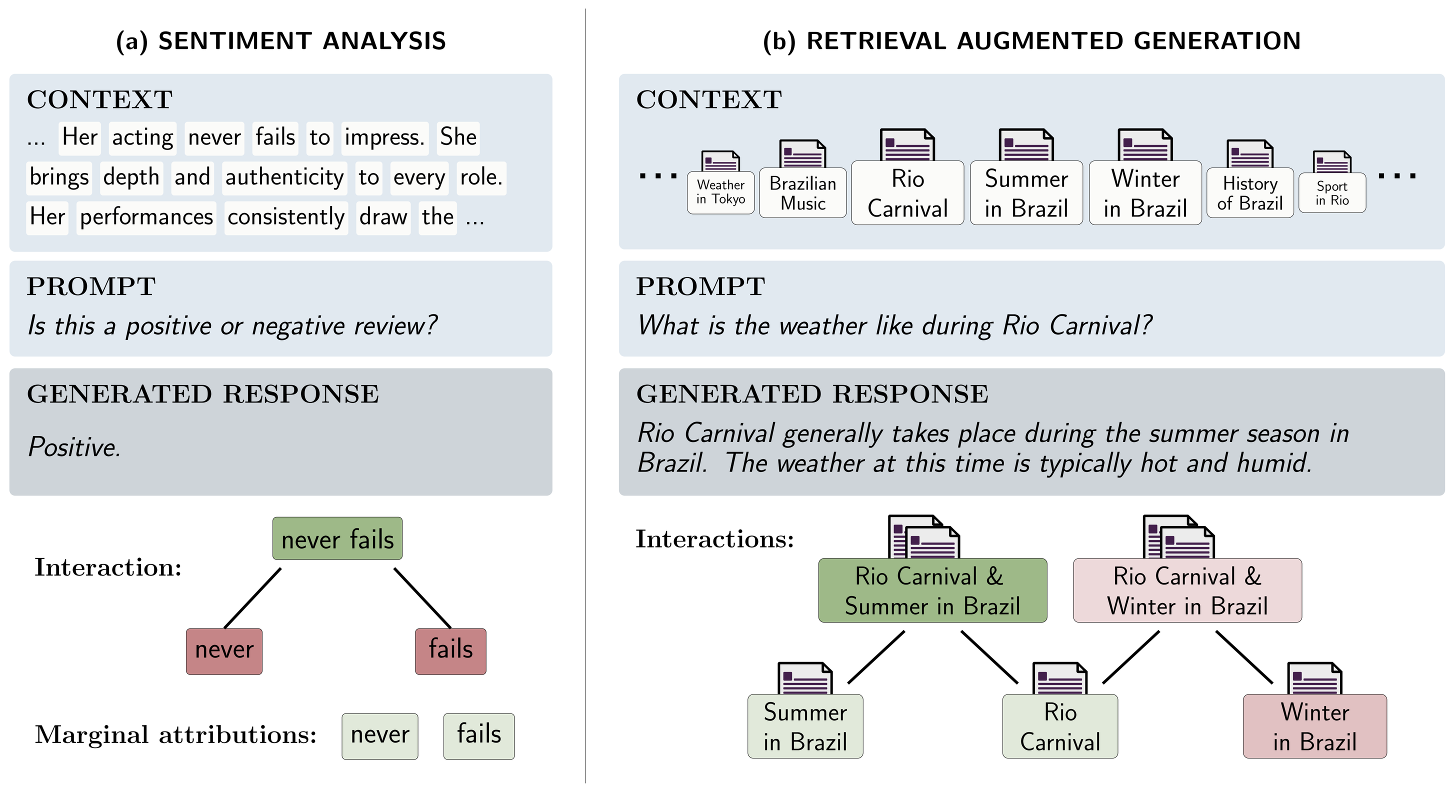
# SPEX: Scaling Feature Interaction Explanations for LLMs

J. S. Kang[1*] L. Butler[1*] A. Agarwal[2*] Y. E. Erginbas[1] R. Pedarsani[3] B. Yu[12] K. Ramchandran[1]

[1]UC Berkeley EECS [2]UC Berkeley Statistics [3]UC Santa Barbara *Equal Contribution

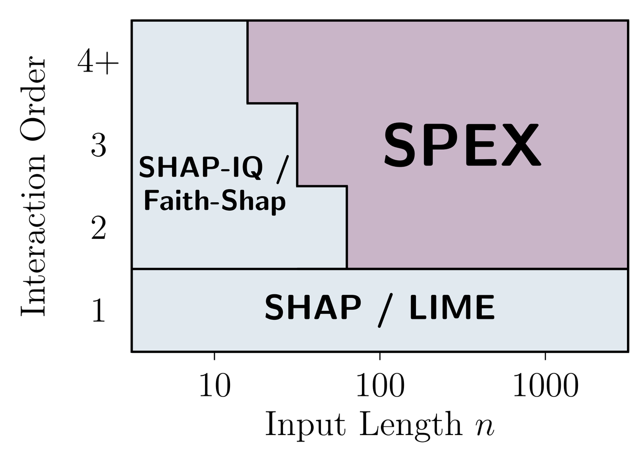
## Problem

LLMs identify important interactions between inputs. Can signal processing and information theory help efficiently identify these interactions using only query access to the LLM?




Example: Tasks can require using interactions between inputs to generate responses.

- Marginal approaches like SHAP/LIME scale, but don't capture important interactions.
- Existing interaction identification approaches are too slow to scale for practical LLM input sizes.
- Our approach, SPEX, scales to large inputs *and* captures interactions.



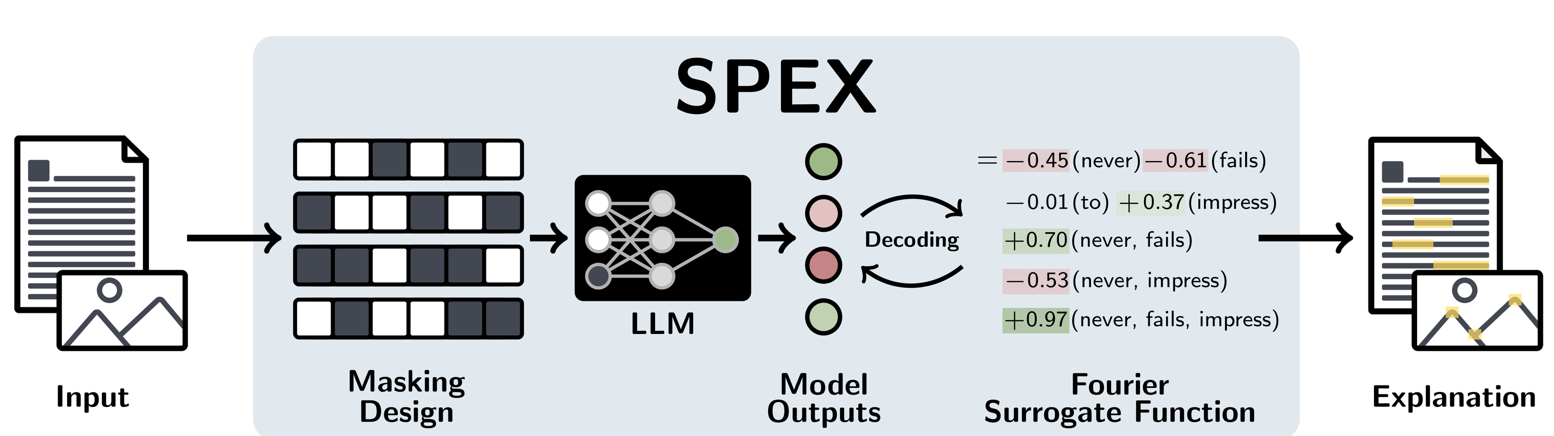

## Formulation as Fourier Transform

- For input $\mathbf{x}$ = "Her acting fails to impress", let $f(\mathbf{x}_S)$ be the output of the LLM under *masking pattern* $S$.
- If $S = \{3\}$, then $\mathbf{x}_S$ is "Her acting `[MASK]` fails to impress", this masking pattern changes the score from positive to negative.
- Equivalently write $f : \mathbb{F}_2^n \to \mathbb{R}$, where $f(\mathbf{x}_S) = f(\mathbf{m})$ with $S = \{i : m_i = 1\}$. Then the Fourier transform is defined as follows:

Forward: $F(\mathbf{k}) = \frac{1}{2^n}\sum_{\mathbf{m}\in\mathbb{F}_2^n}(-1)^{\langle \mathbf{k},\mathbf{m}\rangle} f(\mathbf{m})$ Inverse: $f(\mathbf{m}) = \sum_{\mathbf{k}\in\mathbb{F}_2^n}(-1)^{\langle \mathbf{m},\mathbf{k}\rangle}F(\mathbf{k})$.

We find that $F(\mathbf{k}) \approx 0$ for most $\mathbf{k}$ (sparsity), and most large $F(\mathbf{k})$ are low degree such that $|\mathbf{k}| \le d$ for some small $d$.

- SPEX exploits this sparsity using codes, to compute interactions efficiently, by computing estimates $\hat{F}(\mathbf{k})$ for a small (a-priori unkown) set of $\mathbf{k} \in \mathcal{K}$.
- Inverting our estimated $\hat{F}(\mathbf{k})$ gives us an approximate surrogate function $\hat{f}$.




SPEX utilizes codes to determine masking patterns. We observe the changes in model output depending on the used mask. SPEX uses message passing to learn Fourier coefficients to generate interaction-based explanations.

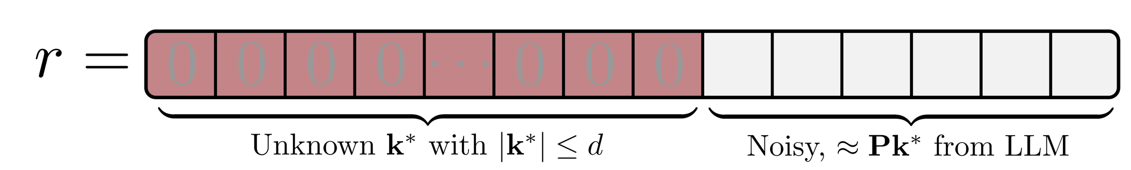
## Algorithm

**Step 1: Masking Design - Embedding Code Structures Through Aliasing**

- We collect samples according to two matricies $\mathbf{M} \in \mathbb{F}_2^{b\times n}$ and $\mathbf{P} \in \mathbb{F}_2^{p\times n}$.

$$u_{c,i}(\boldsymbol{\ell}) = f(\mathbf{M}_c^{\mathrm{T}}\boldsymbol{\ell} + \mathbf{p}_i) \iff U_{c,i}(\mathbf{j}) = \sum_{\mathbf{k}\,:\,\mathbf{M}_c\mathbf{k}=\mathbf{j}} (-1)^{\langle \mathbf{p}_i,\mathbf{k}\rangle}F(\mathbf{k}).$$

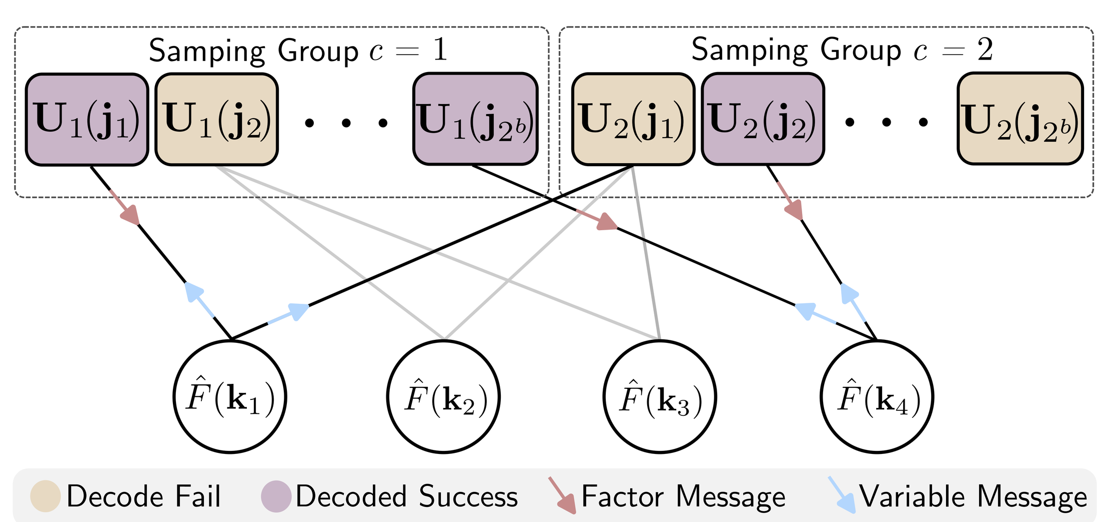
- Depending on $\mathbf{p}_i$, the modulation $(-1)^{\langle \mathbf{p}_i,\mathbf{k}\rangle}$ changes the sign of $F(\mathbf{k})$.
- Each $U_{c,i}(\mathbf{j})$ can be seen as a noisy BPSK message containing a codeword $\mathbf{P}\mathbf{k}^*$ conveying a dominant $\mathbf{k}^*$ in the sum above.




- If $\mathbf{P}$ is a parity matrix of a systematic code, we can decode $r$ to recover dominant $\mathbf{k}^*$. This can be seen as a form of *joint source channel coding*.

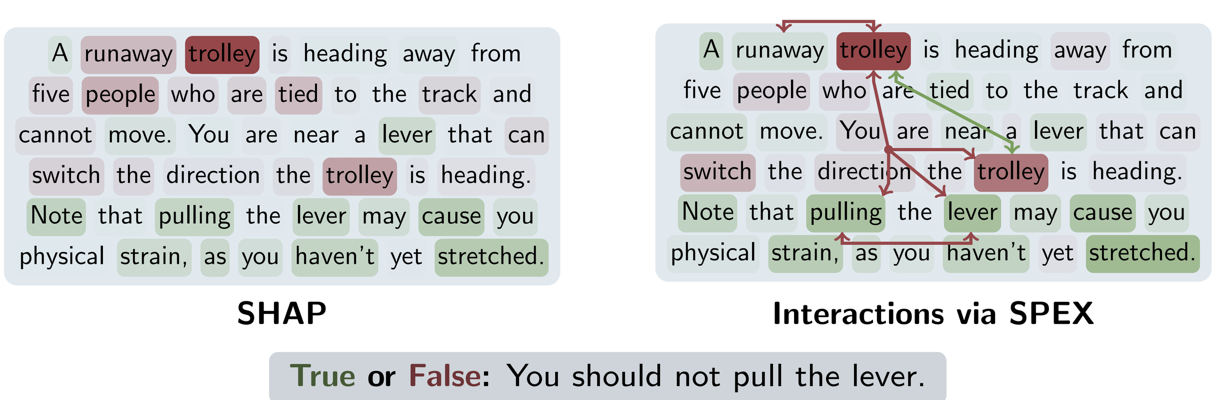
**Step 2: Message Passing - Decoding and Interference Cancellation**

- Defines a bipartite graph connecting the non-zero $F(\mathbf{k})$ and $U$.
- As we recover $\hat{F}(\mathbf{k})$ and $\mathbf{k}$, we can do interference cancellation via message passing. This is inspired by sparse graph codes for robust communication.




- We can analyze the message passing with density evolution theory.

## Case Studies: Applications

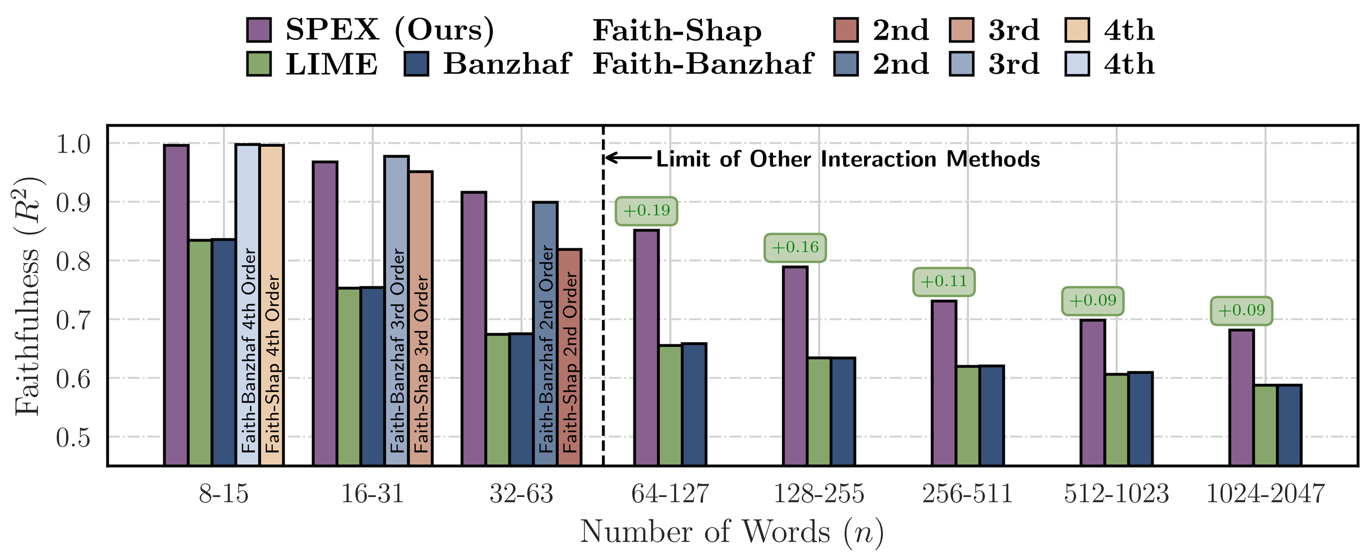




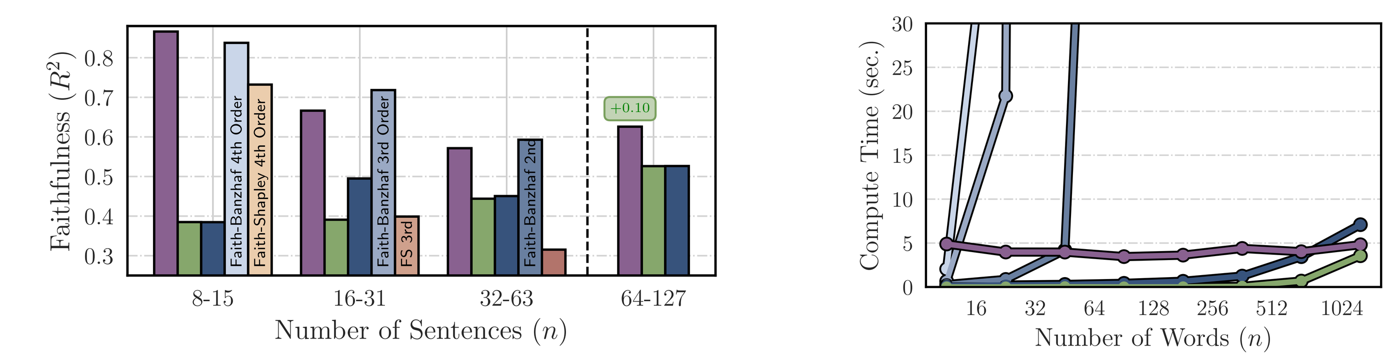
**Abstract Reasoning Errors:** LLMs struggle with modified versions of puzzle questions. We consider a variant of the classic trolley problem. *GPT-4o mini* incorrectly answers. We identify a strong interaction between words that commonly appear in the standard problems.

**Visual Question Answer:** We prompt *LLaVA-NeXT-Mistral* with *"What is shown in this image?"* for the image above. SHAP indicates the importance of image patches containing the ball and the dog. SPEX shows that the presence of *both* the dog and the basketball jointly are critical.

## Experiments



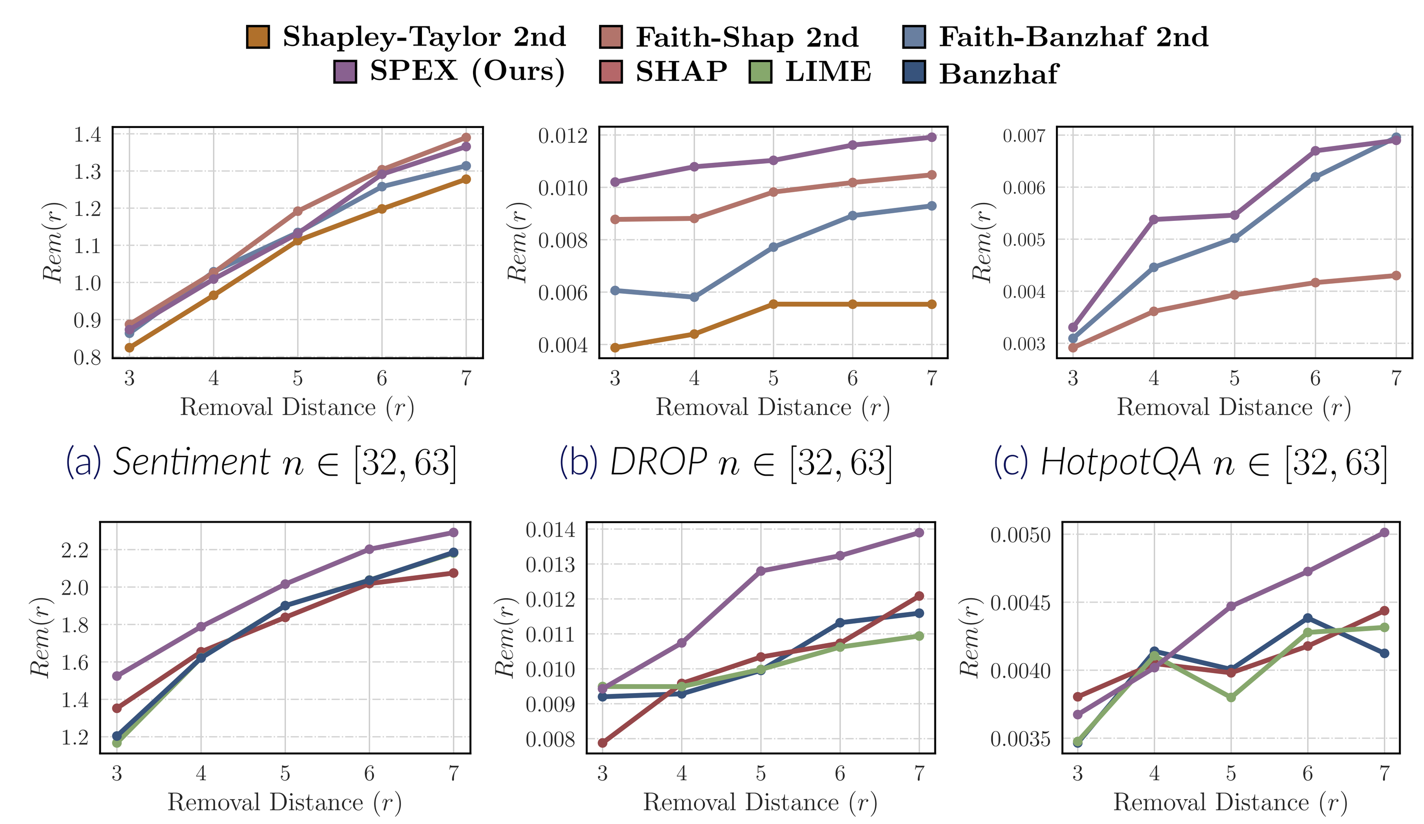

(a) *Sentiment*




(b) *HotpotQA* (c) *Compute Time*

**Faithfulness:** *Faithfulness* to the real function $f$, defined in terms of $R^2$:

$$R^2 = 1 - \frac{\|\hat{f} - f\|^2}{\|f - \bar{f}\|^2}, \quad \|f\|^2 = \sum_{\mathbf{m}\in\mathbb{F}_2^n} f(\mathbf{m})^2 \quad \bar{f} = \frac{1}{2^n}\sum_{\mathbf{m}\in\mathbb{F}_2^n} f(\mathbf{m}).$$

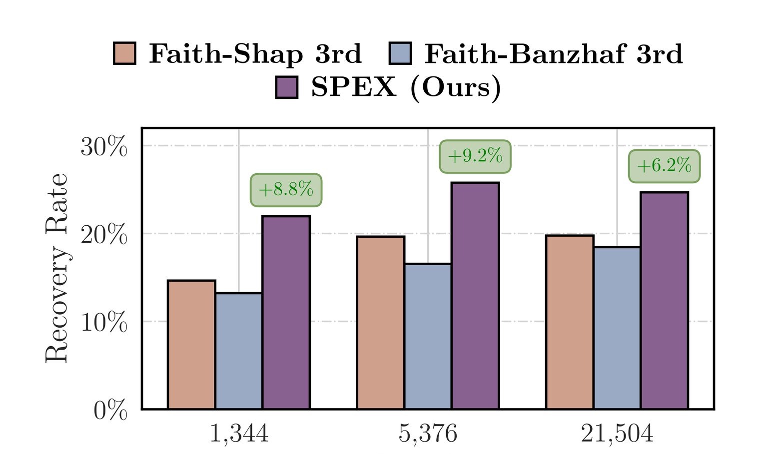


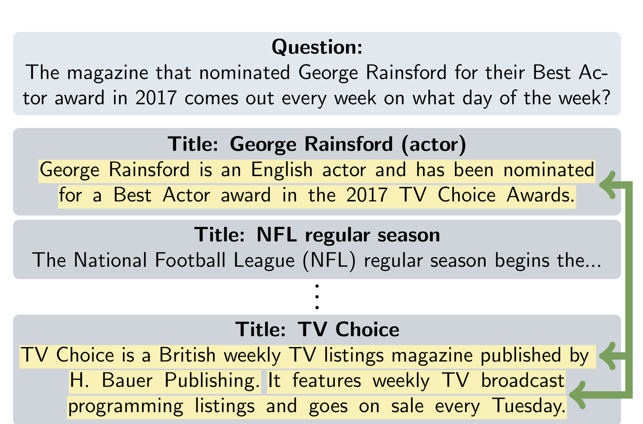

(a) *Sentiment* $n \in [32, 63]$ (b) *DROP* $n \in [32, 63]$ (c) *HotpotQA* $n \in [32, 63]$

(d) *Sentiment* $n \in [64, 127]$ (e) *DROP* $n \in [64, 127]$ (f) *HotpotQA* $n \in [64, 127]$

**Top-$r$ Removal:** We identify the top $r$ influential features to model output:

$$\mathrm{Rem}(r) = \frac{|f(\mathbf{1}) - f(\mathbf{m}^*)|}{|f(\mathbf{1})|}, \quad \mathbf{m}^* = \underset{|\mathbf{m}|=n-r}{\arg\max}\,|\hat{f}(\mathbf{1}) - \hat{f}(\mathbf{m})|.$$




(Left) Recovery of Humal-labeled interactions in *HotpotQA*. (Right) Example interaction.

**Recovery Rate@r:** Let $S_r^* \subseteq [n]$ denote human-annotated sentence. Let $S_i$ denote feature indices of the $i^{\text{th}}$ most important interaction.

$$\text{Recovery@}r = \frac{1}{r}\sum_{i=1}^{r}\frac{|S_r^* \cap S_i|}{|S_i|}.$$

International Conference on Machine Learning, Vancouver, July 2025