Reconstructing the cascade of language processing in the brain using the internal computations of a transformer-based language model

Sreejan Kumar1,*, Theodore R. Sumers2,*, Takateru Yamakoshi3, Ariel Goldstein1,4, Uri Hasson1,4, Kenneth A. Norman1,4, Thomas L. Griffiths2,4, Robert D. Hawkins1,4, Samuel A. Nastase1

1Princeton Neuroscience Institute, Princeton University, Princeton, NJ, 08540, USA
2Department of Computer Science, Princeton University, Princeton, NJ 08540, USA
3Faculty of Medicine, The University of Tokyo, Bunkyo-ku, Tokyo 113-0033, Japan
4Department of Psychology, Princeton University, Princeton, NJ 08540, USA

*Equal contribution. The name listed first in the manuscript was determined by coinflip. Each person reserves the right to put their name first on their CV.

Correspondence: sreejank@princeton.edu, sumers@princeton.edu, snastase@princeton.edu

Abstract
Piecing together the meaning of a narrative requires understanding not only the individual words but also the intricate relationships between them. How does the brain construct this kind of rich, contextual meaning from natural language? Recently, a new class of artificial neural networks—based on the Transformer architecture—has revolutionized the field of language modeling. Transformers integrate information across words via multiple layers of structured circuit computations, forming increasingly contextualized representations of linguistic content. In this paper, we deconstruct these circuit computations and analyze the associated “transformations” (alongside the more commonly studied “embeddings”) at each layer to provide a fine-grained window onto linguistic computations in the human brain. Using functional MRI data acquired while participants listened to naturalistic spoken stories, we find that these transformations capture a hierarchy of linguistic computations across cortex, with transformations at later layers in the model mapping onto higher-level language areas in the brain. We then decompose these transformations into individual, functionally-specialized “attention heads” and demonstrate that the emergent syntactic computations performed by individual heads correlate with predictions of brain activity in specific cortical regions. These heads fall along gradients corresponding to different layers, contextual distances, and syntactic dependencies in a low-dimensional cortical space. Our findings provide a new basis for using the internal structure of large language models to better capture the cascade of cortical computations that support natural language comprehension.

Introduction
Language comprehension is a fundamentally constructive process. We resolve local dependencies among words to assemble lower-level linguistic units into higher-level units of meaning (Chomsky, 1965; MacDonald et al., 1994; Partee, 1995; Goldberg, 2006; Berwick et al., 2013; Christiansen & Chater, 2016), ultimately arriving at the kind of narratives we use to understand the world (Bruner, 1985; Graesser et al., 1994). For example, if a speaker refers to “the secret plan,” we implicitly process the relationships between words in this construction to understand that “secret” modifies “plan.” At a higher level, we use the context of the surrounding narrative to understand the meaning of this phrase—what does the plan entail, who is keeping the secret, and who are they keeping the secret from? This context may comprise hundreds of words...
unfolding over the course of several minutes. The human brain is thought to implement these processes via a series of computations that transform acoustic speech signals into actionable representations of meaning (Hickok & Poeppel, 2007; Hasson et al., 2015; Ding et al., 2016; Friederici et al., 2017; Martin & Doumas, 2017; Pylkkänen, 2019; Martin, 2020).

Traditionally, neuroimaging research has used targeted experimental manipulations to isolate particular linguistic computations—for example, by manipulating the presence/absence or complexity of a given syntactic structure—and mapped these computations onto brain activity in controlled settings (Bookheimer, 2002; Vigneau et al., 2006; Hickok & Poeppel, 2007; Friederici, 2011; Price, 2012). While these findings laid the groundwork for a neurobiology of language, they have limited generalizability outside the laboratory setting, and it has proven difficult to synthesize them into a holistic model that can cope with the full complexity of natural language. This has prompted the field to move toward more naturalistic comprehension paradigms (Hamilton & Huth, 2020; Nastase et al., 2020; Willems et al., 2020). However, these paradigms introduce their own challenges: principally, how to explicitly quantify the linguistic content and computations supporting the richness and expressivity of natural language.

In recent years, the field of natural language processing (NLP) has been revolutionized by a new generation of deep neural networks capitalizing on the Transformer architecture (Vaswani et al., 2017; Devlin et al., 2019; Radford et al., 2019). Transformers are deep neural networks that forgo recurrent connections (Elman, 1990; Hochreiter & Schmidhuber, 1997) in favor of layered “attention head” circuits, facilitating self-supervised training on massive real-world text corpora. At a high level, the Transformer architecture represents the meaning of words as numerical vectors in a high-dimensional “embedding” space where closely related words are located nearer to each other. Unlike the previous generation of word embeddings (Landauer & Dumais, 1997; Mikolov et al., 2013; Pennington et al., 2014), which assign each word a single static (i.e. non-contextual) meaning, Transformers process large blocks of text simultaneously to assign each word a context-sensitive meaning. The core circuit motif of the Transformer—the attention head—incorporates a weighted sum of information exposed by other words, where the relative weighting “attends” more strongly to some words than others. The initial embeddings used as input to the Transformer are non-contextual. Within the Transformer, attention heads in each layer operate in parallel to update the contextual embedding, resulting in surprisingly sophisticated representations of linguistic structure (Manning et al., 2020; Linzen & Baroni, 2021).

The success of Transformers has inspired a growing body of neuroscientific work using them to model human brain activity during natural language comprehension (Jain & Huth, 2018; Toneva & Wehbe, 2019; Antonello et al., 2021; Caucheteux et al., 2021a, 2021b, 2021c; Heilbron et al., 2021; Lyu et al., 2021; Caucheteux & King, 2022; Goldstein et al., 2022). These efforts have focused exclusively on the “embeddings”—the Transformer’s representation of linguistic content—and have largely overlooked the “transformations”—the computations performed by the attention heads. Recent work in NLP, however, has revealed that particular linguistic operations are approximated by particular transformations (Clark et al., 2019; Tenney et al., 2019): for example, attention head 10 in the eighth layer of BERT is specialized for resolving direct object relations, whereas head 11 in the same layer closely tracks noun modifiers (Clark et al., 2019). Although the individual heads that implement these computations operate independently, their transformations are ultimately “fused” together to form the resulting embedding. Thus, unlike the embeddings, the transformations at a given layer can be disassembled into the specialized computations performed by the constituent heads. Can we leverage
the emergent functional specialization of BERT’s internal computations to better understand linguistic computations in the brain?

**Figure 1.** Encoding models for predicting brain activity from language models. (A) Various features are used to predict parcelwise fMRI time series acquired while subjects listened to spoken stories. In addition to classical linguistic features (e.g. parts of speech; black) and static semantic features (e.g. GloVe vectors; gray) extracted from the story transcript, we extracted internal features from a widely-studied Transformer model (BERT-base). The encoding model is estimated from a training subset of each story using banded ridge regression and evaluated on a left-out test segment of each story using three-fold cross-validation. The weights learned from the training set are used to predict the parcelwise time series for the test set; predictions are evaluated by computing the correlation between the predicted and actual parcelwise time series for the test set. (B) We consider two core components of the Transformer architecture at each layer (BERT-base has 12 layers): the embeddings (blue) and the transformations (red). Embeddings represent the contextualized semantic content of the text. Transformations are the output of the self-attention mechanism for each attention head (BERT-base...
has 12 heads per layer, each producing a 64-dimensional vector). Due to the residual connection, transformations represent the incremental information content added to the embedding in that layer. Finally, we also consider the transformation magnitudes (yellow; the L2 norm of each attention head’s 64-dimensional transformation vector), which represent how active each attention head was at that layer. MLP: multilayer perceptron. (C) Attention heads use learned matrices to produce content-sensitive updates to each token. Here, we illustrate the process for a single token (“plan”) passing through a single attention head (layer 7, head 12). First, the input token vectors are multiplied by the head’s learned weight matrices (which are invariant across inputs) tokens to produce query (Q), key (K), and value (V) vectors for each input token. The inner product between the query vector for this token (“plan”, Q) and the key vector (K) for each other token yields a set of “attention weights” that describe how relevant the other tokens are to “plan.” These “attention weights” are used to linearly combine the value vectors (V) from the other tokens. The summed output is the transformation for each head (here, Z_{12}). The results from each attention head in this layer are concatenated and added back to the token’s original representation.

In the current work, we argue that the headwise transformations—the internal computations implemented by the functionally-specialized attention heads—provide a more direct window onto linguistic processing in the brain (Fig. 1A) than the embeddings. Using the widely-studied BERT model (Devlin et al., 2019; Rogers et al., 2020), we extract two internal features of the Transformer architecture (Fig. 1B, C): (1) the layerwise embeddings capturing the contextual linguistic content of each word, with contextual information accumulating across successive layers; and (2) the headwise transformations capturing the incremental information contributed by the computation at each attention head. We use encoding models to evaluate how well different components of the Transformer predict fMRI data acquired during natural language comprehension. We first compare these Transformer features, as well as a transformation magnitude metric summarizing the overall contribution of each head, against two other classes of language features (classical linguistic features and static semantic embeddings), finding that both embeddings and transformations outperform other models. We then examine the pattern of performance across different layers, revealing that the transformations better correspond to the cortical hierarchy for linguistic/narrative construction. Finally, we decompose these transformations into individual attention heads—functionally specialized for particular linguistic operations—and find that certain properties of these heads fall along gradients in a low-dimensional cortical space, yielding new insights into the corresponding cascade of language processing in the brain.

**Results**

We adopted a model-based encoding framework (Naselaris et al., 2011; Yamins & DiCarlo, 2016; Richards et al., 2019) in order to map Transformer features onto brain activity measured using fMRI while subjects listened to naturalistic spoken stories (Fig. 1A). The brain data were downsampled according to a fine-grained functional atlas comprising 1,000 cortical parcels (Schaefer et al., 2018), which were grouped into a variety of regions of interest (ROIs) spanning early auditory cortex to high-level language areas (Fedorenko et al., 2010). Parcelwise encoding models were estimated using banded ridge regression with three-fold cross-validation for each subject and each story (Dupré la Tour et al., 2022). Phonemes, phoneme rate, word rate, and a silence indicator were included as confound variables during model estimation and discarded during model evaluation (Huth et al., 2016). Encoding models were evaluated by computing the correlation between the predicted and actual time series for test partitions; correlations were then converted to the proportion of a noise ceiling estimated via intersubject correlation (Nastase et al., 2019).

**Functional anatomy of a Transformer model**

We take BERT-base (Devlin et al., 2019) as a representative Transformer model due to its success across a broad suite of NLP tasks and the burgeoning literature studying what it learns (Hewitt & Manning, 2019; Liu et
Words are first assigned non-contextual static embeddings, which are submitted as input to the model. Unlike recurrent architectures (Elman, 1990; Hochreiter & Schmidhuber, 1997), which process words serially, BERT considers long “context windows” of up to 512 tokens and processes all words in the context window in parallel. Like most Transformer architectures, BERT is composed of a repeated self-attention motif: 12 sequential layers, each consisting of 12 parallel attention heads. A single attention head consists of three separate query, key, and value matrices. At each layer, the input word embeddings are multiplied by these matrices to produce a query, key, and value vector for each word. For an individual word, self-attention then computes the dot product of that word’s query vector with the key vector from all words in the context window, resulting in a vector of attention weights quantifying the relevance of other words. These attention weights are then used to produce a weighted sum of all value vectors: the 64-dimensional “transformation” vector produced by that attention head. These vectors are then concatenated across all heads within a layer and passed through a feed-forward module (a multilayer perceptron or MLP) to produce a fused 768-dimensional output embedding, which serves as the input for the next layer (Fig. 1B). Embeddings are sometimes referred to as the “residual stream”: the transformations at one layer are added to the embedding from the previous layer, so the embeddings accumulate previous computations that subsequent computations may access (Elhage et al., 2021).

The self-attention mechanism can be thought of as a “soft,” or weighted, key-value store. Each word in the input issues a query which is checked against the keys of all context words. However, unlike a traditional key-value store (which would only return a single, exact query-key match), the attention head first computes how well the query matches with all keys (the attention weights), and returns a weighted sum of each word’s value based on how closely they match. The query, key, and value matrices are learned, and recent work has revealed an emergent functional specialization where specific attention heads approximate certain syntactic relationships (Clark et al., 2019; Tenney et al., 2019). For example, prior work has discovered that a specific head in BERT reliably represents the direct object relationship between tokens (Clark et al., 2019). In a phrase such as “hatch the secret plan,” the “plan” token would attend heavily to the “hatch” token and update its representation accordingly. More precisely, \( Q_{\text{plan}} \times K_{\text{hatch}} \) will yield a large attention weight from “plan” to “hatch.” As a result, the transformation for “plan,” \( Z_{\text{plan}} \), will be heavily weighted towards \( V_{\text{hatch}} \) (Eq. 1). This allows the attention head to update the “plan” token’s representation to reflect the fact that it is being “hatched” (as opposed to being executed, revised, or abandoned; or being used in another sense entirely, e.g. an architectural blueprint or “floor plan”).

**Transformers outperform other linguistic features**

We first compared the encoding performance of features from three families of language models: (1) traditional linguistic features comprising parts of speech and syntactic dependencies; (2) GloVe word embeddings (Pennington et al., 2014) that capture the “static” or non-contextual meanings of words; and (3) contextualized Transformer features extracted from BERT—namely, layer-wise embeddings, transformations, and transformation magnitudes. To summarize the overall performance of these different Transformer features, we concatenated features across all heads and layers, allowing the regularized encoding model to select the

---

1 BERT was trained using context windows from both before and after a given word (i.e. “past” and “future”). By contrast, GPT (24) has a more biologically-plausible unidirectional training objective (e.g. predicting the next word using past words). However, this distinction is only relevant for the pre-training regime: when we extract embeddings and transformations from BERT in the current work, we only allow previous tokens to contextualize the current TR, we do not provide BERT with future tokens.
best-performing combination of those features across the entire network. We use these features to predict response time series in cortical parcels comprising ten language ROIs ranging from early auditory cortex to high-level, left-hemisphere language areas (Fig. 2; see Fig. S1 for right-hemisphere results). Based on prior work (Schrimpf et al., 2021; Goldstein et al., 2022), we expected the BERT embeddings to outperform the GloVe and linguistic features. We further hypothesized that the transformations would perform on par with the embeddings. Finally, the transformation magnitudes, intended to capture the relative contribution of each head, abstracted away from the semantic content, would more closely match the performance of linguistic features.

Figure 2. Comparing three classes of language models across cortical language areas. We used encoding models to evaluate the performance of three different classes of language models: traditional linguistic features, non-contextual word embeddings (GloVe), and contextual Transformer features (BERT). Among the Transformer features, embeddings capture the contextual semantic content of words, transformations capture the contextual transformations that yield these embeddings, and transformation magnitudes capture the non-semantic contribution of each head to a given token. For this analysis, Transformer features were concatenated across all heads and layers. Only left-hemisphere language ROIs are included here; right-hemisphere language ROIs yielded qualitatively similar results (Fig. S1). Model performance is evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation (see “Noise ceiling estimation” in “Materials and methods” for further details). Markers indicate median performance and error bars indicate 95% bootstrap confidence intervals.

First, we confirmed that Transformer embeddings and transformations outperform traditional linguistic features in most language ROIs ($p < .005$ in HG, PostTemp, AntTemp, AngG, IFG, IFGorb, vmPFC, dmPFC, and PMC for both embeddings and transformations; permutation test; FDR corrected; Table S1). Contextual Transformer embeddings also outperform non-contextual GloVe embeddings across several ROIs in keeping with prior work (Schrimpf et al., 2021; Goldstein et al., 2022). Interestingly, transformation magnitudes
outperform GloVe embeddings and traditional linguistic features in lateral temporal areas but not in higher-level language areas. Note, for example, that transformation magnitudes outperform GloVe embeddings in posterior and anterior temporal areas, but this pattern is reversed in the posterior medial cortex (PMC). Finally, we found that the transformations roughly match the embeddings across all ROIs. Note that despite yielding similar encoding performance, the embeddings and transformations are fundamentally different; for example, the average TR-by-TR correlation between embeddings and transformations across both stimuli is effectively zero (-.004 ± .009 SD), the embeddings and transformations yield visibly different TR-by-TR representational geometries (Fig. S2), and the transformations have considerably higher temporal autocorrelation than the embeddings (Fig. S3). As a control analysis, we evaluated these features in a non-language ROI (early visual cortex) and found that no models captured a significant amount of variance (Fig. S4).

Figure 3. Layerwise performance of embeddings and transformations. (A) Layerwise model performance for embeddings (blue) and transformations (red) in ten left-hemisphere language ROIs (see Fig. S8 for right-hemisphere language areas). Model performance for embeddings tends to increase monotonically across layers, whereas performance for embeddings tends to peak for intermediate layers. Markers indicate median performance and error bars indicate 95% bootstrap.
confidence intervals. (B) Model performance for each layer of embeddings and (C) transformations across all cortical parcels. Cortical maps are thresholded to display only parcels with statistically significant model performance (nonparametric bootstrap hypothesis test; FDR controlled at $p < .05$). Model performance is evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation.

Layerwise performance of embeddings and transformations

We next segregated the Transformer features into separate layers. Layerwise embeddings are increasingly contextualized, with word representations in later layers reflecting more complex linguistic relationships (Tenney et al., 2019). Transformations, on the other hand, have a very different layerwise structure. While the embeddings generally accumulate information across layers, the transformations are largely independent from layer to layer (Fig. S5) and produce more layer-specific representational geometries (Figs. S6, S7). We hypothesized that the layer-specificity of transformations may provide finer-grained localization of linguistic computations in the brain. We found that, across language ROIs, the performance of contextual embeddings increased roughly monotonically across layers, peaking in late-intermediate or final layers (Fig. 3A), replicating prior work (Toneva & Wehbe, 2019; Caucheteux et al., 2021a, 2021c). Interestingly, this pattern was observed across most ROIs, suggesting that the hierarchy of layerwise embeddings does not cleanly map onto the cortical hierarchy for language comprehension. Transformations, on the other hand, yield more layer-specific fluctuations in performance than embeddings, suggesting that the computations implemented at particular layers map onto the brain in a more specific way than do the layerwise embeddings. Performance for transformations tended to peak at intermediate layers with reduced performance for the final layers. Beyond the language areas, we see a similar pattern across cortical parcels: embedding performance tends to peak in the final layers, while transformation performance peaks at intermediate layers (Fig. 3B, C).

Figure 4. Layer preferences for embeddings and transformations. (A) Layer preferences are visualized on the cortical surface for embeddings (upper) and transformations (lower). While most cortical parcels prefer the final embedding layers, the transformations reveal a cortical hierarchy of increasing layer preference. Only cortical parcels with encoding performance greater than 20% of the noise ceiling for both embeddings and transformations are included for visualization purposes. The same color map for preferred layer is used for both embeddings and transformations. (B) Histogram of the preferred layer across cortical parcels in language ROIs. Across language parcels, performance for transformations (red) peaks at intermediate layers, while performance for embeddings (blue) peaks in later layers. (C) Distribution of the preferred layer across cortical parcels in language ROIs.
magnitude of layer-to-layer differences in encoding performance for embeddings and transformations; this metric of layer specificity is quantified as the L2 norm of the first differences between encoding performance for neighboring layers. Transformations (red) yield more layer-specific deviations in performance than embeddings (blue).

We next visualized layer preference across cortex—that is, which layer yielded the peak performance for a given cortical parcel—revealing a cortical hierarchy of layer preferences proceeding from low- to high-level language areas (Fig. 4A). Across language parcels, the average performance across participants for embeddings peaked significantly later than performance for transformations (mean preferred embedding layer = 8.9, mean preferred transformation layer = 7.2; \(p < .001\), permutation test; Fig. 4B). To examine the relative specificity of these preferences, we examined the entropy of the softmax distribution across layer performances for language ROIs (see Fig. S9 for details), where higher entropy corresponds to a relatively uniform distribution while lower entropy corresponds to a relatively concentrated distribution. We found that performance scores for transformations were significantly more layer-specific than embeddings overall (\(p < .05\) for all language ROIs; Table S2) where performance scores for embeddings were relatively diffuse across final layers (Fig. S9). Finally, we quantified the magnitude of difference in predictive performance from layer to layer and found that transformations have larger differences in performance between neighboring layers (mean layerwise embedding difference = 7.6, mean layerwise transformation difference = 14.3; \(p < .001\), permutation test; Fig. 4C). This suggests that the computations implemented by the transformations are considerably more layer-specific than the semantic content of the embeddings.

Interpreting transformations via headwise analysis

Despite a large body of work using experimental manipulations of phrases and sentences to dissociate syntax from semantics and localize syntactic operations in the brain (Dapretto & Bookheimer, 1999; Embick et al., 2000; Kuperberg et al., 2000; Ni et al., 2000; Friederici et al., 2003; Glaser et al., 2013; Schell et al., 2017), results have been mixed, leading some authors to suggest that syntactic computations may be fundamentally entangled with semantic representation and distributed throughout the language network (Fedorenko et al., 2012, 2020; I. Blank et al., 2016; Reddy & Wehbe, 2020; Caucheteux et al., 2021c). Along these lines, we found that classical linguistic features—indicator variables identifying parts of speech and syntactic dependencies—are poor predictors of brain activity and did not generally map onto localized brain regions in the context of naturalistic narratives (Figs. 2, S10). Unlike the hand-crafted syntactic features of classical linguistics, BERT’s training regime yields an emergent headwise functional specialization for particular linguistic operations (Clark et al., 2019; Tenney et al., 2019). BERT is not explicitly instructed to represent syntactic dependencies, but nonetheless learns functionally comparable operations from the structure of real-world language. We next split the transformations at each layer into their functionally specialized components—the constituent transformations implemented by each attention head. In the following analyses, we leverage these functionally-specialized headwise transformations to map between syntactic operations and the brain in a way that retains some level of interpretability, but respects the contextual nature of real-world language. Note that the embeddings incorporate information from all the transformations at a given layer (and prior layers), and therefore cannot be meaningfully disassembled in this way. We trained an encoding model on all transformations and then evaluated the performance for each head individually, yielding an estimate of how well each head predicts each cortical parcel (or headwise brain prediction score). For each attention head, we also trained a set of decoding models to determine how much information that head contains about a given syntactic dependency (or headwise dependency prediction score; Fig. S11).
For each classical syntactic dependency, we first identified the attention head that best predicts that dependency (for example, head 11 of layer 6 best predicts the direct object dependency, Fig. S10). We compared the encoding performance for each classical dependency with the encoding performance for the headwise transformation that best predicts that dependency. We found that the head most associated with a given dependency generally outperformed the dependency itself (Fig. S10; see Fig. S12 replication using functionally specialized heads derived from larger text corpora) (Clark et al., 2019). This confirmed our expectation that the dense, emergent headwise transformations are better predictors of brain activity than the sparse, classical linguistic indicator variables. Note that the headwise transformations are considerably higher-dimensional (64 dimensions) than the corresponding one-dimensional dependency indicators. However, we found that even after reducing a given transformation to a single dimension that best predicts the corresponding dependency, the one-dimensional transformation still better predicts brain activity than the dependency itself (Fig. S13). Although the computations performed by these heads approximate particular syntactic operations, they capture a more holistic relationship between words in the context of the narrative. That is, the transformations do not simply indicate the presence, for example, of a direct object relationship; rather, they capture an approximation of the direct object relationship in the context of the ongoing narrative.

Next, we sought to summarize the contributions of all headwise transformations across the entire language network (Fig. 5A). We first obtained the trained encoding model for all transformations (Fig. 1, red) and averaged the regression coefficients (i.e. weight matrices) assigned to the transformation features across subjects and stimuli. To summarize the importance of each head for a given parcel, we segmented the learned weight matrix from the encoding model for that parcel into the individual attention heads at each layer and computed the L2 norm of the headwise encoding weights. This results in a single value for each of the 144 heads reflecting the magnitude of each head’s contribution to encoding performance at each parcel. In order to summarize the contribution of headwise transformations across the language network, we aggregated the headwise encoding weight vectors across parcels in language ROIs and used principal component analysis (PCA) to reduce the dimensionality of this transformation weight matrix across parcels, following Huth and colleagues (2016). This yields a basis set of orthogonal (uncorrelated), 144-dimensional weight vectors capturing the most variance in the headwise transformation weights across all language parcels; each head corresponds to a location in this low-dimensional brain space. The first two principal components (PCs) accounted for 90% of the variance across parcels, while the first nine PCs accounted for 95% of the variance. A given PC can be projected into (i.e. reconstructed in) the original space of cortical parcels, yielding a brain map where positive and negative values indicate positive and negative transformation weights along that PC (Fig. S14). Visualizing these PCs directly on the brain reveals that PC1 ranges from strongly positive values (red) in bilateral posterior temporal parcels and left lateral prefrontal cortex to widespread negative values (blue) in medial prefrontal cortex (Fig. 5B). PC2 ranged from positive values (red) in prefrontal cortex and left anterior temporal areas to negative values (blue) in partially right-lateralized temporal areas (Fig. 5C). Note that the polarity of these PCs is consistent across all analyses, but is otherwise arbitrary.
Figure 5. Headwise transformations in a low-dimensional brain space. (A) We applied PCA to the weight vectors for transformation encoding models across language parcels, effectively projecting the transformation weights into a

[Diagram showing A: Summarizing headwise transformations across the language network, B: PC1 parcelwise weights, C: PC2 parcelwise weights, D: Layer, E: Backward attention, F: Specialized heads, and G: Syntactic dependency prediction]
low-dimensional brain space (Huth et al., 2016). We first obtain the parcelwise weight vectors for the encoding model trained to predict brain activity from BERT transformations. This transformation weight matrix is shaped (768 features \times 12 layers) 9,216 features \times 192 language parcels. We use the L2 norm to summarize the weights within each head, reducing this matrix to (12 heads \times 12 layers) 144 heads \times 192 language parcels. We next summarize these headwise weights across language parcels using PCA. At right, we visualize the headwise transformation weights projected onto the fi rst two PCs. Each data point corresponds to one of 144 heads. Furthermore, each PC can be projected back onto the language network. (B, C) PC1 and PC2 projected back onto the language parcels; red indicates positive weights and blue indicates negative weights along the corresponding PC (see Fig. S14 for higher-order PC maps). (D) Heads colored according to their layer in BERT in the reduced-dimension space of PC1 and PC2. (E) Heads colored according to their average backward attention distance in the story stimuli (look-back token distance is colored according to a log scale). (F) Heads highlighted in red have been reported as functionally specialized by Clark and colleagues (Clark et al., 2019). (G) Heads colored according to their headwise dependency prediction scores for the nominal subject (nsubj), direct object (dobj), and clausal complement (ccomp) syntactic dependencies. Dependency prediction scores reflect the classification accuracy of a cross-validated logistic regression model trained to predict the occurrence of a given linguistic dependency at each TR from the 64-dimensional transformation vector for a given attention head.

We next examined whether there is any meaningful structure in the “geometry” of headwise transformations in this reduced-dimension cortical space. To do this, we visualized several structural and functional properties of the heads in the two-dimensional projection of the language network. We first visualized the layer of each head in this low-dimensional brain space and found a layer gradient across heads in PC1 and PC2 (Fig. 5D). PCs 9, 5, and 1 were the PCs most correlated with layer assignment with $r = .45$, .40, and .26, respectively (Figs. S14, S15). Intermediate layers were generally located in the negative quadrant of both PC1 and PC2 (corresponding to blue parcels in Fig. 5B, C; e.g. posterolateral temporal cortex), with early and late layers located in the positive quadrant (red parcels). For each head, we next computed the average backward attention distance across stimuli. PCs 2, 1, and 3 were the PCs most correlated with backward attention distance with $r = .65$, .20, .19, respectively (Figs. S14, S16). We observed a strong gradient of look-back distance increasing along PC2 (Fig. 5E); that is, prefrontal and left anterior temporal parcels (red parcels in Fig. 5C) correspond to heads with longer look-back distances. Note that the upper quartile of headwise attention distances exceeds 30 tokens, corresponding to look-back distances on the scale of multiple sentences. We also found that the functionally specialized heads previously reported in the literature (Clark et al., 2019) span PC1 and cluster at the negative end of PC2 (corresponding to intermediate layers and relatively recent look-back distance; Fig. 5F). Finally, we visualized the headwise dependency prediction scores in this low-dimensional brain space and observed gradients in different directions along PC1 and PC2 for different dependencies (Figs. 5G, S17). Note that none of the aforementioned structural or functional properties of the heads that we visualize in this low-dimensional brain space are derived from the brain data; that is, the encoding models do not “know” the layer or backward attention distance of any given head.

To quantify the correspondence between the syntactic information contained in a given head and that head’s prediction performance in the brain, we computed the correlation, across attention heads, between the brain prediction and dependency prediction scores (Fig. 6A, B). We repeated this analysis for each syntactic dependency and the parcels comprising each language ROI (Fig. S18). This headwise correspondence between dependencies and ROIs indicates that attention heads containing information about a given dependency also tend to contain information about brain activity for a given ROI—which suggests that the ROI is involved in computing that dependency. We found that the correspondence between brain prediction and dependency prediction scores varied considerably across ROIs. For example, in posterior superior temporal cortex, we observed headwise prediction correspondence for clausal complement (ccomp), direct
object (dobj), and preposition object (pobj) relations. In IFG, on the other hand, headwise prediction correspondence was observed only for the clausal complement relation. Headwise prediction correspondence was high in the angular gyrus and MFG across dependencies: for example, attention heads that predict the existence of a nominal subject relationship (nsubj) also tend to predict the MFG, but not the dmPFC; heads that predict direct object (dobj) tend to predict the angular gyrus, but this relationship is weaker in the vmPFC. In the case of MFG, this is consistent with prior work implicating MFG in both language comprehension and more general cognitive demand (e.g. working memory) (Fedorenko et al., 2011; Minero et al., 2018). Collapsing across dependencies highlights the discrepancy between ROIs (Fig. 6C). The angular gyrus and MFG display a relatively high correspondence; in contrast, the vmPFC and dmPFC display virtually no correspondence.

While transformations explain significant variance in these ROIs at the scale of the full model (Fig. 2), individual layers (Fig. 3A), and individual heads (Fig. 6A), their prediction performance for the brain does not correlate with their prediction performance for classic syntactic dependencies—suggesting that the shared information between transformations and certain ROIs may be semantic in nature or reflect contextual relationships beyond the scope of classical syntax.

**Figure 6.** Correspondence between headwise brain and dependency predictions. (A) Correlation between headwise brain prediction and dependency prediction scores for example ROIs and dependencies; nominal subject (nsubj) in MFG and dmPFC, direct object (dobj) in AngG and vmPFC (see Fig. S18 for correlations plotted for all ROIs and dependencies).
Each point in the scatter plot represents the dependency prediction (x-axis) and brain prediction (y-axis) scores for each of the 144 heads. Brain prediction scores reflect cross-validated encoding model performance evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation. Dependency prediction scores reflect the classification accuracy of a cross-validated logistic regression model trained to predict the occurrence of a given linguistic dependency at each TR from the 64-dimensional transformation vector for a given attention head. Each of these plots corresponds to a labeled cell in the dependencies-by-ROI correlation matrix in panel B. (B) Correlation between headwise brain prediction and dependency prediction scores for each language ROI and syntactic dependency. Dependencies (y-axis) are ordered by their token distance; e.g. the adjectival modifier (amod) spans fewer tokens on average than the clausal complement (ccomp; see “Materials and methods” for details). Labeled cells correspond to the example correlations in panel A. (C) We summarize the brain–dependency prediction correspondence for each ROI by averaging across syntactic dependencies (i.e. averaging each column of panel B).

Discussion
While a growing body of work has used the Transformer architecture (Vaswani et al., 2017; Devlin et al., 2019) to model the neural basis of human language processing (Toneva & Wehbe, 2019; Caucheteux et al., 2021c; Heilbron et al., 2021; Schrimpf et al., 2021; Goldstein et al., 2022), we contribute a fundamentally new perspective on how to relate these models to human brain activity. Prior work has focused on the Transformer’s representation of linguistic content (the embeddings); we instead interrogate the computations implemented by functionally-specialized internal components of the model (the transformations). We demonstrate that the transformations yield comparable performance to the embeddings in predicting brain activity, and that both transformations and embeddings outperform classical linguistic features and non-contextual word embeddings in all language areas. However, the transformations provide novel insights. First, they yield more layer-specific predictive performance and better reflect the cortical hierarchy for linguistic and narrative processing (Lerner et al., 2011; Hasson et al., 2015; Caucheteux et al., 2021b) than the embeddings. Second, we can segregate the transformations into the computations performed by separate, functionally-specialized heads. Examining the contribution of headwise transformations to encoding performance reveals gradients in a low-dimensional cortical space that reflect structural and functional properties of the headwise transformations, including layer, look-back distance, and predictive performance on syntactic dependencies. Finally, we quantified the correspondence between headwise predictions of brain activity and syntactic dependencies for a variety of cortical language areas and dependencies, and found that certain ROIs (e.g. PostTemp, AngG, MFG) display strong headwise prediction correspondence across a variety of dependencies.

To build an intuition for why the transformations may provide complementary insights to the embeddings, we can compare Transformers to convolutional neural networks (CNNs) commonly used in visual neuroscience (Kriegeskorte, 2015; Yamins & DiCarlo, 2016). CNNs typically reduce dimensionality across layers (Krizhevsky et al., 2012; He et al., 2016), putting pressure on the model to gradually discard task-irrelevant, low-level information and retain only high-level semantic content. In contrast, popular Transformer architectures maintain the same dimensionality across layers. Thus Transformer embeddings can aggregate information (from context words) across layers, such that later layers tend to contain the most information (Tenney et al., 2019). In this light, it is unsurprising that encoding performance tends to peak at later embedding layers. Indeed, unlike the structural correspondence between CNN layers and the visual processing hierarchy (Güçlü & van Gerven, 2015; Yamins & DiCarlo, 2016; Dupré la Tour et al., 2021), Transformer embeddings are highly predictive but relatively uninformative for localizing stages of language processing. Unlike the embeddings,

2 Albeit overspecialized for a particular downstream training objective; i.e. the cloze task for BERT.
the transformations reflect updates to word meanings at each layer. Encoding models based on the transformations must choose a step in the contextualization process, rather than “have it all” by simply using later layers. Our discovery that these transformations are equally predictive overall (Fig. 2) but display a more obvious progression of layer preferences across language areas (Fig. 4) suggests that brains and Transformers may follow a more similar overall pattern of processing than was previously believed—a finding which was potentially obfuscated by prior work’s focus on the embeddings.

Although Transformers perform remarkably well on NLP tasks, they are not positioned to serve as implementational models of cortical circuitry. Nonetheless, the human language system and end-to-end deep language models do—to some extent—share common computational principles aimed at producing well-formed, context-sensitive linguistic outputs (Hasson et al., 2020; Goldstein et al., 2022). In both cases, some level of functional specialization appears to emerge (McClelland et al., 2010; Nasr et al., 2019; Yang et al., 2019; Dobs et al., 2022), likely reflecting the statistical structure of natural language. That said, the syntactic operations formalized by classical linguistics may not directly correspond to particular neural computations, and these computations may not be localized to macroanatomical cortical circuits (Mesulam et al., 2015; I. Blank et al., 2016). In practice, we found that classical linguistic features (i.e. parts of speech and syntactic dependencies) are generally poor predictors of brain activity during natural language comprehension (Figs. 2, S10, S13).

The functional properties of the headwise transformations do, however, map onto certain cortical localization trends previously reported in the literature. For example, analyzing the encoding weights for transformations (Fig. 5) revealed that posterior temporal areas assign higher weights to heads at earlier layers (positive values along PC1) with shorter look-back distance (negative values along PC2), consistent with previous work suggesting that posterior temporal areas perform early-stage syntactic (and lexico-semantic) processing (Hickok & Poeppel, 2000, 2007; Flick & Pylkkänen, 2020; Murphy et al., 2022). Headwise correspondence in posterior temporal cortex was high for the ccomp and dobj dependencies (Fig. 6), which are involved in resolving the meaning of verb phrases, corroborating prior work implicating posterior temporal areas in verb–argument integration (Ben-Shachar et al., 2003; Friederici et al., 2003; Bornkessel et al., 2005). Left-lateralized anterior temporal and medial prefrontal areas instead preferred longer look-back distances (Fig. 5), perhaps corresponding to longer-range narrative or event-level relations (Baldassano et al., 2018). Interestingly, the IFG (pars opercularis and triangularis; i.e. Broca’s area) was not strongly associated with heads specialized for particular syntactic operations (Fig. 6B, C), despite being well-predicted by both BERT embeddings and transformations (Fig. 2). There are several possible explanations for this: (1) the natural language stimuli used here may not contain sufficient syntactic complexity to tax IFG; (2) the cortical parcellation used here may yield imprecise functional localization of IFG (Fedorenko & Blank, 2020); and (3) the IFG may be more involved in language production than comprehension (Matchin & Hickok, 2020).

Despite the formal distinction between syntax and lexico-semantics in linguistics, the neural computations supporting human language may not so cleanly dissociate syntactic and semantic processing (Fedorenko et al., 2020), especially during natural language comprehension where syntax and semantics are typically intertwined. Indeed, although Transformer models implicitly learn syntactic and compositional operations in order to produce well-formed linguistic outputs, these emergent structures are generally entangled with semantic content (Baroni, 2020; Manning et al., 2020; Linzen & Baroni, 2021). While the transformations used in the current analysis capture syntactic and contextual operations entangled with semantic content, the transformation magnitudes can serve to disentangle syntax and semantics. Prior work has sought to isolate
the two using artificial stimuli and vector subtraction on the embeddings (Caucheteux et al., 2021c); the transformation magnitudes instead reduce the transformations down to the “activation” of individual attention heads. Insights from NLP (Clark et al., 2019) suggest this metric, which circumvents the stimulus representation entirely, nonetheless contains an emergent form of syntactic information. Our comparison to classical linguistic features (Fig. 2) suggests this is the case: transformation magnitudes outperform classical linguistic features in temporal language areas, and perform comparably elsewhere. Interestingly, the transformation magnitudes also outperform the non-contextual word embeddings in temporal areas (Fig. 2), while this relationship is reversed in angular gyrus, a putative high-level convergence zone for semantic representation (Binder et al., 2009).

There are several limitations to our work. First, we adopt the pretrained BERT-base as our primary model, as its functional specialization has been most extensively characterized by the NLP community (Rogers et al., 2020). BERT, however, is not trained in a biologically plausible manner: it allows for bi-directional information flow and therefore has access to both past and future tokens (Goldstein et al., 2022). More broadly, we hope that language models with more biologically-motivated architectures and human-like objectives will provide deeper insights into the human language faculty (Hasson et al., 2020). Additionally, the temporal resolution of fMRI is not high enough to fully capture language processing that occurs on rapid timescales. The current fMRI results provide complementary insights to measurement modalities with higher temporal resolution (Heilbron et al., 2021; Goldstein et al., 2022) because the superior spatial coverage of fMRI allows us to map language encoding throughout the entire brain.

Our results suggest several future lines of research. Prior work has explored different Transformer architectures (Caucheteux et al., 2021c; Schrimpf et al., 2021) aiming to establish a structural mapping between Transformers and the brain. Toward this end, training “bottlenecked” Transformer models that successively reduce the dimensionality of linguistic representations—similar to CNNs—may produce more hierarchical embeddings and provide a better structural mapping onto cortical language circuits (Raghu et al., 2021). Second, the current work sidesteps the acoustic and prosodic features of natural speech (Santoro et al., 2014; de Heer et al., 2017); future work, however, may benefit from models that extract high-level contextual semantic content directly from the speech signal (in the same way that CNNs operate directly on pixel values) (Li et al., 2022; Millet et al., 2022; Vaidya et al., 2022). Third, a large body of prior work has utilized Transformer embeddings to predict the brain; it may be worth revisiting some of these findings with an emphasis on transformations instead. Finally, current neurobiological models of language highlight the importance of long-range fiber tracts connecting the nodes of the language network (Catani et al., 2005; Saur et al., 2008; Dick & Tremblay, 2012); we suspect that future language models with more biologically-inspired circuit connectivity may provide insights not only into functional specialization but also functional integration across specialized modules.

Transformer-based deep language models like BERT obtain state-of-the-art performance on multiple NLP tasks. BERT’s attention heads are functionally specialized and learn to approximate classical syntactic operations in order to produce contextualized natural language (Clark et al., 2019; Tenney et al., 2019). The rapidly developing field of BERTology (Rogers et al., 2020) seeks to characterize this emergent functional specialization. In this work, we took a first step towards bridging BERTology’s insights to language processing in the brain. Mapping the internal structure of deep language models to cortical language circuits can bring us closer to a mechanistic understanding of human language processing, and may ultimately provide insights into how and why this kind of functional specialization emerges in both deep language models and the brain.
Materials and methods

Experimental data

Models were evaluated on two story datasets from the publicly available “Narratives” collection of fMRI datasets acquired while subjects listened to naturalistic spoken stories (Nastase et al., 2021). Code used to analyze the data are available at the accompanying GitHub repository: https://github.com/tsumers/bert-brains. The “Slumlord” and “Reach for the Stars One Small Step at a Time” dataset includes 18 subjects (ages: 18–27 years, mean age: 21 years, 9 reported female) and comprises two separate stories roughly 13 minutes (550 TRs) and 2,600 words in total. The “I Knew You Were Black” dataset includes 45 subjects (ages: 18-53 years, mean age: 23.3 years, 33 reported female); the story is roughly 13 minutes (534 TRs) long and contains roughly 1,500 words. All functional MRI datasets were acquired with a 1.5 s TR (Nastase et al., 2021), and were organized in compliance with the Brain Imaging Data Structure (Gorgolewski et al., 2016).

Preprocessed MRI data were obtained from the Narratives derivatives release (Nastase et al., 2021). Briefly, the following preprocessing steps were applied to the functional MRI data using fMRIPrep (Esteban et al., 2019): susceptibility distortion correction (using fMRIPrep's fieldmap-less approach), slice-timing correction, volume registration, and spatial normalization to MNI space (MNI152NLin2009cAsym template). Confound regression was then implemented using AFNI's 3dTproject (Cox, 1996) with the following confound regressors: six head motion parameters, the first five aCompCor components from CSF and from white matter (Behzadi et al., 2007), cosine bases for high-pass filtering (cutoff: 128 s), and first- and second-order polynomial trends. Non-smoothed functional data were used for all analyses in the current study. To harmonize datasets with differing spatial resolution and reduce computational demands, we resampled all functional data to a fine-grained 1000-parcel cortical parcellation derived from intrinsic functional connectivity (Schaefer et al., 2018). That is, time series were averaged across voxels within each parcel to yield a single average response time series per parcel (within each subject and story dataset).

Baseline language features

Language model representations were derived from the time-locked phoneme- and word-level transcripts available in the Narratives dataset (Nastase et al., 2021). Words were assigned to the fMRI volumes based on the ending timestamps; e.g. if a word began in one TR and ended in the following TR, it was assigned to the second TR. The following low-level acoustic and linguistic features were extracted for each TR to serve as confound variables in subsequent analyses: (a) the number of words per TR, (b) number of phonemes per TR, and (c) a binary vector indicating the presence of individual phonemes per TR.

We extracted part-of-speech and dependency relations to serve as classical linguistic features. These features were annotated using the en_core_web_lg (v2.3.1) model from spaCy (v2.3.7) (Honnibal et al., 2020). For each TR, we created a binary feature vector across all parts-of-speech/dependency relations, indicating whether or not a given part-of-speech/dependency relation appeared within that TR. Part of speech (e.g. noun, verb, adjective) describes the function of each word in a sentence. We used 14 part-of-speech labels: pronoun, verb, noun, determiner, auxiliary, adposition, adverb, coordinating conjunction, adjective, particle, proper noun, subordinating conjunction, numeral, and interjection (Table S3). A dependency relation describes the syntactic relation between two words. For each word, a parser defines another word in the same sentence, called the “head,” to which the word is syntactically related; the dependency relation describes the way the word is related to its head. We used 25 dependency relations: nsubj, ROOT, advmod, prep, det, pobj,
aux, dobj, cc, ccomp, amod, compound, acomp, poss, xcomp, conj, relcl, attr, mark, npadvmod, advcl, neg, prt, nummod, and intj (Table S4; https://github.com/clir/clearnlp-guidelines/blob/master/md/specifications/dependency_labels.md). For visualization, we focus on 12 dependencies with particularly high correspondence to particular heads reported by Clark and colleagues (Clark et al., 2019).

Finally, to serve as a baseline for Transformer-based language models, we used GloVe vectors (Pennington et al., 2014) which capture the “static” semantic content of a word across contexts. Conceptually, GloVe vectors are similar to the vector representations of text input to BERT prior to any contextualization applied by the Transformer architecture. We obtained GloVe vectors for each word using the en_core_web_lg model from spaCy, and averaged vectors for multiple words occurring within a TR to obtain a single vector per TR.

**Transformer self-attention mechanism**

While language models such as GloVe (Pennington et al., 2014) assign a single “global” or “static” embedding (i.e. meaning) to a given word across all contexts, the Transformer architecture (Vaswani et al., 2017) introduced the self-attention mechanism which yields context-specific representations. Just as convolutional neural nets (LeCun et al., 1998; Krizhevsky et al., 2012) use convolutional filters to encode spatial inductive biases, Transformers use self-attention blocks as a sophisticated computational motif or “circuit” that is repeated both within and across layers. Self-attention represents a significant architectural shift from sequential processing of language via recurrent connections (Elman, 1990) to simultaneously processing multiple tokens. Variations on the Transformer architecture with different dimensionality and training objectives currently dominate major tasks in NLP, with BERT (Devlin et al., 2019) and GPT (Radford et al., 2019) being two of the most prominent examples.

Self-attention operates as follows. A single attention head consists of three separate (learned) matrices: a query matrix, a key matrix, and a value matrix, each of dimensionality $d_{model} \times d_{head}$, where $d_{model}$ indicates the dimensionality of the model’s embedding layers, and $d_{head}$ indicates the dimensionality of the attention head. Input word vectors are multiplied by each of these three matrices independently, producing a query, key, and value vector for each word. To determine the contextualized representation for a given word vector, the self-attention operation takes the dot product of that word’s query vector with the key vector from all words. The resulting values are then scaled and softmaxed, producing the “attention weights” for that word.

Formally, for a Transformer head of dimensionality $d_{head}$ and sets of query, key, and value vectors $Q, K, V$, the self-attention mechanism operates as follows:

$$Attention(Q, K, V) = softmax\left(\frac{Q \times K}{\sqrt{d_{head}}}\right)V \quad (1)$$

The $i$th token “attends” to tokens based on the inner product of its query vector $Q$, with the key vectors for all tokens, $K$. When the query vector matches a given key, the inner product will be large; the softmax ensures the resulting “attention weights” sum to one. These attention weights are then used to generate a weighted sum of the value vectors, $V$, which is the final output of the self-attention operation (Eq. 1). We refer to this output as the “transformation” produced by an individual head. Each attention head produces a separate transformation for each input token.

State-of-the-art Transformer architectures scale up the core self-attention mechanism described above in two ways. First, multiple attention heads are assembled in parallel within a given layer (“multi-headed attention”).
For example, BERT-base-uncased (Devlin et al., 2019), used in most of our analyses, contains 12 attention heads in each layer. The 12 attention heads are each 64-dimensional and act in parallel to produce independent 64-dimensional “transformation” vectors. These vectors are concatenated to produce a single 768-dimensional vector which is then passed to the feed-forward module. Second, BERT-base consists of 12 identical layers stacked on top of each other, allowing the model to learn a hierarchy of transformations for increasingly complex contextualization (Tenney et al., 2019). The full model has 12 layers × 12 heads = 144 total attention heads and a total of 110 million learned parameters. The original input vectors are analogous to the static GloVe vectors; however, as they pass through successive layers in the model, repeated applications of the self-attention mechanism allow their meanings to contextualize each other.

The phenomenal success of Transformer-based models has generated an entire sub-field of NLP research, dubbed “BERTology,” dedicated to reverse-engineering their internal representations (Manning et al., 2020; Rogers et al., 2020). Researchers have linked different aspects of the model to classic NLP concepts, including functional specialization of individual “attention heads” to syntactic dependencies (Clark et al., 2019), representational subspaces corresponding to parse trees (Hewitt & Manning, 2019), and an overall layerwise progression of representations that parallels traditional linguistic pipelines (Tenney et al., 2019). Our approach builds on this work, using internal Transformer features as a bridge from these classical linguistic concepts to brain data.

**Transformer-based features**

We used three separate components from the Transformer models to predict brain activity. The first features we extract are the layerwise “embeddings,” which are the de facto Transformer feature used for most applications, including prior work in neuroscience (e.g. Schrimpf et al., 2021). The embeddings represent the contextualized semantic content, with information accumulating across successive layers as the Transformer blocks extract increasingly nuanced relationships between tokens (Tenney et al., 2019). As a result, later layers often represent a superset of information available in earlier layers, while the final layers are optimized to the specific pretraining task (for BERT, causal language modeling). Prior work using such models typically finds that the mid-to-late layers best predict brain activity (Caucheteux et al., 2021c; Schrimpf et al., 2021).

The second set of features we extract are the headwise “transformations” (Eq. 1), which capture the operation of the self-attention mechanism. Each head’s transformation reflects the information content added by that particular head. While the transformations represent the most crucial aspect of the self-attention mechanism, to our knowledge they have not been previously studied in human neuroscience.

Finally, the third set of features we extract are the “transformation magnitudes,” which are the L2 norm of each attention head’s transformation. This is effectively a “univariate” metric for how active each attention head was: how much its update affected each word’s representation, without any information about the direction of its influence. Prior work has shown that individual attention heads learn to implement particular syntactic relationships (Clark et al., 2019); therefore, the transformation magnitudes provide information about the relationships between tokens in the TR divorced from the semantic content of the TR.

**Generating Transformer features**

Transformer models were accessed via the HuggingFace library (Wolf et al., 2020). We used the BERT-base-uncased model. We generated “embeddings” and “transformations” as follows. For each TR, we concatenated words assigned to that TR (3.75 words on average) together with words from the preceding 20 TRs (corresponding to the preceding 30 seconds of the auditory story stimulus). We passed the resulting set
of words through the Transformer tokenizer and then model. This procedure allowed information from the preceding time window to “contextualize” the meaning of the words occurring in the present TR. At each layer, we extracted the “transformations” (output of the self-attention submodule, Eq. 1) and the “embeddings” (output of the final feedforward layer) for only the words in that TR. We excluded the automatically appended “SEP” token. For each TR, this left us with a tensor of dimension \( n_{\text{layers}} \times d_{\text{model}} \times n_{\text{tokens}} \), where \( n_{\text{layers}} \) indicates the number of layers in the Transformer model, \( d_{\text{model}} \) indicates the dimensionality of the model’s embedding layers, and \( n_{\text{tokens}} \) the number of tokens spoken in that TR. For BERT, this resulted in a 12 layer \( \times 768 \) dimension \( \times n_{\text{tokens}} \) tensor. To reduce this to a consistent dimensionality, we averaged over the tokens occurring within each TR, resulting in a \( 12 \times 768 \times 1 \) tensor for each TR. TRs with no words were assigned zeros. Finally, to generate “transformation magnitudes,” we computed the L2 norm of each headwise transformation.

To generate the “backwards attention” metric (Fig. 5), we followed a procedure similar to the “attention distance” measure in (Vig & Belinkov, 2019). Unlike the previous analyses, this required a fixed number of Transformer tokens per TR. Rather than using the preceding 20 TRs, we first encoded the entire story using the Transformer tokenizer, and for each TR selected the 128 tokens preceding the end of the TR. This corresponded to approximately 30 seconds of the auditory story stimulus. We processed each TR and extracted each head’s matrix of token-to-token attention weights (Eq. 1). We selected the token-to-token attention weights corresponding to information flow from earlier words in the stimulus into tokens in the present TR.\(^3\) We multiplied each token-to-token attention weight by the distance between the two tokens, and divided by the number of tokens in the TR to obtain the per-head attention distance in that TR. Finally, we averaged this metric over all TRs in the stimulus to obtain the headwise attention distances.

**Decoding dependency relations**

In addition to using the linguistic features to predict brain activity, we used the “transformation” representations to predict dependencies. For each TR, the “transformation” consists of a \( n_{\text{layers}} \times d_{\text{model}} \) tensor; for BERT, this yields a 12 layers \( \times 768 \) dimension tensor. Each of the layers is thus a 768-dimensional vector, which itself consists of 12 concatenated 64-dimensional vectors, each corresponding to the output of a single attention head. These 64-dimensional headwise vectors were used as predictors. Note that this decomposition is only valid for transformations, which are the direct results of the multi-head attention mechanism; due to the feedforward layer after the multi-head attention mechanism, there is no trivial way to reduce embeddings to headwise representations.

We performed logistic regression with the L2 penalty (implemented using scikit-learn; Pedregosa et al., 2011) to predict the occurrences of each binary dependency relation over the course of each story from the headwise transformations. The regularization hyperparameter was determined for each head and each dependency relation using nested five-fold cross-validation over a log-scale grid with 11 values ranging from \( 10^{-30} \) to \( 10^{30} \). Since some dependency relations are relatively rare, labels are imbalanced. We corrected for this imbalance by weighting samples according to the inverse frequency of occurrence during training and by using balanced accuracy for evaluation (Brodersen et al., 2010).

**Encoding model estimation and evaluation**

Encoding models were estimated using banded ridge regression with three-fold cross-validation (Dupré la Tour et al., 2022). Ridge regression is a formulation of regularized linear regression using a penalty term to

---

\(^3\) We excluded the special [SEP] token.
ensure that the learned coefficients minimize the squared L2 norm of the model parameters. This effectively imposes a multivariate normal prior with zero mean and spherical covariance on the model parameters (Nunez-Elizalde et al., 2019). Relative to ordinary least squares, ridge regression tends to better accommodate collinear parameters and better generalize to unseen data (by reducing overfitting). In banded ridge regression, the full model is composed of several concatenated submodels including both features of interest (e.g. BERT features) and confound variables. The following confound variables were included: phoneme and word rate per TR, a 32-dimensional phoneme indicator matrix, and an indicator vector indicating silent TRs. Each of these submodels was assigned to a separate “band” in the banded ridge regression and thus received a different regularization parameter when fitting the full model. Specifically, we have separate bands for: the main model (e.g. BERT features), the silence indicator vector, word count and phoneme count vectors, and the phoneme indicator matrix. When evaluating the model, we generate predictions using only the main model band, discarding any confound features. The regularization parameters were selected via a random search of 100 iterations. In each iteration, we sample regularization parameters for each band uniformly from the simplex by sampling from a dirichlet distribution (as implemented in the himalaya package) (Dupré la Tour et al., 2022). We performed nested three-fold cross-validation within each training set of the outer cross-validation partition. Encoding models were fit for each of the 1,000 parcels within each subject and each story dataset. The cross-validation procedure was implemented so as to partition the time series into temporally contiguous segments. When conducting the headwise encoding analyses (Fig. 6), we also discard the learned coefficients corresponding to all heads except for the particular head of interest for prediction and evaluation on the test set (Lee Masson & Isik, 2021).

Encoding model performance was evaluated by computing the Pearson correlation between the predicted and actual time series for the test partition. Correlation was used as the evaluation metric in both the nested cross-validation loop for regularization hyperparameter optimization, and in the outer cross-validation loop. For each partition of the outer cross-validation loop, the regularization parameter with the highest correlation from the nested cross-validation loop within the training set was selected. This procedure yields a correlation value for each test set of the outer cross-validation loop for each parcel and subject. These correlation values were then averaged across cross-validation folds and Fisher-z transformed prior to statistical assessment.

**Noise ceiling estimation**

The raw correlation values described above depend on the signal-to-noise ratio (SNR), duration, and other particular features of the data. In order to provide a more interpretable metric of model performance, we compute the proportion of the correlation value relative to a noise ceiling—effectively the proportion of explained variance relative to the total variance available. Acquiring replicates of brain responses to the same stimulus in the same subjects is effectively impossible under naturalistic contexts due to repetition effects; i.e. listening to a narrative for a second time does not engage the same cognitive processes as listening to the narrative for the first time (Aly et al., 2018). To circumvent this challenge, we used intersubject correlation (ISC) as an estimate of the noise ceiling (Nili et al., 2014; Nastase et al., 2019). In this approach, time series from all subjects are averaged to derive a surrogate model intended to represent the upper limit of potential model performance. For each subject, the test time series for each outer cross-validation fold is first averaged with the test time series of all other subjects in the dataset, then the test time series for that subject is correlated with the average time series. Including each subject in the average biases the noise ceiling upward, thus yielding more conservative estimates of the proportion. Note that under circumstances where individual subjects are expected to vary considerably in their functional architecture, ISC may provide a suboptimal noise ceiling. However, in the current context, we do not hypothesize or model any such differences.
Statistical assessment

To assess the statistical significance of encoding model performance, we used two nonparametric randomization tests. First, when testing whether model performance was significantly greater than zero, we used a one-sample bootstrap hypothesis test (Hall & Wilson, 1991). For each iteration of the bootstrap, we randomly sampled subject-level correlation values (averaged across cross-validation folds), then computed the Fisher-transformed mean across the bootstrap sample of subjects to construct a bootstrap distribution around the mean model performance value across subjects for each parcel. We then subtracted the observed mean performance value from this bootstrap distribution, thus shifting the mean roughly to zero (the null hypothesis). Finally, we computed a one-sided $p$-value by determining how many samples from the shifted bootstrap distribution exceeded the observed mean.

Second, when comparing performance between two models, we used a permutation test. For each iteration of the permutation test, we took the subject-wise differences in performance between the two models, randomly flipped their signs, then recomputed the mean difference in correlation across subjects to populate a null distribution. We then computed a two-sided $p$-value by determining how many samples from either tail of the distribution exceeded the mean observed difference.

Statistical tests for population inference were performed by concatenating subjects across story datasets prior to randomization in order to produce one $p$-value across stories; however, randomization was stratified within each story. When assessing $p$-values across parcels or ROIs, we corrected for multiple tests by controlling the false discovery rate (FDR) at $q < .05$ (Benjamini & Hochberg, 1995).

Summarizing headwise transformation weights

To summarize the contribution of headwise transformations across the language network, we first obtained the regression coefficients—i.e. weight matrices—from the encoding model trained to predict brain activity from the BERT transformations concatenated across layers (Fig. 1, red; corresponding to the performance in Fig. 2, red). We averaged the parcelwise weight vectors across both subjects and stimuli. We next computed the L2 norm of the regression coefficients within each head at each layer, summarizing the contribution of the transformation at each head for each parcel. Following Huth and colleagues (Huth et al., 2012, 2016), we then used principal component analysis (PCA) to summarize these headwise transformation weights across all parcels in the language ROIs. This yields a reduced-dimension brain space where each data point corresponds to the transformation implemented by each of the 144 attention heads. To visualize the structure of these headwise transformations in the reduced-dimension brain space, we colored the data points according to structural and functional properties of the heads, including their layer, backward attention distance, and dependency prediction scores.

Acknowledgments and funding sources

S.K. is supported by NIH T32MH065214. T.S. is supported by an NDSEG Fellowship. R.H. is supported by a C.V. Starr Fellowship. S.A.N. is supported by NIH R01MH112566.

References


Dick, A. S., & Tremblay, P. (2012). Beyond the arcuate fasciculus: consensus and controversy in the


Huth, A. G., Nishimoto, S., Vu, A. T., & Gallant, J. L. (2012). A continuous semantic space describes the


Supporting information

Reconstructing the cascade of language processing in the brain using the internal computations of Transformer language models

Sreejan Kumar¹,*, Theodore R. Sumers²,*, Takateru Yamakoshi³, Uri Hasson¹, Uri Hasson, Kenneth A. Norman¹, Thomas L. Griffiths², Robert D. Hawkins, Samuel A. Nastase¹

¹Princeton Neuroscience Institute, Princeton University, Princeton, NJ, 08540, USA
²Department of Computer Science, Princeton University, Princeton, NJ 08540, USA
³Faculty of Medicine, The University of Tokyo, Bunkyo-ku, Tokyo 113-0033, Japan
⁴Department of Psychology, Princeton University, Princeton, NJ 08540, USA

*Equal contribution.

Correspondence: sreejank@princeton.edu, sumers@princeton.edu, snastase@princeton.edu
Figure S1. Comparing three classes of language models in right-hemisphere language areas (cf. Fig. 2). Right-hemisphere language areas yield qualitatively similar encoding results to left-hemisphere language areas. Model performance is evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation. Markers indicate median performance and error bars indicate 95% bootstrap confidence intervals.
Figure S2. Time-point-by-time-point representational dissimilarity matrices (RDMs) for BERT embeddings (left) and transformations (right) concatenated across all layers (cf. Fig. 2). Despite yielding comparable encoding performance across many ROIs, the embeddings and transformations yield different representational geometries. The time-point-by-time-point RDMs are correlated at Spearman $r = .768$ and $r = .815$ for the “black” and “slumlord” stimuli, respectively. Relatively high correlations are expected when comparing representational geometries for the entire model (all 12 layers) because the transformations contextually sculpt the embeddings layer-by-layer; the layerwise representational geometries reveal a more obvious difference between embeddings and transformations (Figs. S6, S7). Dissimilarities between the vectors at each TR are measured using correlation distance ($1 - $Pearson correlation$)$. Dissimilarities are colored according to percentiles.
Figure S3. Transformations (red) have higher temporal autocorrelation than embeddings (blue) in both the “black” (left) and “slumlord” (right) story stimuli. We computed the TR-by-TR autocorrelation up to a lag of 30 TRs (45 s) separately for each feature in the embedding and transformation vectors. The blue and red lines indicate the mean temporal autocorrelation across features for embeddings and transformations, respectively. Bootstrap 95% confidence intervals are plotted around the means, but are not visible given the highly consistent temporal autocorrelation functions across features.
Figure S4. Encoding performance for three classes of language models in an early visual control ROI (cf. Fig. 2). Early visual cortex was anatomically defined as parcels overlapping with Brodmann Area 17 (primary visual cortex). As expected, none of these language features yield statistically significant prediction performance in early visual parcels. Model performance is evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation. Markers indicate median performance and error bars indicate 95% bootstrap confidence intervals.
Figure S5. Correlation between transformations across layers (left) and embeddings across layers (right). Each word of the story stimulus was propagated through the model and the transformations and embeddings were extracted at each layer. We averaged transformations and embeddings for each word within a TR (see “Generating Transformer features” in Methods). We computed the pairwise correlation of the transformations and embeddings across the 12 layers for each TR (each TR yields a 12 × 12 correlation matrix), then averaged the resulting correlation matrices across TRs. Embeddings are much more similar across layers because embeddings accumulate contextual information layer-by-layer; on the other hand, transformations capture layer-by-layer “updates” to the embedding and are largely uncorrelated across layers.
Figure S6. Time-point-by-time-point RDMs for embeddings (left) and transformations (right) within and across each layer of BERT. To compute the RDMs, we first split the embeddings and transformations by layer, then stacked these layers along the time axis; for example, the 12 diagonal blocks in the transformation RDMs (right) correspond to time-point-by-time-point RDMs at each layer. Cross-layer dissimilarities assume that the 768 features at each layer are shared across layers, and this is not strictly true by design. However, the embedding vectors are much more similar across layers than the transformation vectors. Overall, the RDMs for embeddings and transformations are highly dissimilar; Spearman \( r = .080 \) and \( .083 \) for the “black” and “slumlord” story stimuli, respectively. Dissimilarities between the vectors at each TR are measured using correlation distance \( (1 - \text{Pearson correlation}) \). Dissimilarities are colored according to percentiles.
Figure S7. Second-order layer-by-layer representational geometry of the time-point-by-time-point RDMs for embeddings (left) and transformations (right). We first computed a time-point-by-time-point RDM for each of the 12 layers. We computed the pairwise dissimilarity of these time-point RDMs across all 12 layers, resulting in a 12-by-12 RDM. The layerwise embedding RDMs (left) demonstrate that the layerwise representational geometries evolve sequentially across layers; i.e. the representational geometry at each layer is most similar to neighboring layers and most distinct from the most distant layers. The layerwise transformation RDMs (right), on the other hand, demonstrate that the layerwise representational geometries for transformations are not as sequentially organized; for example, early-layer transformations are more similar to particular late-layer transformations than they are to particular intermediate-layer transformations. The second-order layer-by-layer RDMs are correlated at Spearman $r = .750$ and $r = .848$ for the “black” and “slumlord” stimuli, respectively. Second-order dissimilarities between the time-point-by-time-point RDMs are measured using correlation distance ($1 –$ Pearson correlation). Dissimilarities are colored according to percentiles.
Figure S8. Layerwise encoding performance for embeddings (blue) and transformations (red) in right-hemisphere language areas (cf. Fig. 3A). Model performance is evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation. Markers indicate median performance and error bars indicate 95% bootstrap confidence intervals.
Figure S9. Softmax distribution of transformation and embedding performances across different layers (Tenney et al., 2019). The distribution of layer preferences for embeddings is more diffuse across layers than that of the transformations. To quantify this layer specificity, we quantified the entropy of these softmax distributions and found that the entropy of the embedding distribution was significantly higher than that of the transformation distribution in all of the ROIs (Table S2).
Figure S10. Comparison between encoding performance using the headwise transformation that best decodes a particular classical linguistic dependency (left) and the indicator variable for the linguistic dependency itself (right). The best-performing head is labeled (layer-head) for each dependency. Cortical maps are thresholded to display only parcels with statistically significant model performance (nonparametric bootstrap hypothesis test; FDR controlled at p < .05). Model performance is evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation. Note that for the poss, prt, and mark dependencies, no significant parcels were found.
**Figure S11.** Headwise brain prediction scores (top) and headwise dependency prediction scores (bottom). The headwise brain prediction scores are aggregated across 12 layers of BERT (x-axis) and 10 language ROIs (y-axis). There are 12 heads per layer in BERT. The histogram at right reflects the number of parcels in each language ROI. Brain prediction scores reflect cross-validated encoding model performance evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation. The headwise dependency prediction scores are aggregated across 12 layers of BERT (x-axis) and 12 classical linguistic dependencies. Dependency prediction scores reflect the classification accuracy of a cross-validated logistic regression model trained to predict the occurrence of a given linguistic dependency at each time point from the 64-dimensional transformation vector for a given attention head. The histogram at right reflects the number of occurrences of each linguistic dependency across both story stimuli (Table S5).
**Figure S12.** Encoding performance for heads previously highlighted in the literature for functional specialization (Clark et al., 2019). Heads are labeled according to the linguistic phenomenon for which they are specialized as well as their index in BERT (layer-head). Model performance is evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation.
Figure S13. Difference in encoding performance between headwise transformations reduced to a one-dimensional scalar vector and the dependency indicator variable (unthresholded). We reduced the 64-dimensional transformation vector for the head that best decodes a particular dependency down to a single scalar by computing a weighted across dimensions based on the weights estimated from the logistic regression used to decode that dependency (Fig. S10). Despite the matched dimensionality, warm colors indicate that the head-specific transformation values at each TR still outperform the corresponding classical dependency indicator variable in a cross-validated encoding analysis.
Figure S14. Projection of six PCs summarizing regression coefficients for the transformation-based encoding models (i.e. headwise transformation weights) onto the parcels of the language network (cf. Fig. 5). The PCs are orthogonal and capture non-redundant structure in the transformation weight matrix across the language network. Color bars reflect transformation weights for each PC (the polarity of each PC is arbitrary).
Figure S15. Headwise transformations weights projected into a two-dimensional space summarizing the language network colored according to the layer of each head (cf. Fig. 6D). PCs 9, 5, and 1 were the PCs most correlated with layer assignment with $r = .45$, .40, and .26, respectively. The PCs are orthogonal and capture non-redundant structure in the transformation weight matrix across the language network. Each PC can be projected back onto the cortical language network (Fig. S14). Each point in a scatter plot corresponds to one of the 144 attention heads.
Figure S16. Headwise transformation weights projected into a two-dimensional space summarizing the language network colored according to the backward attention distance of each head (cf. Fig. 6E). PCs 2, 1, and 3 were the PCs most correlated with backward attention distance with $r = .65, .20, .19$, respectively. The PCs are orthogonal and capture non-redundant structure in the transformation weight matrix across the language network. Each PC can be projected back onto the cortical language network (Fig. S14). Each point in a scatter plot corresponds to one of the 144 attention heads.
Figure S17. Headwise transformation weights projected into a two-dimensional space summarizing the language network colored according to each head’s dependency prediction score for each classical linguistic dependency (cf. Fig. 6G). Dependency prediction scores reflect the classification accuracy of a cross-validated logistic regression model trained to predict the occurrence of a given linguistic dependency at each TR from the 64-dimensional transformation vector for a given attention head. The PCs are orthogonal and capture non-redundant structure in the transformation weight matrix across the language network. Each PC can be projected back onto the cortical language network (Fig. S14). Each point in a scatter plot corresponds to one of the 144 attention heads.
Figure S18. Correspondence between headwise brain prediction and dependency prediction scores for all dependencies (x-axis) and language ROIs (y-axis; cf. Fig. 6). Each point in a given scatter plot represents the dependency prediction (x-axis) and brain prediction (y-axis) scores for each of the 144 heads. Brain prediction scores reflect cross-validated encoding model performance evaluated in terms of the percent of a noise ceiling estimated using intersubject correlation. Dependency prediction scores reflect the classification accuracy of a cross-validated logistic regression model trained to predict the occurrence of a given linguistic dependency at each time point from the 64-dimensional transformation vector for a given attention head.
<table>
<thead>
<tr>
<th>Comparison</th>
<th>HG</th>
<th>PostTemp</th>
<th>AntTem</th>
<th>AngG</th>
<th>MFG</th>
<th>IFG</th>
<th>IFGorb</th>
<th>vmPFC</th>
<th>dmPFC</th>
<th>PMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT transformations vs BERT embeddings</td>
<td>0.6620</td>
<td>0.0567</td>
<td>0.6700</td>
<td>0.9140</td>
<td>0.7887</td>
<td>0.7322</td>
<td>0.1517</td>
<td>1.0000</td>
<td>0.1867</td>
<td>0.0660</td>
</tr>
<tr>
<td>BERT transformations vs BERT transformation magnitudes</td>
<td>0.0114</td>
<td>0.0057</td>
<td>0.0025</td>
<td>0.0017</td>
<td>0.7000</td>
<td>0.0250</td>
<td>0.0025</td>
<td>0.0071</td>
<td>0.0017</td>
<td>0.0533</td>
</tr>
<tr>
<td>BERT transformations vs linguistic features</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0025</td>
<td>0.0017</td>
<td>0.7000</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0020</td>
<td>0.0017</td>
<td>0.0017</td>
</tr>
<tr>
<td>BERT transformations vs GloVe embeddings</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0500</td>
<td>0.0017</td>
<td>0.7887</td>
<td>0.0025</td>
<td>1.0000</td>
<td>0.0020</td>
<td>0.0171</td>
<td>0.0017</td>
</tr>
<tr>
<td>BERT embeddings vs BERT transformation magnitudes</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0025</td>
<td>0.0017</td>
<td>1.0000</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0020</td>
<td>0.0017</td>
<td>0.0425</td>
</tr>
<tr>
<td>BERT embeddings vs linguistic features</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0025</td>
<td>0.0017</td>
<td>0.7887</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0020</td>
<td>0.0017</td>
<td>0.0017</td>
</tr>
<tr>
<td>BERT embedding vs GloVe embeddings</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.0140</td>
<td>0.0017</td>
<td>0.7887</td>
<td>0.0140</td>
<td>0.2256</td>
<td>0.0020</td>
<td>0.0017</td>
<td>0.0017</td>
</tr>
<tr>
<td>BERT transformation magnitudes vs linguistic features</td>
<td>0.0017</td>
<td>0.0017</td>
<td>0.6089</td>
<td>0.8087</td>
<td>0.7000</td>
<td>0.7322</td>
<td>0.1986</td>
<td>0.1212</td>
<td>0.7500</td>
<td>0.0017</td>
</tr>
<tr>
<td>BERT transformation magnitudes vs GloVe embeddings</td>
<td>0.0175</td>
<td>0.0488</td>
<td>0.0267</td>
<td>0.4814</td>
<td>0.8678</td>
<td>0.8540</td>
<td>0.2256</td>
<td>1.0000</td>
<td>0.1375</td>
<td>0.0017</td>
</tr>
<tr>
<td>Linguistic features vs GloVe embeddings</td>
<td>0.2633</td>
<td>0.0940</td>
<td>0.0663</td>
<td>0.8144</td>
<td>0.7000</td>
<td>0.6014</td>
<td>0.0200</td>
<td>0.0033</td>
<td>0.0017</td>
<td>0.0229</td>
</tr>
</tbody>
</table>

**Table S1.** Statistical significance of comparisons between BERT features (embeddings, transformations, transformation magnitudes), static GloVe embeddings, and classical linguistic features at each language ROI. For each comparison, two-tailed p-values reflect the statistical significance of the difference in encoding performance across subjects for a given ROI (permutation test; FDR controlled at p < .05).
Table S2. Mean entropy of the distribution of encoding performance across layers for embeddings and transformations at each language ROIs. Statistical significance for the difference between entropy for embeddings and transformations was evaluated by bootstrap resampling the differences (FDR controlled at p < .05).

<table>
<thead>
<tr>
<th>ROI</th>
<th>Embedding Entropy</th>
<th>Transformation Entropy</th>
<th>Difference</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PostTemp</td>
<td>1.2130171</td>
<td>0.68240984</td>
<td>5.126</td>
<td>0.00016667</td>
</tr>
<tr>
<td>AntTemp</td>
<td>0.967746731</td>
<td>0.3861663</td>
<td>6.664</td>
<td>0.00016667</td>
</tr>
<tr>
<td>AngG</td>
<td>0.635415239</td>
<td>0.38440445</td>
<td>2.635</td>
<td>0.0122</td>
</tr>
<tr>
<td>IFG</td>
<td>0.849934771</td>
<td>0.37969937</td>
<td>4.214</td>
<td>0.00342857</td>
</tr>
<tr>
<td>MFG</td>
<td>0.642895319</td>
<td>0.39010882</td>
<td>3.005</td>
<td>0.0122</td>
</tr>
<tr>
<td>IFGorb</td>
<td>0.710395586</td>
<td>0.3873914</td>
<td>3.356</td>
<td>0.0122</td>
</tr>
<tr>
<td>vmPFC</td>
<td>0.819666132</td>
<td>0.41612847</td>
<td>3.975</td>
<td>0.00016667</td>
</tr>
<tr>
<td>dmPFC</td>
<td>0.767067125</td>
<td>0.37410996</td>
<td>5.21</td>
<td>0.00016667</td>
</tr>
<tr>
<td>PMC</td>
<td>1.005453546</td>
<td>0.6119654</td>
<td>3.604</td>
<td>0.00016667</td>
</tr>
<tr>
<td>HG</td>
<td>1.067141906</td>
<td>0.51129539</td>
<td>4.432</td>
<td>0.00016667</td>
</tr>
<tr>
<td>Part-of-speech</td>
<td>Description</td>
<td>Examples</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>------------------</td>
<td>----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRON</td>
<td>pronoun</td>
<td>she, somebody, mine</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VERB</td>
<td>verb</td>
<td>run, eating</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOUN</td>
<td>noun</td>
<td>girl, tree, air</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DET</td>
<td>determiner</td>
<td>a, the, this</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUX</td>
<td>auxiliary</td>
<td>is, will, should</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADP</td>
<td>adposition</td>
<td>in, to, during</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADV</td>
<td>adverb</td>
<td>well, tomorrow, very</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCONJ</td>
<td>coordinating conjunction</td>
<td>and, or, but</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADJ</td>
<td>adjective</td>
<td>big, green, first</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PART*</td>
<td>particle</td>
<td>’s, not</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROPN</td>
<td>proper noun</td>
<td>Mary, John, London</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCONJ</td>
<td>subordinating conjunction</td>
<td>if, while</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NUM</td>
<td>numeral</td>
<td>one, two, three</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INTJ</td>
<td>interjection</td>
<td>psst, bravo</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table S3.** Descriptions and examples of the parts-of-speech. Examples are excerpted from [https://universaldependencies.org/u/pos/](https://universaldependencies.org/u/pos/). Phrasal verb particles, e.g. [give] in, are not included; they are tagged as ADP or ADV.
Table S4. Descriptions and examples of the dependency relations in Figure 5. The labels are part of ClearNLP dependency labels (https://github.com/clir/clearnlp-guidelines/blob/master/md/specifications/dependency_labels.md). Examples are adapted from https://universaldependencies.org/u/dep/all.html. Note that poss, prt, and dobj in the table correspond to nmod:poss, compound:prt, and obj in the universal dependencies, respectively; pobj and prep are not part of the universal dependencies.
Table S5. Number of TRs in which linguistic dependency occurs in the two story stimuli. The column headers indicate the total number of TRs for that stimulus.

<table>
<thead>
<tr>
<th>Dependency</th>
<th>“black” (~534)</th>
<th>“slumlord” (~619)</th>
</tr>
</thead>
<tbody>
<tr>
<td>prep</td>
<td>121</td>
<td>214</td>
</tr>
<tr>
<td>pobj</td>
<td>109</td>
<td>195</td>
</tr>
<tr>
<td>det</td>
<td>117</td>
<td>206</td>
</tr>
<tr>
<td>nsubj</td>
<td>245</td>
<td>392</td>
</tr>
<tr>
<td>amod</td>
<td>43</td>
<td>66</td>
</tr>
<tr>
<td>dobj</td>
<td>95</td>
<td>153</td>
</tr>
<tr>
<td>advmod</td>
<td>122</td>
<td>177</td>
</tr>
<tr>
<td>aux</td>
<td>99</td>
<td>150</td>
</tr>
<tr>
<td>poss</td>
<td>35</td>
<td>45</td>
</tr>
<tr>
<td>ccomp</td>
<td>46</td>
<td>61</td>
</tr>
<tr>
<td>mark</td>
<td>19</td>
<td>35</td>
</tr>
<tr>
<td>prt</td>
<td>13</td>
<td>29</td>
</tr>
</tbody>
</table>