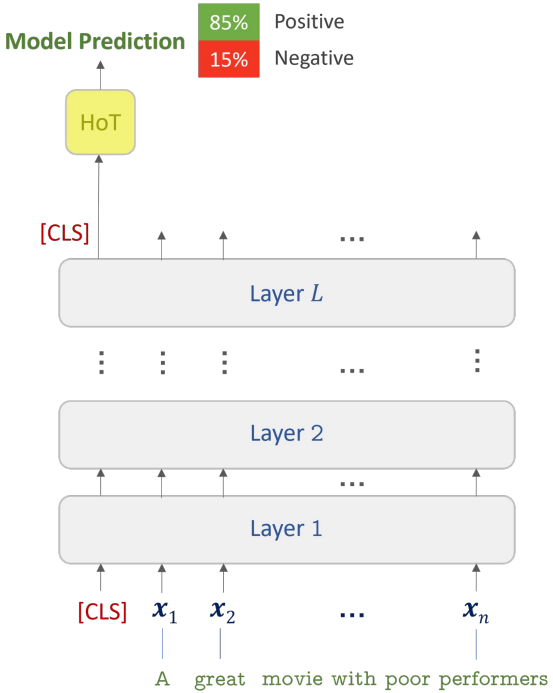
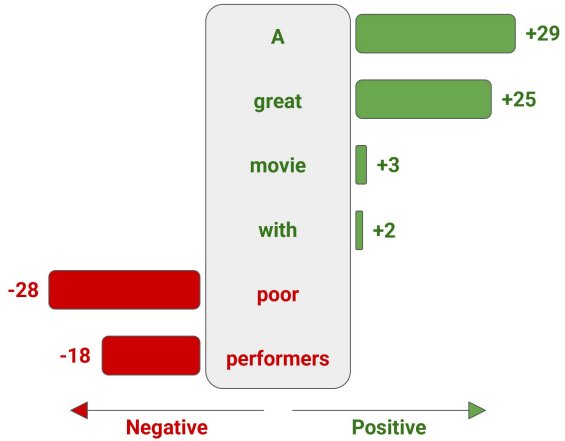
NormXLogit : The *Head-on-Top* Never Lies

Sina Abbasi, Mohammad Reza Modarres, Mohammad Taher Pilehvar

Problem Statement

- A simple classification task: sentiment analysis



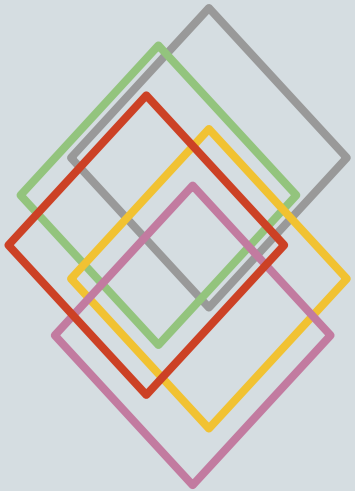


Related Work

- Gradient-based Methods
  - Gradient×Input (Kindermans et al., 2016), and Integrated Gradients (Sundararajan et al., 2017)
  - **Limitation:** Computational overhead, easily manipulable and not reliable (Wang et al., 2020)
- Perturbation-based Methods
  - SHAP (Lundberg and Lee, 2017), and LIME (Ribeiro et al., 2016)
  - **Limitation:** Computational overhead, significantly less faithful (Atanasova et al., 2020)
- Vector-based Methods
  - DecompX (Modarressi et al., 2023), ALTI (Ferrando et al., 2022), and GlobEnc (Modarressi et al., 2022)
  - **Limitation:** Computational overhead, architecture-specific

Proposed Approach:

NormXLogit



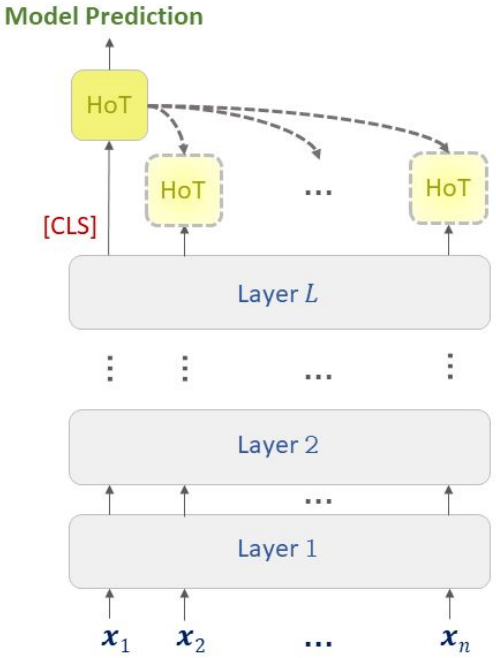
- LogAt: Logit Attribution
- Norm of Word Embedding

Methodology: LogAt

- Intuition behind the attention mechanism:
  - More important tokens → greater contribution in [CLS] → [CLS] has higher degree of similarity

Importance of token x_i

the extent to which final representation of x_i can resemble the model's final prediction



Methodology: LogAt

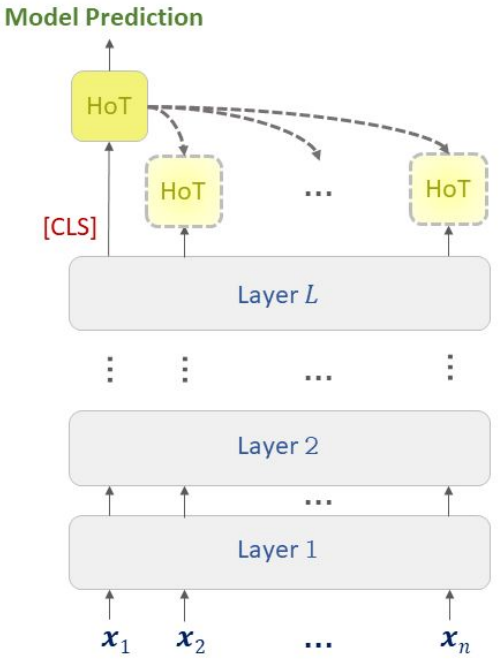
- Intuition behind the attention mechanism:
  - More important tokens → greater contribution in [CLS] → [CLS] has higher degree of similarity

Classification / Language Modeling

$$\mathrm{Att}_{\mathrm{LogAt}}(x_i) = \mathrm{HoT}_{\mathrm{clas}}(x_i^L)[\hat{p}]$$

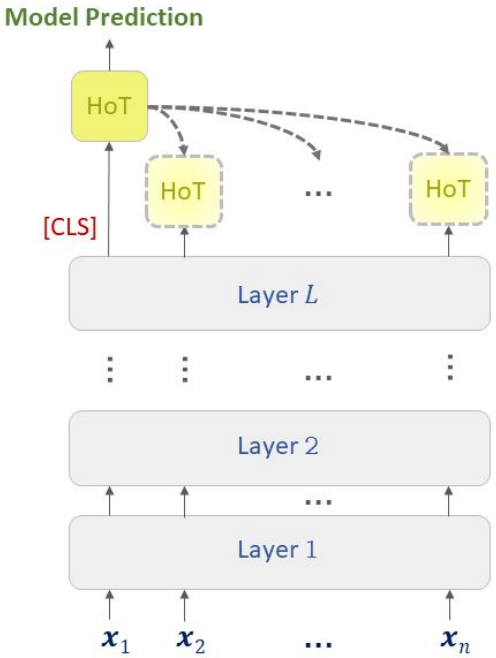
Regression

$$\mathrm{Att}_{\mathrm{LogAt}}(x_i) = \left|\mathrm{HoT}_{\mathrm{reg}}(x_i^L) - \mathrm{HoT}_{\mathrm{reg}}(\texttt{[CLS]}^L)\right|$$



Methodology: Norm of Word Embedding

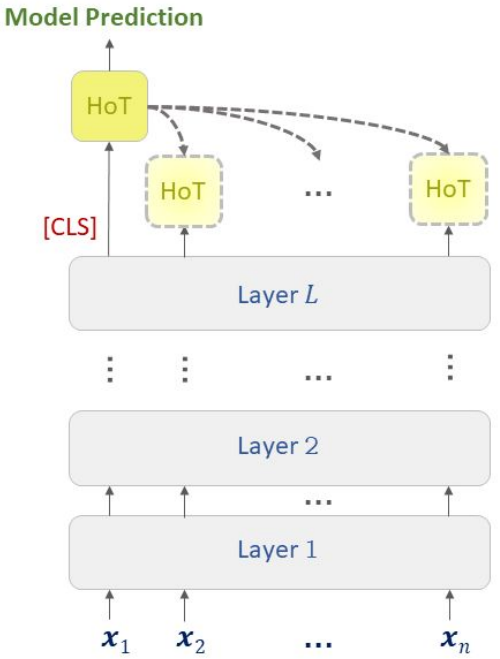
- Based on the self-attention mechanism:
  - Tokens with **greater norms** are expected to **contribute more** to the final representation of the target token.
- Previous work has shown that tokens with **greater ℓ^2 norm** carry **more information** .
  (Oyama et al., 2023)



Methodology: NormXLogit

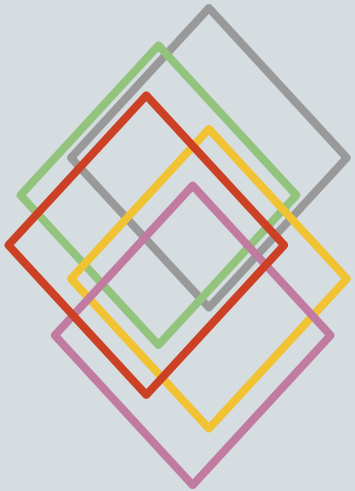
- The attribution of token x_i using NormXLogit is obtained as:

$$\mathrm{Att}_{\mathrm{NormXLogit}}(x_i) = \left\|x_i^0\right\|_2 \times \mathrm{Att}_{\mathrm{LogAt}}(x_i)$$



Experiment 1:

Faithfulness Analysis



- Experimental Setup
- Evaluation Metrics
- Results
- Computational Efficiency

Experimental Setup

- Data:
  - SST-2 (Socher et al., 2013), MultiNLI (Williams et al., 2018), QNLI (Wang et al., 2018), and STS-B (Cer et al., 2017)
- Models:
  - LLAMA 2 (Touvron et al., 2023), DeBERTa (He et al., 2023), and BERT (Devlin et al., 2019)
- Input Attribution Methods:
  - Gradient Norm (Grad. Norm, Simonyan et al., 2014), Gradient×Input (G×I), and Integrated Gradients (IG)
  - DecompX
  - Random Baseline

Evaluation Metrics

- **Comprehensiveness** → necessity of highlighted tokens for the model's prediction
- **Sufficiency** → adequacy of highlighted tokens to preserve the model's prediction
- Each criterion is evaluated through two metrics:
    a. Confidence Drop
    b. Accuracy

Evaluation Metrics (cont.)

- Comprehensiveness - Confidence Drop:

$$\text{Comp}_{\text{CD}}(K\%) = \frac{1}{m}\sum_{i=1}^{m}\left[f_{\hat{y}}(X_i) - f_{\hat{y}}(X_i \backslash K)\right]$$

[CLS]	A	great	movie	[SEP]
5.2	3.1	14.2	8.0	1.7

Positive	Negative
85%	15%
65%	35%

Evaluation Metrics (cont.)

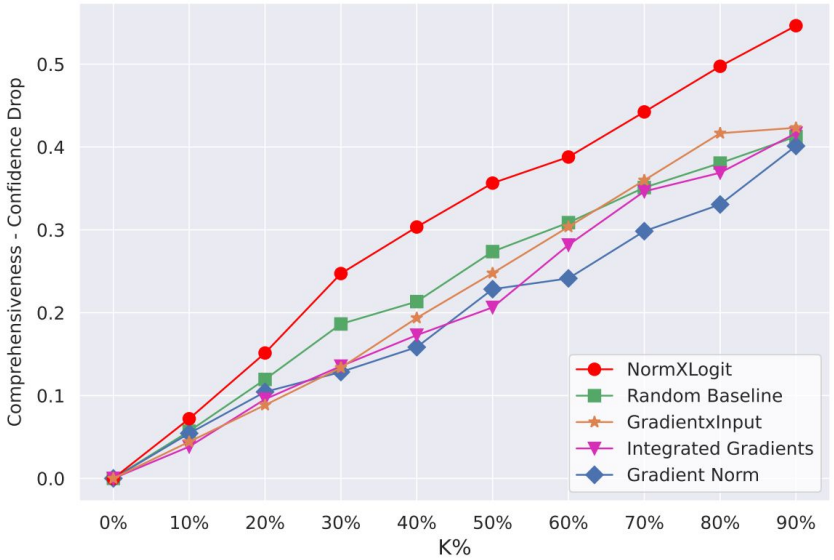
- Comprehensiveness - Accuracy:

$$\text{Comp}_{\text{ACC}}(K\%) = \text{Acc}(\mathcal{D}^{\setminus K})$$

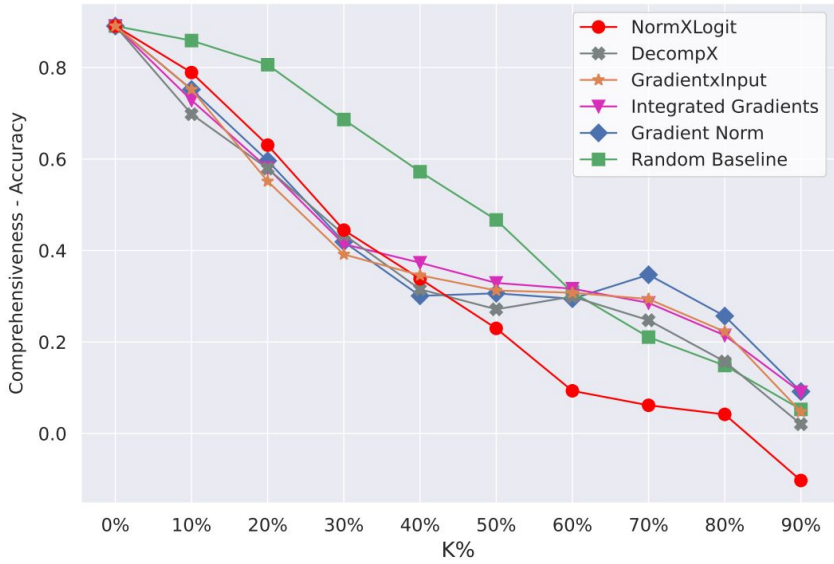
[CLS]	A	great	movie	[SEP]
5.2	3.1	14.2	8.0	1.7

Positive	Negative
85%	15%
65%	35%

Results: Comprehensiveness



Comprehensiveness Confidence Drop of different attribution methods for LLAMA 2 fine-tuned on SST-2 (higher values are better).



Comprehensiveness Accuracy of different attribution methods for BERT fine-tuned on STS-B (lower values are better).

Results: Comprehensiveness

- In most cases, our proposed method performs better than or is competitive with other approaches.

	SST-2 ($\text{Comp}_{\text{CD}}\uparrow$)			MNLI ($\text{Comp}_{\text{CD}}\uparrow$)			QNLI ($\text{Comp}_{\text{CD}}\uparrow$)			STS-B ($\text{Comp}_{\text{Acc}}\downarrow$)		
	LLAMA 2	DeBERTa	BERT	LLAMA 2	DeBERTa	BERT	LLAMA 2	DeBERTa	BERT	LLAMA 2	DeBERTa	BERT
Random	0.256	0.266	0.245	0.421	0.445	0.361	0.284	0.306	0.273	0.283	0.430	0.457
Grad. Norm	0.216	0.320	0.331	0.364	0.535	0.460	0.334	0.365	0.360	0.351	0.338	0.374
G×I	0.236	0.345	0.339	0.442	0.565	0.456	0.353	0.382	0.364	0.255	<u>0.214</u>	0.358
IG	0.220	0.346	0.367	0.448	**0.571**	0.466	0.336	0.381	0.364	0.237	0.227	0.370
DecompX	N/A	N/A	**0.574**	N/A	N/A	**0.585**	N/A	N/A	**0.460**	N/A	N/A	0.336
ℓ^2 norm	0.299	0.360	0.311	0.420	0.473	0.393	0.272	0.339	0.304	0.251	**0.199**	0.321
LogAt	<u>0.341</u>	<u>0.377</u>	0.364	<u>0.518</u>	0.548	<u>0.566</u>	**0.378**	<u>0.435</u>	0.394	**0.167**	0.423	<u>0.313</u>
NormXLogit	**0.341**	**0.386**	<u>0.423</u>	**0.519**	<u>0.566</u>	0.556	<u>0.363</u>	**0.474**	<u>0.402</u>	<u>0.233</u>	0.320	**0.281**

Comprehensiveness of NormXLogit against other methods across various model and dataset configurations. Each value is computed by averaging across all perturbation ratios (higher Confidence Drop and lower Accuracy are better). Best values are in **bold**, and second-best values are <u>underlined</u>.

Computational Efficiency

- NormXLogit demonstrates significantly lower computational costs compared to other methods and remains unaffected by input sequence length.

	Attribution Method				
Input Length	**Gradient Norm**	**Gradient×Input**	**Integrated Gradients**	**DecompX**	**NormXLogit**
40	$0.77_{\pm 0.03}$	$0.80_{\pm 0.02}$	$0.82_{\pm 0.02}$	$0.39_{\pm 0.01}$	$\mathbf{0.36}_{\pm 0.00}$
120	$0.77_{\pm 0.02}$	$0.81_{\pm 0.02}$	$0.96_{\pm 0.03}$	$0.38_{\pm 0.00}$	$\mathbf{0.36}_{\pm 0.00}$
360	$0.76_{\pm 0.03}$	$0.80_{\pm 0.02}$	$1.47_{\pm 0.05}$	$0.99_{\pm 0.02}$	$\mathbf{0.38}_{\pm 0.01}$
512	$0.78_{\pm 0.02}$	$0.80_{\pm 0.02}$	$1.79_{\pm 0.06}$	$2.36_{\pm 0.02}$	$\mathbf{0.36}_{\pm 0.00}$

Average computation time (in seconds) per instance for different attribution methods across various input lengths.

Computational Efficiency

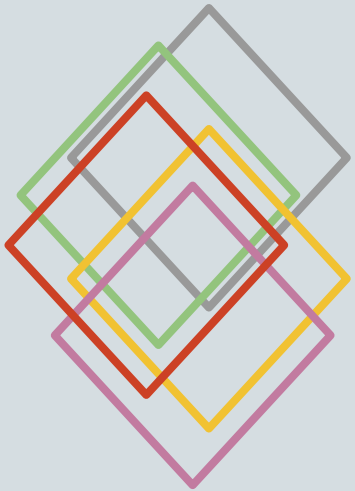
- NormXLogit supports substantially larger batch sizes and achieves the lowest per-instance time, underscoring its scalability and memory efficiency.

	Attribution Method				
Evaluation Criteria	**Gradient Norm**	**Gradient×Input**	**Integrated Gradients**	**DecompX**	**NormXLogit**
Maximum Batch Size	100	100	2	1	**750**
Average Time per Instance (s)	$0.0076_{\pm 0.00}$	$0.0081_{\pm 0.00}$	$1.4104_{\pm 0.01}$	$2.3625_{\pm 0.02}$	$\mathbf{0.0005}_{\pm 0.00}$

Efficiency of attribution methods when maximizing batch size under a 48GB memory constraint (input length = 512).

Experiment 2:

Evidence Alignment



- Experimental Setup
- Alignment Metrics
- Results

Experimental Setup

- Data:
  - BLiMP (Warstadt et al., 2020)
  - contains sentence pairs where the true label is uniquely determined by a single word

Phenomenon	**UID**	**Example** (Target ✔/Foil ✘)
Anaphor Number Agreement	ana	This government alarms itself ✔/themselves ✘.
Determiner-Noun Agreement	dna	Russell explored this ✔/these ✘ mall.
	dnaa	Patients scan this ✔/these ✘ orange brochure.
Subject-Verb Agreement	darn	The sister of doctors writes ✔/write ✘.
	rpsv	The pedestrian has ✔/have ✘ forgotten Grace.

Examples of various linguistic phenomena that have been investigated in our experiments. Each paradigm is represented by a unique identifier (UID) from the BLiMP dataset. The target and foil words are denoted using check and cross marks. In each instance, the relevant evidence is underlined.

Experimental Setup

- Model:
  - RoBERTa (Liu et al., 2019)
- Attribution Methods:
  - GlobEnc, ALTI, and Value Zeroing (Mohebbi et al., 2023)
  - Random Baseline

Alignment Metrics

- Consider the following example:

"Karla thinks/think about it" ⟹ [Karla, thinks, about, it]

$$\mathcal{E} = [1, 0, 0, 0]$$

$$\mathcal{S} = [0.3, 0.1, 0.5, 0.1]$$

- Dot Product:
  - The dot product $\mathcal{E} \cdot \mathcal{S}$ measures the total score that the target attribution method assigns to the evidence tokens.

$$\begin{aligned} \mathcal{E} \cdot \mathcal{S} &= 1 \cdot 0.3 + 0 \cdot 0.1 + 0 \cdot 0.5 + 0 \cdot 0.1 \\ &= 0.3 \end{aligned}$$

Alignment Metrics

- Consider the following example:

"Karla thinks/think about it" ⟹ [Karla, thinks, about, it]

$\boldsymbol{\mathcal{E}} = [1, 0, 0, 0]$

$\boldsymbol{\mathcal{S}} = [0.3, 0.1, 0.5, 0.1]$

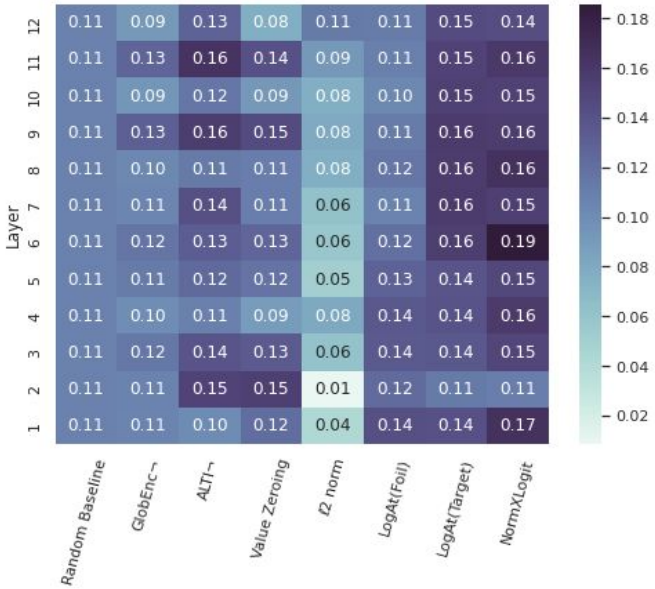
- Average Precision:
  - Based on ranking rather than raw scores

$\text{Rank}(\boldsymbol{\mathcal{S}}) = [2, 0, 1, 3]$

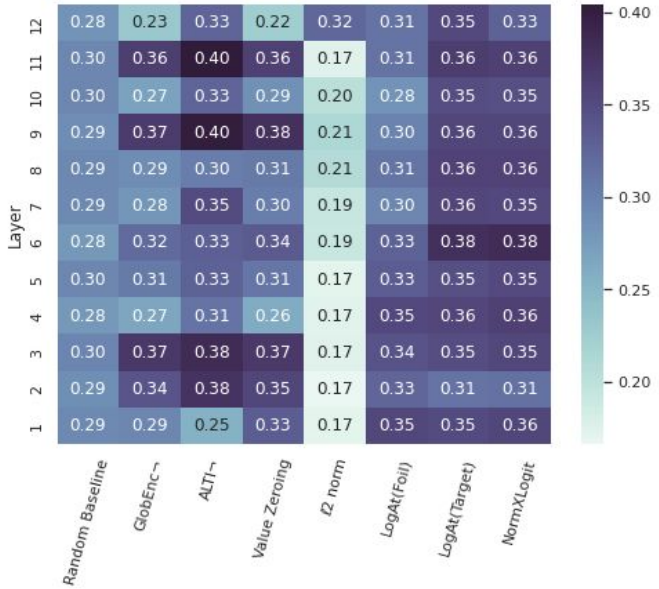
$\text{Evidence Index} = \{0\}$

$$\text{AP} = \sum_{k=1}^{n} (R_k - R_{k-1}) P_k$$

Results



Per-layer alignment between evidence and explanation vectors for the fine-tuned version of RoBERTa, calculated using **Dot Product** metric (higher values are better).



Per-layer alignment between evidence and explanation vectors for the fine-tuned version of RoBERTa, calculated using **Average Precision** metric (higher values are better).