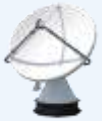
Exploring and Explaining Attention Models from Language to Vision

Dr. Elizabeth Hou

elizabeth.hou@str.us

cyber analytics sensors systems impact.

STR is a national security company






RF, Acoustics, Optics

- Radar/EW HW systems
- Acoustic payloads
- Algorithms & modes
- Open architecture processors
- Sensing autonomy, orchestration
- Counter-sensing technologies

Cyber

- Vulnerability research
- Reverse engineering
- Protection & attack mitigation
- Agent framework
- Full system M&S
- Attack vector generation

Analytics, C2

- Target recognition, tracking
- Pattern of life, behavioral analysis
- Counter-AI/ML
- Info ops
- Resource allocation
- Weapon-target-pairing; Autonomy

Systems Dev

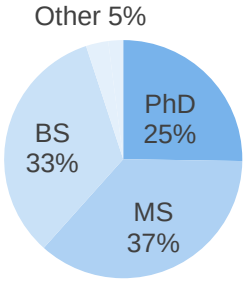
- Product Lifecycle Support
- Systems Eng. & Analysis
- MBSE / Digital Eng. solutions
- Quality Assurance
- Configuration Management

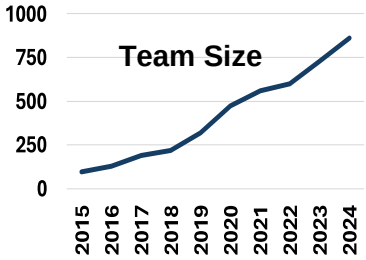
C5ISR&T Capabilities | **C-C5ISR&T Capabilities** | **C-/C5ISR&T Test Capabilities**

Develop and transition advanced technology for DoD and IC customers

- DARPA contractor, supporting all offices
- IARPA, DoD services, IC member programs
- Independent – 95% of work as prime, with 100+ small biz & university subs
- Advanced technology; mission focused; S&T → operations

Talent: Scientific & engineering staff with broad range of skills





Independent: mission understanding, cleared staff, growing impact

Internship* Program

We look out for your whole experience. Our summer internship program even includes housing, which takes the stress out of apartment hunting and lets you hang out with your fellow interns. We also provide social events so you can meet a lot of other people across the company and discover their stories.

What will the day-to-day work consist of during an STR internship?
Students will work on real programs with STR, no fake problems or concepts. We want our interns to gain real experience.

When do the internship and co-op programs happen?
STR offers Spring and Fall Co-ops in addition to our Summer Internship program.

How would an STR internship help me grow professionally?
Each intern is matched with a mentor who will assist with an assigned summer project. Interns will have the opportunity to give a company presentation on what they've learned at the end of the program.

SCAN HERE TO LEARN MORE AND APPLY

Or visit str.us/internships/

*Interns must be US Citizens

Woburn, MA | Arlington, VA | Dayton, OH | Melbourne, FL | Carlsbad, CA

Outline

Decoding Layer Saliency in Language Transformers

Elizabeth M. Hou [1] Gregory Castanon [1]

Abstract

In this paper, we introduce a strategy for identifying textual saliency in large-scale language models applied to classification tasks. In visual networks where saliency is more well-studied, saliency is naturally localized through the convolutional layers of the network; however, the same is not true in modern transformer-stack networks used to process natural language. We adapt gradient-based saliency methods for these networks, propose a method for evaluating the degree of semantic coherence of each layer, and demonstrate consistent improvement over numerous other methods for textual saliency on multiple benchmark classification datasets. Our approach requires no additional training or access to labelled data, and is comparatively very computationally efficient.

architectures, with convolutional heads to visual networks encouraging local associations while stacks of fully connected transformer layers allow natural language tokens to associate more globally. This free association, combined with the degree of model complexity in transformer architectures, leads to challenges in interpretability, as not all feature spaces within the hidden layers of the network map cleanly to natural language.

Current methods that explain the decision making processes of transformer-stack architectures focus on the embedding layer. However, these methods often result in confusing or redundant explanations, as information gets muddled passing through multiple layers of transformers in the stack. Along with (Rogers et al., 2020), we hypothesize that a more meaningful, clear, decision-oriented representation exists in solely the later layers of the network. In this paper, we propose a method that only captures the signal of the later layers of the transformer stack and projects it back

ICML 2023

https://proceedings.mlr.press/v202/hou23a/hou23a.pdf

Ties to Recent Advances in LLMs

Generative LLMs

Vision Transformers

Summary of Decoding Layer Saliency in Language Transformers

- **Problem:** Saliency (importance of words for a task) in natural language isn't local and transformer stack language models capture a lot of "other" information (language structure, syntax, etc)
- **Goal:** Provide better explainability through more meaningful, clear, decision-oriented representations from the information encoded in the hidden layers of an encoder based transformer stack
- **Method:** Assign explanatory power to tokens in an input sequence from layer specific saliency scores calculated using information only downstream from that layer
- **Algorithm:** Computationally efficient projection of a layer's saliency score using a pre-trained language model head as the mapping function

Final Solution Example

Our method (green) is able to more accurately assign explanatory power (highlights) to tokens

Simple	**Negative: 0.9809374213218689** unflinchingly bleak and desperate	**Negative: 0.9997243285179138** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.
Smooth	**Negative: 0.9809374213218689** unflinchingly bleak and desperate	**Negative: 0.9997243285179138** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.
Integrated	**Negative: 0.9809374213218689** unflinchingly bleak and desperate	**Negative: 0.9997243285179138** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.
Decoded Grad-CAM l_7	**Negative: 0.94347584** unflinchingly bleak and desperate	**Negative: 0.99634856** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.

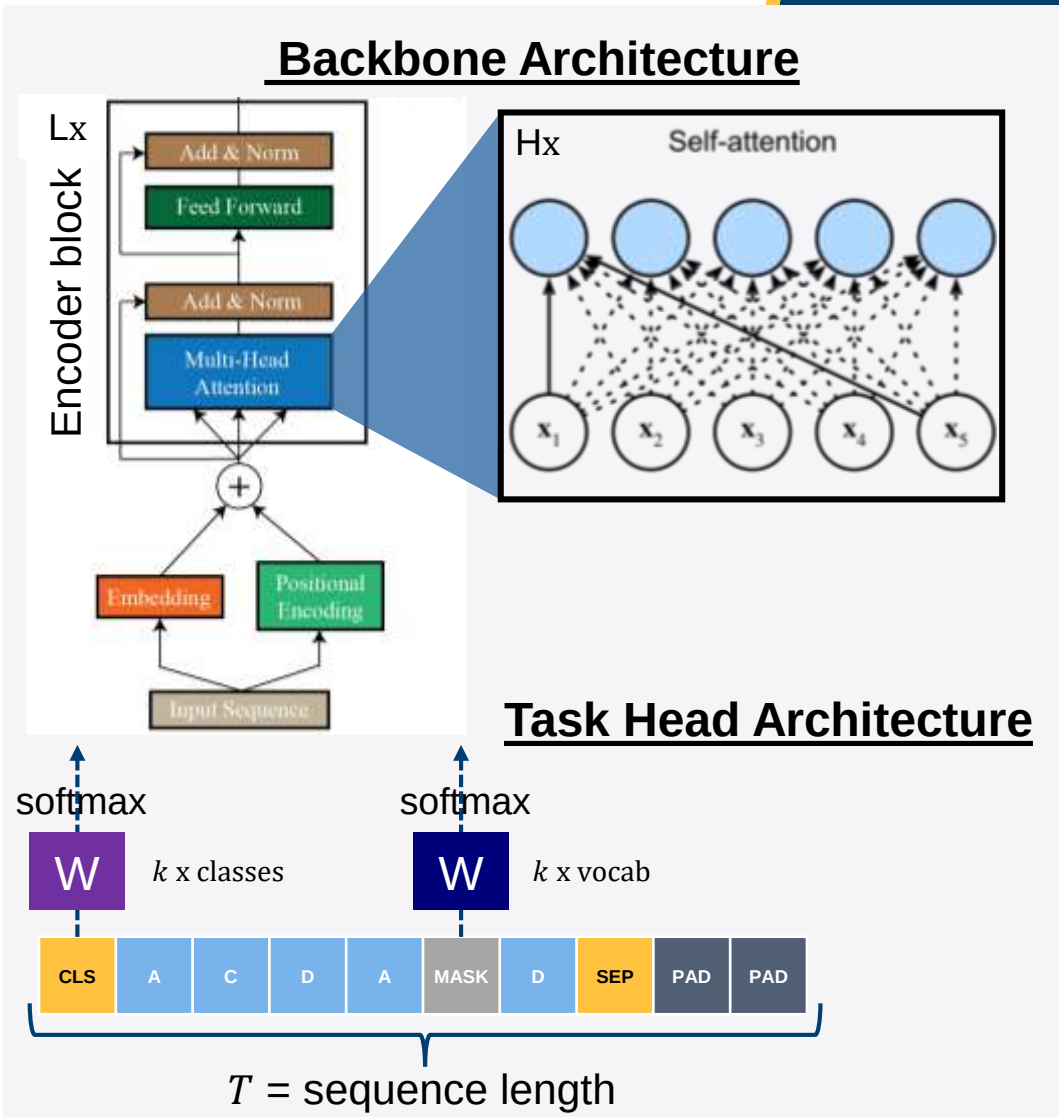
Figure 12. STT-2 Example 1: For the left column example, Decoded Grad-CAM l_7 and to an extent AllenNLP Interpret's Simple highlight the negative sentiment words "bleak" and "desperate", but all three of AllenNLP Interpret's methods also focus on "and". For the right column example, all four methods focus on "awful" and "unrem"(ittingly), but the AllenNLP Interpret's methods are more noisy with highlights on unrelated terms such as "it", "dog" and "constitutes".

Review: Saliency

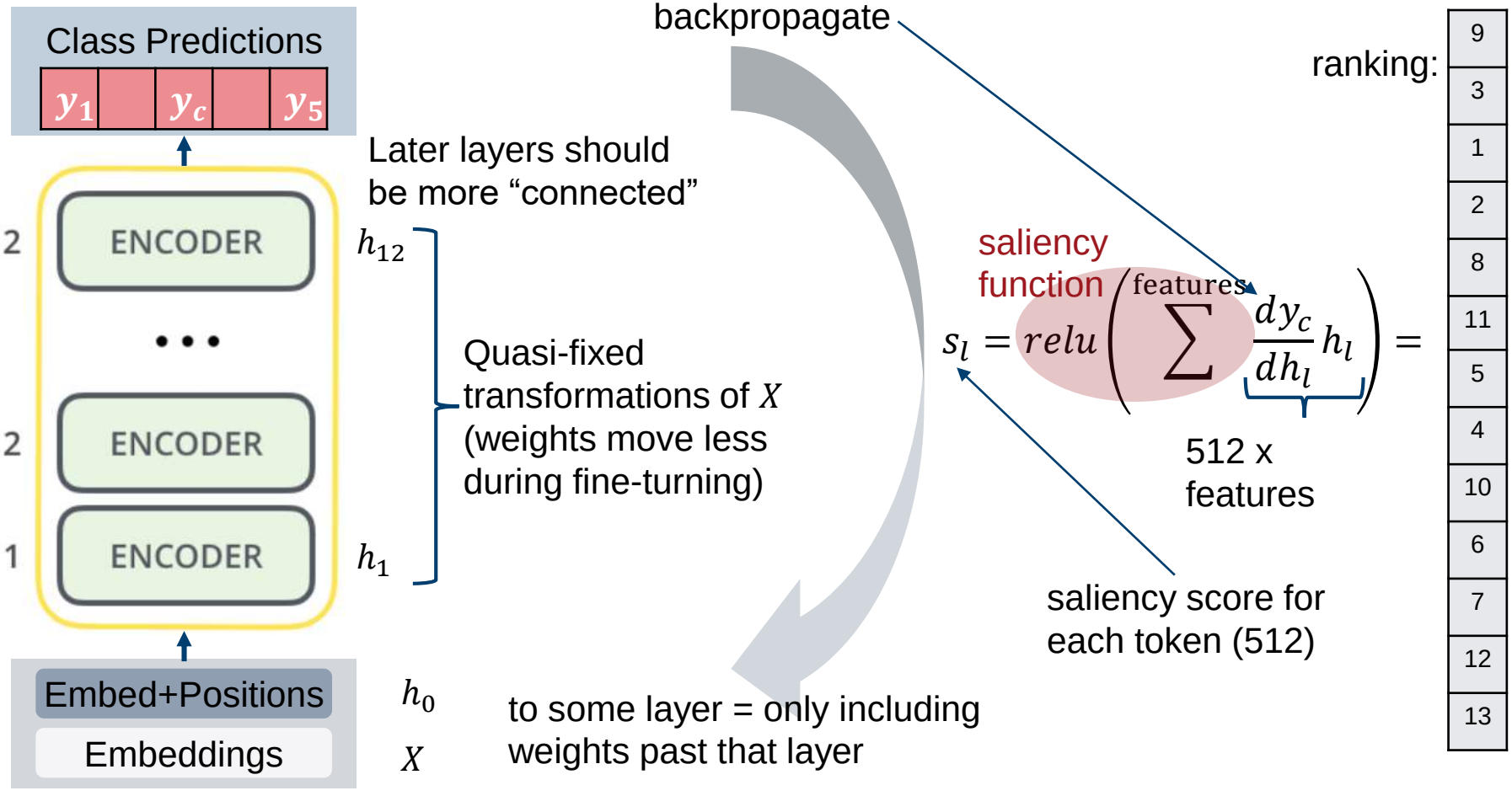
- **Saliency backpropagation techniques originate from Computer Vision (CV) literature**
  - Well studied problem in vision with CNN architectures
  - More open question in NLP
- **We are borrowing the "theory" from CV**
  - CV and NLP problems don't not align completely
- **State of the art NLP models are much more complex than the CNNs and Language is a much more difficult medium to work with**
  - Language has a lot more nuances than images
    - Parts of speech, syntax, vocabulary choice
  - Easier to interpret significance of patches of pixels in images, location matters more
    - 2 (normalized) images of cats have ears, eyes, whiskers in relatively the same area of the image
    - 2 sentences about cats can have the "cat" words in very different parts of the sentence
  - Pixels are 256 x 3 color values and are ordinal i.e. lower number means darker
    - Pixels can be "treated" as continuous variables
  - Tokens occupy discrete extremely large vocab with no ordered meaning (values in a set)
    - Embedding space featurization includes more than just token information (dependent on entire sentence)

Introduction: Transformer Models

- **Transformers are sequence transduction (inputs / outputs are same size) models first introduced in NLP domain**
- **Transformer Encoder Stack (TES) is a series (L) of encoder block layers where each layer consists of:**
  - H **Multi-headed Self-Attention:** each head learns relationships within layer's input
  - **Feed Forward:** uses outputs from each self-attention head to predict layer's output
  - Results in learning a k dim feature embedding
- **Task Heads: Classification / Mask Language Modelling**
  - Linear layer + softmax

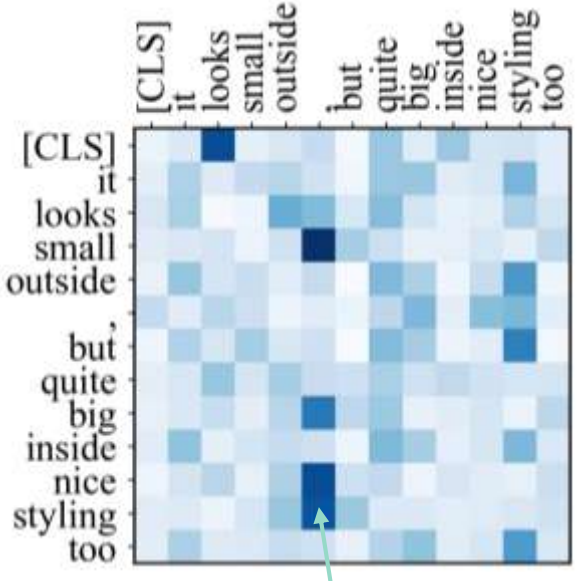


GradCAM (from CV) as Saliency Algorithm



Self-Attention Mechanism Is Nonlinear

- **Inputs / prior layer output: X (**token length x embedding**)**
- **Unnormalized Self-Attention Matrix: $U = \exp(XW_Q W_K^{\mathrm{T}} X^{\mathrm{T}})$**
  - Kernel-like structure
  - $XW_Q W_K^{\mathrm{T}} X^{\mathrm{T}}$ is a weighted similarity
- **Normalized Self-Attention Matrix: $A = \mathrm{diag}(U\mathbf{1})^{-1} U$**
  - divide rows of U by their row sum
  - A is a right stochastic matrix
  - A is a transition matrix
- **Self-Attention Output: $X' = AXW_V$**
  - XW_V is new featurization: token length x V feature length
  - Rows of X' are weighted combination of original inputs
  - Rows of X' *no longer* represent the tokens i.e. do not preserve the same meaning as the feature embedding in the rows of X

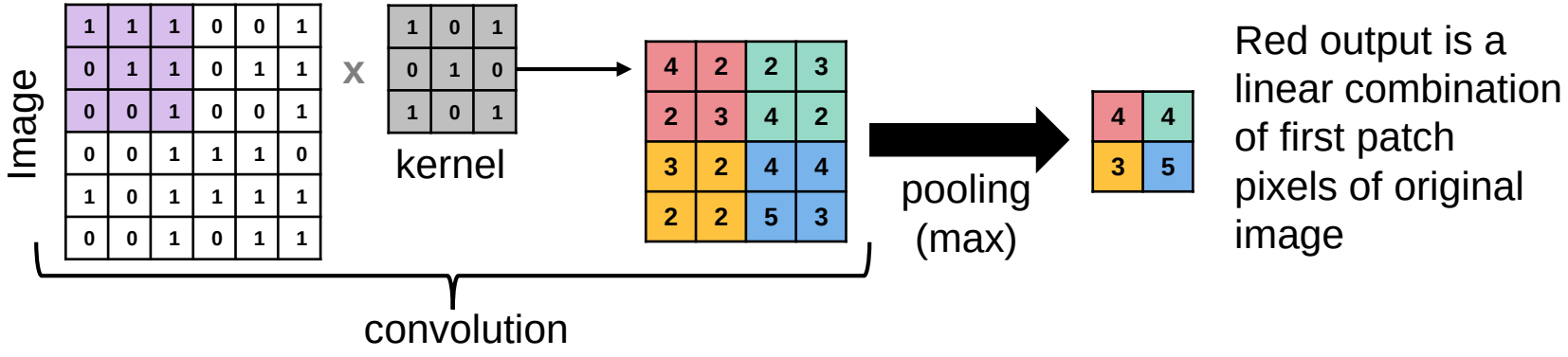


Self-Attention Matrix A

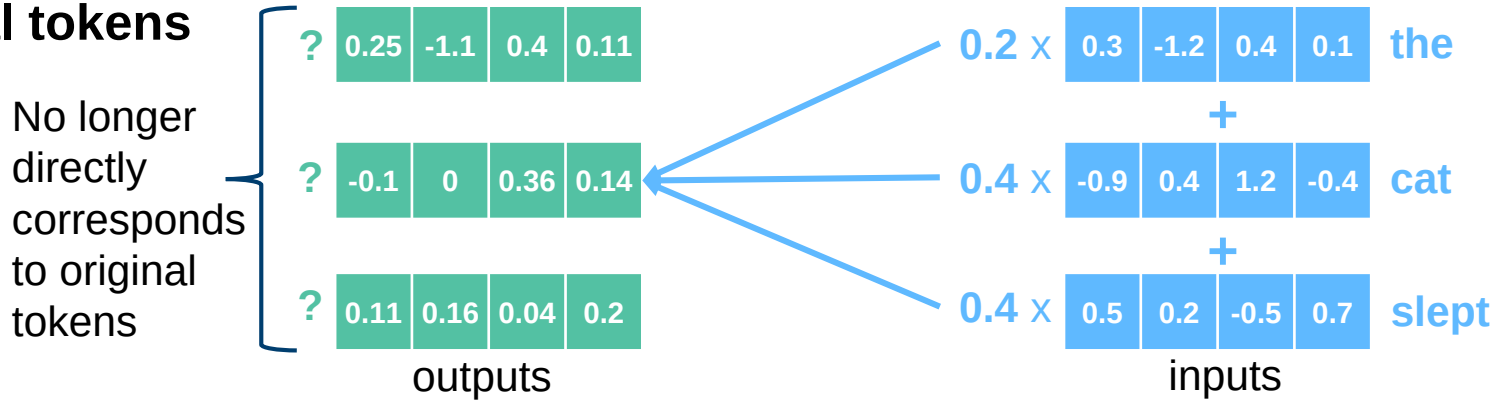
Function of inputs and weights learned during pre-training

From CNN to Transformers

- **CNNs have local structure in the convolutions and pooling**

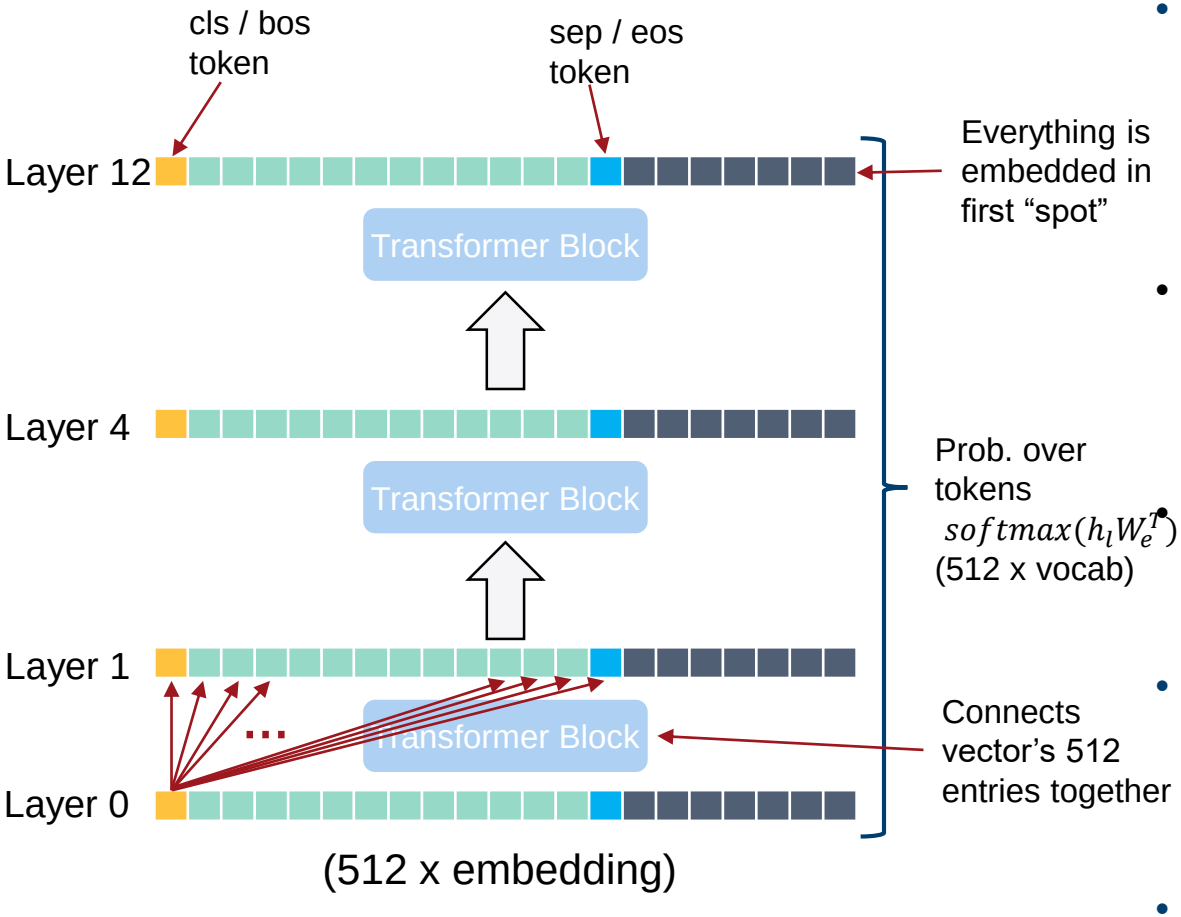


- **Self-attention has global connections that give a weighted average over all original tokens**



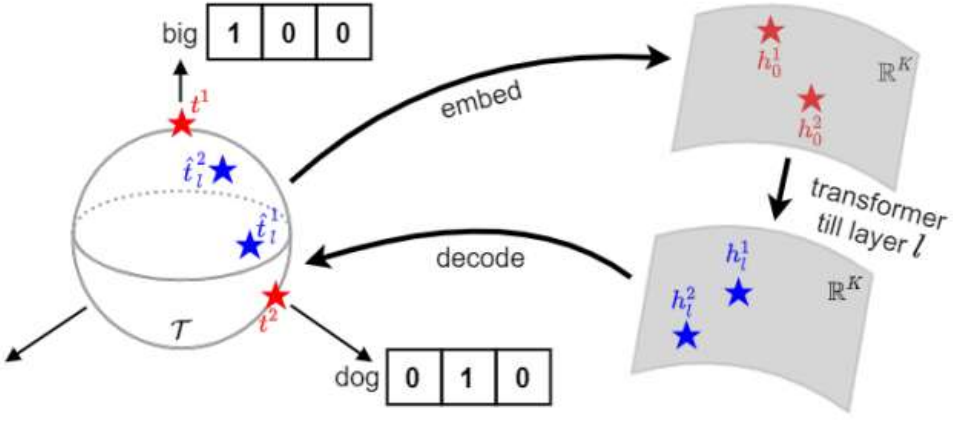
Outputs of attention block are linear combinations of all tokens, so their position in a sequence may not directly correspond to the original tokens

Elements of Vector ≠ Original Tokens



- From CV we know that different layers may produce meaningful, but different saliency "heatmaps"
  - But unlike in CV, it is not easy to "project" the hidden layers back onto original space
- Self-attention uses entire input to learn new representations for some subset of features making each element of output a function of all elements of input
- Problem because the *k* "spot" being indicated as salient ≠ the original token for that spot being salient
- Important to explore saliency of layers > 0 because then only information in the network downstream from that layer is included in the score
- Allows us to ignore potentially task-irrelevant information in earlier layers of the network

Toy Example for Decoding Layer Outputs



1. Input sequence "big dog" tokenized into t^1 and t^2 that lie on unit hypersphere $\mathcal{T}$
2. Tokens are embedded into having K continuous features (h_0^1 and h_0^2)
3. Series of l transformer blocks applied to the embedded input sequence to produce h_l^1 and h_l^2
4. Use $f^{lm}(\cdot)$ to decode onto $\mathcal{T}$
5. Outputs ($\hat{t}_l^1$ and $\hat{t}_l^2$) can be interpreted as weighted combinations of original tokens t^1 and t^2

How do we find $f^{lm}(\cdot)$?

Interpreting Saliency for Intermediate Layers

- Let α_l^c be a $n \times K$ score matrix for layer l indicating the contribution of each element of h_l^k (K features of output of l-th transformer block) to the final classification decision c
- Need to project scores for layers $l > 0$ back into a space where elements of projected vector correspond directly to locations of input sequence tokens
- Language model head $f^{lm}(\cdot)$ minimizes loss between output of transformer block and its closest possible token space representation

$$\arg\min_f \mathcal{L}(t, f(h_l))$$

  where $\mathcal{L}(\cdot)$ is a cross-entropy function, t is the original input sequence, and layer $l = L$ is the final transformer block in the stack
- Conjecture: $f^{lm}(\cdot)$ is estimating the map between $\mathbb{R}^K$ and $\mathcal{T}$ *in general* because the pre-training tasks are performed over an enormous corpus

Calculating Layer Saliency Scores

(Grad-CAM as example score metric)

original score

$$\alpha_l^c = \left[\ldots, \frac{\partial y^c}{\partial h_l^k} h_l^k, \ldots \right] \quad \forall k = 1, \ldots, K$$

$$s_l^c = g(\hat{D}_l \alpha_l^c) = ReLU(\sum_{k=1}^{K} \hat{D}_l \alpha_l^{c,k})$$

- y^c are predictions for class c
- $\hat{D}_l$ is $T \times n$ matrix: rows are columns of $\hat{P}_l = f^{lm}(h_l)$ corresponding to tokens in the input sequence
- s_l^c are saliency scores aggregated with function $g(\cdot)$ over all features for layer l

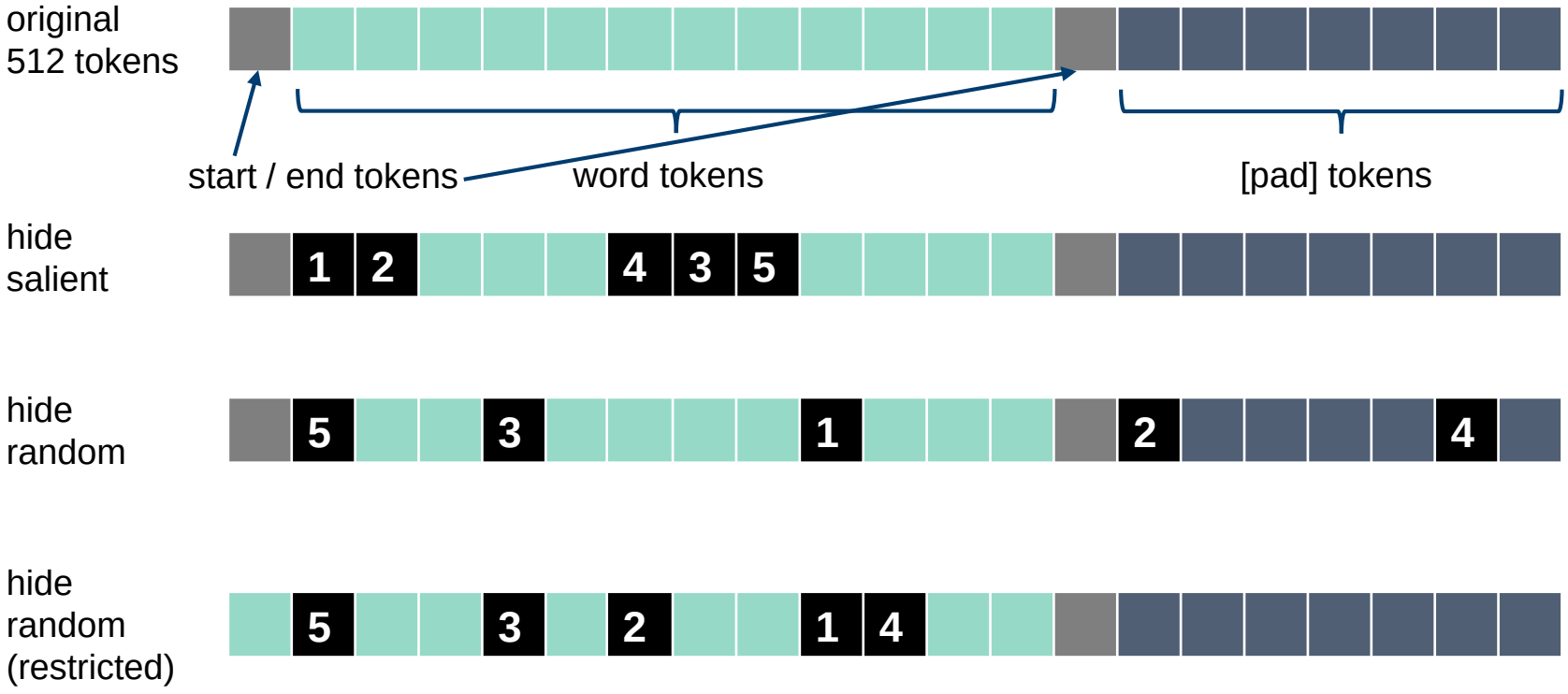
- By using $\hat{D}_l$ to project α_l^c back onto $\mathcal{T}$, elements of s_l^c correspond directly to contributions of each token in original input sequence to LM's final decision
- s_l^c capture *only* information in the model that is downstream from a specific layer l (controls amount of model information used)

The Importance of Intermediate Layers

- **The saliency score at a specific intermediate layer only contains information in the model downstream from it**
  - So we can control for the amount of model information used in a saliency score with the choice in intermediate layer
- **(Rogers et al., 2020) surveys 150 papers and derives potential explanations for the roles of the layers of the BERT (encoder stack transformer) model**
  - Lower layers have the most information about linear word order (i.e. the linear position of a word in a sentence)
  - Middle layers contain syntactic information
  - Final layers are the most task-specific
- **So by only capturing information downstream from a specific layer, we ignore potentially task irrelevant information in the earlier layers of the network**

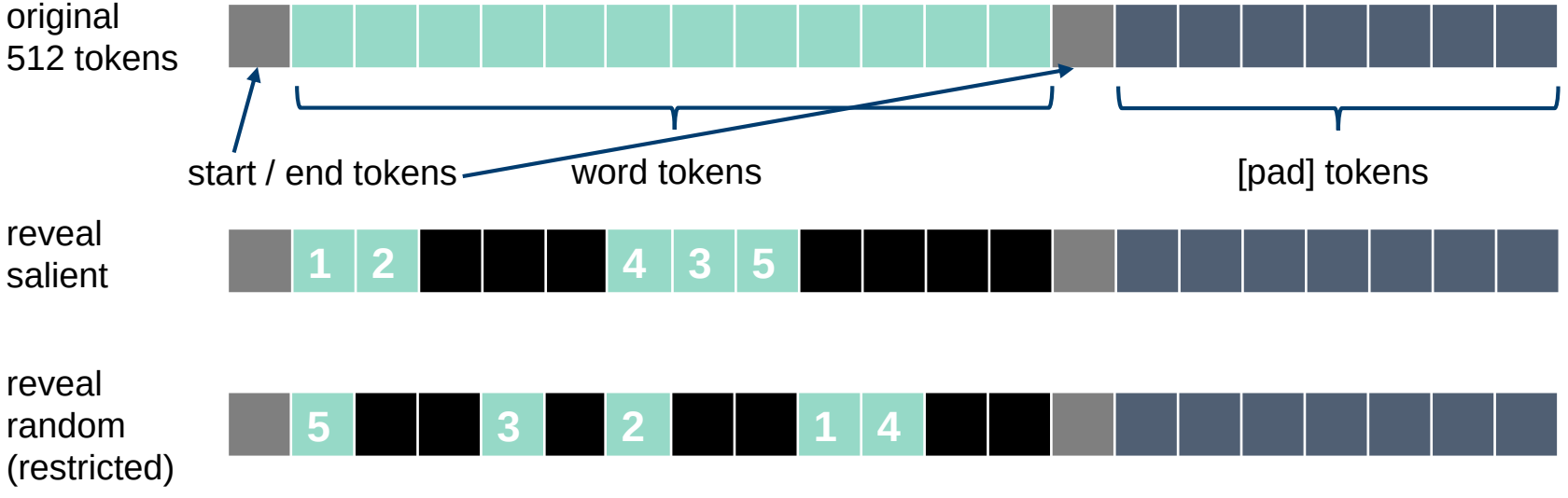
Rogers, Anna, Olga Kovaleva, and Anna Rumshisky. "A primer in BERTology: What we know about how BERT works." Transactions of the association for computational linguistics 8 (2021): 842-866.

Evaluation Scheme: The Hiding Game



- Goal is to observe a decrease in performance when more important tokens are hidden
- In general, creating more out of distribution sentences (strange artifacts due to masking) will *also* decrease performance

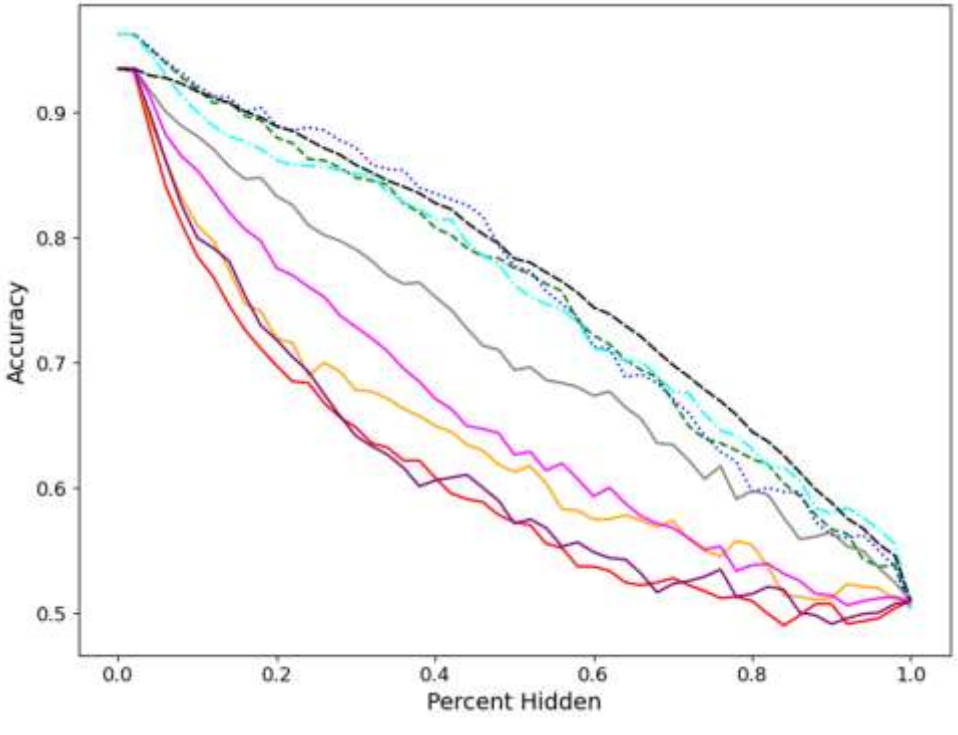
Evaluation Scheme: The Revealing Game

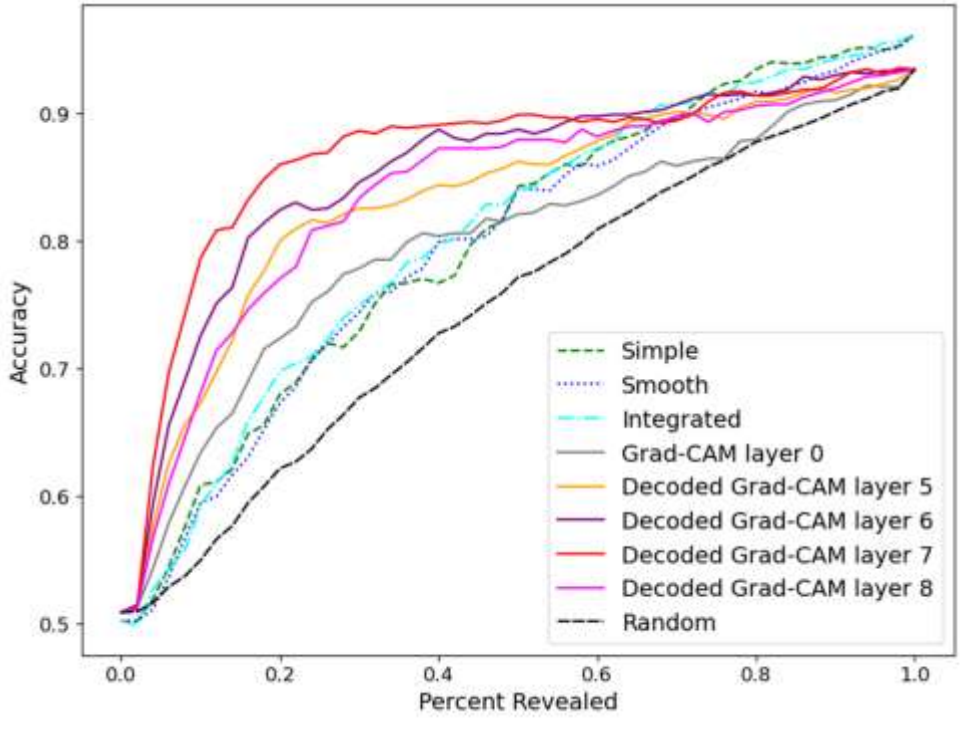


- Goal is to observe an increase in performance when more important tokens are revealed
- Neutralizes out of distribution sentence affect
  - Revealing salient tokens creates less artifacts -> good as this implies some “sense” in the tokens revealed
  - Revealing salient and random create equally as strange artifacts -> no affect on relative performance
- More clear idea of whether salient tokens are *useful* for classification vs hiding game that shows what is *absolutely necessary* for correct classification

Experiments: SST-2 Dataset

Binary classification task - each sample is a sentence from a movie review labeled as either Negative or Positive sentiment

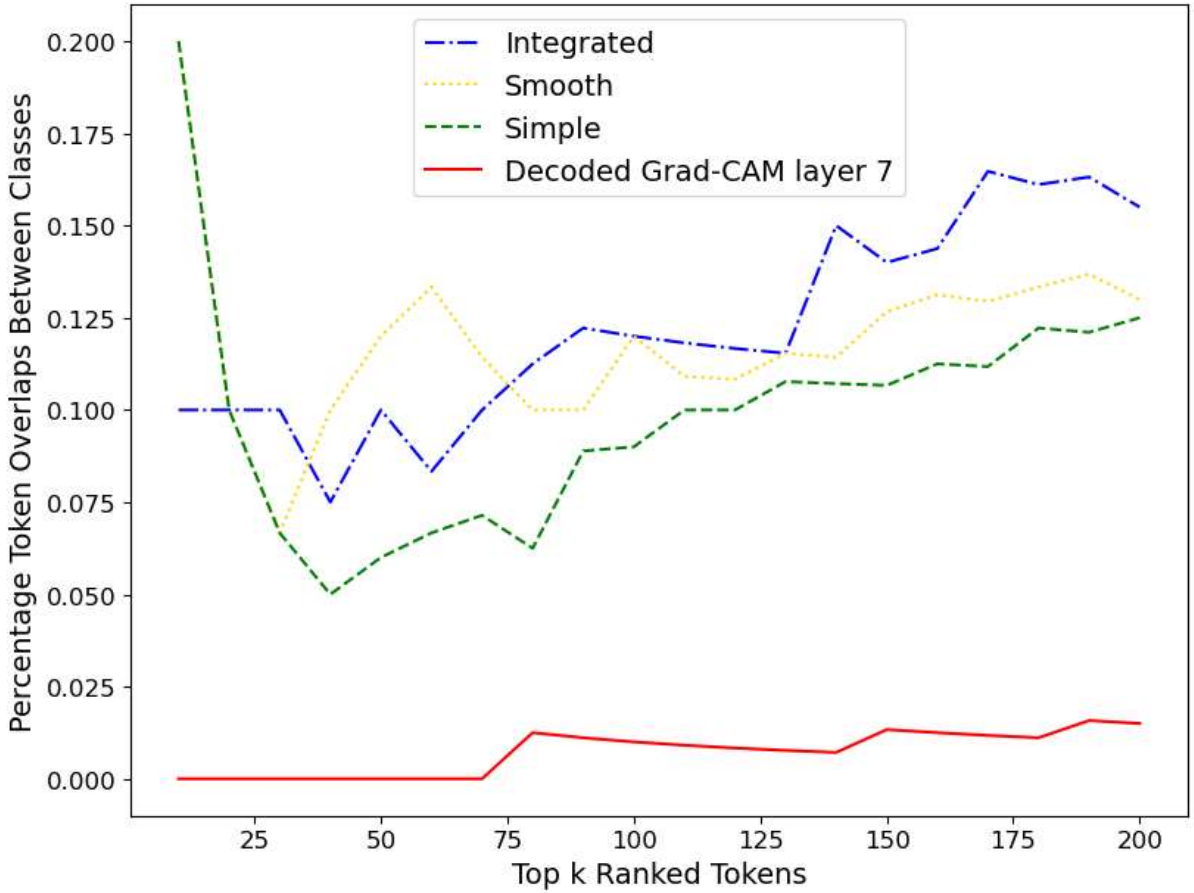




Layer 7 only captures information downstream from it and significantly outperforms Layer 0

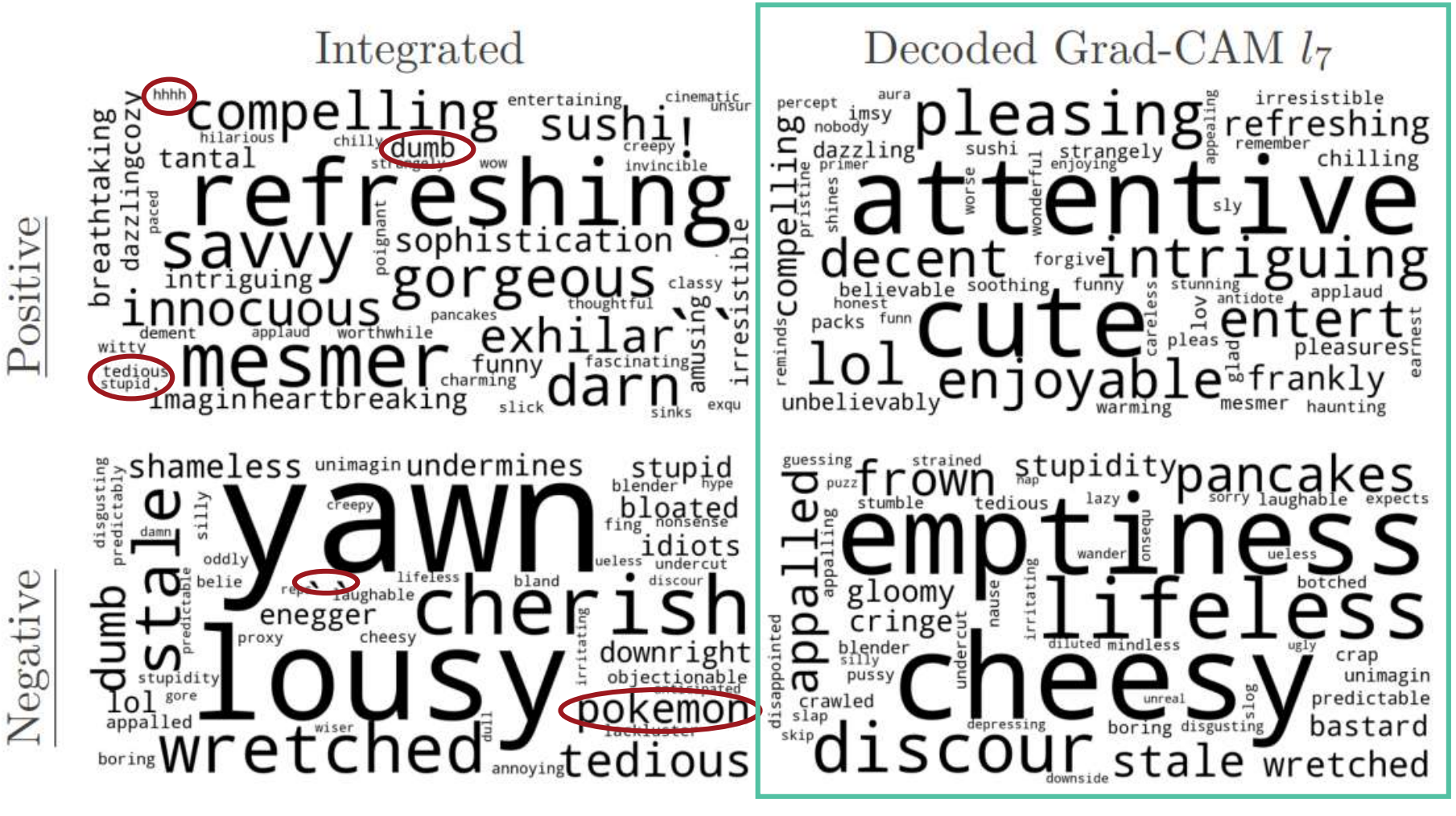
Experiments: SST-2 Dataset

- **Aggregate scores of all tokens in a predicted class and weigh their total score by how rarely they occur in everyday language (the inverse document frequency w.r.t Wikipedia)**
- **Intuition: tokens with high importance scores that are rare are representative of that class**
- **Tokens should disambiguate classes so important tokens should be unique (low overlap between classes)**
- **Count number of top k ranked tokens that appear in every pair of classes**



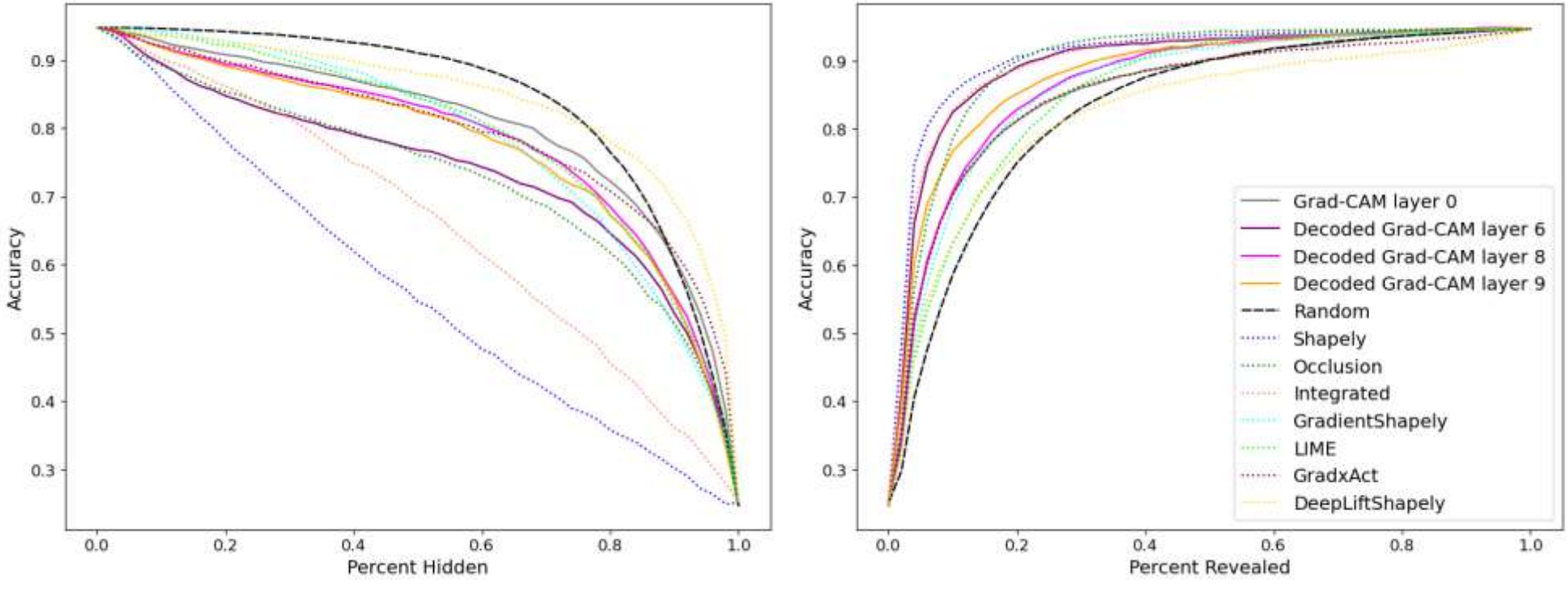
Lower is Better (more representative tokens)

Experiments: SST-2 Dataset



Experiments: AG News Dataset

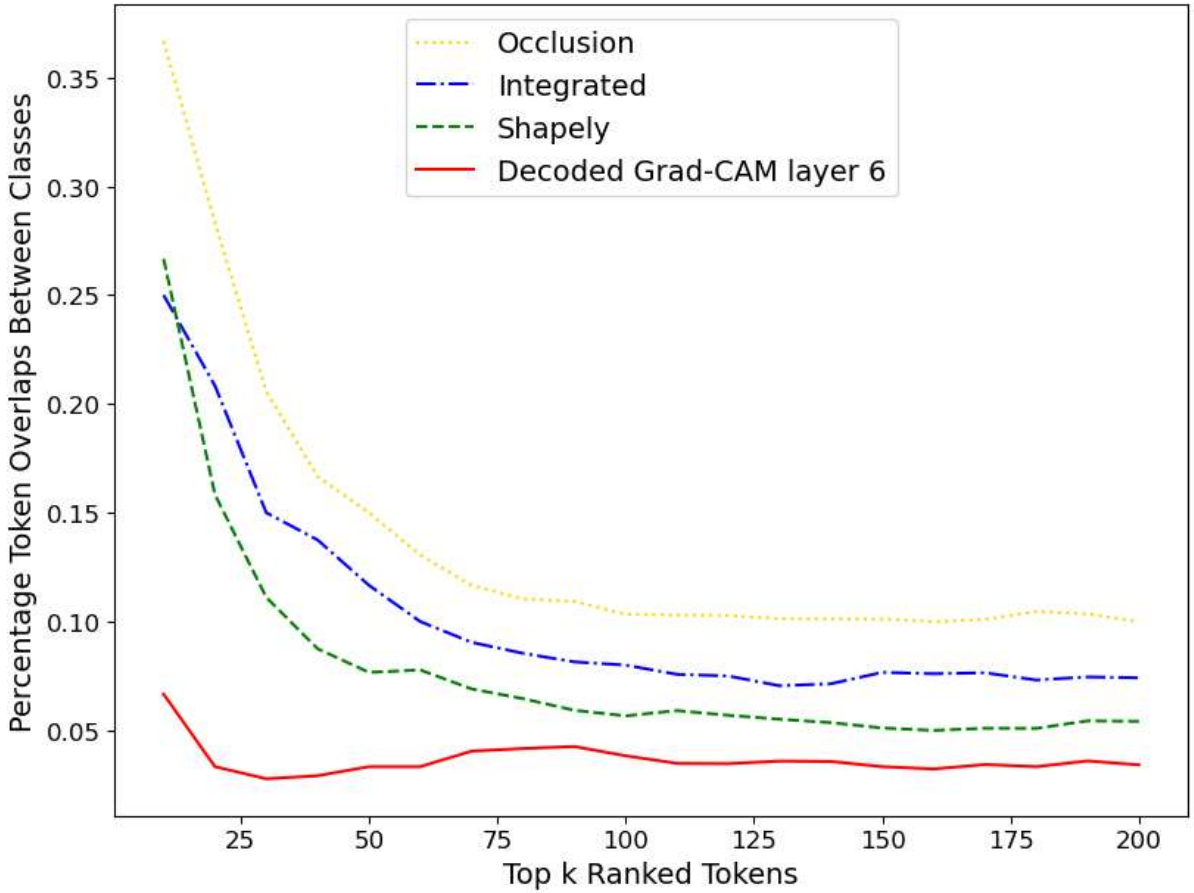
Multiclass classification task - each sample is a sentence from a news article labeled as belonging to World, Sports, Business, or Sci/Tech topics



Layer 8 only captures information downstream from it and significantly outperforms Layer 0

Experiments: AG News Dataset

- **Aggregate scores of all tokens in a predicted class and weight their total score by how rarely they occur in everyday language (the inverse document frequency w.r.t Wikipedia)**
- **Intuition: tokens with high importance scores that rare are representative of that class**
- **Tokens should disambiguate classes so important tokens should be unique (low overlap between classes)**
- **Count number of top k ranked tokens that appear in every pair of classes**

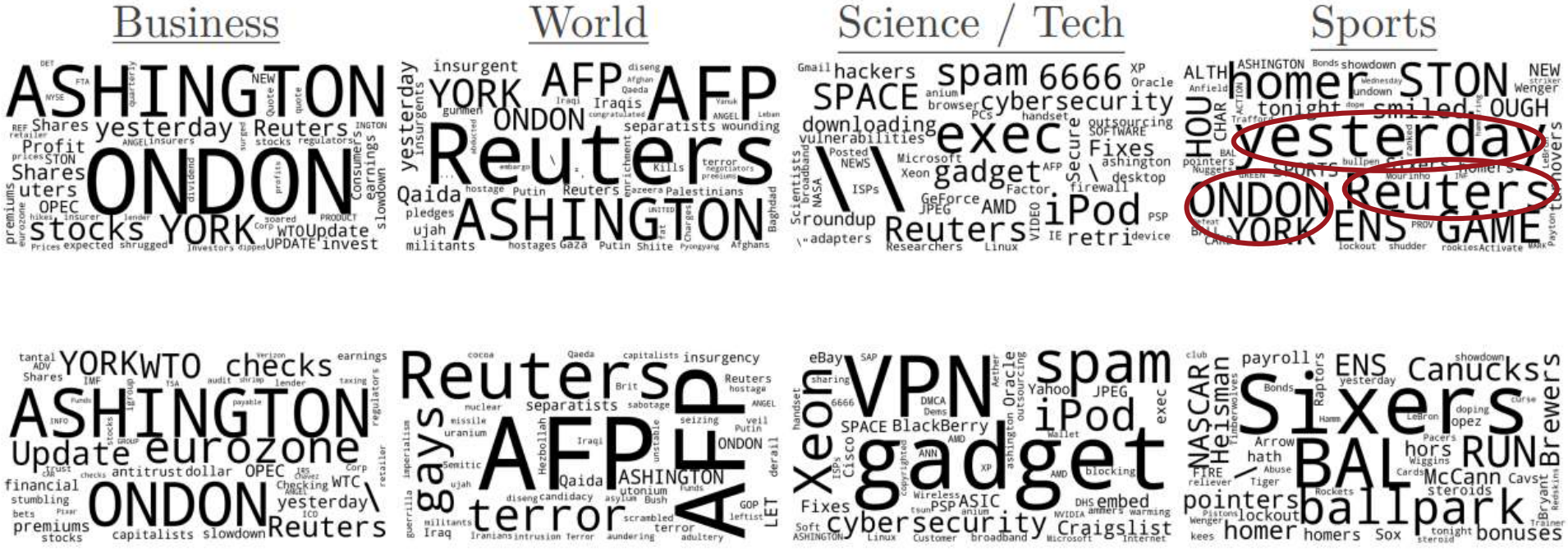


Lower is Better (more representative tokens)

Experiments: AG News Dataset

Shapely

Decoded Grad-CAM l_6



Examples of Highlighted Explanations

Simple	**Negative: 0.9809374213218689** unflinchingly bleak and desperate	**Negative: 0.9997243285179138** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.
Smooth	**Negative: 0.9809374213218689** unflinchingly bleak and desperate	**Negative: 0.9997243285179138** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.
Integrated	**Negative: 0.9809374213218689** unflinchingly bleak and desperate	**Negative: 0.9997243285179138** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.
Decoded Grad-CAM l_7	**Negative: 0.94347584** unflinchingly bleak and desperate	**Negative: 0.99634856** so unremittingly awful that labeling it a dog probably constitutes cruelty to canines.

Figure 12. STT-2 Example 1: For the left column example, Decoded Grad-CAM l_7 and to an extent AllenNLP Interpret's Simple highlight the negative sentiment words "bleak" and "desperate", but all three of AllenNLP Interpret's methods also focus on "and". For the right column example, all four methods focus on "awful" and "unrem"(ittingly), but the AllenNLP Interpret's methods are more noisy with highlights on unrelated terms such as "it", "dog" and "constitutes".

Examples of Highlighted Explanations

Simple	**Positive: 0.9998635053634644** birthday girl is an amusing joy ride, with some surprisingly violent moments.	**Positive: 0.9998799562454224** more romantic, more emotional and ultimately more satisfying than the teary-eyed original.
Smooth	**Positive: 0.9998635053634644** birthday girl is an amusing joy ride, with some surprisingly violent moments.	**Positive: 0.9998799562454224** more romantic, more emotional and ultimately more satisfying than the teary-eyed original.
Integrated	**Positive: 0.9998635053634644** birthday girl is an amusing joy ride, with some surprisingly violent moments.	**Positive: 0.9998799562454224** more romantic, more emotional and ultimately more satisfying than the teary-eyed original.
Decoded Grad-CAM l_7	**Positive: 0.9993205** birthday girl is an amusing joy ride, with some surprisingly violent moments.	**Positive: 0.99926156** more romantic, more emotional and ultimately more satisfying than the teary-eyed original.

Figure 12. STT-2 Example 2: For the left column example, Decoded Grad-CAM l_7 focuses strongly on the positive phrases "is an amusing joy" and "surprising". The other methods also highlight these terms, but less clearly with unrelated words such as "moment" and potentially negative words such as "violent". For the right column example, all methods focus on the positive words "romantic", "satisfying", "original", and "emotional"; however the AllenNLP Interpret's methods are more noisy and highlight many other words too.

Examples of Highlighted Explanations

Shapely	**SciTech: 0.9973194766244139** Dutch Retailer Beats Apple to Local Download Market AMSTERDAM (Reuters) - Free Record Shop, a Dutch music retail chain, beat Apple Computer Inc. to market on Tuesday with the launch of a new download service in Europe's latest battleground for digital song services.
Occlusion	**SciTech: 0.9973194754393703** Dutch Retailer Beats Apple to Local Download Market AMSTERDAM (Reuters) - Free Record Shop, a Dutch music retail chain, beat Apple Computer Inc. to market on Tuesday with the launch of a new download service in Europe's latest battleground for digital song services.
Integrated	**SciTech: 0.9973194754393703** Dutch Retailer Beats Apple to Local Download Market AMSTERDAM (Reuters) - Free Record Shop, a Dutch music retail chain, beat Apple Computer Inc. to market on Tuesday with the launch of a new download service in Europe's latest battleground for digital song services.
Decoded Grad-CAM l_6	**SciTech: 0.99106675** Dutch Retailer Beats Apple to Local Download Market AMSTERDAM (Reuters) - Free Record Shop, a Dutch music retail chain, beat Apple Computer Inc. to market on Tuesday with the launch of a new download service in Europe's latest battleground for digital song services.

Figure 12. AG News Example 1: Decoded Grad-CAM l_6 and Shapely focus on highlighting "Apple" (a tech company) along with technology terms like "Download" and "Computer". Occlusion focuses on terms related to the Netherlands such as "AMSTERDAM" and "Dutch", which do not have an obvious connection to technology. Integrated lightly highlights a large number of words, but some are technology related ones.

Examples of Highlighted Explanations

Shapely	**World: 0.9992575257269619** Charity chief kidnapped in Iraq Care International charity says its chief of operations in Iraq has been kidnapped in Baghdad. A spokeswoman told Reuters on Tuesday that Margaret Hassan, who has been working
Occlusion	**World: 0.9992575260579086** Charity chief kidnapped in Iraq Care International charity says its chief of operations in Iraq has been kidnapped in Baghdad. A spokeswoman told Reuters on Tuesday that Margaret Hassan, who has been working
Integrated	**World: 0.9992575260579086** Charity chief kidnapped in Iraq Care International charity says its chief of operations in Iraq has been kidnapped in Baghdad. A spokeswoman told Reuters on Tuesday that Margaret Hassan, who has been working
Decoded Grad-CAM l_6	**World: 0.99931705** Charity chief kidnapped in Iraq Care International charity says its chief of operations in Iraq has been kidnapped in Baghdad. A spokeswoman told Reuters on Tuesday that Margaret Hassan, who has been working

Figure 12. AG News Example 2: Decoded Grad-CAM l_6 highlights the words "chief" and "kidnapped" along with terms related to the Middle East region ("Iraq" and "Baghdad"). Occlusion lightly highlights the phrases "kidnapped in Iraq" and "kidnapped in Baghdad", which also are meaningful. Shapely and Integrated have less clear explanations with focus on the words "in", "Care", and the punctuation.

Examples of Highlighted Explanations

Shapely	**Sports: 0.9991075460048449** Hockey Labor Talks Broken Off TORONTO -- National Hockey League labor talks came to a halt Tuesday after each side rejected the other #39;s proposal. The talks lasted more than three hours, with the league making a one-hour presentation on
Occlusion	**Sports: 0.9991075473661293** Hockey Labor Talks Broken Off TORONTO -- National Hockey League labor talks came to a halt Tuesday after each side rejected the other #39;s proposal. The talks lasted more than three hours, with the league making a one-hour presentation on
Integrated	**Sports: 0.9991075473661293** Hockey Labor Talks Broken Off TORONTO -- National Hockey League labor talks came to a halt Tuesday after each side rejected the other #39;s proposal. The talks lasted more than three hours, with the league making a one-hour presentation on
Decoded Grad-CAM l_6	**Sports: 0.99970883** Hockey Labor Talks Broken Off TORONTO -- National Hockey League labor talks came to a halt Tuesday after each side rejected the other #39;s proposal. The talks lasted more than three hours, with the league making a one-hour presentation on

Figure 12. AG News Example 3: Decoded Grad-CAM l_6 heavily highlights the word "Hockey". The other methods also have some focus on hockey terms such as the phrase "National Hockey League labor", but are noisy and also highlight many unrelated terms such as "The talks" and "after each".

Examples of Highlighted Explanations

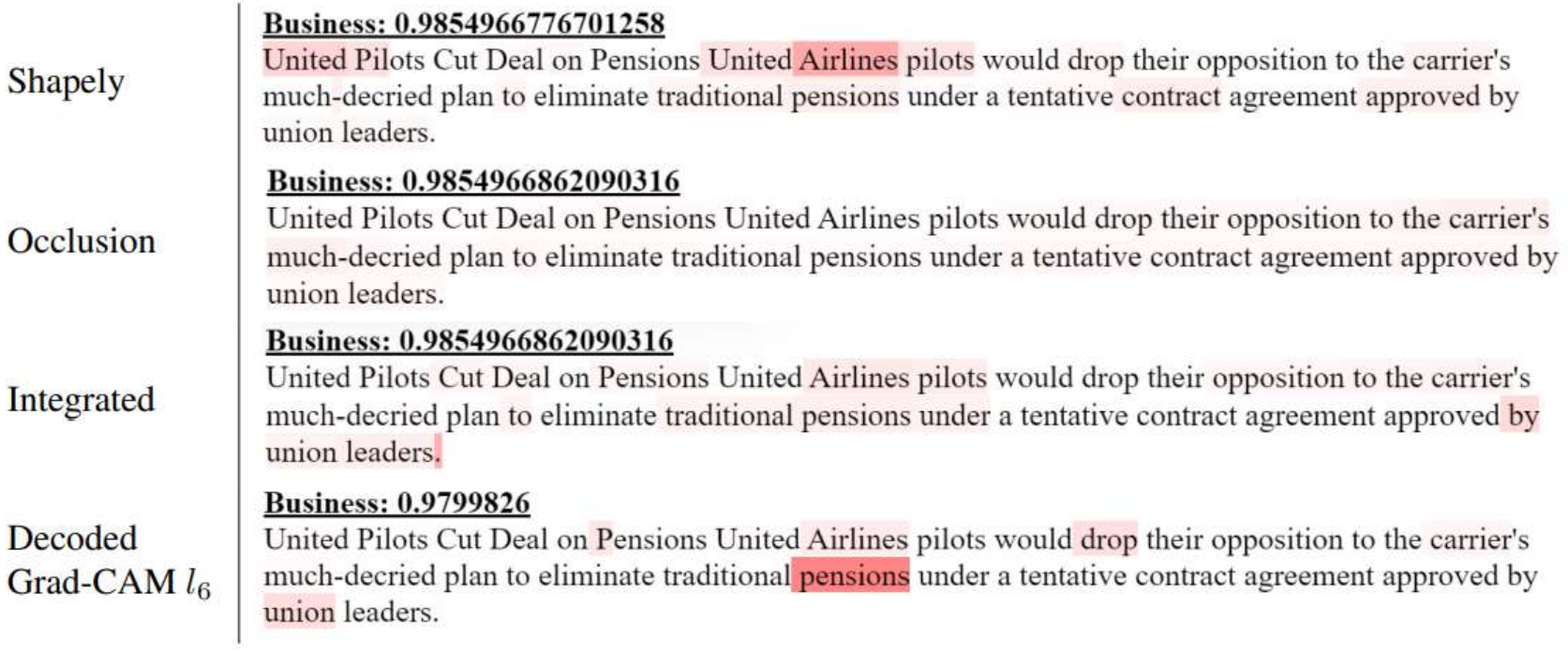
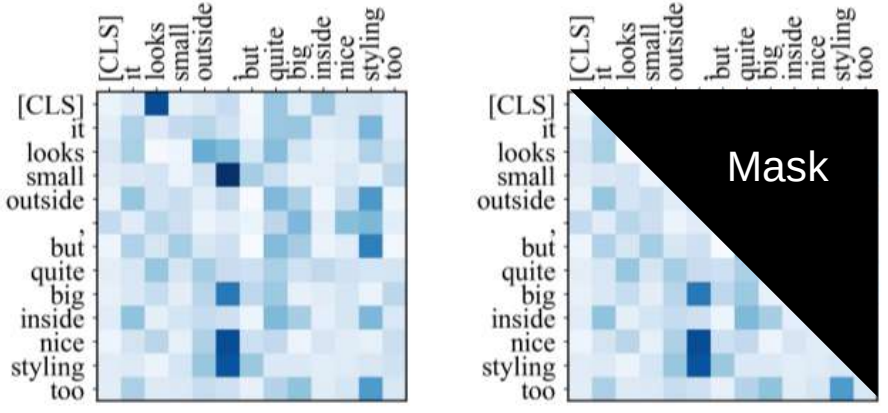


Figure 12. AG News Example 4: Decoded Grad-CAM l_6 heavily highlights the word ”pensions” with some additional focus on ”union”; however, it also does highlight some less clear terms such as ”carrier”, ”drop”, and ”Airlines”. Shapely and Integrated also highlight key business terms such as ”traditional pensions”, ”contract”, and ”union leaders”; although Shapely also puts a lot of emphasis on ”United” and Integrated on ”Airlines pilots”. Occlusion lightly highlights everything and does not have an clear explanations.

What about Decoder Stack Models?

- **Previous work is on Transformers that are a stack of encoders**
- **Most of the recent models are a stack of decoders (generative models)**
  - Qwen, Mistral, LLAMA, DeepSeek
  - Can we still apply the same techniques?
- **Difference in Decoder Stack Models**
  - Trained for next token prediction like tasks
  - Casual self-attention allows for text generation without peaking



Bi-directional Self-Attention

Casual Self-Attention

- **We can apply the same saliency techniques to generative LLMs**
  - Caveat: many of the newest models (Mixtral 8x7B, Qwen2-57B-A14B) are mixture of experts models, tracking the saliency of multiple models can become more difficult (a lot more engineering)

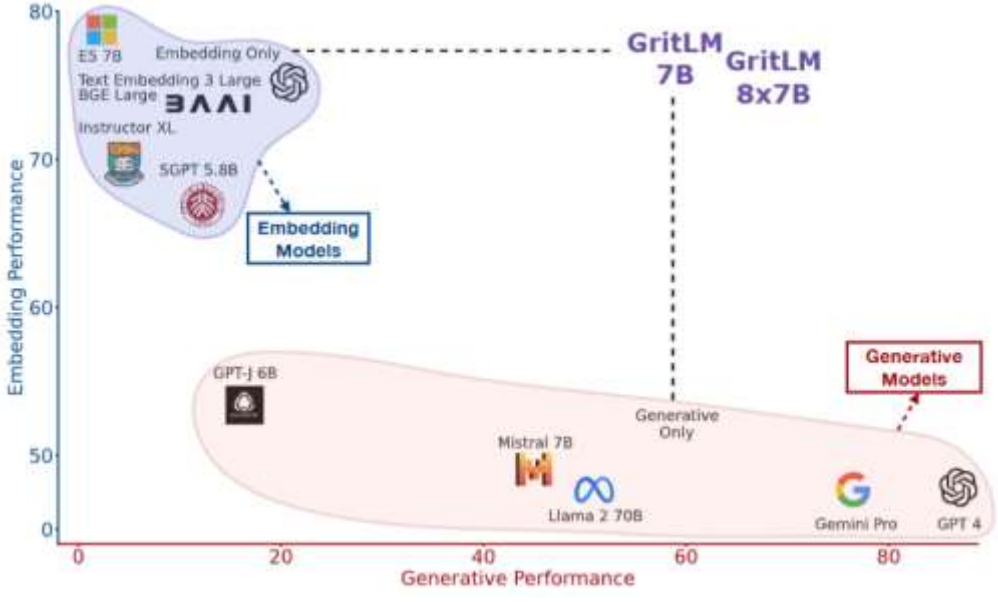
Are Encoders dead?

Massive Text Embedding Benchmark (MTEB) Leaderboard

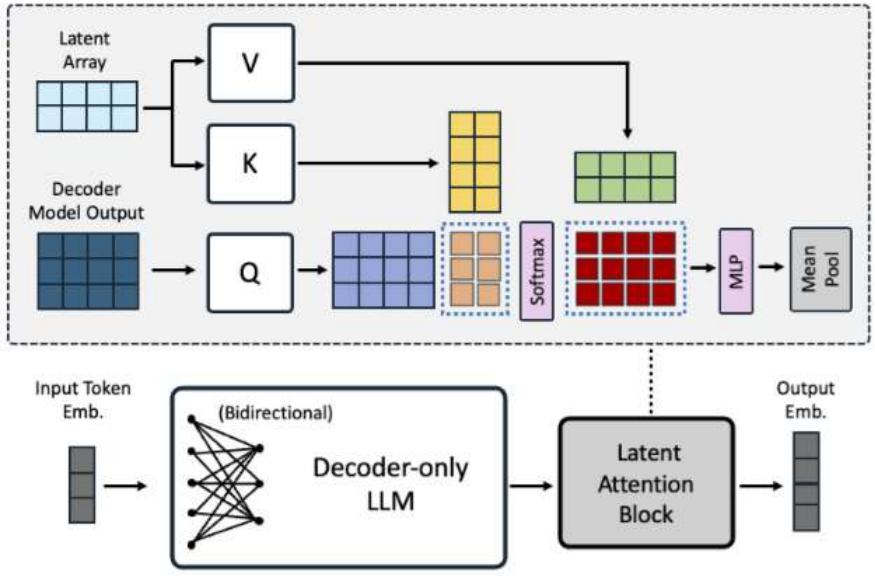
Rank (Bor…	Model	Zero-shot	Memory U…	Number of P…	Embedding D…	Max Tokens	Mean … ↓	Mean (TaskT…	Bitext …	Classification	Clustering	Instruc
2	Qwen3-Embedding-8B	99%	28866	7B	4096	32768	**70.58**	**61.69**	**80.89**	**74.00**	**57.65**	10.06
3	Qwen3-Embedding-4B	99%	15341	4B	2560	32768	69.45	60.86	79.36	72.33	57.15	**11.56**
1	gemini-embedding-001	99%	Unknown	Unknown	3072	2048	68.37	59.59	79.28	71.82	54.59	5.18
4	Qwen3-Embedding-0.6B	99%	2272	595M	1024	32768	64.34	56.01	72.23	66.83	52.33	5.09
7	multilingual-e5-large-instruct	99%	1068	560M	1024	514	63.22	55.08	80.13	64.94	50.75	-0.40
5	gte-Qwen2-7B-instruct	⚠ NA	29040	7B	3584	32768	62.51	55.93	73.92	61.55	52.77	4.94
10	text-multilingual-embedding-002	99%	Unknown	Unknown	768	2048	62.16	54.25	70.73	64.64	47.84	4.08
6	Linq-Embed-Mistral	99%	13563	7B	4096	32768	61.47	54.14	70.34	62.24	50.60	0.94
8	embeddinggemma-300m	99%	578	307M	768	2048	61.15	54.31	64.40	60.90	51.17	5.61
14	Cohere-embed-multilingual-v3.0	⚠ NA	Unknown	Unknown	1024	Unknown	61.12	53.23	70.50	62.95	46.89	-1.89

- **Top models are all based on decoder-only architectures (despite encoders being trained for embedding)**
- **Leverages the benefits of these highly performant pretrained LLMs**
- **But… these decoder-only models are fine-tuned to become encoders for embedding tasks!**

Nvidia NV-Embed & GRIT



Muennighoff, Niklas, et al. "Generative representational instruction tuning." *The Thirteenth International Conference on Learning Representations*. 2024.



Lee, Chankyu, et al. "NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models." The Thirteenth International Conference on Learning Representations. 2024.

- **Both papers train models using bi-directional attention and contrastive style loss function for fine-tuning on sentence embedding tasks**
  - This is literally the definition of an encoder stack

Can We Explain Sentence Embeddings?

Tokenized Vector

Start token holds “task” embedding (e.g. classification)

Each text token holds a d dimensional vector representing that text in continuous space

- inverting this embedding allows for recovery of the text
- synonyms are close together in this embedding space
- this is the basis of our prev. paper

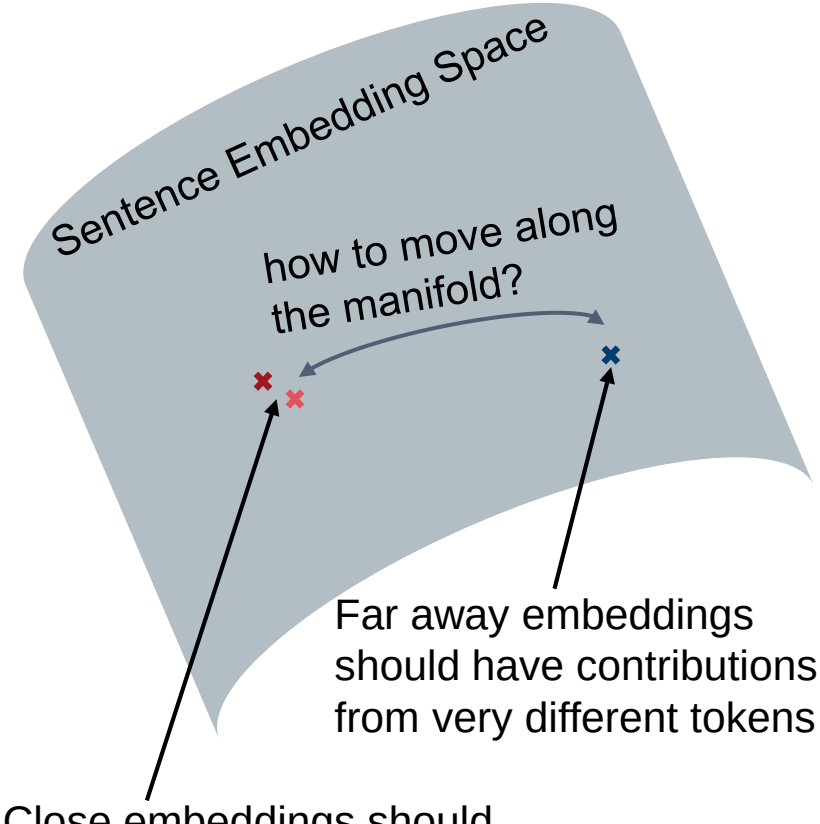
End token holds a representation of the input text as a whole

- Note: mean pooling for the embedding is equivalent to attention with the prev. layer’s outputs

Padding holds nothing

- **Applying the Decoding Layer Saliency technique will show which tokens contribute significantly to the final sentence embedding**
- **What about the embedding as a whole?**
  - The final layer’s text token embeddings (mint) are trained for Masked Language Modelling (in encoder stacks) and Next Token Prediction (in decoder stacks) so it’s embedding should represent the masked/next token in an input
  - The sentence embedding token is trained for Contrastive Loss i.e. given pairs of similar / different text, ensure the similar ones are close and the different ones are far away

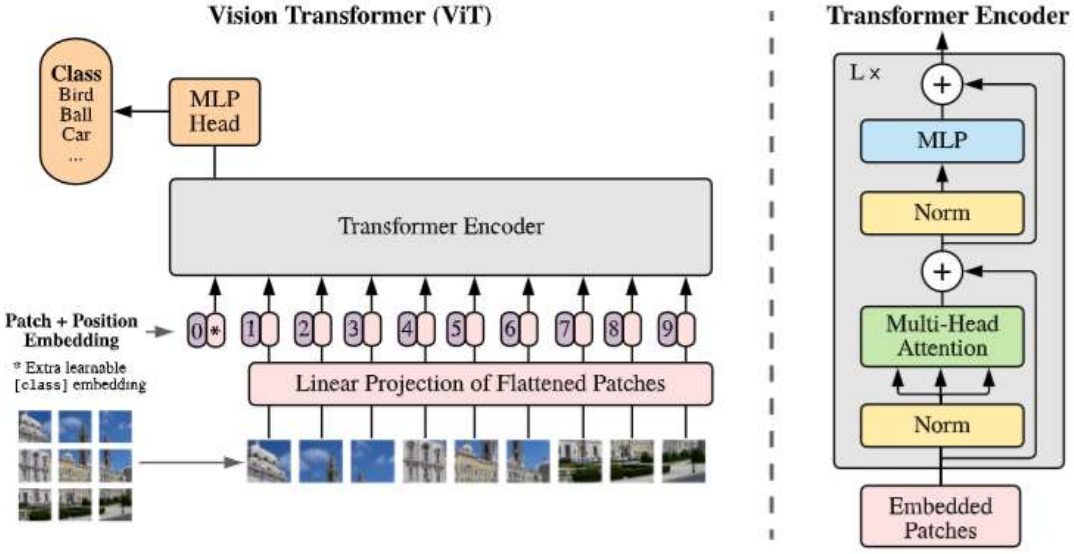
The Sentence Embedding Space



- **We should expect the Red and Pink sentences to have contributions from similar tokens**
  - We see a similar phenomena in classification tasks e.g. text classified in the Sports category have tokens related to sports

  - Text classified in the same class have “task” embedding vectors that are close to each other (necessary to form a decision boundary)
- **But what about Red and Blue?**
  - Dissimilar embeddings should not share tokens, but what are “far away tokens”? Can we define “orthogonal” tokens?
- **Can we move along a manifold in this space to generate new embeddings and recover their NLP text?**

What about Vision Attention Models?

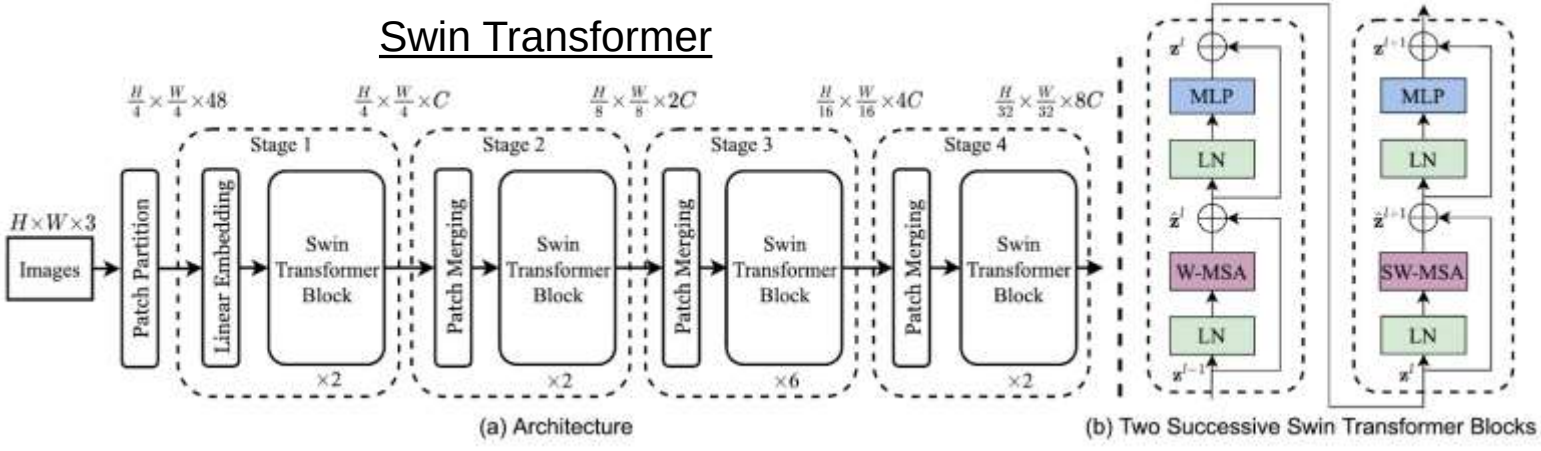


Dosovitskiy, Alexey, et al. “An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale.” International Conference on Learning Representations, 2021.

- **ViT models embed each patch of an image into a “token” embedding using a feedforward network (similar to the embedding matrix)**
  - Back to a stack of encoders
- **But they are trained for specific tasks and not with self-supervision**
  - No “decoding” matrix learned from Masked Language Modelling
  - So unable to interpret intermediate hidden layer outputs as being a combination of patches

What about Vision Attention Models?

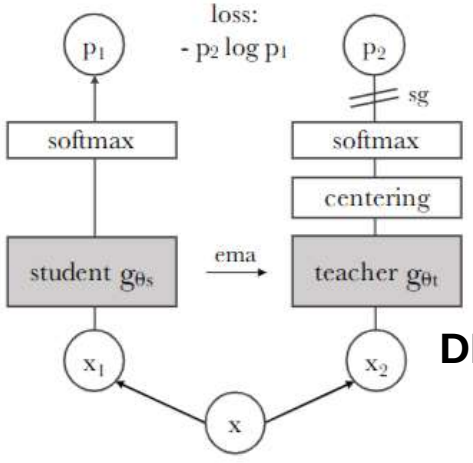
Swin Transformer



Liu, Ze, et al. "Swin transformer: Hierarchical vision transformer using shifted windows." *Proceedings of the IEEE/CVF international conference on computer vision*. 2021.

- **Instead of global self-attention over the image, it alternates between regular and shifted self-attention *within* each window**
  - Shifted self-attention shifts the windows to allow for cross-window attention
- **Patch merging pools patch tokens together, reducing dimensionality as the layer increases**
- **Still only trained for specific tasks (classification on ImageNet) during pre-training so still no “decoding” matrix**

What about Vision Attention Models?



Caron, Mathilde, et al. "Emerging properties in self-supervised vision transformers." *Proceedings of the IEEE/CVF international conference on computer vision*. 2021.

DINO (Denoising Self-Supervised Learning) Model

- **Self-supervised through student – teacher co-distillation model**
  - Want the predicted feature embeddings to be close to each w.r.t. cross-entropy loss
  - Teacher model is not pre-trained (both learning at the same time), but has a few more components to the architecture and a larger view (more pixels in a cropped image)

- Claim: self-attention map (weights) for the final layer's [CLS] token projected back onto the original image (and thresholded to keep 60% of the mass) creates an image segmentation mask
- But the vector elements (from penultimate layer) attending to [CLS] in the final layer do not correspond to the initial patches

Questions?

elizabeth.hou@str.us