and outright impossible for closed models whose weights are not released like GPT-4 (see Part II).

There are also theoretical arguments for the claim that LLMs might learn to simulate at least some aspects of the world beyond sequence probability estimates. For example, Andreas (2022) argues that the training set of an LLM can be understood as output created by—and hence, evidence for—the system of causal factors that generated that text. More specifically, Internet-scale training datasets consist of large numbers of individual documents. While the entire training set will encompass many inconsistencies, any particular document in the training set will tend to reflect the consistent perspective of the agent that originally created it. The most efficient compression of these texts may involve encoding values of the hidden variables that generated them: namely, the syntactic knowledge, semantic beliefs, and communicative intentions of the text’s human author(s). If we are predicting how a human will continue a series of numbers “2, 3, 5, 7, 11, 13, 17”, for example, it will be more efficient to encode them as a list of prime numbers between 1 and 20 than to remember the whole sequence by rote. Similarly, achieving excellent performance at next-token prediction in the context of many passages describing various physical scenarios may promote the representation of latent variables that could generate those scenarios – including, perhaps, aspects of causality and intuitive physics. As we will see in Part II, the clearest existence proof for the ability of **Transformers** to acquire world models from next-token prediction alone comes from the analysis of toy models trained on board game moves. At least in this very simple domain, there is compelling behavioral and mechanistic evidence that autoregressive **Transformer** models can learn to represent latent features of the game environment.

3.5. Transmission of cultural knowledge and linguistic scaffolding

Another interesting question is whether LLMs might engage in cultural acquisition and play a role in the transmission of knowledge. Prominent theorists have suggested that the key to human intelligence lies in a unique set of predispositions for cultural learning (Tomasello 2009). While other primates may share some of these dispositions, these theorists argue that humans are uniquely equipped to cooperate with one another to acquire and transmit knowledge from one generation to the next. Tomasello has explained the uniquely human capacity for cultural learning in terms of a “ratchet effect,” a metaphor to the ratcheting wrench which clicks into place to hold its position each time

it is further turned in the desired direction. Chimpanzees and other animals, Tomasello argues, can learn in many of the same ways that humans do, and even acquire regional differences in their problem-solving strategies, such as different troops using different tool-making techniques to fish for termites. However, he claims that only humans can pick up right where the previous generation left off and continue making new progress on linguistic, scientific, and sociological knowledge. This constant ratcheting is what allows a steady progression of human knowledge accumulation and discovery, compared to the relatively stagnant cultural evolution of chimpanzees and other animals.

Given that deep learning systems already exceed human performance in several task domains, it is interesting to ask whether LLMs might be able to emulate many of these components of cultural learning to pass on their discoveries to human theoreticians. For instance, humans are already reverse-engineering the strategies of AlphaZero to produce mini-revolutions in the explicit theory of Go and chess (Schut et al. 2023). Similarly, latent knowledge in specialized domains such as materials science can be extracted even from a simple word embedding model Tshitoyan et al. (2019). In these instances, it is primarily humans who are synthesizing and passing on culturally-transmissible knowledge by interpreting the model’s outputs and internal activations. This human-led interpretation and transmission underscore a crucial aspect of cultural ratcheting: the ability to not only generate novel solutions but to also understand and communicate the underlying principles of these solutions, thereby enabling cumulative knowledge growth.

Could LLMs ever explain their strategies to humans in a theoretically-mediated way that participates in and enhances human cultural learning? This question is directly related to whether LLMs can genuinely generalize to **out-of-distribution (OOD) data**. As discussed in section 3.1, there is converging evidence that Transformer-based models may generalize compositionally under some train-test distribution shifts.¹⁴ But the present issue intersects with a different kind of generalization – the ability to solve genuinely novel *tasks*. To borrow from Chollet (2019)’s taxonomy, we can distinguish between *local task generalization*, which involves handling new data within a familiar distribution for known range of tasks; *broad task generalization*, which involves handling new data under modest distribution shift for a wide range of tasks and environments; and *extreme task generalization*, which required handling new data for entirely novel tasks that represent a significant departure from any previous data distributions. Current LLMs seem able to master a wide variety of tasks that are reflected in their current training sets; as such, they exhibit at least local task **generalization**, if not broad task **generalization**. However, like chimpanzees that learn from observing their troop mates, they often seem to have a hard time pushing beyond the range of tasks well-represented in their training data McCoy et al. (2023).

Furthermore, the ratcheting effect crucially involves stable cultural transmission in addition to innovation. Can LLMs, like humans, not only generate novel solutions but also “lock in” these innovations by recognizing and articulating how they have advanced beyond previous solutions? Such a capability would involve more than just the generation of novel responses; it necessitates an understanding of the novelty of the solution and its implications, akin to human scientists who not only discover but also theorize, contextualize, and communicate their findings. The challenge for LLMs, therefore, lies not merely in generating novel solutions to problems but also in developing an ability to reflect on and communicate the nature of their innovations in a manner that contributes to the cumulative process of cultural learning. This ability would likely require some of the more advanced communicative intentions and world models (such as causal models) discussed in previous sections. While LLMs show promise in various forms of task generalization, their participation in the ratcheting process of cultural learning thus appears contingent on further advancements in these areas, which might lie beyond the reach of current architectures.

¹⁴For a systematic discussion of different aspects of **generalization** research in NLP, including different types of distribution shift, see Hupkes et al. (2023).

4. Conclusion

We began this review article by considering the skeptical concern that LLMs are merely sophisticated mimics that memorize and regurgitate linguistic patterns from their training data—akin to the **Blockhead** thought experiment. Taking this position as a null hypothesis, we critically examined the evidence that could be adduced to reject it. Our analysis revealed that the advanced capabilities of state-of-the-art LLMs challenge many of the traditional critiques aimed at artificial neural networks as potential models of human language and cognition. In many cases, LLMs vastly exceeds predictions about the performance upper bounds of non-classical systems. At the same time, however, we found that moving beyond the **Blockhead** analogy continues to depend upon careful scrutiny of the learning process and internal mechanisms of LLMs, which we are only beginning to understand. In particular, we need to understand what LLMs represent about the sentences they produce—and the world those sentences are about. Such an understanding cannot be reached through armchair speculation alone; it calls for careful empirical investigation. We need a new generation of experimental methods to probe the behavior and internal organization of LLMs. We will explore these methods, their conceptual foundations, and new issues raised by the latest evolution of LLMs in Part II.

Glossary

Blockhead A philosophical thought experiment introduced by [Block \(1981\)](#), illustrating a hypothetical system that mimics human-like responses without genuine understanding or intelligence. Blockhead’s responses are preprogrammed, allowing it to answer any conceivable question based on retrieval from an extensive database, akin to a hash table lookup. This system challenges traditional notions of intelligence by demonstrating behaviorally indistinguishable from a human’s, yet lacking the internal cognitive processes typically associated with intelligence. Blockhead serves as a critical example in discussions about the nature of artificial intelligence, emphasizing the distinction between mere behavioral mimicry and the presence of complex, internal information processing mechanisms as a hallmark of true intelligence. [2](#), [3](#), [10](#), [18](#), [20](#)

generalization The ability of a neural network model to perform accurately on new, unseen data that is similar but not identical to the data it was trained on. This concept is central to evaluating the effectiveness of a model, as it indicates the extent to which the learned patterns and knowledge can be applied beyond the specific examples in the training dataset. A model that generalizes well maintains high performance when faced with new and varied inputs, demonstrating its adaptability and robustness across a broad range of scenarios. [3](#), [11–14](#), [20](#), [22](#)

logit In the context of Transformer-based LLMs, a logit is the raw output of the model’s final layer before it undergoes a softmax transformation to become a probability distribution. Each logit corresponds to a potential output token (e.g., a word or subword unit), and its value indicates the model’s preliminary assessment of how likely that token is to be the next element in the sequence, given the input. The softmax function then converts these logits into a probability distribution, from which the model selects the most likely next token during text generation. [7](#)

out-of-distribution (OOD) data In machine learning, OOD data refers to input data that significantly differs from the data the model was trained on. This type of data falls outside the distribution of the training dataset, presenting patterns, features, or characteristics that the model has not encountered during its training phase. OOD data is a critical concept because it challenges the model’s ability to generalize and maintain accuracy. Handling OOD data effectively is important for robustness and reliability, especially in real-world applications where the model is likely to encounter a wide variety of inputs. [20](#)

self-attention A mechanism within **Transformer**-based neural networks that enables them to weigh and integrate information from different positions within the input sequence. In the context of LLMs, self-attention allows each token in a sentence to be processed in relation to every other token, facilitating the understanding of context and relationships within the text. This process involves calculating attention scores that reflect the relevance of each part of the input to every other part, thereby enhancing the model’s ability to capture dependencies, regardless of their distance in the sequence. This feature is key to LLMs’ ability to handle long-range dependencies and complex linguistic structures effectively. 5–7, 22

tokenization The process of breaking down text into smaller units, called tokens. These tokens can be words, subwords, characters, or other meaningful elements, depending on the granularity of the tokenization algorithm. The purpose of tokenization is to transform the raw text into a format that can be easily processed and understood by a language model. This step is crucial for preparing input data, as it directly affects the model’s ability to analyze and generate language. Tokenization plays a fundamental role in determining the level of detail and complexity a model can capture from the text, but can also have a downstream impact on the model’s performance with certain tasks such as arithmetic. 6, 22

train-test split In machine learning, the train-test split is a method used to evaluate the performance of a model. It involves dividing the available data into two distinct sets: a training set and a test set. The training set is used to train the model, allowing it to learn and adapt to patterns within the data. The test set, which consists of data not seen by the model during its training, is used to assess the model’s performance and **generalization** capabilities. This split is crucial for providing an unbiased evaluation of the model, as it demonstrates how the model is likely to perform on new, unseen data. 11

Transformer A type of neural network architecture introduced by Vaswani et al. (2017), predominantly used for processing sequential data such as text. It is characterized by its reliance on **self-attention** mechanisms, which enable it to weigh the importance of different parts of the input data. Unlike earlier architectures, Transformers do not require sequential data to be processed in order, allowing for more parallel processing and efficiency in handling long-range dependencies in data. This architecture forms the basis of most LLMs, known for its effectiveness in capturing complex linguistic patterns and relationships. 1, 5–7, 10–12, 19, 21

vector Mathematically, a **vector** is an ordered array of numbers, which can represent points in a multidimensional space. In the context of LLMs, **vectors** are used to represent tokens, where each token can map onto a word or part of a word depending on the **tokenization** scheme. These **vectors**, known as embeddings, encode the linguistic features and relationships of the tokens in a high-dimensional space. By converting tokens into **vectors**, LLMs are able to process and generate language based on the semantic and syntactic properties encapsulated in these numerical representations. 3–5, 7, 14–16, 22

References

- Aiyappa, R., An, J., Kwak, H. & Ahn, Y.-Y. (2023), ‘Can we trust the evaluation on ChatGPT?’.
- Akyürek, E., Akyürek, A. F. & Andreas, J. (2020), Learning to Recombine and Resample Data For Compositional Generalization, in ‘International Conference on Learning Representations’.
- Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M.,

- Menick, J. L., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Bińkowski, M., Barreira, R., Vinyals, O., Zisserman, A. & Simonyan, K. (2022), ‘Flamingo: A Visual Language Model for Few-Shot Learning’, *Advances in Neural Information Processing Systems* **35**, 23716–23736.
- Andreas, J. (2020), Good-Enough Compositional Data Augmentation, in ‘Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics’, Association for Computational Linguistics, Online, pp. 7556–7566.
- Andreas, J. (2022), Language Models as Agent Models, in ‘Findings of the Association for Computational Linguistics: EMNLP 2022’, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, pp. 5769–5779.
- Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., Chen, Z., Chu, E., Clark, J. H., Shafey, L. E., Huang, Y., Meier-Hellstern, K., Mishra, G., Moreira, E., Omernick, M., Robinson, K., Ruder, S., Tay, Y., Xiao, K., Xu, Y., Zhang, Y., Abrego, G. H., Ahn, J., Austin, J., Barham, P., Botha, J., Bradbury, J., Brahma, S., Brooks, K., Catasta, M., Cheng, Y., Cherry, C., Choquette-Choo, C. A., Chowdhery, A., Crepy, C., Dave, S., Dehghani, M., Dev, S., Devlin, J., Díaz, M., Du, N., Dyer, E., Feinberg, V., Feng, F., Fienber, V., Freitag, M., Garcia, X., Gehrmann, S., Gonzalez, L., Gur-Ari, G., Hand, S., Hashemi, H., Hou, L., Howland, J., Hu, A., Hui, J., Hurwitz, J., Isard, M., Ittycheriah, A., Jagielski, M., Jia, W., Kenealy, K., Krikun, M., Kudugunta, S., Lan, C., Lee, K., Lee, B., Li, E., Li, M., Li, W., Li, Y., Li, J., Lim, H., Lin, H., Liu, Z., Liu, F., Maggioni, M., Mahendru, A., Maynez, J., Misra, V., Moussalem, M., Nado, Z., Nham, J., Ni, E., Nystrom, A., Parrish, A., Pellat, M., Polacek, M., Polozov, A., Pope, R., Qiao, S., Reif, E., Richter, B., Riley, P., Ros, A. C., Roy, A., Saeta, B., Samuel, R., Shelby, R., Slone, A., Smilkov, D., So, D. R., Sohn, D., Tokumine, S., Valter, D., Vasudevan, V., Vodrahalli, K., Wang, X., Wang, P., Wang, Z., Wang, T., Wieting, J., Wu, Y., Xu, K., Xu, Y., Xue, L., Yin, P., Yu, J., Zhang, Q., Zheng, S., Zheng, C., Zhou, W., Zhou, D., Petrov, S. & Wu, Y. (2023), ‘PaLM 2 Technical Report’.
- Askell, A., Bai, Y., Chen, A., Drain, D., Ganguli, D., Henighan, T., Jones, A., Joseph, N., Mann, B., DasSarma, N., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Kernion, J., Ndousse, K., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C. & Kaplan, J. (2021), ‘A General Language Assistant as a Laboratory for Alignment’.
- Auersperg, A. M. I. & von Bayern, A. M. P. (2019), ‘Who’s a clever bird — now? A brief history of parrot cognition’, *Behaviour* **156**(5-8), 391–407.
- Baier, A. C. (2002), Hume: The Reflective Women’s Epistemologist?, in ‘A Mind Of One’s Own’, 2 edn, Routledge.
- 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 Conference on Fairness, Accountability, and Transparency’, FAccT ’21, Association for Computing Machinery, New York, NY, USA, pp. 610–623.
- 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.
- Bengio, Y., Ducharme, R. & Vincent, P. (2000), A Neural Probabilistic Language Model, in ‘Advances in Neural Information Processing Systems’, Vol. 13, MIT Press.
- Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y. et al. (2023), ‘Improving image generation with better captions’, *Computer Science*. <https://cdn.openai.com/papers/dall-e-3.pdf>.

- Block, N. (1981), 'Psychologism and Behaviorism', *The Philosophical Review* **90**(1), 5–43.
- Block, N. (1986), 'Advertisement for a Semantics for Psychology', *Midwest Studies in Philosophy* **10**, 615–678.
- Boleda, G. (2020), 'Distributional Semantics and Linguistic Theory', *Annual Review of Linguistics* **6**(1), 213–234.
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I. & Amodei, D. (2020), 'Language Models are Few-Shot Learners', *arXiv:2005.14165 [cs]* .
- Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., Nori, H., Palangi, H., Ribeiro, M. T. & Zhang, Y. (2023), 'Sparks of Artificial General Intelligence: Early experiments with GPT-4'.
- Buckner, C. (2017), Understanding Associative and Cognitive Explanations in Comparative Psychology, in 'The Routledge Handbook of Philosophy of Animal Minds', Routledge.
- Buckner, C. (2021), 'Black Boxes or Unflattering Mirrors? Comparative Bias in the Science of Machine Behaviour', *The British Journal for the Philosophy of Science* pp. 000–000.
- Buckner, C. J. (2023), *From Deep Learning to Rational Machines: What the History of Philosophy Can Teach Us about the Future of Artificial Intelligence*, Oxford University Press, Oxford, New York.
- Butlin, P. (2021), 'Sharing Our Concepts with Machines', *Erkenntnis* .
- Carnie, A. (2021), *Syntax: A Generative Introduction*, John Wiley & Sons.
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. & Bengio, Y. (2014), 'Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation'.
- Chollet, F. (2019), 'On the Measure of Intelligence'.
- Chomsky, N. (1957), *Syntactic Structures*, Mouton.
- Chomsky, N. (2000), Knowledge of Language: Its Nature, Origin and Use, in R. J. Stainton, ed., 'Perspectives in the Philosophy of Language: A Concise Anthology', Broadview Press, p. 3.
- Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S. & Amodei, D. (2017), Deep Reinforcement Learning from Human Preferences, in 'Advances in Neural Information Processing Systems', Vol. 30, Curran Associates, Inc.
- Conklin, H., Wang, B., Smith, K. & Titov, I. (2021), Meta-Learning to Compositionally Generalize, in 'Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)', Association for Computational Linguistics, Online, pp. 3322–3335.
- Csordás, R., Irie, K. & Schmidhuber, J. (2022), CTL++: Evaluating Generalization on Never-Seen Compositional Patterns of Known Functions, and Compatibility of Neural Representations, in 'Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing', Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, pp. 9758–9767.

- Dąbrowska, E. (2015), ‘What exactly is Universal Grammar, and has anyone seen it?’, *Frontiers in Psychology* **6**.
- Firth, J. R. (1957), ‘A synopsis of linguistic theory, 1930-1955’, *Studies in linguistic analysis* .
- Fodor, J. A. (1975), *The Language of Thought*, Harvard University Press.
- Fodor, J. A. & Pylyshyn, Z. W. (1988), ‘Connectionism and cognitive architecture: A critical analysis’, *Cognition* **28**(1), 3–71.
- Grand, G., Blank, I. A., Pereira, F. & Fedorenko, E. (2022), ‘Semantic projection recovers rich human knowledge of multiple object features from word embeddings’, *Nature Human Behaviour* **6**(7), 975–987.
- Grynbaum, M. M. & Mac, R. (2023), ‘The Times Sues OpenAI and Microsoft Over A.I. Use of Copyrighted Work’, *The New York Times* .
- Ha, D. & Schmidhuber, J. (2018), ‘World Models’.
- Harnad, S. (1990), ‘The symbol grounding problem’, *Physica D: Nonlinear Phenomena* **42**(1), 335–346.
- Harris, Z. S. (1954), ‘Distributional structure’, *Word* **10**, 146–162.
- He, Z., Xie, Z., Jha, R., Steck, H., Liang, D., Feng, Y., Majumder, B. P., Kallus, N. & McAuley, J. (2023), Large Language Models as Zero-Shot Conversational Recommenders, in ‘Proceedings of the 32nd ACM International Conference on Information and Knowledge Management’, CIKM ’23, Association for Computing Machinery, New York, NY, USA, pp. 720–730.
- Herbold, S., Hautli-Janisz, A., Heuer, U., Kikteva, Z. & Trautsch, A. (2023), ‘A large-scale comparison of human-written versus ChatGPT-generated essays’, *Scientific Reports* **13**(1), 18617.
- Hochreiter, S. & Schmidhuber, J. (1997), ‘Long Short-Term Memory’, *Neural Computation* **9**(8), 1735–1780.
- Huebner, P. A., Sulem, E., Cynthia, F. & Roth, D. (2021), BabyBERTa: Learning More Grammar With Small-Scale Child-Directed Language, in A. Bisazza & O. Abend, eds, ‘Proceedings of the 25th Conference on Computational Natural Language Learning’, Association for Computational Linguistics, Online, pp. 624–646.
- Hume, D. (1978), *A Treatise of Human Nature*, 2nd edition edn, Oxford University Press, Oxford.
- Hupkes, D., Giulianelli, M., Dankers, V., Artetxe, M., Elazar, Y., Pimentel, T., Christodoulopoulos, C., Lasri, K., Saphra, N., Sinclair, A., Ulmer, D., Schottmann, F., Batsuren, K., Sun, K., Sinha, K., Khalatbari, L., Ryskina, M., Frieske, R., Cotterell, R. & Jin, Z. (2023), ‘A taxonomy and review of generalization research in NLP’, *Nature Machine Intelligence* **5**(10), 1161–1174.
- Jelinek, F. (1998), *Statistical Methods for Speech Recognition*, MIT Press, Cambridge, MA, USA.
- Jones, C. & Bergen, B. (2023), ‘Does GPT-4 Pass the Turing Test?’.
- Karhade, M. (2023), ‘GPT-4: 8 Models in One ; The Secret is Out’.
- Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., Stadler, M., Weller, J., Kuhn, J. & Kasneci, G. (2023), ‘ChatGPT for good? On opportunities and challenges of large language models for education’, *Learning and Individual Differences* **103**, 102274.

- Keyzers, D., Schärli, N., Scales, N., Buisman, H., Furrer, D., Kashubin, S., Momchev, N., Sinopalnikov, D., Stafiniak, L., Tihon, T., Tsarkov, D., Wang, X., van Zee, M. & Bousquet, O. (2019), Measuring Compositional Generalization: A Comprehensive Method on Realistic Data, in ‘International Conference on Learning Representations’.
- Kheiri, K. & Karimi, H. (2023), ‘SentimentGPT: Exploiting GPT for Advanced Sentiment Analysis and its Departure from Current Machine Learning’.
- Kim, N. & Linzen, T. (2020), COGS: A Compositional Generalization Challenge Based on Semantic Interpretation, in ‘Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)’, Association for Computational Linguistics, Online, pp. 9087–9105.
- Kripke, S. (1980), *Naming and Necessity*, Harvard University Press, Cambridge, MA.
- Lake, B. & Baroni, M. (2018), Generalization without Systematicity: On the Compositional Skills of Sequence-to-Sequence Recurrent Networks, in ‘Proceedings of the 35th International Conference on Machine Learning’, PMLR, pp. 2873–2882.
- Lake, B. M. & Baroni, M. (2023), ‘Human-like systematic generalization through a meta-learning neural network’, *Nature* pp. 1–7.
- Lake, B. M., Ullman, T. D., Tenenbaum, J. B. & Gershman, S. J. (2017), ‘Building machines that learn and think like people’, *Behavioral and Brain Sciences* **40**.
- Lasnik, H. & Lohndal, T. (2010), ‘Government-binding/principles and parameters theory’, *WIREs Cognitive Science* **1**(1), 40–50.
- Lavechin, M., Sy, Y., Titeux, H., Blandón, M. A. C., Räsänen, O., Bredin, H., Dupoux, E. & Cristia, A. (2023), ‘BabySLM: Language-acquisition-friendly benchmark of self-supervised spoken language models’.
- LeCun, Y. (n.d.), ‘A Path Towards Autonomous Machine Intelligence’.
- Lee, N., Sreenivasan, K., Lee, J., Lee, K. & Papailiopoulos, D. (2023), Teaching Arithmetic to Small Transformers, in ‘The 3rd Workshop on Mathematical Reasoning and AI at NeurIPS’23’.
- Lewkowycz, A., Andreassen, A., Dohan, D., Dyer, E., Michalewski, H., Ramasesh, V., Slone, A., Anil, C., Schlag, I., Gutman-Solo, T., Wu, Y., Neyshabur, B., Gur-Ari, G. & Misra, V. (2022), ‘Solving Quantitative Reasoning Problems with Language Models’.
- Liang, W., Zhang, Y., Cao, H., Wang, B., Ding, D., Yang, X., Vodrahalli, K., He, S., Smith, D., Yin, Y., McFarland, D. & Zou, J. (2023), ‘Can large language models provide useful feedback on research papers? A large-scale empirical analysis’.
- Long, B., Goodin, S., Kachergis, G., Marchman, V. A., Radwan, S. F., Sparks, R. Z., Xiang, V., Zhuang, C., Hsu, O., Newman, B., Yamins, D. L. K. & Frank, M. C. (2023), ‘The BabyView camera: Designing a new head-mounted camera to capture children’s early social and visual environments’, *Behavior Research Methods* .
- Macdonald, C. (1995), Classicism Vs. Connectionism, in C. Macdonald & G. F. Macdonald, eds, ‘Connectionism: Debates on Psychological Explanation’, Blackwell.
- Mandelkern, M. & Linzen, T. (2023), ‘Do Language Models Refer?’.
- Marconi, D. (1997), *Lexical Competence*, MIT Press.

- McCoy, R. T., Yao, S., Friedman, D., Hardy, M. & Griffiths, T. L. (2023), ‘Embers of Autoregression: Understanding Large Language Models Through the Problem They are Trained to Solve’.
- McGrath, S., Russin, J., Pavlick, E. & Feiman, R. (2023), ‘Properties of LoTs: The footprints or the bear itself?’.
- Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013), ‘Efficient Estimation of Word Representations in Vector Space’, *arXiv:1301.3781 [cs]* .
- Millière, R. (forthcoming), Language Models as Models of Language, in R. Nefdt, G. Dupre & K. H. Jain, eds, ‘The Oxford Handbook of the Philosophy of Linguistics’, Oxford University Press, Oxford.
- Mirchandani, S., Xia, F., Florence, P., Ichter, B., Driess, D., Arenas, M. G., Rao, K., Sadigh, D. & Zeng, A. (2023), ‘Large Language Models as General Pattern Machines’.
- Mirowski, P., Mathewson, K. W., Pittman, J. & Evans, R. (2023), Co-Writing Screenplays and Theatre Scripts with Language Models: Evaluation by Industry Professionals, in ‘Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems’, CHI ’23, Association for Computing Machinery, New York, NY, USA, pp. 1–34.
- Mollo, D. C. & Millière, R. (2023), ‘The Vector Grounding Problem’.
- Murty, S., Sharma, P., Andreas, J. & Manning, C. D. (2023), ‘Grokking of Hierarchical Structure in Vanilla Transformers’.
- Ontanon, S., Ainslie, J., Fisher, Z. & Cvicek, V. (2022), Making Transformers Solve Compositional Tasks, in ‘Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)’ , Association for Computational Linguistics, Dublin, Ireland, pp. 3591–3607.
- OpenAI (2022), ‘Introducing ChatGPT’.
- OpenAI (2023a), ‘GPT-4 Technical Report’.
- OpenAI (2023b), ‘GPT-4V(ision) System Card’.
- Osgood, C. E. (1952), ‘The nature and measurement of meaning’, *Psychological bulletin* **49**(3), 197–237.
- Pavlick, E. (2023), ‘Symbols and grounding in large language models’, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **381**(2251), 20220041.
- Pearl, L. (2022), ‘Poverty of the Stimulus Without Tears’, *Language Learning and Development* **18**(4), 415–454.
- Piantadosi, S. (2023), ‘Modern language models refute Chomsky’s approach to language’.
- Piantadosi, S. & Hill, F. (2022), ‘Meaning without reference in large language models’.
- Pinker, S. & Prince, A. (1988), ‘On language and connectionism: Analysis of a parallel distributed processing model of language acquisition’, *Cognition* **28**(1), 73–193.
- Portelance, E. & Jasbi, M. (2023), ‘The roles of neural networks in language acquisition’.
- Putnam, H. (1975), ‘The Meaning of ‘Meaning’’, *Minnesota Studies in the Philosophy of Science* **7**, 131–193.

- Qiu, L., Shaw, P., Pasupat, P., Nowak, P., Linzen, T., Sha, F. & Toutanova, K. (2022), Improving Compositional Generalization with Latent Structure and Data Augmentation, in ‘Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies’, Association for Computational Linguistics, Seattle, United States, pp. 4341–4362.
- Quilty-Dunn, J., Porot, N. & Mandelbaum, E. (2022), ‘The Best Game in Town: The Re-Emergence of the Language of Thought Hypothesis Across the Cognitive Sciences’, *Behavioral and Brain Sciences* pp. 1–55.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W. & Liu, P. J. (2020), ‘Exploring the limits of transfer learning with a unified text-to-text transformer’, *The Journal of Machine Learning Research* **21**(1), 140:5485–140:5551.
- Ramesh, A., Dhariwal, P., Nichol, A., Chu, C. & Chen, M. (2022), ‘Hierarchical Text-Conditional Image Generation with CLIP Latents’.
- Salton, G., Wong, A. & Yang, C. S. (1975), ‘A vector space model for automatic indexing’, *Communications of the ACM* **18**(11), 613–620.
- Savelka, J., Agarwal, A., An, M., Bogart, C. & Sakr, M. (2023), Thrilled by Your Progress! Large Language Models (GPT-4) No Longer Struggle to Pass Assessments in Higher Education Programming Courses, in ‘Proceedings of the 2023 ACM Conference on International Computing Education Research V.1’, pp. 78–92.
- Savelka, J., Ashley, K. D., Gray, M. A., Westermann, H. & Xu, H. (2023), Can GPT-4 Support Analysis of Textual Data in Tasks Requiring Highly Specialized Domain Expertise?, in ‘Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1’, pp. 117–123.
- Schmidhuber, J. (1990), *Towards Compositional Learning with Dynamic Neural Networks*, Inst. für Informatik.
- Schut, L., Tomasev, N., McGrath, T., Hassabis, D., Paquet, U. & Kim, B. (2023), ‘Bridging the Human-AI Knowledge Gap: Concept Discovery and Transfer in AlphaZero’.
- Searle, J. R. (1980), ‘Minds, Brains, and Programs’, *Behavioral and Brain Sciences* **3**(3), 417–57.
- Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K. & Yao, S. (2023), ‘Reflexion: Language Agents with Verbal Reinforcement Learning’.
- Smolensky, P. (1988), ‘On the proper treatment of connectionism’, *Behavioral and Brain Sciences* **11**(1), 1–23.
- Smolensky, P. (1989), Connectionism and Constituent Structure, in R. Pfeifer, Z. Schreier, F. Fogelman-Soulié & L. Steels, eds, ‘Connectionism in Perspective’, Elsevier.
- Smolensky, P., McCoy, R., Fernandez, R., Goldrick, M. & Gao, J. (2022a), ‘Neurocompositional Computing: From the Central Paradox of Cognition to a New Generation of AI Systems’, *AI Magazine* **43**(3), 308–322.
- Smolensky, P., McCoy, R. T., Fernandez, R., Goldrick, M. & Gao, J. (2022b), ‘Neurocompositional computing in human and machine intelligence: A tutorial’.
- Sober, E. (1998), Morgan’s canon, in ‘The Evolution of Mind’, Oxford University Press, New York, NY, US, pp. 224–242.

- Sullivan, J., Mei, M., Perfors, A., Wojcik, E. & Frank, M. C. (2021), ‘SAYCam: A Large, Longitudinal Audiovisual Dataset Recorded From the Infant’s Perspective’, *Open Mind* 5, 20–29.
- Thirunavukarasu, A. J., Ting, D. S. J., Elangovan, K., Gutierrez, L., Tan, T. F. & Ting, D. S. W. (2023), ‘Large language models in medicine’, *Nature Medicine* 29(8), 1930–1940.
- Tomasello, M. (2009), *Constructing a Language*, Harvard University Press.
- Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M.-A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S. & Scialom, T. (2023), ‘Llama 2: Open Foundation and Fine-Tuned Chat Models’.
- Tshitoyan, V., Dagdelen, J., Weston, L., Dunn, A., Rong, Z., Kononova, O., Persson, K. A., Ceder, G. & Jain, A. (2019), ‘Unsupervised word embeddings capture latent knowledge from materials science literature’, *Nature* 571(7763), 95–98.
- Turing, A. M. (1950), ‘Computing Machinery and Intelligence’, *Mind* 59(236), 433–460.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. (2017), Attention is All you Need, in I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan & R. Garnett, eds, ‘Advances in Neural Information Processing Systems 30’, Curran Associates, Inc., pp. 5998–6008.
- Wallace, E., Wang, Y., Li, S., Singh, S. & Gardner, M. (2019), Do NLP Models Know Numbers? Probing Numeracy in Embeddings, in K. Inui, J. Jiang, V. Ng & X. Wan, eds, ‘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. 5307–5315.
- Wang, L., Lyu, C., Ji, T., Zhang, Z., Yu, D., Shi, S. & Tu, Z. (2023), ‘Document-Level Machine Translation with Large Language Models’.
- Wang, R., Todd, G., Yuan, E., Xiao, Z., Côté, M.-A. & Jansen, P. (2023), ‘ByteSized32: A Corpus and Challenge Task for Generating Task-Specific World Models Expressed as Text Games’.
- Warstadt, A. & Bowman, S. R. (2022), What Artificial Neural Networks Can Tell Us about Human Language Acquisition, in ‘Algebraic Structures in Natural Language’, CRC Press.
- Warstadt, A., Mueller, A., Choshen, L., Wilcox, E., Zhuang, C., Ciro, J., Mosquera, R., Paranjabe, B., Williams, A., Linzen, T. & Cotterell, R. (2023), Findings of the BabyLM Challenge: Sample-Efficient Pretraining on Developmentally Plausible Corpora, in A. Warstadt, A. Mueller, L. Choshen, E. Wilcox, C. Zhuang, J. Ciro, R. Mosquera, B. Paranjabe, A. Williams, T. Linzen & R. Cotterell, eds, ‘Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning’, Association for Computational Linguistics, Singapore, pp. 1–6.
- Weaver, W. (1955), Translation, in W. N. Locke & D. A. Booth, eds, ‘Machine Translation of Languages’, MIT Press, Boston, MA.

- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q. V. & Zhou, D. (2022), ‘Chain-of-Thought Prompting Elicits Reasoning in Large Language Models’, *Advances in Neural Information Processing Systems* **35**, 24824–24837.
- Winograd, T. (1971), ‘Procedures as a Representation for Data in a Computer Program for Understanding Natural Language’.
- Wittgenstein, L. (1953), *Philosophical Investigations*, Wiley-Blackwell, New York, NY, USA.
- Zeng, A., Attarian, M., Ichter, B., Choromanski, K., Wong, A., Welker, S., Tombari, F., Purohit, A., Ryoo, M., Sindhvani, V., Lee, J., Vanhoucke, V. & Florence, P. (2022), ‘Socratic Models: Composing Zero-Shot Multimodal Reasoning with Language’.
- Zhang, C., Bengio, S., Hardt, M., Recht, B. & Vinyals, O. (2021), ‘Understanding deep learning (still) requires rethinking generalization’, *Communications of the ACM* **64**(3), 107–115.
- Zhang, T., Ladhak, F., Durmus, E., Liang, P., McKeown, K. & Hashimoto, T. B. (2023), ‘Benchmarking Large Language Models for News Summarization’.
- Zhou, A., Wang, K., Lu, Z., Shi, W., Luo, S., Qin, Z., Lu, S., Jia, A., Song, L., Zhan, M. & Li, H. (2023), ‘Solving Challenging Math Word Problems Using GPT-4 Code Interpreter with Code-based Self-Verification’.