Investigating Representation Geometry in Neural Language Models

Lucia Domenichelli

First year Research Proposal

September 2025

Supervisor Prof. Felice Dell'Orletta

Cosupervisor Prof.sa Dominique Pierina Brunato

Contents

1 Introduction **5**
1.1 Research questions 6

2 Related work and SOTA **8**
2.0.1 Architectural foundations 8
2.0.2 Pretraining objectives 11
2.1 Representation space: from black-box evaluation to mechanistic analyses 12
2.1.1 Probing 13
2.1.2 Mechanistic approach: Sparse Autoencoders 13
2.1.3 Dynamic approach 15
2.1.4 Geometric approach 16
2.2 Representation space in Language Models: what we already know 19

3 Preliminary results **21**
3.0.1 Study 1: From Human Reading to NLM Understanding: Evaluating the Role of Eye-Tracking Data in Encoder-Based Models 21
3.0.2 Study 2: The Role of Eye-Tracking Data in Encoder-Based Models: an In-depth Linguistic Analysis 24
3.0.3 Study 3: Curriculum learning shapes layers geometry 28

List of Tables

3.1 Performance on downstream tasks for different eye-tracking (ET) injection strategies. Shaded cells (🟦) are within 0.02 of the dst-only score for that task. For MNLI we report Matched (m) and MisMatched (mm) separately. Metrics follow GLUE conventions (e.g., Matthews corr. for cola, accuracy for mnli, qnli, rte, wnli, sst-2; combined F1/Acc for mrpc and qqp; Spearman/Pearson for stsb); comp is sentence-complexity prediction (Spearman). 24

3.2 Spearman correlation coefficients between model attention and user attention for the model fine-tuned on predicting user ET features (eye-only) and RoBERTa-base (base). The scores are averaged across all users. Differently from eye-only, only half of users lead to significant correlations with base; the others were excluded from the mean. 24

3.3 Correlations between human attention (*TRT*) and model attention on the **last layer** for each injection strategy. The scores are averaged across all readers. Highlighted cells indicate that the correlation score of the ET injected model exceeds that of dst-only by at least 0.02 points. Bold scores are the highest correlation coefficients: those exceeding 0.27, which is 0.02 points lower than the last-layer correlation of eye-only. 25

List of Figures

2.1 The Transformer architecture [From: Vaswani et al. 2017]. 10

2.2 Residual-stream dataflow within one Transformer block and common extraction points: $\tilde{R}_\ell$, A_ℓ, R_ℓ^{attn}, $\tilde{R}_\ell^{\text{attn}}$, M_ℓ, and the block outputs $R_\ell/R_{\ell+1}$. 11

2.3 Comparison of masked (bidirectional) and causal (autoregressive) language modeling objectives. Arrows indicate the permitted flow of contextual information for predicting the target token. 12

2.4 Probing Transformer internal representations. [From: "Notebook 1.2: Probing" in the XNLM Lab (AILC LCL 2023).] 14

2.5 Topological data analysis methods [From: ICML-2025 Slide Deck by Gardinazzi et al.] 16

2.6 Isotropy in $\mathbb{R}^{1,2,3}$. 19

2.7 Schematic layerwise intrinsic-dimension (ID) profile in encoders. 20

3.1 Eye-tracking data fixations. 22

3.2 Correlation between model attention and human attention (p-value < 0.05). . . . 27

3.3 Attention shift for word length and for UD Parts of Speech. 27

3.4 Attention shift for word position in sentence and distance from syntactic head. . 28

3.5 IsoScore (left) and Linear intrinsic dimensionality (right) of word embeddings from all model layers, before and after fine-tuning, averaged across users. 28

3.6 Isotropy before (left) and after (right) fine-tuning, shown for the 13 most frequent POS classes. 28

3.7 Linear-ID before (left) and after (right) fine- tuning, shown for the 13 most frequent POS classes. 29

3.8 Isotropy before (left) and after (right) fine-tuning, shown for syntax head distance (up to 7 tokens distance). 29

3.9 Linear-ID before (left) and after (right) fine- tuning, shown for syntax head distance (up to 7 tokens dis- tance). 29

3.10 Performance scores (accuracy, F1, and Spearman correlation) on sentiment analysis, POS tagging and complexity. Many seeds were used so results are averaged on each checkpoint. 31

3.11 Geometric scores (Linear ID, IsoScore and TwoNN) at last over checkpoints. . . 32

Chapter 1

Introduction

Natural language processing (NLP) seeks to build algorithms that analyze, understand, and generate human language. Traditional systems depended on hand-engineered features and task-specific models (for example, SVMs and early neural models such as RNNs). Before the current wave of progress, gains in NLP tended to be gradual and tied to specific tasks. That changed with the arrival of the *Transformer* architecture (Vaswani et al. 2017) and large-scale pretraining. In simple terms, Transformers are neural language models that use self-attention to decide which parts of a sequence should influence each other. They can be built with an encoder (which reads text and builds rich contextual representations), a decoder (which generates text one token at a time), or both. Well-known encoder-only models like BERT learn to fill in masked words using context from both sides (Devlin et al. 2019), while decoder-only models like GPT-2 learn to predict the next token from left-to-right. Trained on massive corpora, these models acquire general-purpose representations that can be adapted to many tasks with minimal engineering, and they capture a surprising amount of syntactic and semantic structure (Rogers et al. 2020). This paradigm shift has made the Transformer family the default choice across a wide range of benchmarks and real-world applications, redirecting attention from bespoke feature design to the reuse and adaptation of general-purpose models. The Transformer family has become the default architecture across a wide range of benchmarks and real-world applications, shifting the field's focus from bespoke feature engineering to the reuse and adaptation of general-purpose models. Despite this success and the breadth of applications, substantial research challenges remain. Contemporary models are exceptionally large and hungry for data, compute, and energy, constraints that limit accessibility and raise environmental concerns. Their internal decision processes also remain difficult to explain, which is why they are often described as "black boxes." Ethical and governance questions naturally follow from this scale and opacity; while these are not the focus of this work, we note that the EU AI Act entered into force on 1 August 2024, with key obligations for general-purpose AI set to apply from 2 August 2025 (European Union 2025).

Within this landscape, my thesis concentrates on two themes that currently dominate the discussion: efficiency and interpretability.

- Efficiency is about enabling NLP models to do more, faster, and with fewer resources, whether that means less training data, less compute power, lower memory usage, shorter inference time, or lower energy consumption. The goal is to scale performance not just by making models bigger but by making them smarter in how they learn and operate. Techniques such as model distillation, cascading (using smaller models first and larger ones only when necessary), pruning, quantization, and more efficient attention mechanisms are some ways to push forward.

- Interpretability is concerned with our ability to understand why a model behaves the way it does what internal patterns, representations, or features lead to a certain output. This includes both global interpretability (what the model has learned overall) and local interpretability (why a specific prediction was made). This is especially important when dealing with language. Interpretability helps surface bias, unfair treatment, or errors in under-represented linguistic settings. Surveys of fairness in large language models show how structural or training data biases get amplified and how interpretability tools are needed to diagnose and mitigate such issues.

My research focuses principally on the second point, with the aim of influencing the first as well. Interpretability in large language models (LLMs) refers to the quest for understanding how and why these models produce their outputs. This covers not just what goes in and what comes out, but internal representations, decision pathways, and how linguistic phenomena are encoded. LLMs are often treated as black boxes, but interpretability aims to open them up: to reveal which neurons respond to which features, how layers transform meaning, and how model components interact causally. Research in this area includes surveys of explainability methods that address input-output behaviour, probing and representation analysis to see what information layers register, and mechanistic interpretability, which attempts to trace internal circuits or causal paths in the network to map model behaviour more transparently. A core challenge *faithfulness*: explanations must accurately reflect what the model is actually doing, not just provide plausible narratives. Because LLMs are complex and may mix or "superpose" features, disentangling them in a reliable way remains a difficult problem.

1.1 Research questions

This proposal adopts a *geometric* view of representation learning in large language models. We treat hidden activations as point clouds and study how their shape changes across depth, training, and adaptation. Our analyses rely on compact, basis-independent descriptors that summarize both the effective complexity of a representation set and how broadly it occupies the ambient space. The aim is to link these metrics to modeling choices (objective, architecture, curriculum, supervision), to observable outcomes (transfer, retrieval, calibration), and to interpretable accounts of model computation.

Core research questions

RQ1: How do shape and spread evolve across layers in pretrained encoders and decoders, and is there a consistent trajectory (e.g., expansion, compression)? How do these profiles vary with training objective and model scale?

RQ2: How does representation geometry develop over the course of pretraining, and does ordering the same data by curriculum (rather than at random) change the speed or the shape of consolidation? Can early geometric signals predict downstream performance?

RQ3: How do different fine-tuning strategies reshape geometry across the stack, and are these changes transient or persistent when tasks or domains change?

RQ4: Can geometric properties of representations reliably characterize and distinguish linguistic phenomena (e.g. semantics , syntax, morphology), and are these patterns consistent across languages?

RQ5: When we steer models toward human-like learning (e.g., by incorporating eye-tracking and other cognitive signals during adaptation) how are representation geometry and attention allocation affected? Do these shifts align with improvements in accuracy or calibration?

RQ6: To what extent do global and local geometric properties correlate with downstream usefulness, such as linear transfer, retrieval and clustering quality, and confidence during generation, and can geometry act as an early indicator of improvement or degradation?

List of published works

- *From Human Reading to NLM Understanding: Evaluating the Role of Eye-Tracking Data in Encoder-Based Models* (Luca Dini, Lucia Domenichelli, Dominique Brunato, Felice Dell'Orletta). In Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 17796–17813, Vienna, Austria. Association for Computational Linguistics.

- *The Role of Eye-Tracking Data in Encoder-Based Models: an In-depth Linguistic Analysis* (Lucia Domenichelli, Luca Dini, Dominique Brunato, Felice Dell'Orletta). In Proceedings of the 11th Italian Conference on Computational Linguistics (CLiC-it 2025) (ACCEPTED).

Chapter 2

Related work and SOTA

This chapter introduces three broad perspectives to explore large language models (LLMs): *architecture*, *interpretability*, and the *geometry of embedding spaces*. The goal is to look beyond benchmark scores and give a clear, high-level account of how models are built, how their internal behavior can be examined, and how their learned representations are organized. First, we take a systems view of current LLMs, outlining the main building blocks and how information moves through them. The aim is not to catalog variants, but to establish a common vocabulary for later analyses. Then we summarize practical approaches for relating internal states to model behavior. The focus is on methods that provide testable evidence about what is encoded and how it is used, without committing to a specific technique or task. To conclude, we treat model activations as point sets in a high-dimensional space and describe their global structure using simple geometric descriptors (e.g., how many directions matter and how evenly they are used). This view offers compact summaries that are comparable across models and training regimes.

2.0.1 Architectural foundations

The *Transformer* (see Figure 2.1) is the backbone of modern language models and many multi-modal systems (Vaswani et al. 2017). A Transformer processes a sequence of token embeddings with a stack of identical blocks. Each block has two parts. *Self-attention* mixes information across positions based on content so that a token can consult other tokens in the same sequence. A *feed-forward network* then applies a learned non-linear transformation to each position. Residual connections carry a running representation across depth and normalization keeps scales stable. We will refer to this running representation as the *residual stream*, since it is the pathway where information accumulates layer by layer. Practical models use *multi-head* attention to capture different interaction patterns in parallel, and they inject positional information so the model knows the order of tokens. Decoder-only LMs use a causal mask that prevents access to future tokens and supports left-to-right generation. Encoder–decoder models add cross-attention so the decoder can query encoder states, which is useful for tasks like translation and summarization. Later we detail the block internals and implementation choices, but the basic picture is: attention routes information, feed-forward layers transform it, and the residual stream aggregates it through depth.

Layer normalization and the residual stream Throughout the stack, the state propagated along the residual pathway encodes the model's current working representation for each token. In *pre-norm* layouts, this state is normalized before each sublayer, which improves gradient flow in deep networks. Writing $R_\ell \in \mathbb{R}^{n\times d_{\text{model}}}$ for the residual stream entering layer ℓ, a canonical pre-norm block is

$$\tilde{R}_\ell = \text{Norm}(R_\ell), \quad A_\ell = \text{MHA}(\tilde{R}_\ell), \quad R_\ell^{\text{attn}} = R_\ell + A_\ell,$$

$$\tilde{R}_\ell^{\text{attn}} = \text{Norm}(R_\ell^{\text{attn}}), \quad M_\ell = \text{FFN}(\tilde{R}_\ell^{\text{attn}}), \quad R_{\ell+1} = R_\ell^{\text{attn}} + M_\ell.$$

where MHA stands for Multi-Head Attention and FFN for Feed-Forward Network. Layer normalization is applied per token along the feature dimension: for $x \in \mathbb{R}^{d_{\text{model}}}$,

$$\text{LayerNorm}(x) = g \odot \frac{x - \mu(x)}{\sqrt{\sigma^2(x) + \varepsilon}} + b,$$

with learned gain g and bias b. RMSNorm variants omit mean subtraction and rescale by the root-mean-square, $\text{RMSNorm}(x) = g \odot \frac{x}{\sqrt{\frac{1}{d_{\text{model}}}\|x\|_2^2 + \varepsilon}}$. In all cases, the residual stream remains the single carrier of information across depth; attention and the FFN *read* from its normalized form and *write* updates back via residual addition.

Attention Given a sequence $X \in \mathbb{R}^{n\times d_{\text{model}}}$, learned projections produce queries, keys, and values: $Q = XW_Q$, $K = XW_K$, $V = XW_V$, with $W_Q, W_K \in \mathbb{R}^{d_{\text{model}}\times d_k}$ and $W_V \in \mathbb{R}^{d_{\text{model}}\times d_v}$. Scaled dot-product attention computes a content-addressed read from V using similarities between Q and K:

$$\text{Attention}(Q, K, V) = \text{Softmax}\Big(\frac{QK^\top + M}{\sqrt{d_k}}\Big)V,$$

where $M \in \mathbb{R}^{n\times n}$ encodes padding or causal masks (disallowing attention to future positions in decoder-only LMs). The softmax acts row-wise so that each query forms a distribution over keys, the resulting weights yield a convex combination of value vectors and an output in $\mathbb{R}^{n\times d_v}$.

Multi-head and block structure. Rather than a single map, MHA runs h heads with separate projections $\{W_Q^{(i)}, W_K^{(i)}, W_V^{(i)}\}_{i=1}^h$, their outputs are concatenated and linearly projected:

$$\text{MHA}(X) = \text{Concat}\big(\text{Attention}(XW_Q^{(i)}, XW_K^{(i)}, XW_V^{(i)})\big)_{i=1}^h W_O,$$

with $W_O \in \mathbb{R}^{(h\,d_v)\times d_{\text{model}}}$. The FFN applies a two-layer MLP with expansion factor $d_{\text{ff}} \gg d_{\text{model}}$ and a gated or smooth nonlinearity (e.g., GELU or SwiGLU), independently at each position. In decoder-only LLMs, causal masking enforces left-to-right factorization, encoder–decoder models additionally include cross-attention in the decoder to access encoder states.

Positional information and efficiency variants Because attention is permutation-equivariant, position must be injected explicitly, commonly via absolute or relative encodings, or rotary position embeddings (RoPE) that rotate Q, K in a position-dependent manner and help length extrapolation. Efficiency-oriented refinements adjust this template without changing the core dataflow: multi-query or grouped attention reduces memory bandwidth at decode time by sharing K, V across heads, sparse or block-local patterns trade global context for subquadratic complexity, mixture-of-experts (MoE) layers route tokens through a small subset of expert FFNs to increase capacity at roughly constant per-token FLOPs (Shazeer et al. 2017), optimized kernels such as FlashAttention minimize memory traffic via tiling while preserving exact results (Dao et al. 2022).

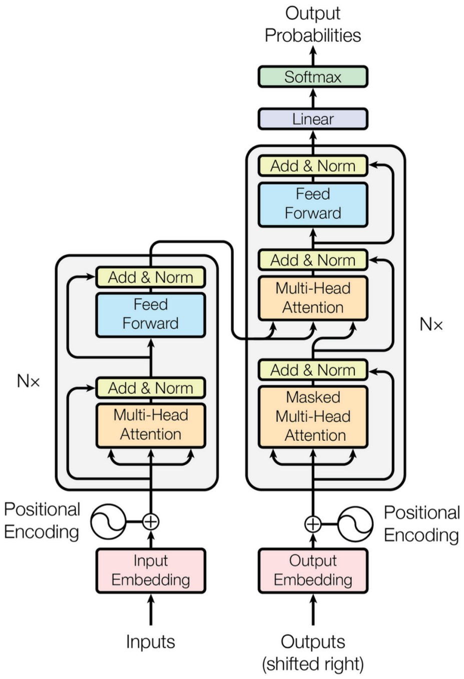

Figure 1: The Transformer - model architecture.

Figure 2.1: The Transformer architecture [From: Vaswani et al. 2017].

Hidden states and where to extract them With the residual-stream perspective in place, it is useful to make the extraction sites explicit, as they correspond to distinct semantics. Let R_0 denote the input embeddings (after positional encoding). At layer ℓ, the pre-attention normalized state $\tilde{R}_\ell = \text{Norm}(R_\ell)$ is the *input* seen by the attention mechanism, its content reflects what the model has computed up to that depth, before any new contextual mixing. The attention *write* A_ℓ (after output projection) encodes context-dependent updates assembled by comparing queries and keys, the updated residual $R_\ell^{\text{attn}} = R_\ell + A_\ell$ aggregates all contributions up to and including attention at layer ℓ. The pre-FFN normalized state $\tilde{R}_\ell^{\text{attn}} = \text{Norm}(R_\ell^{\text{attn}})$ is then consumed by the MLP, whose *write* M_ℓ introduces new nonlinear features, the block output $R_{\ell+1} = R_\ell^{\text{attn}} + M_\ell$ is the natural choice when a single per-layer representation is desired. A representation of this ”information highway” can be seen in Figure 2.2.

In encoder models, sentence-level vectors are often formed from the final residual stream by pooling (e.g., mean or attention pooling) or by selecting a designated token, in decoder-only LMs, token-level states at the final position align with next-token prediction, while span- or

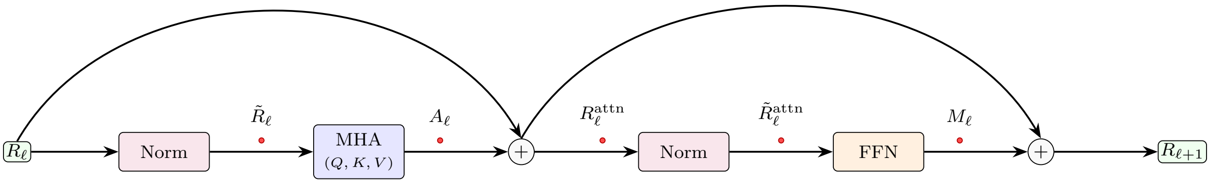

Figure 2.2: Residual-stream dataflow within one Transformer block and common extraction points: $\tilde{R}_\ell$, A_ℓ, R_ℓ^{attn}, $\tilde{R}_\ell^{\text{attn}}$, M_ℓ, and the block outputs $R_\ell/R_{\ell+1}$.

sentence-level embeddings are typically obtained by pooling across positions or by extracting intermediate layers to trade off semantic abstraction and syntactic detail. For analyses targeting specific computations, finer-grained options such as per-head outputs before concatenation, or the Q, K, V projections themselves can be informative, but the residual stream remains the canonical locus for "hidden states," since it is the sole pathway that persists across depth and accumulates all writes.

2.0.2 Pretraining objectives

Modern language models share the common backbone of the *Transformer*, but the *pretraining objective* determines what their internal representations come to encode. Formally, given a corpus $\mathcal{D}$, a corruption/conditioning operator $\mathcal{C}$, and parameters θ, pretraining minimizes:

$$\mathcal{L}(\theta) \;=\; \mathbb{E}_{x\sim\mathcal{D}}\big[\; \ell\big(f_\theta(\mathcal{C}(x)),\, x\big) \;\big],$$

so the hidden states $\{\phi_\ell(x;\theta)\}$ are pressured to become *sufficient statistics* for predicting the targets specified by $\mathcal{L}$. In this sense, the objective, not the architecture, primarily *defines the representations*: it decides which information is preserved, which invariances are preferred, and which contextual dependencies are emphasized. The Transformer itself is agnostic and can serve as encoder, decoder, or encoder–decoder, what changes the model's behavior, and the semantics of its hidden states, is the pretraining objective that defines the learning signal.

Let $x = (w_1, \dots, w_T)$ denote a text sequence drawn from the data distribution $\mathcal{D}$, and let p_θ be a parametric model. Three objective families dominate.

(A) Masked language modeling (encoder-only) Mask a random subset $M \subset \{1, \dots, T\}$ and predict only the masked tokens from their bidirectional context:

$$\max_\theta \;\mathbb{E}_x\, \mathbb{E}_M\Big[\sum_{t\in M} \log p_\theta\big(w_t \mid x_{\setminus M}\big)\Big].$$

This yields bidirectional (Figure 2.3 left) contextual encoders. RoBERTa optimizes the same loss with dynamic masking and larger corpora, SpanBERT extends masking to contiguous spans. For intrinsic scoring with MLMs, a common surrogate is the *pseudo-log-likelihood* $\mathrm{PLL}(x) = \sum_t \log p_\theta\big(w_t \mid x_{\setminus t}\big)$.

(B) Autoregressive / causal language modeling (decoder-only) Factorize by the chain rule and maximize the causal log-likelihood:

$$\max_{\theta}\ \mathbb{E}_{x\sim\mathcal{D}}\Big[\log p_\theta(x)\Big]\ =\ \mathbb{E}_x\Big[\sum_{t=1}^{T}\log p_\theta(w_t \mid w_{<t})\Big].$$

This induces a left-to-right conditioning structure and directly supports open-ended generation and perplexity evaluation (Figure 2.3 right). Scaling this objective underpins GPT-style models.

(C) Denoising autoencoder pretraining (encoder–decoder) Use an encoder–decoder and predict the clean target from a corrupted source:

$$\max_{\theta}\ \mathbb{E}_{x\sim\mathcal{D}}\,\mathbb{E}_{\tilde{x}\sim q(\cdot\mid x)}\Big[\log p_\theta\big(x \mid \tilde{x}\big)\Big],$$

where $q(\tilde{x} \mid x)$ is a corruption process. Prominent instantiations differ in q and the target: T5 uses span corruption with sentinel tokens (text-to-text), BART mixes token masking, deletion, permutation, and sentence shuffling, MASS masks a contiguous fragment on the encoder side and predicts it with the decoder. These objectives excel for instruction-following and generation with explicit conditioning.

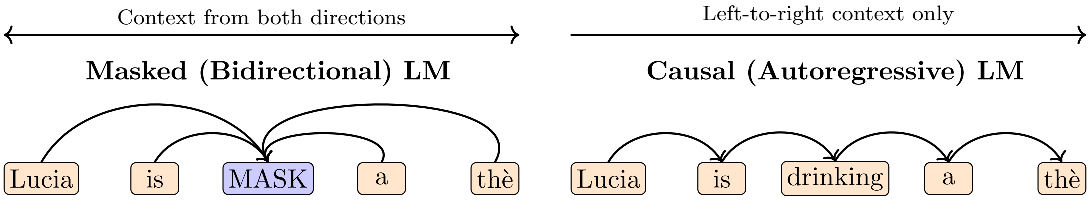
Context from both directions
Masked (Bidirectional) LM
Lucia
is
MASK
a
thè
Left-to-right context only
Causal (Autoregressive) LM
Lucia
is
drinking
a
thè

Figure 2.3: Comparison of masked (bidirectional) and causal (autoregressive) language modeling objectives. Arrows indicate the permitted flow of contextual information for predicting the target token.

2.1 Representation space: from black-box evaluation to mechanistic analyses

The great performance and opacity of the internal computations of Language Models have driven a wide range of interpretability methodologies. At one end are black-box evaluations that systematically benchmark models without assuming access to inner mechanisms (Liang et al. 2022). At the other end, mechanistic studies attempt to open the box and localize computations in specific substructures, including causal circuit discovery and interventions that edit or trace factual associations (Meng et al. 2022; Conmy et al. 2023; Geva et al. 2021; Ferrando et al. 2024). Between these extremes lies a geometric perspective that treats activations as point clouds and analyzes their structure across layers, which is the one we decided to pursue.

2.1.1 Probing

Probing provides a high-level interface for inspecting what information is present in a model's internal states without changing the model itself (see Figure 2.4). The procedure is to freeze the language model, extract hidden representations at a chosen layer and position, and train a small auxiliary predictor to recover a target variable, for example part of speech, dependency distance, sentence polarity (Miaschi and Dell'Orletta 2020). If the predictor succeeds under controlled capacity, the representation is said to *encode* that variable (Alain and Bengio 2016; Belinkov 2022). A probe measures *decodability* from a given representation, not whether the model *uses* that information during inference, availability is not causality. Stronger claims require pairing probes with targeted interventions or ablations.

Two common designs are *linear* probes, which test whether a variable is recoverable by an affine map, and *structural* probes, which ask whether a structured object such as a dependency tree or distances in a latent metric can be recovered by a simple transformation (Hewitt and Manning 2019). An Information-theoretic framings quantify how much information about a variable is present in the representation and use minimum-description-length criteria to balance fit against probe complexity (Pimentel et al. 2020; Voita and Titov 2020). Good practice fixes the extraction protocol (layer, token position or pooling rule, and normalization), constrains probe capacity and regularization, and reports *selectivity* using control tasks that are learnable by the probe but not supported by the representation (Hewitt and Liang 2019; Belinkov 2022). Linear probes offer conservative, comparable tests across layers and models. Shallow non-linear probes can reveal information that is present but not linearly aligned, although higher capacity increases overfitting risk and weakens interpretability.

Probe outcomes depend on the geometry of the embedding space. Global anisotropy can inflate cosine-based scores unless states are centered or dominant directions are removed, local structure often looks more uniform than global summaries suggest (Ethayarajh 2019a; Cai et al. 2021). Conversely, low effective dimensionality can coincide with easier decodability for variables that dominate the available degrees of freedom (Ansuini et al. 2019). For comparability, it is helpful to fix normalization, document isotropy and dimensionality summaries, and interpret probe results in that context (Rudman et al. 2022a; Belinkov 2022).

2.1.2 Mechanistic approach: Sparse Autoencoders

The starting point is the *Linear representation hypothesis*: many high-level variables that matter for model behavior are encoded as approximately linear directions or low-dimensional subspaces inside the residual stream. Classic evidence came from linear regularities in word vectors and from interventions that add or subtract activation directions to steer model outputs. More recent analyses of *superposition* show why single neurons often mix many concepts and why we should expect features to live in sparse directions spread across neurons rather than in individual units. At the same time, there is growing evidence that some variables are only partly linear and may require small subspaces rather than a single direction. These observations motivate a tool that can make such sparse directions explicit and test them with causal interventions (Park

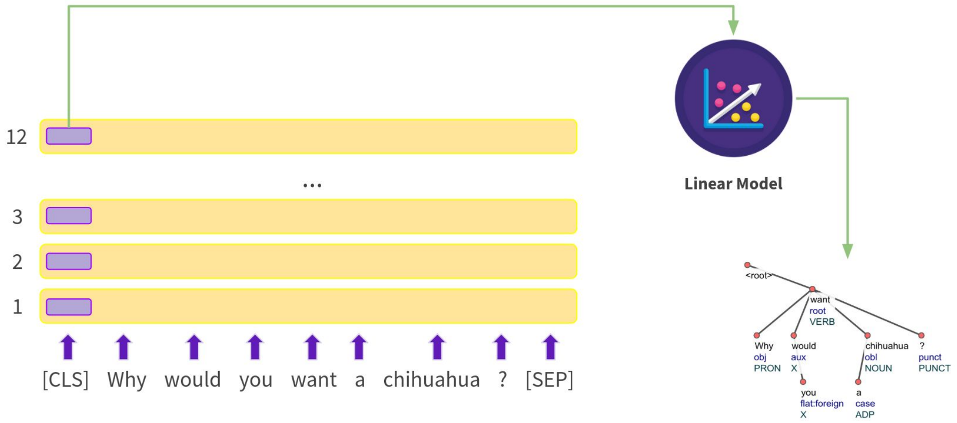

Figure 2.4: Probing Transformer internal representations. [From: "Notebook 1.2: Probing" in the XNLM Lab (AILC LCL 2023).]

et al. 2023; Elhage et al. 2022).

A *sparse autoencoder* (SAE) is a simple way to expose this structure. Given activations $h_{\ell,t}(x) \in \mathbb{R}^d$ from a chosen layer ℓ and token position t, an SAE learns an overcomplete dictionary $D \in \mathbb{R}^{d \times m}$ and sparse codes $a(x) \in \mathbb{R}^m$ such that:

$$h_{\ell,t}(x) \approx D\,a(x), \qquad a(x) = s\big(W\,h_{\ell,t}(x) + b\big),$$

with a sparsifying nonlinearity $s(\cdot)$ that makes only a few latent features active for each input.

In practice, each latent often fires on a coherent pattern in text, such as a particular morphology, a style marker, or an entity class. This factorization turns an opaque vector into a short list of human-checkable features and gives a direct handle for *causal* tests: we can ablate a feature by zeroing its latent, or add it by injecting De_i scaled by a small coefficient, then measure the effect on the model's predictions. Large-scale studies report that as the dictionary grows and training is tuned, more latents become monosemantic and the decomposition captures a larger share of the original signal (Cunningham et al. 2023; Templeton et al. 2024).

There are several practical choices. *Where to tap* the model matters: many works train SAEs on the pre-normalized residual stream or on block writes such as MLP outputs, since these loci aggregate information that is passed forward. *How to enforce sparsity* also matters. Early SAEs used ℓ_1 penalties or fixed k-sparse encoders ($TopK$). Newer variants separate "which features fire" from "by how much" using *Gated SAEs*, introduce discontinuous activations to better match ℓ_0 sparsity with $JumpReLU$, or let the effective k vary across examples with $BatchTopK$. These changes improve the reconstruction-sparsity trade-off and reduce pathologies such as *shrinkage* or dead latents. In practice one monitors both reconstruction quality and sparsity, and inspects feature quality with automated labeling or small human audits (Rajamanoharan et al. 2024; Bussmann et al. 2024).

Scaling studies show that SAEs can recover thousands of human-legible features from large LLMs when trained on diverse text and with careful engineering. Empirically, feature quality improves smoothly with model size, dataset size, and dictionary width, which gives practical guidance for allocating compute. Open-source tooling now makes it feasible to train and audit SAEs, and tutorials demonstrate how to combine them with standard causal analyses such as activation patching (Team 2024). SAE features are not only descriptive. They support precise *steering* at inference time: one can compute "steering vectors" from contrastive data and add them to activations to increase or decrease a behavior, or select specific SAE latents that correspond to the behavior and modulate only those. These interventions can change style, reduce hallucinations on some tasks, or amplify a targeted capability while preserving overall fluency. Such experiments also act as mechanistic tests of whether a feature is causally involved in the behavior of interest (Panickssery et al. 2023).

Two caveats are important. First, not every variable of interest is perfectly linear, some appear to require small subspaces or exhibit context-dependent composition. Second, SAEs do not necessarily produce a unique "canonical" set of atomic features. Different training runs or larger dictionaries can split or refine existing latents while improving reconstruction. These are reasons to pair SAEs with causal evaluations and to report sensitivity to training choices, rather than reasons to discard the method (Engels et al. 2024).

2.1.3 Dynamic approach

To move beyond *static* snapshots of internal representations, we introduce two complementary *dynamic* perspectives that reveal and quantify how geometry evolves across the network.

(i) Topological data analysis Topological Data Analysis (TDA) provides robust, scale-aware descriptors of shape. Its core tool, persistent homology, records the birth and death of connected components, cycles, and higher-dimensional cavities along a filtration $\{K_\nu\}_{\nu\geq 0}$ with $\nu_1 \leq \nu_2 \Rightarrow K_{\nu_1} \subseteq K_{\nu_2}$ (e.g., the Vietoris–Rips filtration built by connecting points within radius ν), see Figure 2.5. Zigzag persistence extends this framework to sequences where inclusions may reverse, making it well-suited for representations that change over time or across network layers and track the formation and destruction of topological features through the model and to quantify their persistence statistics, yielding a concise characterization of the transformations in high-dimensional space (Gardinazzi et al. 2024; Uchendu and Le 2024).

(ii) A particle-dynamics lens One can see the set of token representations at each layer as a cloud of points that moves through the embedding space as the model processes the input. Self-attention behaves like diffusion that mixes information across tokens, while the feed-forward block behaves like a drift that nudges each token along learned directions. A compact summary of the cloud at a given layer is the matrix of pairwise similarities between tokens, tracking this summary through depth reveals how geometry changes across the network. In very wide models, theory shows that these statistics become predictable and evolve according to low-dimensional dynamics, which motivates treating depth as a discrete time axis and connects Transformers

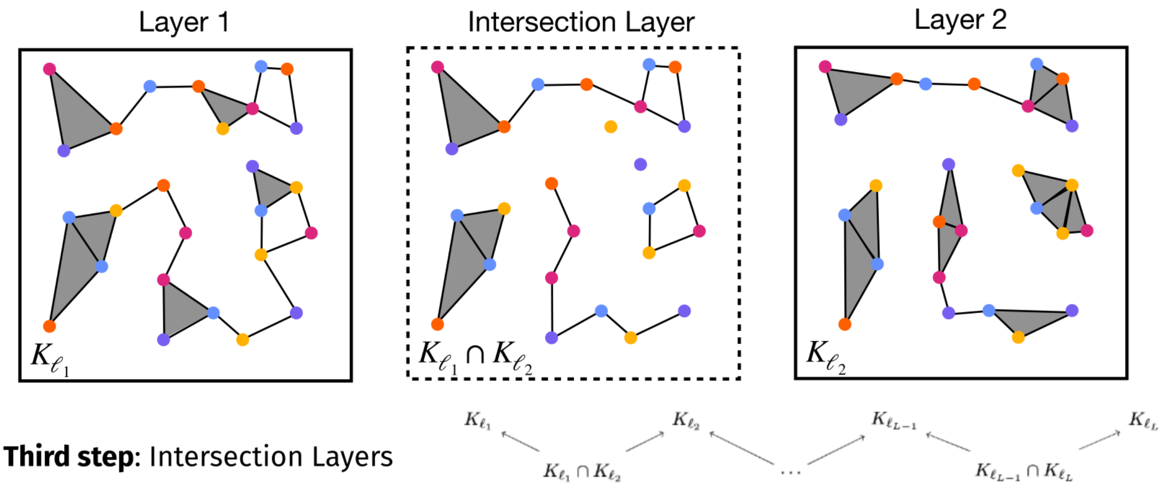
Layer 1
Intersection Layer
Layer 2
K_{ℓ_1}
$K_{\ell_1} \cap K_{\ell_2}$
K_{ℓ_2}
Third step: Intersection Layers
K_{ℓ_1}
K_{ℓ_2}
$K_{\ell_{L-1}}$
K_{ℓ_L}
$K_{\ell_1} \cap K_{\ell_2}$
…
$K_{\ell_{L-1}} \cap K_{\ell_L}$

Figure 2.5: Topological data analysis methods [From: ICML-2025 Slide Deck by Gardinazzi et al.]

to continuous-depth formulations based on neural ordinary differential equations (Schoenholz et al. 2016; R. T. Chen et al. 2018). This view also predicts a stable versus chaotic boundary: in the stable regime nearby inputs become more similar across layers, while in the chaotic regime small differences grow.

2.1.4 Geometric approach

We have deliberately focused our initial studies on the geometric perspective of interpretability, and this remains an ongoing effort. In §2.2 we review in greater depth the relevant literature to clarify what has been done and what remains open

A useful starting point is the *Manifold Hypothesis*: learned activations occupy structured, low-dimensional subsets of the ambient space. Together with the Johnson–Lindenstrauss (JL) lemma, which guarantees that pairwise distances among a finite set of points can be approximately preserved under random projections to a much lower dimension, this motivates estimating the *intrinsic dimensionality* (ID) of representation sets and, jointly, characterizing how evenly those sets use the available directions in space, *isotropy*. ID summarizes the effective degrees of freedom in the activations, while an isotropy index summarizes directional balance.

We start by talking about intrinsic dimensionality.

Intrinsic dimensionality

Formally, a point cloud $H = \{h_i\}_{i=1}^N \subset \mathbb{R}^D$ has intrinsic dimension d if it lies on (or sufficiently near) a d-dimensional manifold $\mathcal{M} \subset \mathbb{R}^D$. Two complementary families of estimators are common:

- Linear intrinsic dimensionality estimators
- Non-linear intrinsic dimensionality estimators

We now briefly introduce both of them.

(A) Linear ID estimators. These estimators look only at the covariance spectrum of H. Two widely used summaries are:

$$d_{\mathrm{PR}} \;=\; \tfrac{(\mathrm{tr}\, C)^2}{\mathrm{tr}\,(C^2)} \qquad \text{and} \qquad d_\tau \;=\; \min\{k: \; \textstyle\sum_{j=1}^{k} \lambda_j / \sum_{j=1}^{D} \lambda_j \geq \tau\},$$

where C is the sample covariance and $\lambda_1 \geq \ldots \geq \lambda_D$ its eigenvalues. The first is the *participation ratio* (effective rank), the second is the PCA "variance-explained at level τ" . These are easy to compute and rotation invariant, but they only capture *linear* structure: curved manifolds with broad spectra can have large d_{PR} even when their *nonlinear* ID is small. In practice, we report a linear ID side-by-side with a nonlinear estimator to separate spectral spread from manifold dimension.

(B) Nonlinear, neighborhood-based ID estimators. These estimators use local geometry, typically k-nearest-neighbor (kNN) distances and angles, and remain valid for curved manifolds. They assume local homogeneity at the scale of the kth neighbor and are sensitive to the choice of k. Below we detail the three estimators we will use: TwoNN, GRIDE, and a maximum-likelihood kNN estimator; we also mention an angle-based alternative.

TwoNN (Two Nearest Neighbors) For each point h_i, let $r_{i,1}$ and $r_{i,2}$ be the distances to the first and second nearest neighbors and define the ratio $\mu_i = r_{i,2}/r_{i,1}$. Under a locally homogeneous Poisson model on a d-dimensional manifold, μ_i follows a Pareto distribution with shape parameter d. TwoNN estimates d by fitting a straight line to the transformed empirical CDF:

$$\log\bigl(1 - \tilde{F}(\mu_{(i)})\bigr) \;=\; -d \,\log \mu_{(i)} \quad \text{(no intercept)},$$

often discarding the largest few percent of ratios to reduce boundary effects. TwoNN is fast, scale-aware, and surprisingly robust for moderate d, with finite samples it tends to *underestimate* high dimensions and therefore is best treated as a lower bound beyond $d \approx 20$. In applied work one typically (a) repeats the estimate over several random subsamples ("decimation," a proxy for checking scale-invariance) and (b) reports a median and a bootstrap interval. This estimator has been used to map layerwise ID profiles in deep networks and to connect the location of "compression valleys" to generalization behavior (Facco et al. 2017; Ansuini et al. 2019).

GRIDE (Generalized Ratios ID Estimator) GRIDE extends TwoNN by using ratios of *generic* neighbor orders, $\dot{\mu}_i = r_{i,n_2}/r_{i,n_1}$ with $1 < n_1 < n_2$. Under the same local Poisson assumptions, closed-form likelihoods are available for $\dot{\mu}_i$, which gives a maximum-likelihood estimate of d and exact confidence intervals. Using higher-order neighbors makes the estimator less sensitive to small-scale noise and allows a principled exploration of how ID changes with neighborhood scale, without throwing away most points as in decimation. In practice we scan a small grid of (n_1, n_2), check stability across scales, and report the value on the earliest plateau (Denti et al. 2022).

MLE kNN estimator (Levina–Bickel) A classic alternative uses the log-ratios of the first k neighbor distances. For each point:

$$\hat{d}_i^{-1} = \frac{1}{k-1} \sum_{j=1}^{k-1} \log \frac{r_{i,k}}{r_{i,j}}, \qquad \hat{d} = \frac{1}{N} \sum_{i=1}^{N} \hat{d}_i.$$

Small k reduces curvature bias but increases variance; k in the range $10-20$ is common, with sensitivity checks across a grid. Weighted variants and bias corrections exist, and asymptotic standard errors can be derived from the MLE. We use this estimator as a secondary check on TwoNN/GRIDE, not as a sole source (Levina and Bickel 2004).

Angle + norm estimator (DANCo) DANCo combines two statistics whose distributions depend on d: the concentration of inter-point angles and the distribution of neighbor radii. It fits d by matching both. DANCo is comparatively robust to curvature and moderate density variation but is more computationally demanding and can be sensitive to high-dimensional noise without careful neighbor selection. We use it to corroborate results on difficult layers or when spectra are very flat. (Ceruti et al. 2012)

Local ID (LID) In some analyses the focus is on the *local* complexity around a point or within a class-conditioned subset. LID formalizes the tail rate of the distance distribution and can be estimated from kNN distances. Local estimates help explain heterogeneity across tokens or labels and are useful for detection tasks, but they are noisier and more sensitive to sampling (Houle et al. 2018) .

Because linear spectra and nonlinear neighborhood geometry answer different questions, we think it is important to report both. Linear ID (participation ratio and variance-explained counts) tracks how *spread out* variance is across directions, while TwoNN/GRIDE/DANCo quantify manifold degrees of freedom. Recent analyses emphasize that the two can diverge systematically during pretraining and fine-tuning, with "linear" dimensionality reflecting superficial or form-like factors and "nonlinear" ID tracking semantic or task-useful complexity (Ansuini et al. 2019).

Isotropy

We now introduce the concept of *isotropy*.

Let $C = \frac{1}{N} \sum_i (h_i - \bar{h})(h_i - \bar{h})^\top$ be the covariance of centered, optionally length-normalized vectors. A representation set is *second-order isotropic* when $C \propto I$, i.e., variance is uniformly distributed across directions (see Figure 2.6). Contextual embeddings in popular LMs are typically *anisotropic*: tokens occupy a narrow cone, which inflates average cosines and can harm similarity search unless post-processed (Ethayarajh 2019a; Mu et al. 2017). The metric *IsoScore* (Rudman et al. 2022a) quantifies how uniformly a point cloud uses the ambient space by comparing the spectrum of C to the spectrum of an ideal isotropic cloud. It is designed to be rotation, mean, and scale invariant, and to remain stable on mini-batches. In contrast to earlier proxies such as average random cosine or the *partition score* (which can be brittle and non-invariant), IsoScore behaves predictably under controlled tests and correlates with true space utilization. We standardize on IsoScore as our primary isotropy measure and report it together with linear ID so that “how many directions” and “how evenly those directions are used” are both visible.

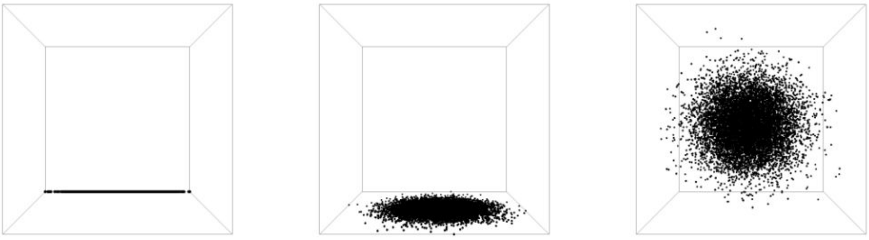

Figure 2.6: Isotropy in $\mathbb{R}^{1,2,3}$.

2.2 Representation space in Language Models: what we already know

As already introduced, two global descriptors are particularly informative for understanding how embedding spaces are organized in large language models (LLMs): the *intrinsic dimensionality* (ID) of activation sets and their *isotropy*. Taken together, they provide model-agnostic summaries of compression, specialization, and the balance between shared and cluster-specific structure. These descriptors were first applied to latent spaces in computer vision and molecular modeling, and more recently to textual embeddings. Most studies analyze sentence-level representations extracted layer by layer, typically via a designated sentence token or by mean-pooling token vectors. In what follows, we review early work that employs these descriptors for language models and summarize the main findings.

Isotropy: what we know Early analyses showed that contextual embeddings produced by ELMo, BERT, and GPT-2 are strongly *anisotropic*, concentrating in narrow cones, this degrades cosine-based similarity and motivates debiasing or re-centering before evaluation (Ethayarajh 2019b). Follow-up work refined this picture. Cai et al. (Cai et al. 2021) showed that conditioning on senses or clusters yields more balanced local geometry than global summaries suggest. IsoScore was introduced as a rotation and scale robust measure of uniform space utilization, addressing weaknesses of average *random cosine* and partition scores (Rudman et al. 2022b). Importantly, isotropy relates to *linguistic* use cases: post-processing that increases isotropy improves unsupervised sentence representations and semantic textual similarity (B. Li et al. 2020), while in multilingual settings both the extent and *location* of anisotropy matter for cross-lingual alignment and transfer, and can change with fine-tuning (Rajaee and Pilehvar 2021). Recent studies debate whether anisotropy is inherent to transformers or contingent on training choices

(Xie et al. 2024; Machina et al. 2024), this nuance is relevant when we interpret geometry shifts induced by domain adaptation or instruction tuning.

Intrinsic dimensionality: what we know Across architectures and domains, trained networks tend to produce activations with ID far below the ambient width, and ID varies systematically with depth. In vision and text models alike, layerwise ID often follows an *expand–compress* trajectory (Figure 2.7): an early rise associated with feature mixing, followed by a compression valley and, sometimes, a late consolidation phase (Ansuini et al. 2019; Valeriani et al. 2023). For LLMs, Cheng et al (Cheng et al. 2024) reported the emergence of a high-dimensional "abstraction" phase that correlates with improved linguistic capability under continued pretraining. Beyond description, ID has practical value: local ID of activations can distinguish human from model-generated text in a model-agnostic fashion (Tulchinskii et al. 2023), and it correlates with factuality during generation, with lower local ID neighborhoods associated with more truthful answers (Yin et al. 2024). ID has also explained why fine-tuning is effective in low-data regimes: many NLP tasks lie in low-dimensional subspaces of the parameter space, which helps account for the success of parameter-efficient adaptation (Aghajanyan et al. 2021).

Implications for linguistic analysis Geometry helps answer linguistic questions without committing to a specific probe architecture. Isotropy affects cosine-based lexical and sentence semantics, so centering and debiasing become part of fair evaluation (Ethayarajh 2019b; B. Li et al. 2020). Layerwise ID trajectories align with the intuition that lower layers encode local form while mid layers consolidate abstract structure (Ansuini et al. 2019; Valeriani et al. 2023). In multilingual models, geometry differences across languages can be quantified and, in some cases, aligned, which supports stronger zero-shot transfer (Rajaee and Pilehvar 2021). Finally, local ID during generation offers a complementary confidence signal for question answering and summarization (Yin et al. 2024), connecting representational complexity to observable behavior.

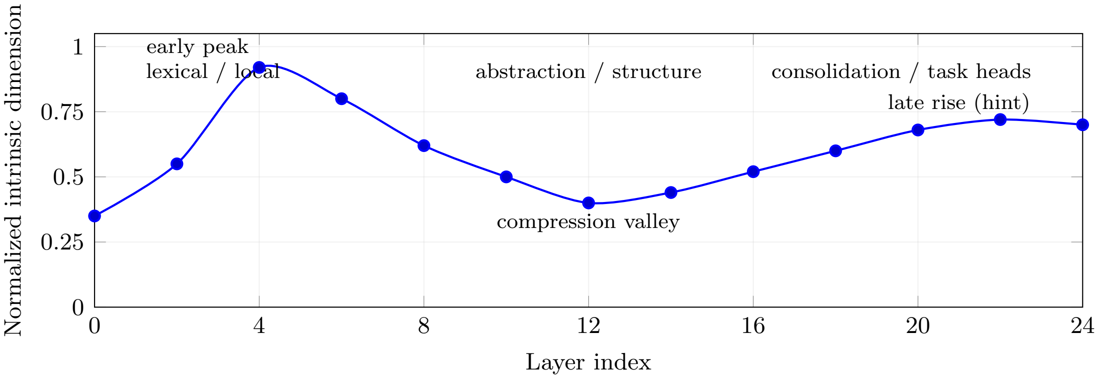

Figure 2.7: Schematic layerwise intrinsic-dimension (ID) profile in encoders.

Chapter 3

Preliminary results

This chapter presents the work, both published and unpublished, developed this year as part of a broader effort to bring Transformer models closer to how human learning. Guided by human-inspired methods, including the use of eye-tracking data, we investigate the geometry of representations in neural language models. Using the metrics introduced in §2.1.4, we study how embedding spaces evolve and reorganize during training and adaptation. We describe three studies. In the *first*, we analyze a pretrained model, comparing its embeddings before and after multiple fine-tuning procedures and under alternative pretraining objectives. In the *second*, we take a cross-lingual view, examining manifolds for linguistic classes in Italian and English and showing that these classes differ systematically under geometric descriptors. In the *third*, we examine how representational geometry emerges during pretraining by training models on the same corpus in different orders and tracking how these permutations shape the trajectories of the learned spaces.

3.0.1 Study 1: From Human Reading to NLM Understanding: Evaluating the Role of Eye-Tracking Data in Encoder-Based Models

This study (Dini, Domenichelli, Brunato, and Dell'Orletta 2025) investigates whether injecting *eye-tracking* (ET) information into an encoder-only language model changes (i) downstream task performance, (ii) the alignment between model attention and human gaze, and (iii) the *geometry* of the model's embedding space. Using RoBERTa-base, we compare four ET–injection strategies (intermediate fine-tuning with several "how many layers to unfreeze" variants, LoRA adapters, interleaved multi-task learning, and multi-task learning with silver ET labels) against strong baselines. We report that ET supervision preserves or only marginally affects task performance, improves attention–gaze correlation, and *compresses* the representation space (lower intrinsic dimensionality and lower isotropy). Before presenting these results, we introduce the concept of eye-tracking data.

Eye-tracking data Eye tracking measures fixations and saccades during reading (Figure 3.1), offering behavioural signals that correlate with linguistic difficulty and processing dynamics (Demberg and Keller 2008). While high-quality data have traditionally required laboratory eye-trackers, recent work demonstrates practical gaze estimation with commodity cameras (Papoutsaki 2015), and dedicated devices are becoming more accessible (San Agustin et al. 2010; D. Li et al. 2006). Several publicly available, annotated corpora facilitate NLP research with gaze features, including the Dundee Corpus Kennedy et al. 2013, GECO (Cop et al. 2017), ZuCo (Hollenstein et al. 2018), and MECO L1 (Siegelman et al. 2022).

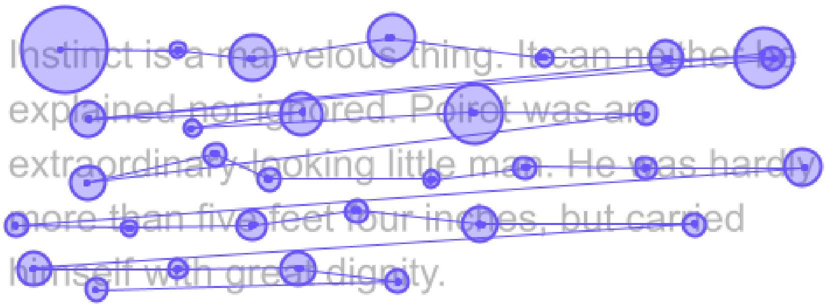
Instinct is a marvelous thing. It can neither be
explained nor ignored. Poirot was an
extraordinary-looking little man. He was hardly
more than five feet four inches, but carried
himself with great dignity.

Figure 3.1: Eye-tracking data fixations.

Experimental settings

Data

- **Eye-tracking (ET):** English GECO corpus, 12 readers after filtering, five word-level gaze features (first fixation duration, gaze duration, first-run #fixations, total reading time, total #fixations), scaled to $[0, 100]$. Split: 80% train, 20% test, consistent across users.
- **Downstream tasks:** GLUE (CoLA, SST-2, MNLI, QNLI, RTE, WNLI, QQP, MRPC, STS-B) and a human complexity judgement dataset (COMP). Task metrics follow official choices (e.g., MCC for CoLA, accuracy for NLI, Spearman/Pearson for STS-B)
- **Out-of-domain geometry analysis:** English UD-EWT sentences length-matched to GECO for sentence-embedding geometry analyses.

Model and training Model used is **RoBERTa-base** (12 layers, $d = 768$). ET prediction is framed as multi-label token-level regression, assigning ET features to the first sub-token of each word, downstream tasks are sentence-level classification or regression. Common hyperparameters include AdamW, lr 1×10^{-5}, warmup 0.06, weight decay 0.1, and 10 epochs (task-dependent exceptions for large GLUE datasets and for LoRA runs).

ET-injection strategies

1. **Intermediate fine-tuning (INT):** first fine-tune on ET, then on the downstream task. Variants: INT-FULL (full model), INT-LAST3, INT-LAST2, INT-CLF (progressively fewer layers updated in the second step).
2. **LoRA:** ET fine-tuning, then downstream fine-tuning with low-rank adapters; adapters are moved back onto the ET-specialized model for evaluation.

3. **Multi-task, interleaved (MT-IL):** alternate batches between ET and downstream data, balancing dataset sizes by repeating ET batches.
4. **Multi-task with silver labels (MT-SILV):** generate ET pseudo-labels on the downstream set using an ET-specialized model and train jointly on task labels and silver ET features.

Representation space analyses

Sentence embeddings are obtained by mean pooling token representations, geometry is measured per layer on GECO and on UD-EWT (out-of-domain), enabling within-model, across-layer comparisons.

Metrics Two complementary descriptors are used.

- **Linear intrinsic dimensionality (Linear ID):** the number of effectively used linear directions in the embedding cloud (see §2.1.4).
- **Isotropy (IsoScore):** a bounded, rotation-invariant index of how uniformly the point cloud uses the ambient space (see §2.1.4).

Results

We now briefly show the results of our work.

i) *Downstream performance is preserved.* Models trained with ET injection retain competitive accuracy. In particular, INT-FULL achieves an average of 0.82 versus 0.83 for DST-ONLY, with most task scores within 0.02 points of the baseline. Multi-task strategies (MT-IL, MT-SILV) also remain robust, while partial unfreezing (INT-LAST3/2/CLF) and LoRA show larger drops (see Table 3.1).

ii) *ET supervision improves attention–gaze alignment.* Fine-tuning on ET features increases correlation between model attention and human gaze for *all* readers. The effect emerges from mid layers onward, with stronger alignment in upper layers compared to the baseline (see Table 3.2).

iii) *Alignment persists after task fine-tuning.* On the last layer, ET-injected models consistently outperform DST-ONLY (e.g., INT-CLF reaches 0.29 correlation vs. 0.08 for DST-ONLY). Partial unfreezing strategies preserve the highest correlations, while INT-FULL erodes them more (see Table 3.3) .

iv) *Representation geometry is compressed.* Compared to RoBERTa-base (Linear ID = 297, IsoScore = 28.69×10^{-3}), ET-injected models show lower dimensionality and reduced isotropy. For example, EYE-ONLY yields Linear ID = 160 and IsoScore = 4.97×10^{-3} (see Tables 3.4a and 3.4b).

v) INT-FULL *balances accuracy and compression.* This strategy achieves strong task performance while producing the lowest or near-lowest geometry metrics (average Linear ID =

117; IsoScore = 4.09×10^{-3}) across tasks.

vi) *Findings replicate out-of-domain.* On English UD-EWT, ET-injected models again show reduced Linear ID and isotropy compared to baselines, confirming that compression effects generalize beyond the training corpus.

vii) *Isotropy and dimensionality correlate.* Across models and tasks, isotropy and Linear ID display a strong positive Spearman correlation ($\rho \approx 0.75$ on GECO, similar on UD-EWT), indicating the metrics co-vary.

viii) *Trade-offs across strategies.* Partial fine-tuning (INT-LAST3/2/CLF) yields the strongest attention–gaze alignment after downstream training, while INT-FULL compresses geometry most. Both remain broadly competitive in downstream accuracy (see Tables 3.1 to 3.3).

	Downstream task											
Fine-tuning	COLA	COMP	MNLI-M	MNLI-MM	MRPC	QNLI	QQP	RTE	SST-2	STSB	WNLI	AVG
int-full	0.56	0.90	0.88	0.88	0.90	0.93	0.90	0.70	0.92	0.91	0.56	0.82
int-last3	0.25	0.88	0.70	0.71	0.80	0.82	0.81	0.54	0.88	0.81	0.56	0.71
int-last2	0.15	0.85	0.62	0.64	0.77	0.75	0.77	0.53	0.86	0.74	0.56	0.66
int-clf	0.00	0.70	0.43	0.44	0.75	0.61	0.61	0.50	0.76	0.12	0.56	0.50
lora	0.41	0.87	0.85	0.85	0.80	0.91	0.86	0.49	0.93	0.88	0.55	0.76
mt-il	0.53	0.91	0.83	0.83	0.90	0.92	0.88	0.75	0.93	0.90	0.52	0.81
mt-silv	0.51	0.91	0.88	0.87	0.88	0.93	0.90	0.60	0.93	0.91	0.50	0.76
dst-only	0.60	0.91	0.88	0.88	0.90	0.93	0.90	0.77	0.93	0.90	0.56	0.83

Table 3.1: Performance on downstream tasks for different eye-tracking (ET) injection strategies. Shaded cells (🟦) are within 0.02 of the DST-ONLY score for that task. For MNLI we report Matched (M) and MisMatched (MM) separately. Metrics follow GLUE conventions (e.g., Matthews corr. for COLA, accuracy for MNLI, QNLI, RTE, WNLI, SST-2; combined F1/Acc for MRPC and QQP; Spearman/Pearson for STSB); COMP is sentence-complexity prediction (Spearman).

	Model Layer												
Model	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**avg**
eye-only	0.23	0.20	0.26	0.23	0.19	0.22	0.25	0.25	**0.32**	0.27	0.24	0.29	0.25
base	0.25	0.21	**0.28**	0.25	0.21	0.17	0.16	0.19	0.18	0.05	0.08	0.12	0.18

Table 3.2: Spearman correlation coefficients between model attention and user attention for the model fine-tuned on predicting user ET features (EYE-ONLY) and RoBERTa-base (BASE). The scores are averaged across all users. Differently from EYE-ONLY, only half of users lead to significant correlations with BASE; the others were excluded from the mean.

3.0.2 Study 2: The Role of Eye-Tracking Data in Encoder-Based Models: an In-depth Linguistic Analysis

This study (Domenichelli et al. 2025) explores how injecting human eye–tracking (ET) signals into a multilingual encoder changes (i) the model's *attention* allocation across linguistically

	Attention correlation (last layer)										
Fine-tuning	**cola**	**comp**	**mnli**	**mrpc**	**qnli**	**qqp**	**rte**	**sst-2**	**stsb**	**wnli**	**avg**
int-full	0.19	**0.35**	0.05	0.16	0.06	0.08	0.12	0.04	0.09	0.18	0.13
int-last3	**0.29**	**0.28**	0.24	**0.31**	0.16	**0.29**	0.26	0.21	0.20	0.23	0.25
int-last2	**0.28**	0.26	0.19	**0.28**	**0.30**	0.24	**0.28**	**0.29**	**0.31**	**0.28**	0.27
int-clf	**0.29**	**0.29**	**0.29**	**0.29**	**0.29**	**0.29**	**0.29**	**0.29**	**0.29**	**0.29**	**0.29**
lora	**0.27**	0.22	0.13	0.20	0.13	0.20	**0.32**	0.16	0.21	**0.30**	0.21
mt-il	0.26	0.23	0.22	**0.27**	0.16	0.21	**0.27**	0.20	**0.27**	**0.28**	0.24
mt-silv	0.25	0.11	**0.28**	0.15	**0.31**	0.23	**0.33**	**0.31**	0.14	**0.27**	0.24
dst-only	0.06	0.08	0.05	0.01	0.07	0.03	0.02	0.07	0.11	0.12	0.08

Table 3.3: Correlations between human attention (*TRT*) and model attention on the **last layer** for each injection strategy. The scores are averaged across all readers. Highlighted cells indicate that the correlation score of the ET injected model exceeds that of DST-ONLY by at least 0.02 points. Bold scores are the highest correlation coefficients: those exceeding 0.27, which is 0.02 points lower than the last-layer correlation of EYE-ONLY.

	Linear ID										
F-T	**cola**	**comp**	**mnli**	**mrpc**	**qnli**	**qqp**	**rte**	**sst-2**	**stsb**	**wnli**	**avg**
int-full	**127**	**89**	191	185	242	11	161	**4**	**32**	127	**117**
int-last3	173	135	194	162	**148**	154	**154**	92	142	154	151
int-last2	162	148	166	160	160	153	157	142	158	158	157
int-clf	160	160	160	160	160	160	160	160	160	160	160
lora	184	144	310	**158**	279	256	166	202	146	163	201
mt-il	232	154	**110**	179	228	88	155	251	228	152	178
mt-silv	249	209	233	268	251	207	206	221	264	209	232
dst-only	289	249	249	249	249	**3**	278	**4**	249	**16**	186
base	297										–
eye-only	160										–

(a) Layer 12 Linear ID values averaged over all users in the GECO dataset. Entries in bold mark the lowest value for each task.

	IsoScore $\times 10^3$										
F-T	**cola**	**comp**	**mnli**	**mrpc**	**qnli**	**qqp**	**rte**	**sst-2**	**stsb**	**wnli**	**avg**
int-full	**0.74**	**1.19**	2.75	15.59	3.03	**0.35**	5.95	0.88	**0.71**	9.74	4.09
int-last3	7.92	2.96	7.40	6.79	**2.10**	3.53	4.36	**0.69**	3.46	5.00	4.42
int-last2	5.89	3.78	7.45	5.05	5.58	4.08	4.35	3.24	5.77	4.69	4.99
int-clf	4.99	4.99	4.99	4.99	4.99	4.99	4.99	4.99	4.99	4.99	4.99
lora	11.26	5.36	30.23	8.47	11.34	28.62	6.01	9.99	2.72	5.27	11.93
mt-il	4.34	5.02	**1.39**	**2.71**	4.06	1.07	**2.52**	5.83	4.66	3.69	**3.53**
mt-silv	17.38	10.76	8.28	12.00	11.89	6.57	10.14	11.26	21.56	11.97	12.18
dst-only	6.53	35.94	4.58	15.08	4.69	0.40	28.03	1.17	11.14	**0.27**	10.78
base	28.69										–
eye-only	4.97										–

(b) Layer 12 IsoScore values ($\times 10^3$) averaged over all users in the GECO dataset. Entries in bold mark the lowest value for each task.

defined categories and (ii) the *geometry* of token representations. Experiments cover Italian and English and quantify both attention–gaze alignment and representation–space shifts. ET fine-tuning increases correlation between model attention and human reading time, reallocates attention toward linguistically informative tokens, and compresses the embedding space in upper layers, with effects replicated across languages.

Experimental settings

Data

- **Eye-tracking (ET):** MECO L1 subsets for Italian (9 readers) and English (25 readers reading a shared subset of sentences). Five word-level targets capture early, late, and contextual reading signals: First Fixation Duration, Gaze Duration, Total Reading Time, First-run #Fixations, and Total #Fixations.
- **Linguistic annotations for analysis:** UD Italian-ISDT and UD English-EWT training sets provide PoS tags, head–dependent distances, and sentence positions used to define linguistic classes for attention and geometry analyses.

Model and training We employed a multilingual encoder **XLM-RoBERTa-base** (12 layers). ET prediction is framed as multi-target token-level regression, with labels attached to the first sub-token of each word. For each participant, one reader-specific model is fine-tuned for 50 epochs with learning rate 5×10^{-5}, weight decay 0.01, warm-up ratio 0.05; hyperparameters are selected by 5-fold CV. Unlike aggregation across readers, training is *individualized*: one model per reader to respect inter-subject variability in reading behavior. This design is central to the subsequent correlation and geometry analyses.

Attention–gaze alignment For each layer, the attention *received* by each word when computing the sentence BOS token `<s>` is correlated (Spearman) with the reader's Total Reading Time, yielding layer-wise attention–gaze alignment curves before and after ET fine-tuning. Attention is L1-normalized per sentence (excluding BOS/EOS). For each class (POS, word length in characters, sentence position, head–dependent distance), the percentage change in average received attention is measured after vs. before ET fine-tuning.

Representation-space analyses

Metrics Two complementary geometry descriptors are computed per layer on UD sentences (first sub-tokens):

- **Linear intrinsic dimensionality (Linear ID):** the number of effectively used linear directions in the embedding cloud (see §2.1.4).
- **Isotropy (IsoScore):** a bounded, rotation-invariant index of how uniformly the point cloud uses the ambient space (see §2.1.4).

Significance of pre-/post-ET differences is tested with Wilcoxon signed-rank tests.

Token representations are taken at every layer for UD sentences, class-conditioned analyses compute IsoScore and Linear-ID within each linguistic group, while global curves summarize layer-wise trajectories across the whole dataset. This is reported separately for Italian and replicated for English. We report (i) global geometry over layers before/after ET; (ii) POS-conditioned manifolds; and (iii) geometry grouped by head–dependent distance.

Results

i) **Attention–gaze alignment** Spearman correlations between model attention and human Total Reading Time increase after ET fine-tuning, with the largest gains in deeper layers for both Italian (see Figure 3.2).

ii) **Attention reallocation by linguistic factors.** Single-character tokens lose attention; short words (2–3 chars) gain attention in several middle and upper layers (Figure 3.3 left). Punctuation is consistently down-weighted, while some function words (ADP/DET/AUX) gain attention, Italian shows notable increases for CCONJ (Figure 3.3 right). Earlier sentence positions gain attention on average, with layer-specific boosts to the first token in layers 2 and 11 (Figure 3.4 left). Tokens closer to their syntactic head, especially when the head follows, receive more attention after ET (see Figure 3.4 right).

iii) **Global geometry.** After ET, upper layers become markedly more anisotropic and lower-dimensional: IsoScore begins to fall from layer 4 rather than only in the very top layers, and Linear-ID drops from ~ 650 to < 100 by layer 12 (Italian, similar trends for English). Differences are significant by Wilcoxon tests (Figure 3.5).

iv) **POS-conditioned geometry.** Before ET, content words (NOUN/VERB/PROPN) occupy higher-ID, more isotropic regions than function words and punctuation. ET compresses *all* POS categories in the upper stack, largely reducing the content–function gap above layer ~ 6 (Figures 3.6 - 3.7).

v) **Syntactic-distance geometry.** Right dependents show higher ID and isotropy than left dependents already before ET, ET applies a near-uniform upper-layer compression from layer 8 while preserving the lower-layer left–right asymmetry (Figures 3.8 - 3.9).

vi)

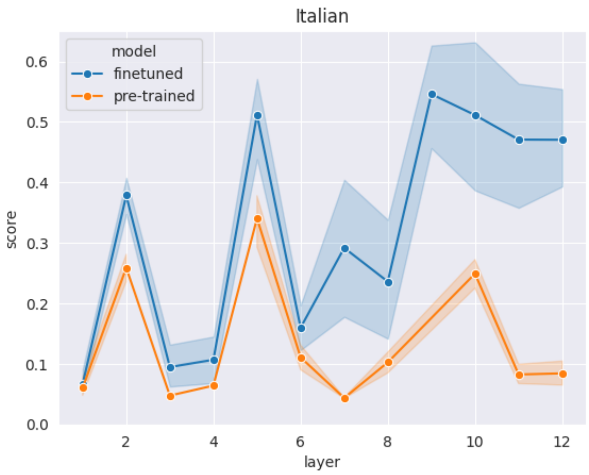

Figure 3.2: Correlation between model attention and human attention (p-value < 0.05).

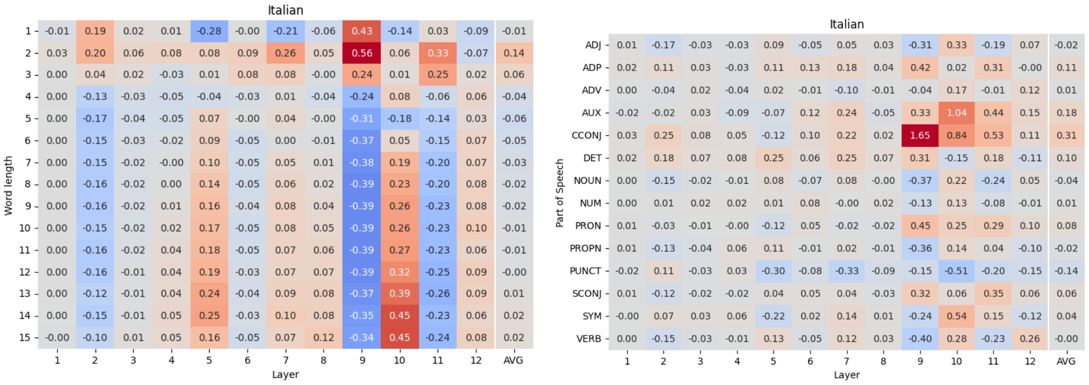

Figure 3.3: Attention shift for word length and for UD Parts of Speech.

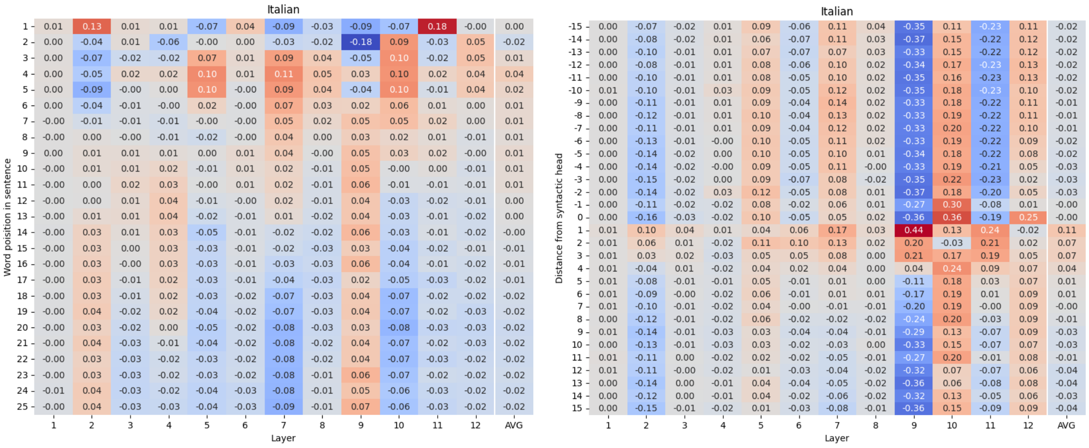

Figure 3.4: Attention shift for word position in sentence and distance from syntactic head.

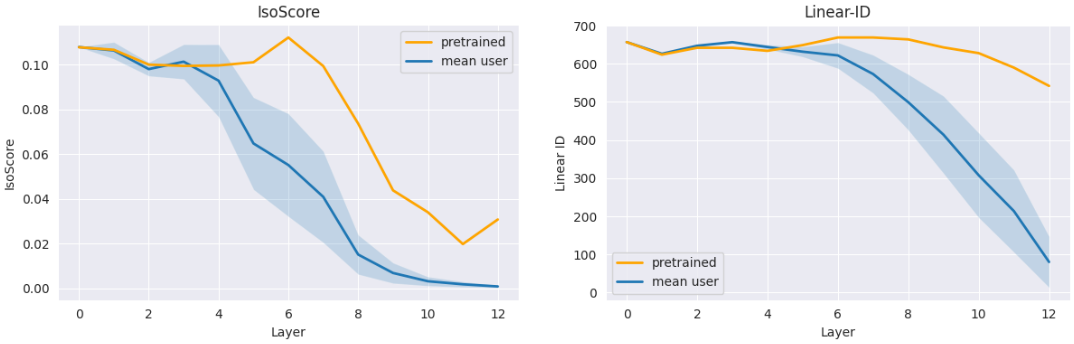

Figure 3.5: IsoScore (left) and Linear intrinsic dimensionality (right) of word embeddings from all model layers, before and after fine-tuning, averaged across users.

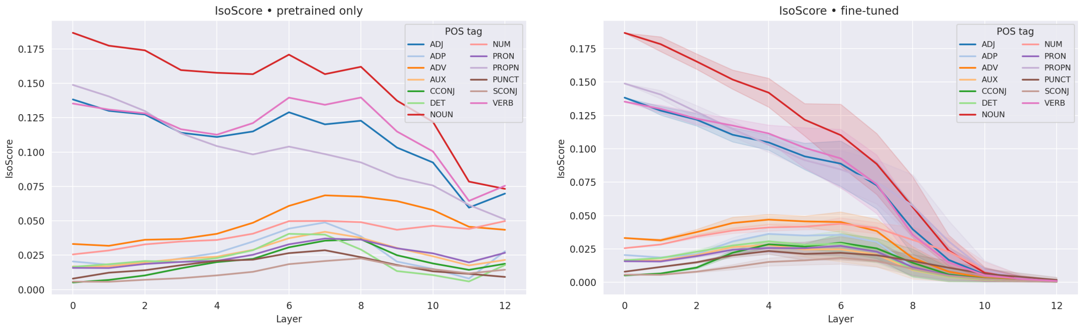

Figure 3.6: Isotropy before (left) and after (right) fine-tuning, shown for the 13 most frequent POS classes.

3.0.3 Study 3: Curriculum learning shapes layers geometry

This study examines how simple *curriculum schedules* during pretraining affect both downstream performance and the *geometry* of internal representations in a **RoBERTa-medium** encoder. We order the *same* pretraining sentences by sentence-level difficulty proxies (length and two Italian readability indices) and compare forward (*easy*→*hard*), inverted (*hard*→*easy*), and `random` orderings. At regular checkpoints we (i) fine-tune on three tasks (complexity prediction, sentiment, POS tagging) under fixed budgets and (ii) monitor geometry on a held-out

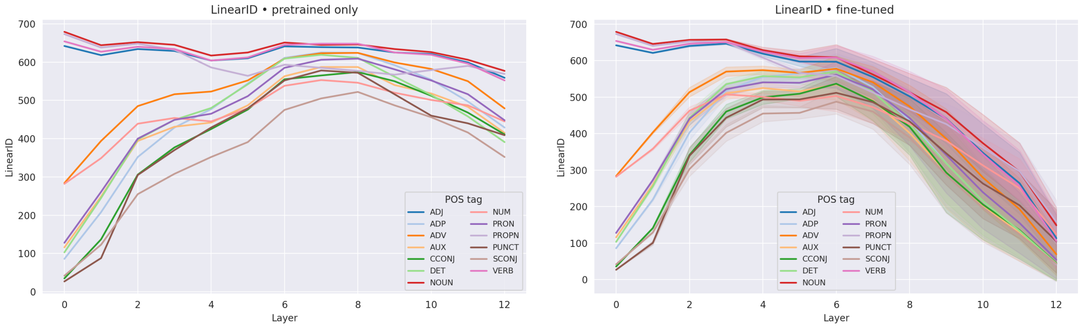

Figure 3.7: Linear-ID before (left) and after (right) fine- tuning, shown for the 13 most frequent POS classes.

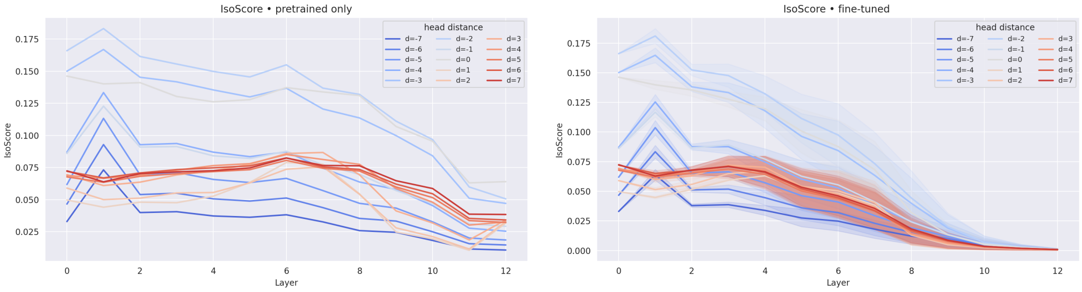

Figure 3.8: Isotropy before (left) and after (right) fine-tuning, shown for syntax head distance (up to 7 tokens distance).

corpus using *IsoScore*, *Linear ID* and *TwoNN* (see §2.1.4). Before presenting our experiments, we introduce the notion of curriculum learning and its relevance for language models training.

Curriculum learning (CL) organises training data. We choose to pursue a ”pedagogically inspired curriculum learning”, organising data from easier to harder examples, mimicking human pedagogy and often yielding faster convergence and better generalisation (Bengio et al. 2009). Although large language models are typically trained on randomly ordered corpora, CL requires defining a difficulty ranking and a pacing schedule, both nontrivial design choices. In neural language modelling, CL has been applied to data-to-text generation (Chang et al. 2021), intent detection (Gong et al. 2021), and even pretraining regimes (Nagatsuka et al. 2021), where staged exposure improved stability and downstream performance. In our setting, CL provides

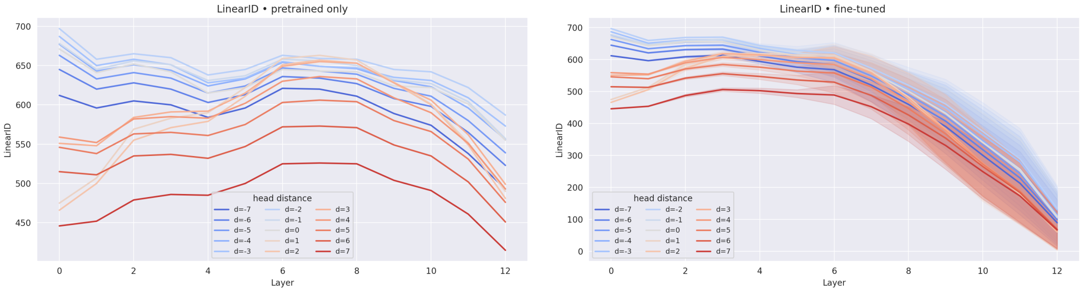

Figure 3.9: Linear-ID before (left) and after (right) fine- tuning, shown for syntax head distance (up to 7 tokens dis- tance).

a controlled way to probe how the latent geometry evolves as ”complexity” is introduced progressively.

Experimental settings

Data We pretrain on a single Wikipedia corpus (identical across schedules) and reserve a held-out set for geometry tracking. Downstream evaluation uses three tasks: (i) sentence *complexity* regression (reporting MAE and Spearman ρ), (ii) *sentiment* classification (accuracy), and (iii) *POS* tagging (F1 and accuracy). These tasks are specifically chosen, since each targets a specific type of linguistic knowledge

Model and training Backbone: **RoBERTa-medium** (8-layer encoder, masked-LM objective). We train up to 120k updates with stage boundaries at 40k and 80k for curricula, checkpoints are evaluated on all downstream tasks under fixed fine-tuning budgets. For geometry, we extract the mean pooled representation of each sentence from all layers and at each checkpoint.

Curricula Curriculum orderings are:

- SENTENCE_LENGTH, the length of a sentence in tokens.
- GULPEASE INDEX (Lucisano, Piemontese, et al. 1988) a readability metric based on raw text features
- READ – IT (Dell’Orletta et al. 2011), a readability metrics that combines traditional raw text features with lexical, morpho-syntactic and syntactic information.

For each difficulty signal we instantiate two schedules:

- **Forward** (*easy*→*hard*): examples sorted from simpler to harder.
- **Inverted** (*hard*→*easy*): reverse order.

A uniform `random` order is the baseline.

Representation space analyses

Extraction protocol At each checkpoint we compute sentence-conditioned token embeddings from all layers and summarize geometry on the held-out corpus.

Metrics. We track three metrics, two *linear* and one *nonlinear*:

- **IsoScore** (see §2.1.4)
- **Linear ID** (see §2.1.4)
- **TwoNN** (see §2.1.4) nearest-neighbour intrinsic dimensionality (lower ⇒ fewer effective degrees of freedom).

These are computed on the same data and protocol across schedules to enable direct comparison.

First results

Although our analysis is still preliminary, the evidence so far suggests that curriculum schedules primarily affect the convergence of the learning curve rather than the final level of performance. Across all tasks, end accuracy under curriculum training is essentially indistinguishable from the random-ordering baseline (Figure 3.10).

By contrast, the representation geometry differs markedly between curriculum-trained and randomly trained models (Figure 3.11). For linear geometry diagnostics, we observe two clear regimes: curriculum runs move more quickly toward a compact and well-spread configuration, whereas random ordering follows a slower, smoother trajectory. The small spikes in the curves align with stage boundaries in the curriculum, as expected when the distribution of training examples shifts. Notably, these trajectories diverge from the very first checkpoints.

For the neighborhood-based intrinsic-dimensionality estimator (TwoNN), differences between curriculum and random training are more subtle. This contrast indicates that linear and non-linear measures capture complementary aspects of the representation space, and that data ordering can reshape geometry even when end-task accuracy remains unchanged.

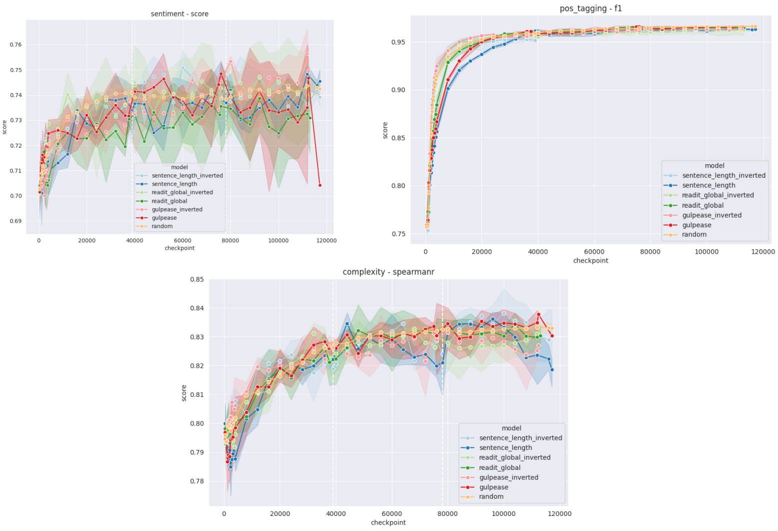

Figure 3.10: Performance scores (accuracy, F1, and Spearman correlation) on sentiment analysis, POS tagging and complexity. Many seeds were used so results are averaged on each checkpoint.

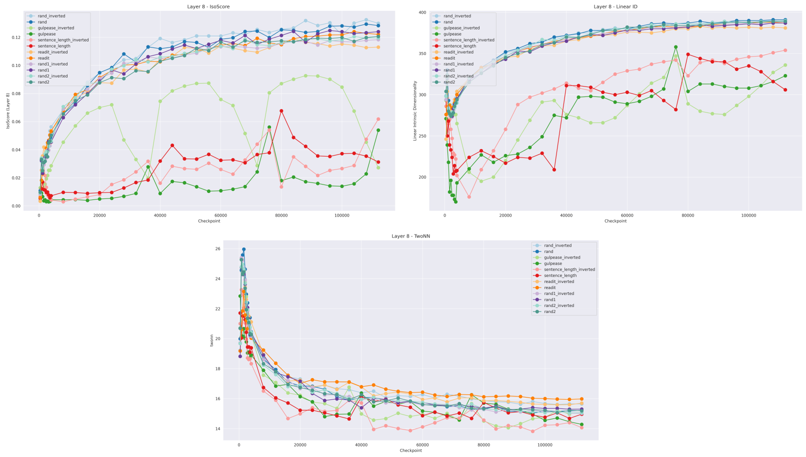

Figure 3.11: Geometric scores (Linear ID, IsoScore and TwoNN) at last over checkpoints.

Bibliography

Aghajanyan, Armen, Sonal Gupta, and Luke Zettlemoyer (2021). “Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning”. In: *Proceedings of ACL*, pp. 7319–7328.

Alain, Guillaume and Yoshua Bengio (2016). “Understanding Intermediate Layers Using Linear Classifier Probes”. In: *arXiv preprint arXiv:1610.01644*. DOI: 10.48550/arXiv.1610.01644. arXiv: 1610.01644 [stat.ML]. URL: https://arxiv.org/abs/1610.01644.

Ansuini, Alessio, Alessandro Laio, Jakob H Macke, and Davide Zoccolan (2019). “Intrinsic dimension of data representations in deep neural networks”. In: *Advances in Neural Information Processing Systems* 32.

Belinkov, Yonatan (2022). “Probing classifiers: Promises, shortcomings, and advances”. In: *Computational Linguistics* 48.1, pp. 207–219.

Bengio, Yoshua, Jérôme Louradour, Ronan Collobert, and Jason Weston (2009). “Curriculum learning”. In: *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48.

Bussmann, Bart, Patrick Leask, and Neel Nanda (2024). “BatchTopK Sparse Autoencoders”. In: *arXiv:2412.06410*.

Cai, Xingyu, Jiaji Huang, Yuchen Bian, and Kenneth Church (2021). “Isotropy in the contextual embedding space: Clusters and manifolds”. In: *International conference on learning representations*.

Ceruti, Claudio, Simone Bassis, Alessandro Rozza, Gabriele Lombardi, Elena Casiraghi, and Paola Campadelli (2012). “DANCo: dimensionality from angle and norm concentration”. In: *arXiv preprint arXiv:1206.3881*.

Chang, Ernie, Hui-Syuan Yeh, and Vera Demberg (2021). “Does the order of training samples matter? improving neural data-to-text generation with curriculum learning”. In: *arXiv preprint arXiv:2102.03554*.

Chen, Ricky TQ, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud (2018). “Neural ordinary differential equations”. In: *Advances in neural information processing systems* 31.

Cheng, Pengcheng, François Fleuret, and Marco Baroni (2024). *Emergence of a High-Dimensional Abstraction Phase in Language Models*. arXiv:2405.15471.

Conmy, Arthur, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso (2023). “Towards automated circuit discovery for mechanistic interpretability”. In: *Advances in Neural Information Processing Systems* 36, pp. 16318–16352.

Cop, Uschi, Nicolas Dirix, Denis Drieghe, and Wouter Duyck (2017). “Presenting GECO: An eyetracking corpus of monolingual and bilingual sentence reading”. In: *Behavior research methods* 49.2, pp. 602–615.

Cunningham, Hoagy, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey (2023). “Sparse Autoencoders Find Highly Interpretable Features in Language Models”. In: *arXiv:2309.08600*.

Dao, Tri, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré (2022). “Flashattention: Fast and memory-efficient exact attention with io-awareness”. In: *Advances in neural information processing systems* 35, pp. 16344–16359.

Dell’Orletta, Felice, Simonetta Montemagni, and Giulia Venturi (2011). “READ–IT: Assessing readability of Italian texts with a view to text simplification”. In: *Proceedings of the second workshop on speech and language processing for assistive technologies*, pp. 73–83.

Demberg, Vera and Frank Keller (2008). “Data from eye-tracking corpora as evidence for theories of syntactic processing complexity”. In: *Cognition* 109.2, pp. 193–210.

Denti, Francesco, Diego Doimo, Alessandro Laio, and Antonietta Mira (2022). “The generalized ratios intrinsic dimension estimator”. In: *Scientific Reports* 12.1, p. 20005.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186.

Dini, Luca, Lucia Domenichelli, Dominique Brunato, and Felice Dell'Orletta (July 2025). "From Human Reading to NLM Understanding: Evaluating the Role of Eye-Tracking Data in Encoder-Based Models". In: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar. Vienna, Austria: Association for Computational Linguistics, pp. 17796–17813. ISBN: 979-8-89176-251-0. DOI: 10.18653/v1/2025.acl-long.870. URL: https://aclanthology.org/2025.acl-long.870/.

Domenichelli, Lucia, Luca Dini, Dominique Brunato, and Felice Dell'Orletta (2025). "The Role of Eye-Tracking Data in Encoder-Based Models: an In-depth Linguistic Analysis". In.

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

Engels, Joshua, Isaac Liao, Eric J. Michaud, Wes Gurnee, and Max Tegmark (2024). "Not All Language Model Features Are Linear". In: *arXiv preprint arXiv:2405.14860*. URL: https://arxiv.org/abs/2405.14860.

Ethayarajh, Kawin (Nov. 2019a). "How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China: Association for Computational Linguistics, pp. 55–65. DOI: 10.18653/v1/D19-1006. URL: https://aclanthology.org/D19-1006/.

– (2019b). "How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings". In: *Proceedings of EMNLP-IJCNLP*. Hong Kong, pp. 55–65.

European Union (2025). *Implementation Timeline of the Artificial Intelligence Act*. https://artificialintelligenceact.eu/implementation-timeline/.

Facco, Elena, Maria d'Errico, Alex Rodriguez, and Alessandro Laio (2017). "Estimating the intrinsic dimension of datasets by a minimal neighborhood information". In: *Scientific reports* 7.1, p. 12140.

Ferrando, Javier, Gabriele Sarti, Arianna Bisazza, and Marta R Costa-Jussà (2024). "A primer on the inner workings of transformer-based language models". In: *arXiv preprint arXiv:2405.00208*.

Gardinazzi, Yuri, Karthik Viswanathan, Giada Panerai, Alessio Ansuini, Alberto Cazzaniga, and Matteo Biagetti (2024). "Persistent topological features in large language models". In: *arXiv preprint arXiv:2410.11042*.

Geva, Mor, Roei Schuster, Jonathan Berant, and Omer Levy (Nov. 2021). "Transformer Feed-Forward Layers Are Key-Value Memories". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Ed. by Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 5484–5495. DOI: 10.18653/v1/2021.emnlp-main.446. URL: https://aclanthology.org/2021.emnlp-main.446/.

Gong, Yantao, Cao Liu, Jiazhen Yuan, Fan Yang, Xunliang Cai, Guanglu Wan, Jiansong Chen, Ruiyao Niu, and Houfeng Wang (2021). "Density-based dynamic curriculum learning for intent detection". In: *Proceedings of the 30th ACM international conference on information & knowledge management*, pp. 3034–3037.

Hewitt, John and Percy Liang (Nov. 2019). "Designing and Interpreting Probes with Control Tasks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China: Association for Computational Linguistics, pp. 2733–2743. DOI: 10.18653/v1/D19-1275. URL: https://aclanthology.org/D19-1275/.

Hewitt, John and Christopher D. Manning (June 2019). "A Structural Probe for Finding Syntax in Word Representations". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4129–4138. DOI: 10.18653/v1/N19-1419. URL: https://aclanthology.org/N19-1419/.

Hollenstein, Nora, Jonathan Rotsztejn, Marius Troendle, Andreas Pedroni, Ce Zhang, and Nicolas Langer (2018). "ZuCo, a simultaneous EEG and eye-tracking resource for natural sentence reading". In: *Scientific data* 5.1, pp. 1–13.

Houle, Michael E, Erich Schubert, and Arthur Zimek (2018). "On the correlation between local intrinsic dimensionality and outlierness". In: *International Conference on Similarity Search and Applications*. Springer, pp. 177–191.

Kennedy, Alan, Joël Pynte, Wayne S Murray, and Shirley-Anne Paul (2013). "Frequency and predictability effects in the Dundee Corpus: An eye movement analysis". In: *Quarterly Journal of Experimental Psychology* 66.3, pp. 601–618.

Levina, Elizaveta and Peter Bickel (2004). "Maximum likelihood estimation of intrinsic dimension". In: *Advances in neural information processing systems* 17.

Li, Bohan, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li (2020). "On the Sentence Embeddings from Pre-trained Language Models". In: *Proceedings of EMNLP*.

Li, Dongheng, Jason Babcock, and Derrick J Parkhurst (2006). "openEyes: a low-cost head-mounted eye-tracking solution". In: *Proceedings of the 2006 symposium on Eye tracking research & applications*, pp. 95–100.

Liang, Percy, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. (2022). "Holistic evaluation of language models". In: *arXiv preprint arXiv:2211.09110*.

Lucisano, Pietro, Maria Emanuela Piemontese, et al. (1988). "Gulpease: una formula per la predizione della leggibilita di testi in lingua italiana". In: *Scuola e città*, pp. 110–124.

Machina, Aaron, Becca Nelson, and Jeremy Gwinnup (2024). "Anisotropy is Not Inherent to Transformers". In: *Proceedings of NAACL*.

Meng, Kevin, David Bau, Alex Andonian, and Yonatan Belinkov (2022). "Locating and editing factual associations in gpt". In: *Advances in neural information processing systems* 35, pp. 17359–17372.

Miaschi, Alessio and Felice Dell'Orletta (2020). "Contextual and non-contextual word embeddings: an in-depth linguistic investigation". In: *Proceedings of the 5th Workshop on Representation Learning for NLP*, pp. 110–119.

Mu, Jiaqi, Suma Bhat, and Pramod Viswanath (2017). "All-but-the-Top: Simple and Effective Postprocessing for Word Representations". In: *arXiv preprint arXiv:1702.01417*. DOI: 10.48550/arXiv.1702.01417. URL: https://arxiv.org/abs/1702.01417.

Nagatsuka, Koichi, Clifford Broni-Bediako, and Masayasu Atsumi (2021). "Pre-training a BERT with curriculum learning by increasing block-size of input text". In: *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pp. 989–996.

Panickssery, Nina, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner (2023). "Steering llama 2 via contrastive activation addition". In: *arXiv preprint arXiv:2312.06681*.

Papoutsaki, Alexandra (2015). "Scalable webcam eye tracking by learning from user interactions". In: *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 219–222.

Park, Kiho, Yo Joong Choe, and Victor Veitch (2023). "The Linear Representation Hypothesis and the Geometry of Large Language Models". In: *arXiv preprint arXiv:2311.03658*. URL: https://openreview.net/pdf?id=T0PoOJg8cK.

Pimentel, Tiago, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell (July 2020). "Information-Theoretic Probing for Linguistic Structure". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault. Online: Association for Computational Linguistics, pp. 4609–4622. DOI: 10.18653/v1/2020.acl-main.420. URL: https://aclanthology.org/2020.acl-main.420/.

Rajaee, Sara and Mohammad Taher Pilehvar (2021). "How Does Fine-tuning Affect the Geometry of Embedding Space? A Case Study on Isotropy". In: *Findings of EMNLP*.

Rajamanoharan, Senthooran, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda (2024). "Improving Dictionary Learning with Gated Sparse Autoencoders". In: *arXiv preprint arXiv:2404.16014*. URL: https://arxiv.org/abs/2404.16014.

Rogers, Anna, Olga Kovaleva, and Anna Rumshisky (2020). "A Primer in BERTology: What We Know About How BERT Works". In: *Transactions of the ACL*.

Rudman, William, Nate Gillman, Taylor Rayne, and Carsten Eickhoff (May 2022a). "IsoScore: Measuring the Uniformity of Embedding Space Utilization". In: *Findings of the Association for Computational Linguistics: ACL 2022*. Ed. by Smaranda Muresan, Preslav Nakov, and Aline Villavicencio. Dublin, Ireland: Association for Computational Linguistics, pp. 3325–3339. DOI: 10.18653/v1/2022.findings-acl.262. URL: https://aclanthology.org/2022.findings-acl.262/.

– (2022b). "IsoScore: Measuring the Uniformity of Embedding Space Utilization". In: *Findings of the Association for Computational Linguistics (ACL)*, pp. 3325–3339.

San Agustin, Javier, Henrik Skovsgaard, Emilie Mollenbach, Maria Barret, Martin Tall, Dan Witzner Hansen, and John Paulin Hansen (2010). "Evaluation of a low-cost open-source gaze tracker". In: *Proceedings of the 2010 symposium on eye-tracking research & applications*, pp. 77–80.

Schoenholz, Samuel S, Justin Gilmer, Surya Ganguli, and Jascha Sohl-Dickstein (2016). "Deep information propagation". In: *arXiv preprint arXiv:1611.01232*.

Shazeer, Noam, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean (2017). "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer". In: *arXiv preprint arXiv:1701.06538*.

Siegelman, Noam, Sascha Schroeder, Cengiz Acartürk, Hee-Don Ahn, Svetlana Alexeeva, Simona Amenta, Raymond Bertram, Rolando Bonandrini, Marc Brysbaert, Daria Chernova, et al. (2022). "Expanding horizons of cross-linguistic research on reading: The Multilingual Eye-movement Corpus (MECO)". In: *Behavior research methods* 54.6, pp. 2843–2863.

Team, Anthropic Interpretability (2024). *The Engineering Challenges of Scaling Interpretability*. Anthropic Research Blog. https://www.anthropic.com/research/engineering-challenges-interpretability.

Templeton, Evan, Jack Lindsey, Chris Olah, and Anthropic Interpretability Team (2024). *Extracting Interpretable Features from Claude 3 Sonnet*. Transformer Circuits Thread. https://transformer-circuits.pub/2024/scaling-monosemanticity/.

Tulchinskii, Eduard, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Sergey Nikolenko, Evgeny Burnaev, Serguei Barannikov, and Irina Piontkovskaya (2023). "Intrinsic dimension estimation for robust detection of ai-generated texts". In: *Advances in Neural Information Processing Systems* 36, pp. 39257–39276.

Uchendu, Adaku and Thai Le (2024). "Unveiling topological structures in text: A comprehensive survey of topological data analysis applications in nlp". In: *arXiv preprint arXiv:2411.10298*.

Valeriani, Lucrezia, Diego Doimo, Francesca Cuturello, Alessandro Laio, Alessio Ansuini, and Alberto Cazzaniga (2023). "The geometry of hidden representations of large transformer models". In: *Advances in Neural Information Processing Systems* 36, pp. 51234–51252.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin (2017). "Attention is all you need". In: *Advances in neural information processing systems* 30.

Voita, Elena and Ivan Titov (Nov. 2020). "Information-Theoretic Probing with Minimum Description Length". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu. Online: Association for Computational Linguistics, pp. 183–196. DOI: 10.18653/v1/2020.emnlp-main.14. URL: https://aclanthology.org/2020.emnlp-main.14/.

Xie, Yang, Seth Ebner, Ian Tenney, Mingqiu Wang, Wen-tau Yih, and Sebastian Gehrmann (2024). *Anisotropy Is Inherent to Self-Attention in Transformers*. arXiv:2401.12143.

Yin, Fan, Jayanth Srinivasa, and Kai-Wei Chang (2024). *Characterizing Truthfulness in Large Language Model Generations with Local Intrinsic Dimension*. arXiv:2402.18048.