Symbolic vs. Continuous Features in Transformers: A Digital Communication System's Explanation

Kan Deng
Oleap AI
dengkan.2007@tsinghua.org.cn

Abstract

The term "feature" in mechanistic interpretability is ambiguous, sometimes referring to symbolic properties (e.g., grammatical number), sometimes to neural activations (e.g., basis vectors). We clarify this distinction using communication theory: symbolic features are the *information* being transmitted, while neural features are the *signals* carrying that information. Through a toy transformer implementing subject-verb agreement, we demonstrate how linguistic properties can be encoded as orthogonal basis vectors, transmitted via attention, and decoded for grammatical decisions. This educational distillation provides a communication-theoretic lens for understanding transformer internals, offering conceptual clarity for mechanistic interpretability.

1 Introduction

The term "feature" in mechanistic interpretability is ambiguous: does it mean abstract linguistic properties (singular/plural) or neural activation patterns? Despite discoveries of circuits Nanda et al. [2023], superposition Elhage et al. [2022], and induction heads Olsson et al. [2022], this confusion persists. Communication theory distinguishes information (message) from signals (carrier)—we apply this to transformers.

Transformers' complex behavior emerges from simple operations, like communication systems that transmit digital information through linear filters and routing. The key is **layered abstraction** Welch [1974], Eldar and Mishali [2009]: separating physical signals from logical information. HPSG Pollard and Sag [1994] similarly factorizes grammar through typed feature structures. We build a white-box toy model showing how transformers transmit symbolic linguistic features via orthogonal basis vectors and attention-based routing, providing ground-truth understanding of symbolic processing through continuous computation.

2 Related Work

Mechanistic interpretability reveals structured mechanisms in Transformers: circuits Nanda et al. [2023], superposition Elhage et al. [2022], induction heads Olsson et al. [2022], and the IOI circuit Wang et al. [2023]. The Transformer Circuits thread Elhage et al. [2021], Bricken et al. [2023] has been especially influential in formalizing the QK/OV decomposition and viewing attention as information routing, while the logit lens nostalgebraist [2020] illustrates progressive refinement across layers. Sparse autoencoders Cunningham et al. [2023] and dictionary learning approaches Bricken et al. [2023] recover interpretable directions from superposition, connecting to compressed sensing Donoho [2006], Candès et al. [2006]. Our use of a feature codebook and matched-filter-style notation follows this line of work and is intended as a minimal, self-contained formalization of the standard Transformer computation graph for our setting, rather than a new theoretical framework.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Mechanistic Interpretability Workshop at NeurIPS 2025.

Tracr Lindner et al. [2023] compiles programs to weights using CFG-style structure, which abstracts away most lexical features; HPSG Pollard and Sag [1994] maintains rich typed feature structures at the lexical level. The English Resource Grammar Flickinger [2000] demonstrates the practical scalability of such grammars. Related neurosymbolic and compositional distributional research Baroni and Zamparelli [2010], Polajnar and Clark [2014], Fried et al. [2015], Smolensky and Legendre [2006], Gauthier and Levy [2020] explores how symbolic constraints can be embedded into continuous representations. Our framing is complementary: we focus on the internal transmission of such symbolic features through neural computations rather than external supervision or structured prediction.

Finally, information-theoretic perspectives such as the information bottleneck Tishby et al. [2000] and classical results on coding and correlation Welch [1974], Eldar and Mishali [2009] provide global views on representation efficiency. We adopt a communication-theoretic view primarily as an organizing lens: we stay close to the standard QK/OV and MLP decomposition from prior work, using consistent notation to support our central conceptual distinction between symbolic "information" (features) and the continuous "signals" (activations) that carry it.

3 Model Transformer Computation Graph as Physical Layer

The transformer architecture can be precisely modeled as a communication system's physical layer. Each operation has an exact mathematical equivalence in signal processing, enabling rigorous analysis of information flow.

Signal Modulation. The embedding layer maps discrete tokens to continuous signals. For vocabulary $\mathcal{V}$ and embedding matrix $\mathbf{E} \in \mathbb{R}^{|\mathcal{V}| \times d}$:

$$\mathbf{s}_t = \mathbf{e}_{x_t} + \mathbf{p}_t \in \mathbb{R}^d \tag{1}$$

where $\mathbf{e}_{x_t}$ is the token embedding (codeword) and $\mathbf{p}_t$ is the position encoding (phase). This is identical to digital modulation in communication systems, mapping symbols to signal constellation points.

Attention as Matched Filtering. The QKV mechanism implements signal detection through correlation. For head h:

$$\alpha_{ij}^{(h)} = \frac{\langle \mathbf{W}_Q^{(h)} \mathbf{x}_i, \mathbf{W}_K^{(h)} \mathbf{x}_j \rangle}{\sqrt{d_k}} \tag{2}$$

This is mathematically equivalent to matched filtering, where $\mathbf{W}_K^{(h)}$ defines reference patterns and the dot product performs correlation detection. The QK circuit Elhage et al. [2021] computes pattern matching, while the OV circuit moves information:

$$\mathbf{z}_i = \sum_j \text{softmax}(\alpha_{ij}) \cdot \mathbf{W}_V^{(h)} \mathbf{x}_j \tag{3}$$

where $\mathbf{W}_V^{(h)}$ acts as feature extraction filters (demodulation carriers).

OV map in residual space. For head h, define $\mathbf{v}_j^{(h)} = \mathbf{W}_V^{(h)} \mathbf{x}_j$ and $\mathbf{u}_i^{(h)} = \sum_j a_{ij}^{(h)} \mathbf{v}_j^{(h)}$. Write the output matrix in block form

$$\mathbf{W}_O = [\mathbf{W}_O^{(1)}, \ldots, \mathbf{W}_O^{(H)}], \qquad \mathbf{W}_O^{(h)} \in \mathbb{R}^{d \times d_h}.$$

Then the multi-head attention update at position i is

$$\mathbf{r}_i^{\text{attn}} = \sum_{h=1}^{H} \mathbf{W}_O^{(h)} \mathbf{u}_i^{(h)} = \sum_{h=1}^{H} \sum_j a_{ij}^{(h)} \, \mathbf{W}_O^{(h)} \mathbf{W}_V^{(h)} \mathbf{x}_j, \tag{4}$$

i.e., each head writes back to the shared residual stream through its standard $\mathbf{W}_O^{(h)} \mathbf{W}_V^{(h)}$ composition, while the QK part is captured by $\mathbf{W}_Q^{(h)\top} \mathbf{W}_K^{(h)}$ as in the previous paragraph.

Multi-Head as Frequency Division. The multi-head mechanism divides the d-dimensional signal space into H parallel channels, each operating on a $d_h = d/H$ dimensional subspace. This is

analogous to frequency division multiplexing, where different frequency bands carry independent information streams.

Residual Stream as Communication Bus. The residual connections create a communication path through layers:

$$\mathbf{x}^{(l+1)} = \mathbf{x}^{(l)} + \text{Attention}^{(l)}(\mathbf{x}^{(l)}) + \text{MLP}^{(l)}(\mathbf{x}^{(l)}) \tag{5}$$

This additive structure ensures information from early layers remains accessible, enabling multi-hop communication across the network.

Feature Encoding. For symbolic features, we can use orthogonal encoding when the feature space is small:

$$\mathbf{B} = [\mathbf{b}_1, ..., \mathbf{b}_n] \in \mathbb{R}^{d \times n}, \quad \mathbf{B}^T\mathbf{B} = \mathbf{I}_n \tag{6}$$

Each feature f_i maps to basis vector $\mathbf{b}_i$, guaranteeing perfect recovery via matched filtering. When $n > d$, sparse overcomplete bases enable superposition Elhage et al. [2022], where k-sparse signals with $k \ll n$ can be recovered despite non-orthogonality, as demonstrated by sparse autoencoders Cunningham et al. [2023].

4 Design a Toy Communication Protocol for Subject-Verb Agreement

4.1 Symbolic Linguistic Features for Grammar Analysis

Consider the minimal pair "cat meows" (grammatical) vs. "cat meow" (ungrammatical). To understand how transformers might process this, we distinguish between **symbolic linguistic features** (abstract grammatical properties) and their **neural representations** (directions in activation space).

Context-Free Grammar (CFG): Category-Level Abstraction. A standard CFG captures agreement at the level of nonterminal categories, e.g.,

$$S \to NP_{sg}\, VP_{sg} \quad | \quad NP_{pl}\, VP_{pl} \tag{7}$$

$$NP_{sg} \to \text{cat} \quad VP_{sg} \to \text{meows}. \tag{8}$$

Here "cat meows" is licensed as $S \Rightarrow NP_{sg}\, VP_{sg}$. This illustrates how discrete categories can encode constraints such as singular vs. plural agreement. However, once we move to the derived tree, lexical identity is largely abstracted into category labels: the grammar does not treat fine-grained lexical features as first-class objects in its computational state.

Universal Dependencies (UD): Lexicalized Cross-Lingual Features. Universal Dependencies Nivre et al. [2016] is a mature, widely-used framework that annotates running text with POS tags, morphological features, and labeled dependency relations in a consistent, cross-lingual scheme. For our running example one might obtain:

```
cat --nsubj--> meows     [Number=Sing on both]
```

UD is explicitly lexicalized: features such as `Number`, `Person`, `Tense` live on the tokens themselves, and agreement can be checked by comparing these attributes along dependency edges. In this sense UD gives a clear, operational handle on symbolic features in real corpora. Compared to HPSG, however, UD is primarily an annotation and representation standard: the constraints and algorithms that enforce or exploit these features are deliberately left to downstream parsers or models, rather than being built into a single declarative grammar. The two perspectives are complementary: UD prioritizes uniform, data-driven annotation, while HPSG makes constraints part of the grammar's internal machinery.

HPSG: Lexical Features as First-Class Citizens. Head-driven Phrase Structure Grammar (HPSG) Pollard and Sag [1994] represents each word with a typed feature structure integrating syntactic and semantic information. For illustration, we use a simplified HPSG-style notation:

$$\textit{cat}: \begin{bmatrix} \text{PHON} & \langle \text{cat} \rangle \\ \text{HEAD} & \text{noun} \\ \text{AGR} & [\text{PERS } 3,\ \text{NUM } sg] \\ \text{VAL|SUBJ} & \langle\,\rangle \end{bmatrix} \quad \textit{meows}: \begin{bmatrix} \text{PHON} & \langle \text{meows} \rangle \\ \text{HEAD} & \text{verb} \\ \text{AGR} & [\text{PERS } 3,\ \text{NUM } sg] \\ \text{VAL|SUBJ} & \langle \text{NP}[\text{AGR } [\text{PERS } 3,\ \text{NUM } sg]] \rangle \end{bmatrix} \tag{9}$$

Agreement is enforced by unifying the verb's AGR specification with that of its subject NP inside the grammar itself. The key observation for our purposes is that such attribute–value pairs (e.g., HEAD=verb, NUM=sg) can be treated as **atomic symbolic information** attached to each lexical item. Each token carries multiple such features (POS, NUM, selectional constraints, etc.) that must be transmitted and compared. This naturally maps to a communication picture in which each token broadcasts multiple feature "carriers"—one per linguistic property.

4.2 From Symbolic Features to Neural Transmission

We now design a transformer that implements HPSG-style feature checking through a communication protocol. The key idea is that each token transmits multiple (approximately) orthogonal signals, one for each linguistic feature.

The Codebook–Basis Connection. In mechanistic interpretability, a "basis" is a set of directions in activation space; in communication theory, a "codebook" maps discrete symbols to continuous signals. The embedding matrix can be viewed as a codebook: column i is the codeword for token i, and orthogonal bases are a special case where codewords occupy independent directions.

Multi-Carrier Transmission Protocol. Instead of encoding only token identity (e.g., "cat"), we encode structured feature information as a superposition of carriers living in disjoint subspaces of the model dimension.

Concretely, decompose the embedding space as a direct sum

$$\mathbb{R}^d = \mathbb{R}^{d_{\text{POS}}} \oplus \mathbb{R}^{d_{\text{NUM}}} \oplus \mathbb{R}^{d_{\text{LEX}}},$$

where each block is reserved for one feature type (e.g., POS, number, lexical identity). For each feature value we choose a basis vector supported only in its block; e.g. $\mathbf{b}_{\text{POS=noun}} \in \mathbb{R}^{d_{\text{POS}}}$ (extended with zeros outside the POS block), and similarly for $\mathbf{b}_{\text{NUM=sg}}$, $\mathbf{b}_{\text{LEX=cat}}$, etc.

Stage 1: Modulation (Embedding Layer). Each word is encoded as the sum of its active feature carriers:

$$\mathbf{e}_{\text{cat}} = \mathbf{b}_{\text{POS=noun}} + \mathbf{b}_{\text{NUM=sg}} + \mathbf{b}_{\text{LEX=cat}}, \tag{10}$$

$$\mathbf{e}_{\text{meows}} = \mathbf{b}_{\text{POS=verb}} + \mathbf{b}_{\text{NUM=sg}} + \mathbf{b}_{\text{LEX=meows}}. \tag{11}$$

Because the POS / NUM / LEX blocks are disjoint (and can be chosen orthogonal), the combined embedding uniquely identifies the bundle of symbolic features: changing any feature corresponds to moving along a different block direction.

Message Encoding as Feature Decomposition. Under this view, each token transmits multiple feature channels in parallel: one coordinate in the POS block, one in the NUM block, one in the LEX block, etc. The rest of the transformer can be interpreted as probing and combining these blocks via linear maps and dot products, exactly as in a multi-carrier communication system.

Stage 2: Channel Probing and Routing (Attention). Attention heads act as specialized receivers that probe specific feature channels:

Head 1 (Syntactic Router): This head probes the POS channel to identify subject-verb pairs:

$$\mathbf{W}_Q^{(1)} = [\mathbf{b}_{\text{POS}}, ...] \quad \text{(probe POS channel in query)} \tag{12}$$

$$\mathbf{W}_K^{(1)} = [\mathbf{b}_{\text{POS}}, ...] \quad \text{(probe POS channel in key)} \tag{13}$$

When query extracts "verb" and key extracts "noun", high attention score triggers information routing. The value matrix then extracts and routes the NUM feature:

$$\mathbf{W}_V^{(1)} = [\mathbf{b}_{\text{NUM}}, ...] \quad \text{(extract NUM channel for routing)} \tag{14}$$

Channel Probing as Feature Extraction: Different attention heads probe different feature channels. This is exactly what sparse autoencoders do—they learn basis vectors (matched filters) to extract specific features from superposed representations.

Head 2 (Agreement Checker): This head probes the NUM channel to check compatibility:

$$\mathbf{W}_Q^{(2)} = [\mathbf{b}_{\text{NUM}}, ...] \quad \text{(probe NUM from verb position)} \tag{15}$$

$$\mathbf{W}_K^{(2)} = [\mathbf{b}_{\text{NUM}}, ...] \quad \text{(probe NUM from routed subject)} \tag{16}$$

The dot product $\langle \mathbf{c}_{\text{sg}}, \mathbf{c}_{\text{sg}} \rangle = 1$ indicates agreement (both singular). For "cat meow", we'd get $\langle \mathbf{c}_{\text{sg}}, \mathbf{c}_{\text{pl}} \rangle = 0$ (mismatch).

Routing Protocols for Feature Unification: Attention implements routing based on feature compatibility. When patterns match (e.g., both singular), information flows. When they mismatch, routing is blocked. This implements HPSG unification through neural computation.

Stage 3: Demodulation (MLP + Output). The MLP acts as a matched filter bank, extracting the agreement signal:

$$\mathbf{W}_{\text{MLP}}[i, :] = \mathbf{b}_{\text{AGREE}}^T \quad \Rightarrow \quad \text{activation}_i = \langle \mathbf{b}_{\text{AGREE}}, \text{residual stream} \rangle \tag{17}$$

High activation indicates agreement detected $\rightarrow$ output "grammatical". Low activation indicates mismatch $\rightarrow$ output "ungrammatical".

Superposition and Overcomplete Coding: When we have more features than dimensions ($n > d$), features share directions:

$$\text{Signal} = \sum_{i \in \text{Active}} \alpha_i \mathbf{b}_i, \quad |\text{Active}| \ll n \tag{18}$$

This is superposition—multiple features encoded in overlapping directions. Recovery works when features are sparse (few active per token), as shown by compressed sensing theory.

The transformer doesn't just process tokens—it transmits and processes structured symbolic information through continuous signals. Understanding this distinction between symbolic features (the information) and neural features (the signal carriers) is key to mechanistic interpretability.

5 Limitations

Our toy model uses designed weights, not learned ones, yielding artificially clean protocols. We demonstrate only simple subject-verb agreement, not long-range dependencies or nested structures. We assume perfect orthogonality while real transformers exhibit partial superposition with interference. The model lacks autoregression and the communication framework may not capture all emergent phenomena in large-scale transformers. Despite providing conceptual clarity, empirical investigation remains essential.

6 Conclusion

This educational distillation clarifies a fundamental ambiguity: symbolic features (linguistic properties) are the *information*, while neural features (basis vectors) are the *signals*. Through a communication lens, we show how transformers transmit structured information via orthogonal carriers and routing protocols, bridging digital communication and neural computation for mechanistic interpretability.

References

Marco Baroni and Roberto Zamparelli. Nouns are vectors, adjectives are matrices: Representing adjective–noun constructions in semantic space. In *Proceedings of EMNLP*, pages 1183–1193, 2010.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable directions in language models. *arXiv preprint arXiv:2309.08600*, 2023.

David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

Yonina C Eldar and Moshe Mishali. Robust recovery of signals from a structured union of subspaces. *IEEE Transactions on Information Theory*, 55(11):5302–5316, 2009.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Dan Flickinger. On building a more efficient grammar by exploiting types. In *Natural Language Engineering*, volume 6, pages 15–28. Cambridge University Press, 2000.

Daniel Fried, Tamara Polajnar, and Stephen Clark. Low-rank tensor methods for verbs in compositional distributional semantics. In *Proceedings of ACL*, pages 731–736, 2015.

Jon Gauthier and Roger Levy. Syntax inside neural networks: Quantifying representational structure. In *Proceedings of the 42nd Annual Meeting of the Cognitive Science Society (CogSci)*, pages 1061–1067, 2020.

David Lindner, János Kramár, Matthew Rahtz, Thomas McGrath, and Vladimir Mikulik. Tracr: Compiled transformers as a laboratory for interpretability. *arXiv preprint arXiv:2301.05062*, 2023.

Neel Nanda, Lawrence Chan, Tom Liberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. *arXiv preprint arXiv:2301.05217*, 2023.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of LREC*, 2016.

nostalgebraist. Interpreting gpt: the logit lens. `https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens`, 2020.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

Tamara Polajnar and Stephen Clark. Improving distributional semantic vectors for Compositionality. In *Proceedings of COLING*, pages 1103–1112, 2014.

Carl Pollard and Ivan A Sag. *Head-driven phrase structure grammar*. University of Chicago Press, 1994.

Paul Smolensky and Géraldine Legendre. *The Harmonic Mind: From Neural Computation to Optimality-Theoretic Grammar*, volume 1–2. MIT Press, 2006.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*, 2023.

Lloyd Welch. Lower bounds on the maximum cross correlation of signals. *IEEE Transactions on Information theory*, 20(3):397–399, 1974.