DecompX:
Explaining Transformers Decisions by Propagating Token Decomposition

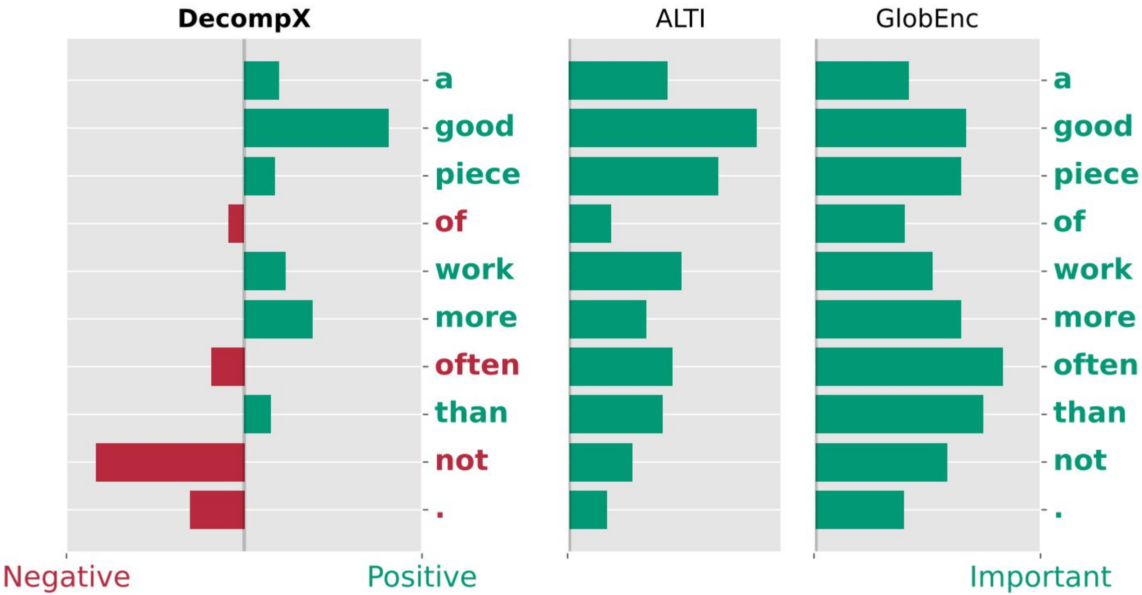
Ali Modarressi*, Mohsen Fayyaz*, Ehsan Aghazadeh,
Yadollah Yaghoobzadeh, Mohammad Taher Pilehvar

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN
mcml Munich Center for Machine Learning
TeIAS | Tehran Institute for Advanced Studies
KHATAM UNIVERSITY

61 ACL 2023

Introduction

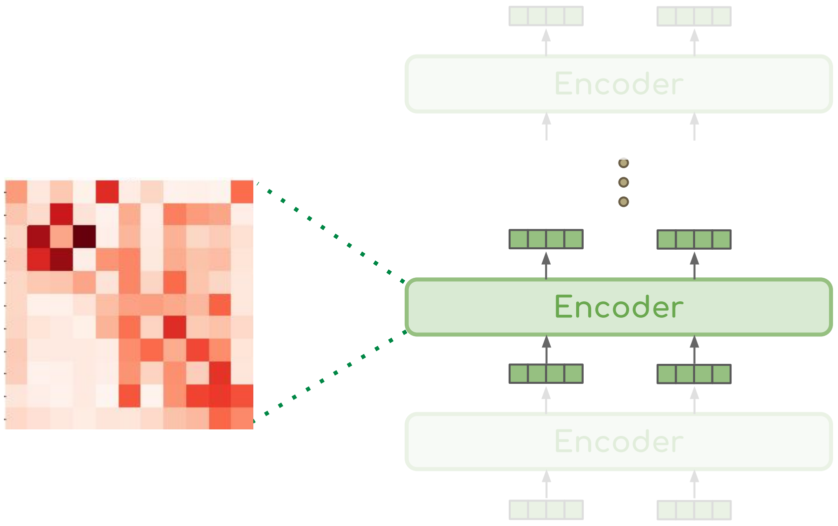
What is Explanation?



[1] Ali Modarressi, Mohsen Fayyaz, Yadollah Yaghoobzadeh, and Mohammad Taher Pilehvar. 2022. GlobEnc: Quantifying global token attribution by incorporating the whole encoder layer in transformers.
[2] Javier Ferrando, Gerard I. Gállego, and Marta R. Costajussà. 2022. Measuring the mixing of contextual information in the transformer.

Existing Methods

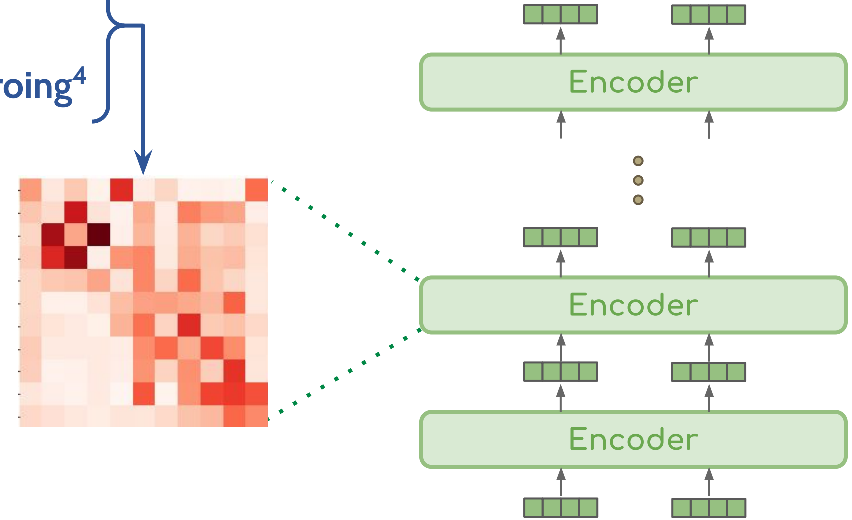
Local Attention Map $\longrightarrow$ Scalar Aggregation (e.g. Rollout, Flow)



Existing Methods

Local Attention Map → Scalar Aggregation (e.g. Rollout, Flow)

- Raw-attention[1]
- ALTI[2]
- Globenc[3]
- Value-Zeroing[4]

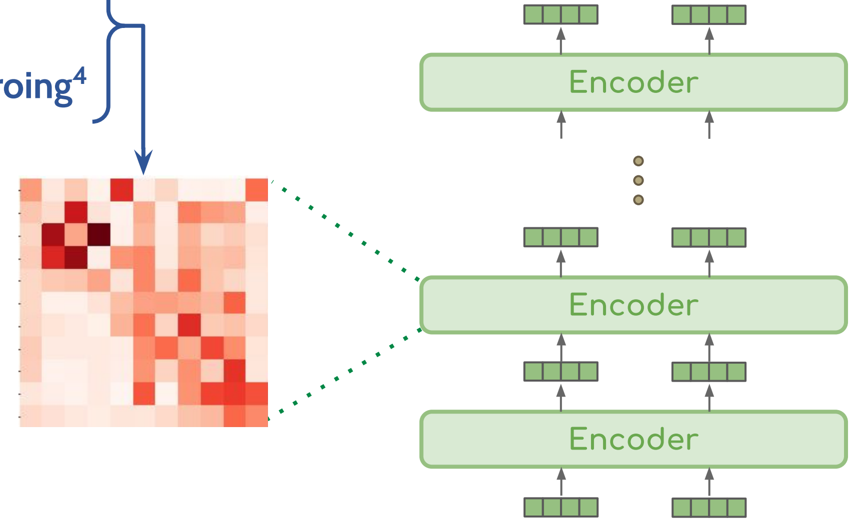


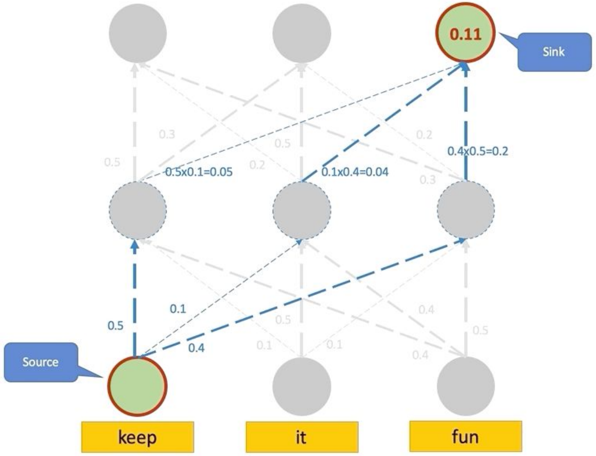
[1] Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4190–4197, Online. Association for Computational Linguistics.
[2] Javier Ferrando, Gerard I. Gállego, and Marta R. Costajussà. 2022. Measuring the mixing of contextual information in the transformer.
[3] Ali Modarressi, Mohsen Fayyaz, Yadollah Yaghoobzadeh, and Mohammad Taher Pilehvar. 2022. GlobEnc: Quantifying global token attribution by incorporating the whole encoder layer in transformers.
[4] Hosein Mohebbi, Willem Zuidema, Grzegorz Chrupała, and Afra Alishahi. 2023. Quantifying context mixing in transformers. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics

Existing Methods

Local Attention Map → Scalar Aggregation (e.g. Rollout, Flow)

- Raw-attention[1]
- ALTI[2]
- Globenc[3]
- Value-Zeroing[4]

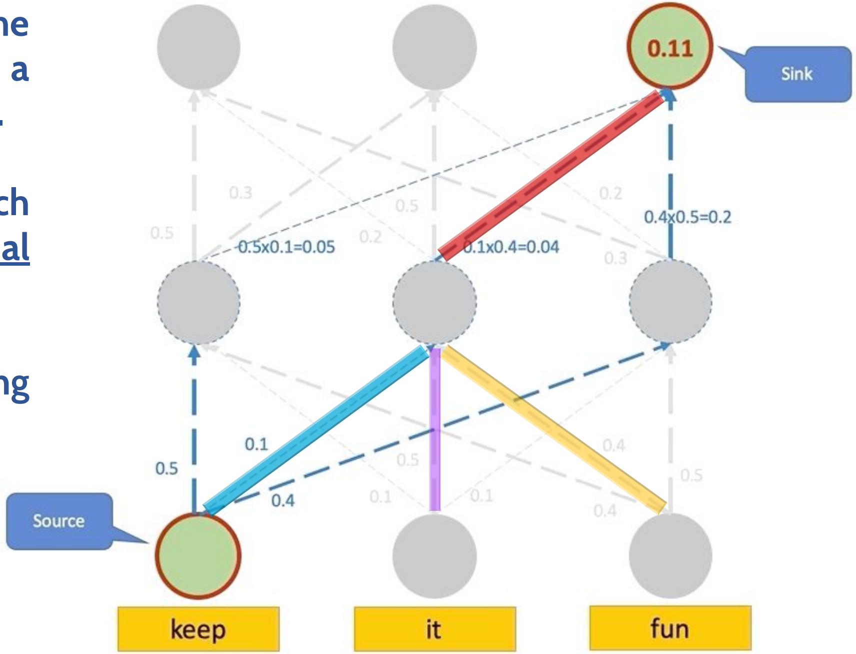




[1] Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4190–4197, Online. Association for Computational Linguistics.
[2] Javier Ferrando, Gerard I. Gállego, and Marta R. Costajussà. 2022. Measuring the mixing of contextual information in the transformer.
[3] Ali Modarressi, Mohsen Fayyaz, Yadollah Yaghoobzadeh, and Mohammad Taher Pilehvar. 2022. GlobEnc: Quantifying global token attribution by incorporating the whole encoder layer in transformers.
[4] Hosein Mohebbi, Willem Zuidema, Grzegorz Chrupała, and Afra Alishahi. 2023. Quantifying context mixing in transformers. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics

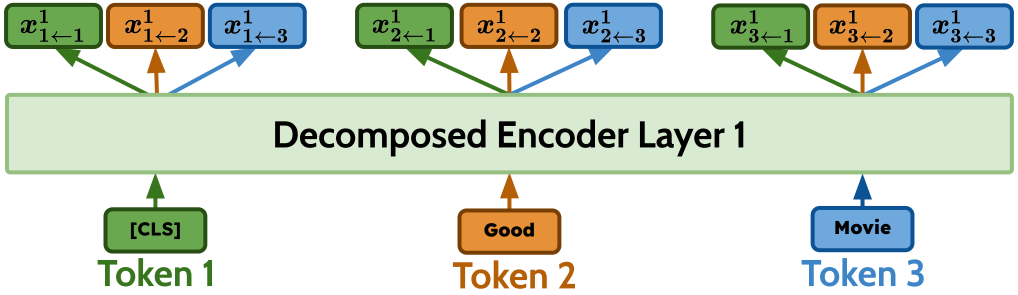
Scalar Aggregation Issues

- ➔ Scalar aggregation methods (e.g. Rollout) assume that the only required information for computing the global flow is a set of <u>scalar cross-token attributions</u>.
- ➔ Nevertheless, this simplifying assumption ignores that each decomposed vector represents the <u>multi-dimensional impact of its inputs</u>.
- ➔ Therefore, losing information is inevitable when reducing these complex vectors into one cross-token weight.



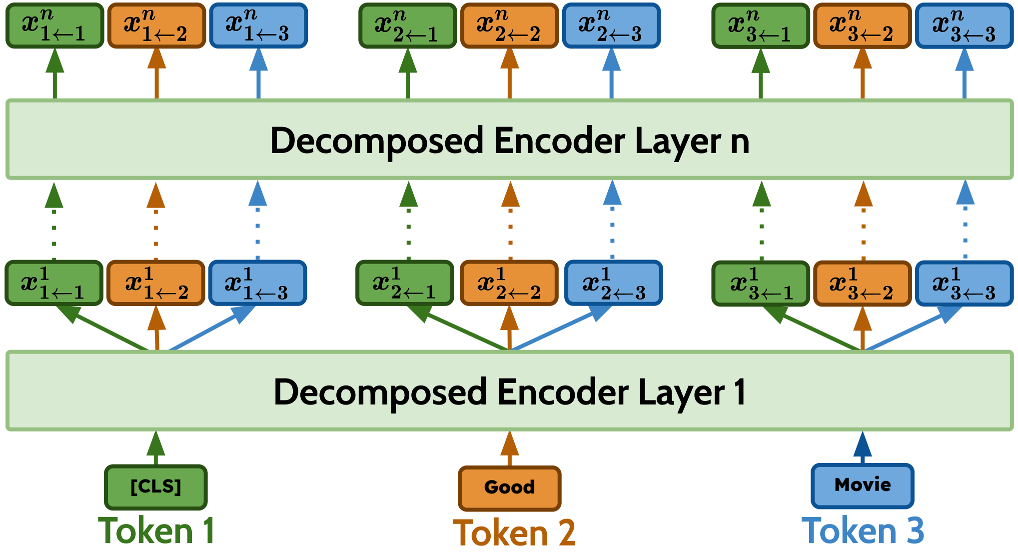
Our Solution: DecompX

Propagating Token Decomposition



Our Solution: DecompX

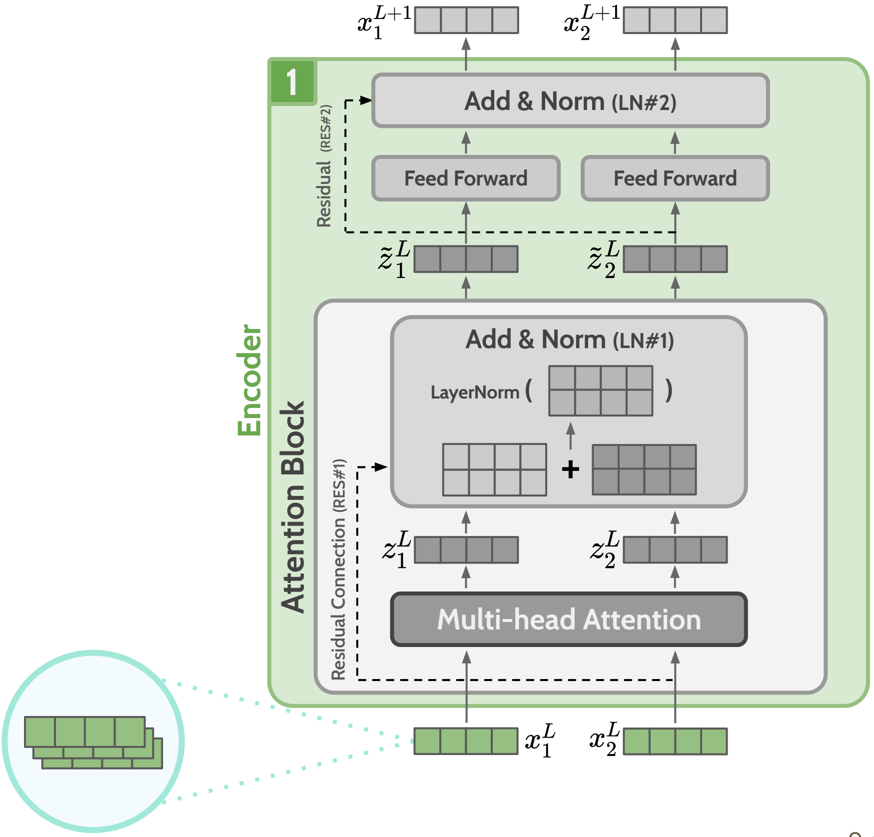
Propagating Token Decomposition



DecompX

Propagating Token Decomposition

$$\boldsymbol{x}_i^{\ell} = \sum_{k=1}^{N} \boldsymbol{x}_{i \Leftarrow k}^{\ell}$$

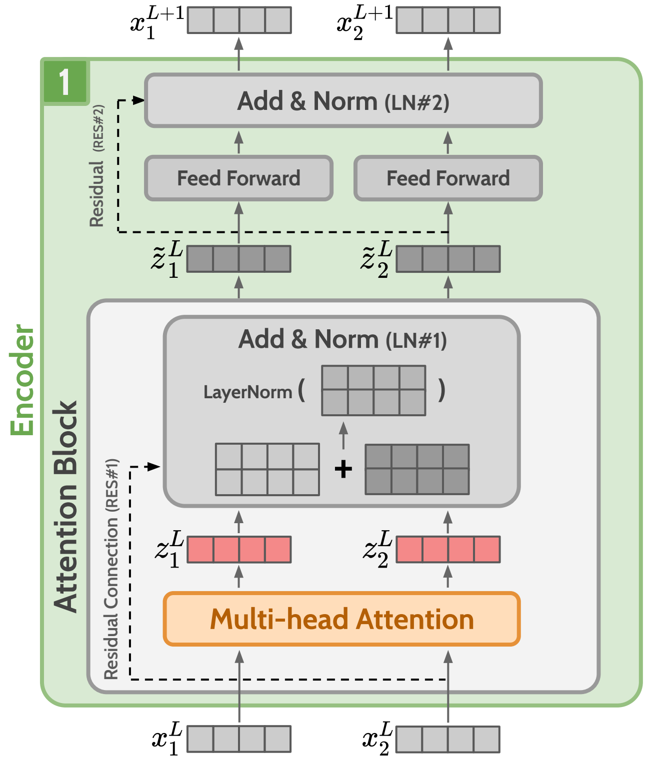


DecompX

Propagating Token Decomposition

$$\boldsymbol{x}^{\ell}_{i\Leftarrow k}$$

$$\boldsymbol{z}_i^{\ell} = \sum_{k=1}^{N} \underbrace{\left(\boxed{\sum_{h=1}^{H}\sum_{j=1}^{N} \alpha^{h}_{i,j}\boldsymbol{x}^{\ell}_{j\Leftarrow k}\boldsymbol{W}^{h}_{\boldsymbol{Att}}} + \omega_k \boldsymbol{b}_{\boldsymbol{Att}} \right)}_{\boldsymbol{z}^{\ell}_{i\Leftarrow k}}$$



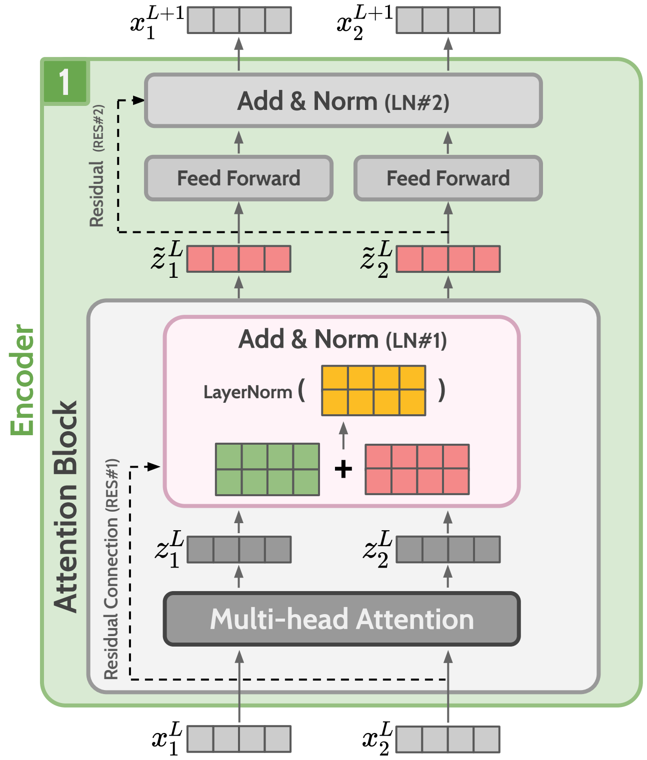
DecompX

Propagating Token Decomposition

$$x^{\ell}_{i\Leftarrow k}$$

$$z_i^{\ell} = \sum_{k=1}^{N} \underbrace{\left(\sum_{h=1}^{H} \sum_{j=1}^{N} \alpha^{h}_{i,j} \boldsymbol{x}^{\ell}_{j\Leftarrow k} \boldsymbol{W}^{h}_{\text{Att}} + \omega_k \boldsymbol{b}_{\text{Att}} \right)}_{z^{\ell}_{i\Leftarrow k}}$$

$$\mathrm{LN}(\boldsymbol{z}^{+\ell}_{i}) = \sum_{k=1}^{N} \underbrace{g_{\boldsymbol{z}^{+\ell}_{i}}(\boldsymbol{z}^{+\ell}_{i\Leftarrow k}) + \boldsymbol{\beta}}_{\tilde{\boldsymbol{z}}^{\ell}_{i\Leftarrow k}}$$



DecompX

Propagating Token Decomposition

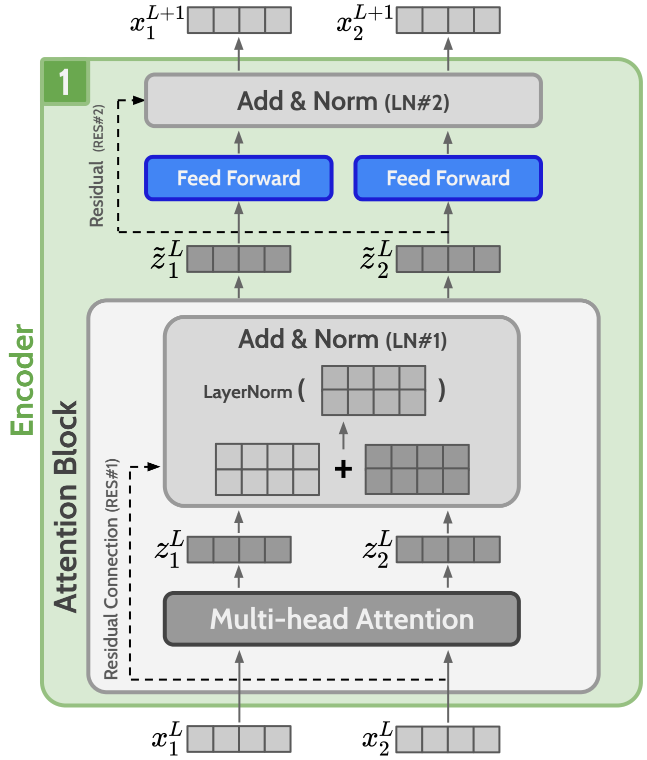
$$x^{\ell}_{i\Leftarrow k}$$

$$z^{\ell}_{i} = \sum_{k=1}^{N} \underbrace{\left(\sum_{h=1}^{H} \sum_{j=1}^{N} \alpha^{h}_{i,j} \boldsymbol{x}^{\ell}_{j\Leftarrow k} \boldsymbol{W}^{h}_{\mathrm{Att}} + \omega_k \boldsymbol{b}_{\mathrm{Att}} \right)}_{z^{\ell}_{i\Leftarrow k}}$$

$$\mathrm{LN}(z^{+\ell}_{i}) = \sum_{k=1}^{N} \underbrace{g_{z^{+\ell}_{i}}(z^{+\ell}_{i\Leftarrow k}) + \beta}_{\tilde{z}^{\ell}_{i\Leftarrow k}}$$

$$\boldsymbol{z}^{\ell}_{\mathrm{FFN},i} = f^{(\zeta^{\ell}_{i})}_{\mathrm{act}} \Big(\sum_{k=1}^{N} \zeta^{\ell}_{i\Leftarrow k} \Big) \boldsymbol{W}^{2}_{\mathrm{FFN}} + \boldsymbol{b}^{2}_{\mathrm{FFN}}$$

$$= \sum_{k=1} \underbrace{\boldsymbol{\theta}^{(\zeta^{\ell}_{i})} \odot \zeta^{\ell}_{i\Leftarrow k} + \boldsymbol{b}^{2}_{\mathrm{FFN}}}_{\boldsymbol{z}^{\ell}_{\mathrm{FFN},i\Leftarrow k}}$$



DecompX

Propagating Token Decomposition

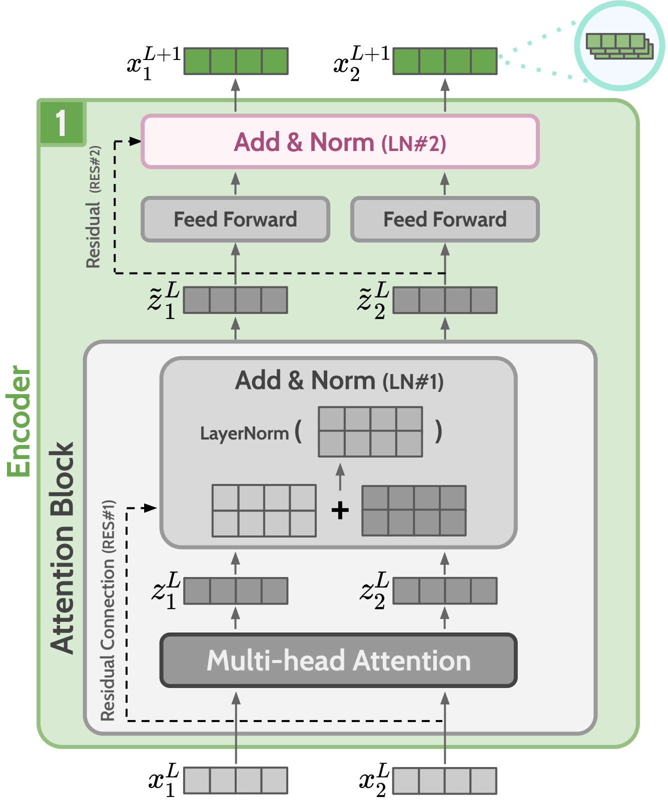
$$x^{\ell}_{i\Leftarrow k}$$

$$z_i^{\ell} = \sum_{k=1}^{N} \underbrace{\Big(\sum_{h=1}^{H}\sum_{j=1}^{N} \alpha^{h}_{i,j}\, x^{\ell}_{j\Leftarrow k} W^{h}_{Att} + \omega_k b_{Att}\Big)}_{z^{\ell}_{i\Leftarrow k}}$$

$$\mathrm{LN}(z^{+\ell}_{i}) = \sum_{k=1}^{N} \underbrace{g_{z^{+\ell}_{i}}(z^{+\ell}_{i\Leftarrow k}) + \beta}_{\tilde{z}^{\ell}_{i\Leftarrow k}}$$

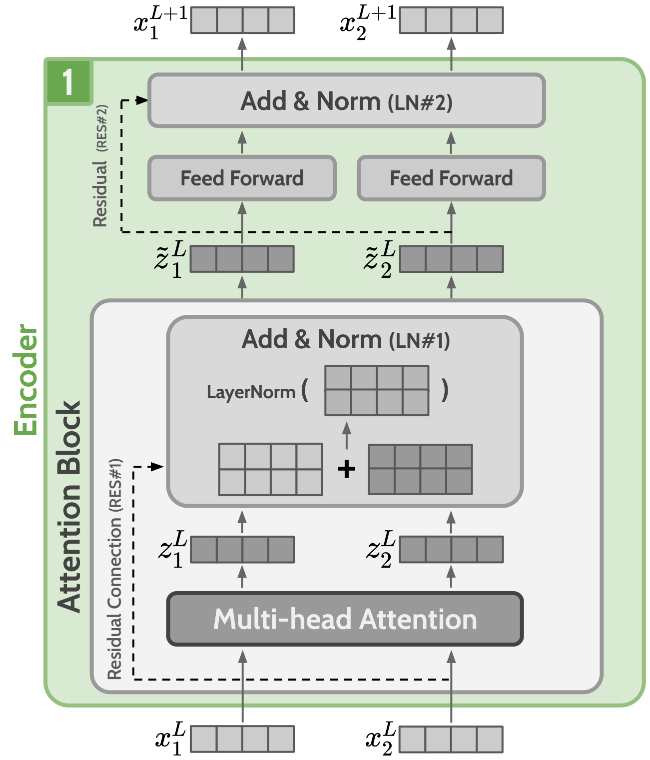
$$z^{\ell}_{\mathrm{FFN},i} = f_{\mathrm{act}}^{(\zeta^{\ell}_{i})}\Big(\sum_{k=1}^{N} \zeta^{\ell}_{i\Leftarrow k}\Big) W^{2}_{\mathrm{FFN}} + b^{2}_{\mathrm{FFN}} = \sum_{k=1}^{N} \underbrace{\theta^{(\zeta^{\ell}_{i})} \odot \zeta^{\ell}_{i\Leftarrow k} + b^{2}_{\mathrm{FFN}}}_{z^{\ell}_{\mathrm{FFN},i\Leftarrow k}}$$

$$x_i^{\ell+1} = \mathrm{LN}\Big(\sum_{k=1}^{N}\big[\underbrace{\tilde{z}^{\ell}_{i\Leftarrow k} + z^{\ell}_{\mathrm{FFN},i\Leftarrow k}}_{z^{\ell}_{\mathrm{FFN}^{+},i\Leftarrow k}}\big]\Big) = \sum_{k=1}^{N} \underbrace{g_{z^{\ell}_{\mathrm{FFN}^{+},i}}(z^{\ell}_{\mathrm{FFN}^{+},i\Leftarrow k}) + \beta}_{x^{\ell+1}_{i\Leftarrow k}}$$



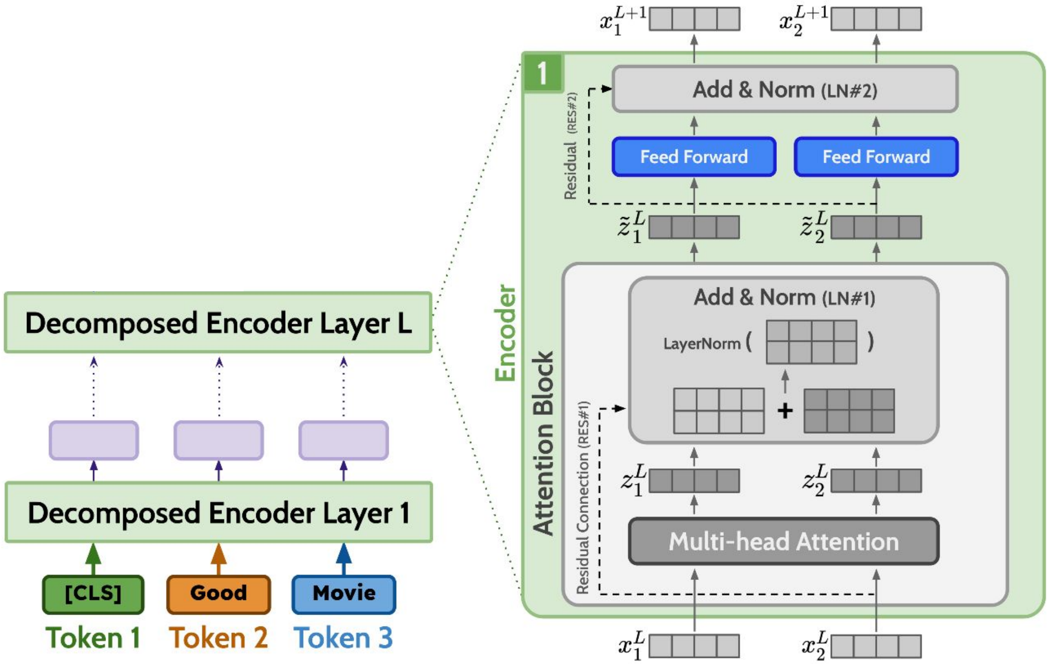
DecompX

Propagating Token Decomposition



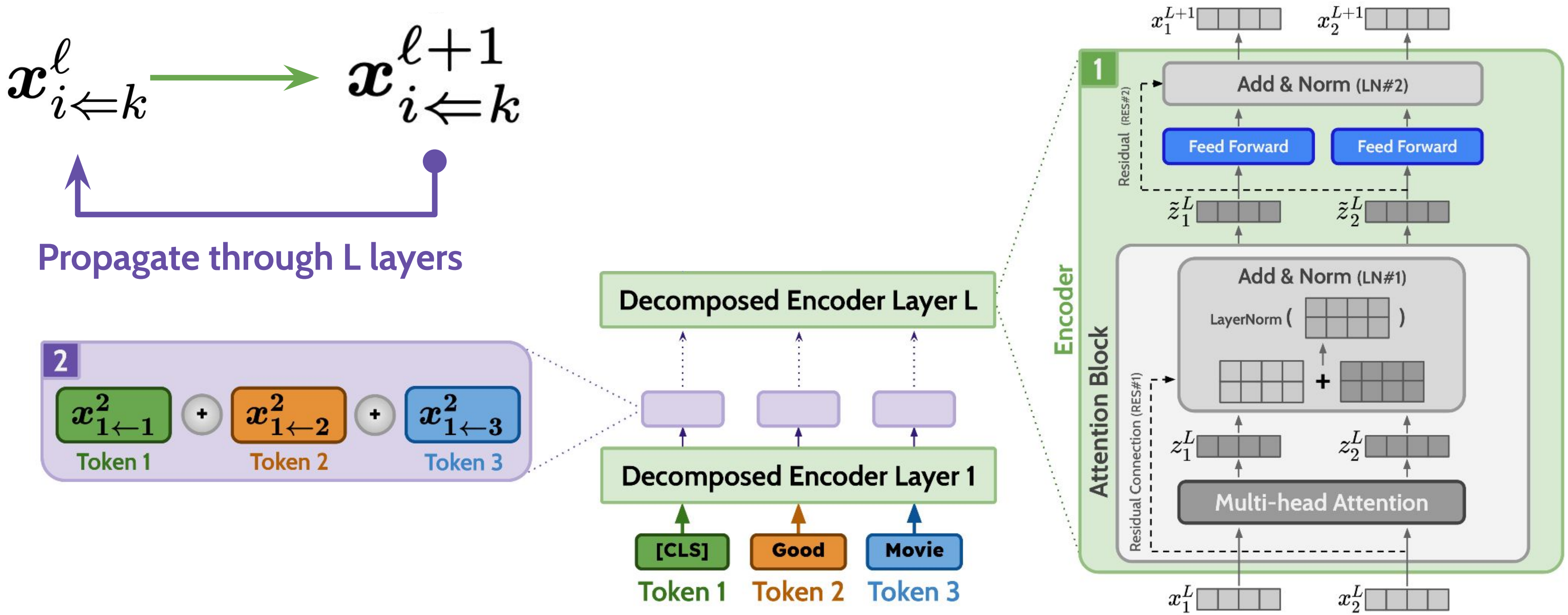
DecompX

Propagating Token Decomposition



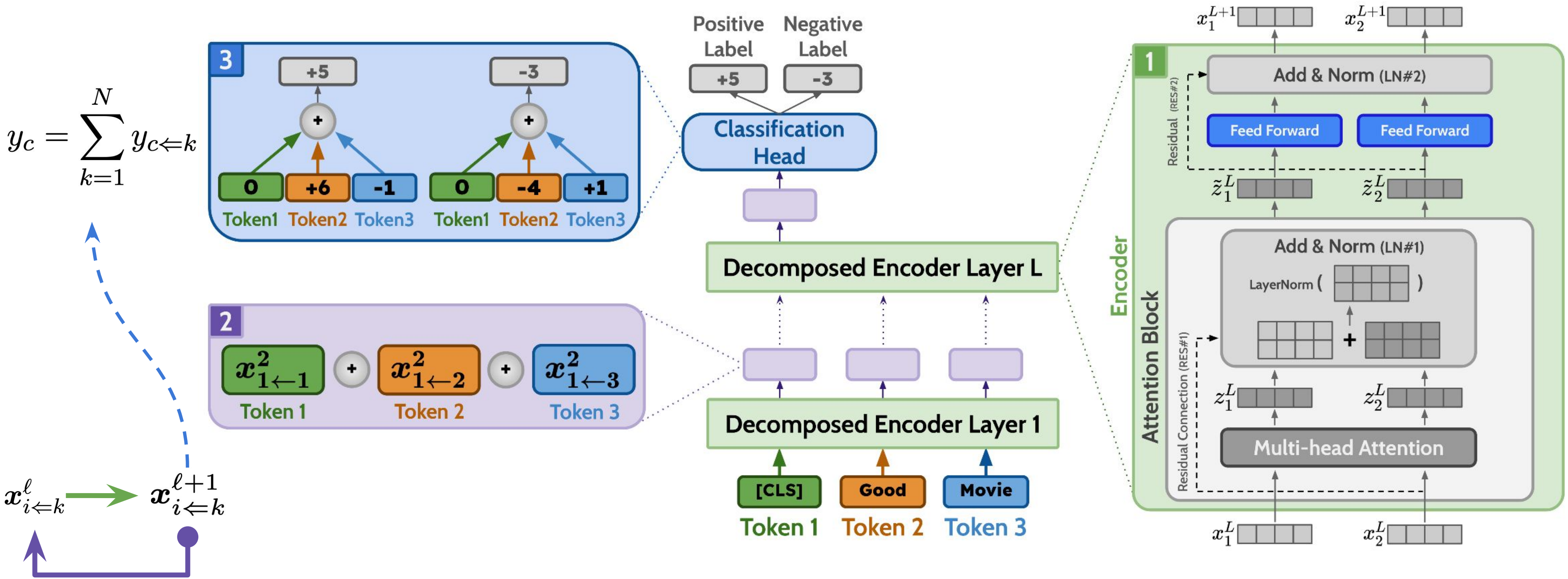
DecompX

Propagating Token Decomposition



DecompX

Propagating Token Decomposition



DecompX

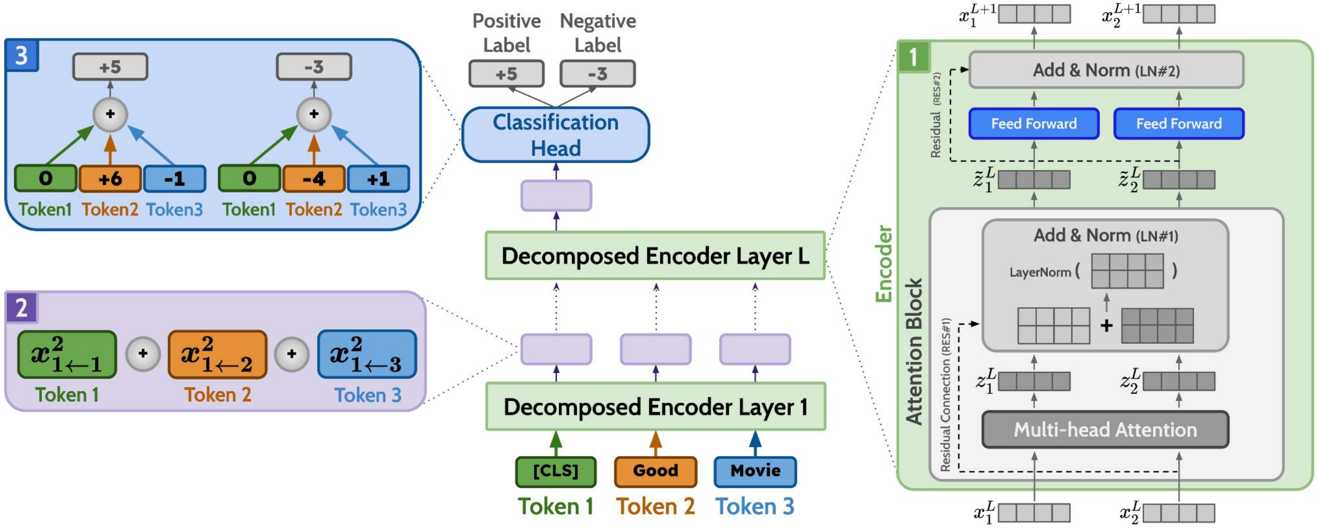
Overview

Our contributions:

1) We incorporated all the encoder layer components including nonlinear functions

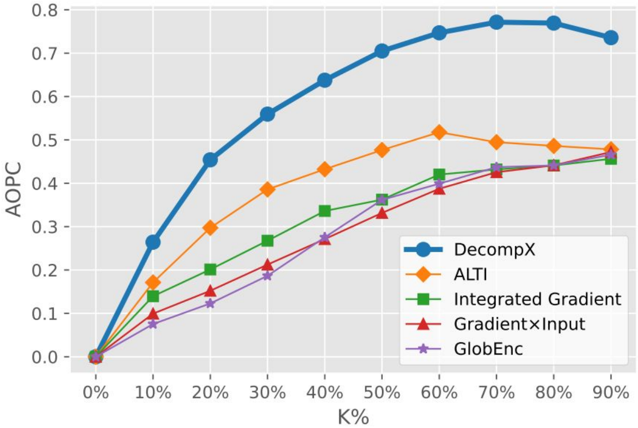
2) Propagated the decomposed vectors throughout the whole model

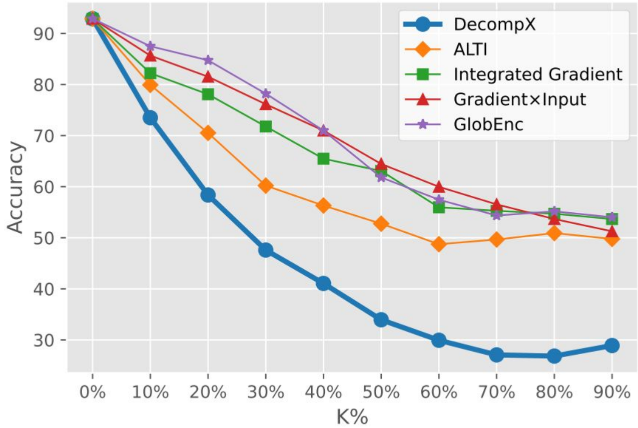
3) Incorporated the classification head



Evaluation

Results





$$\mathrm{AOPC}(K) = \frac{1}{N}\sum_{i=1}^{N} p(\hat{y} \mid x_i) - p(\hat{y} \mid \tilde{x}_i^{(K)})$$

- ➔ **AOPC and Accuracy of different explanation methods on SST2 upon masking K% of the most important tokens.**
- ➔ **DecompX outperforms existing explanation methods, both vector- and gradient-based, by a large margin at every corruption ratio.**

Evaluation

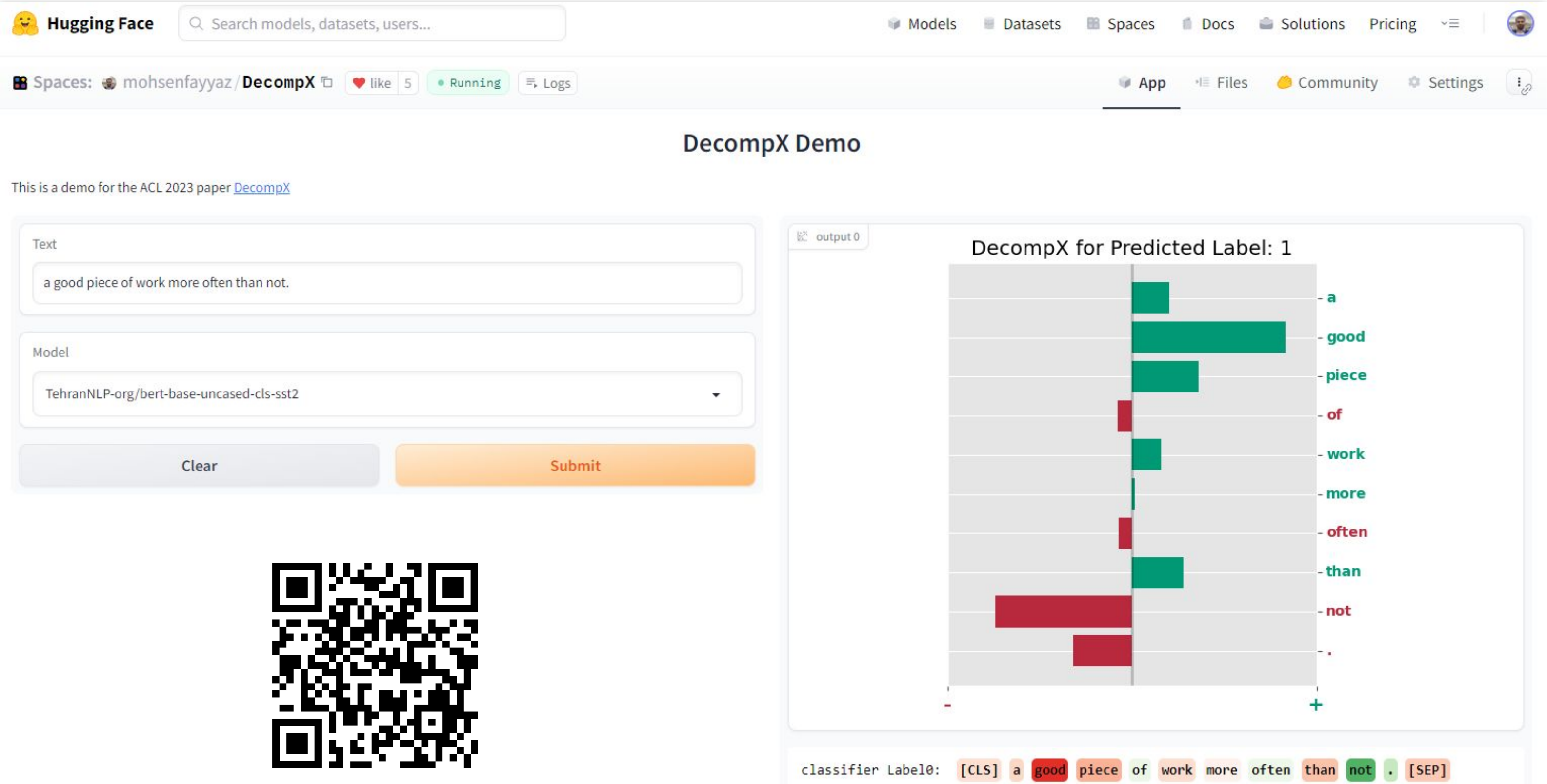
Aggregated Results

	SST2			MNLI			QNLI			HateXplain		
	Acc↓	AOPC↑	Pred↑	Acc↓	AOPC↑	Pred↑	Acc↓	AOPC↑	Pred↑	Acc↓	AOPC↑	Pred↑
GlobEnc (Modarressi et al., 2022)	67.14	0.307	72.36	48.07	0.498	70.43	64.93	0.342	84.00	47.65	0.401	56.50
+ FFN	64.90	0.326	79.01	45.05	0.533	75.15	63.74	0.354	84.97	46.89	0.406	59.52
ALTI (Ferrando et al., 2022)	57.65	0.416	88.30	45.89	0.515	74.24	63.85	0.355	85.69	43.30	0.469	64.67
Gradient×Input	66.69	0.310	67.20	44.21	0.544	76.05	62.93	0.366	86.27	46.28	0.433	60.67
Integrated Gradients	64.48	0.340	64.56	40.80	0.579	73.94	61.12	0.381	86.27	45.19	0.445	64.46
DecompX	**40.80**	**0.627**	**92.20**	**32.64**	**0.703**	**80.95**	**57.50**	**0.453**	**89.84**	**38.71**	**0.612**	**66.34**

➔ Accuracy, AOPC, and Prediction Performance of DecompX compared with the existing methods on different datasets.

➔ DecompX consistently outperforms other methods, which confirms that a holistic vector-based approach can present higher-quality explanations.

Online Demo



Github.com/mohsenfayyaz/DecompX

THANK YOU!

amodaresi@cis.lmu.de
mohsen.fayyaz77@ut.ac.ir
Eaghazade1998@ut.ac.ir
y.yaghoobzadeh@ut.ac.ir
mp792@cam.ac.uk

Github.com/mohsenfayyaz/DecompX