Guest lecture for CS 329X: Human-Centered NLP

Model Visualization

Sherry Tongshuang Wu

Human Computer Interaction Institute

@tongshuangwu / sherryw@cs.cmu.edu

Logistics: Final presentation

Mon, Apr 24	**Final project presentation - 1** `(Presentation)`	Slides
Wed, Apr 26	**Final project presentation - 2** `(Presentation)`	Slides

You should cover:

A quick review on motivation and your project objective

Your method and result

Some discussion on what you learned from your project (limitation, implication for future work, how it could be done differently, etc.)

You will be graded based on:

Presentation clarity, project completeness (an estimation of the effort you put in), and thoughtfulness

Overview

Key things to consider in model visualizations

Common techniques for getting the information to visualize

Local feature attribution

More global dimensionality reduction

And their visual encodings:

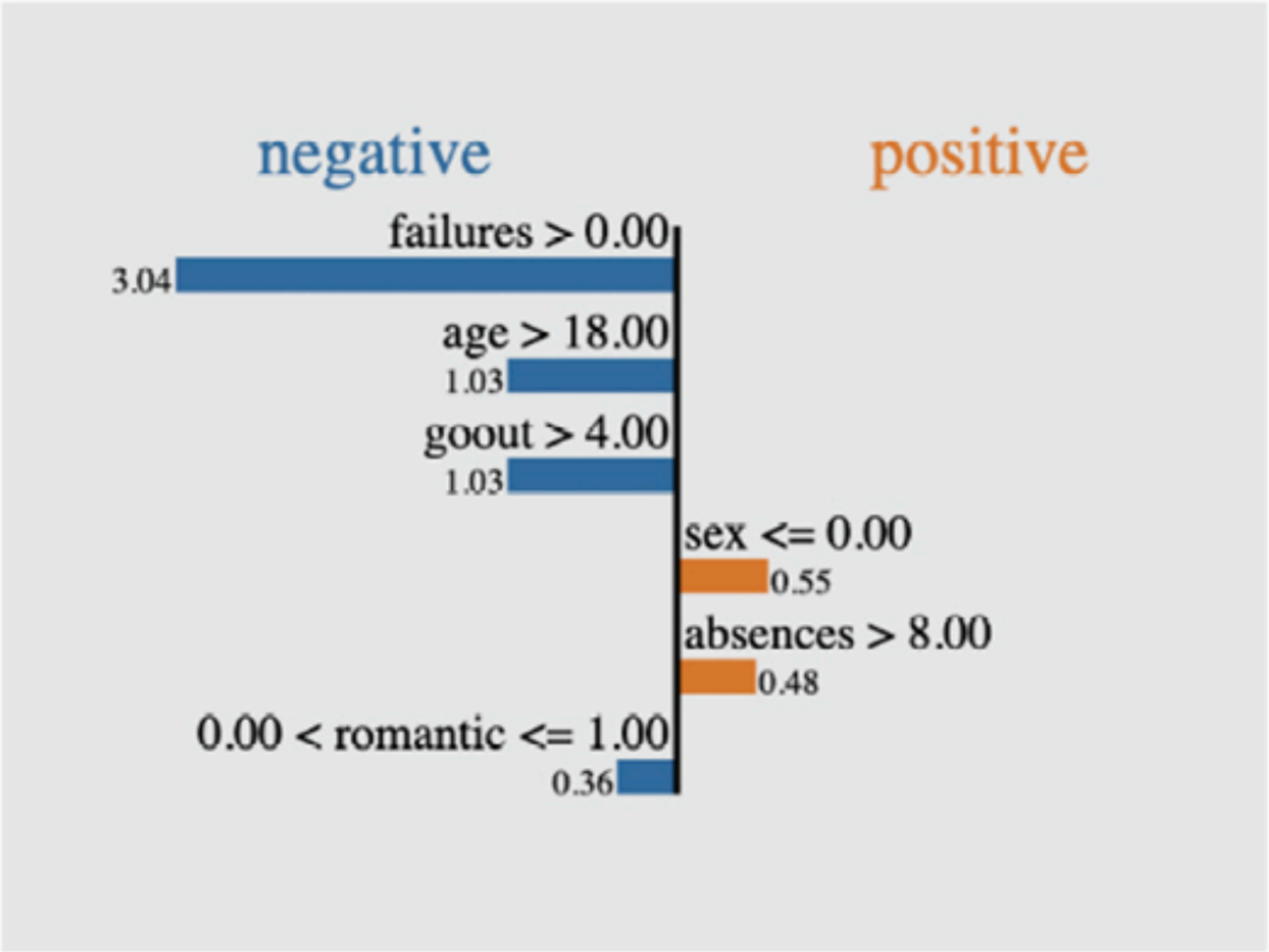
Why certain visualization is more effective than others

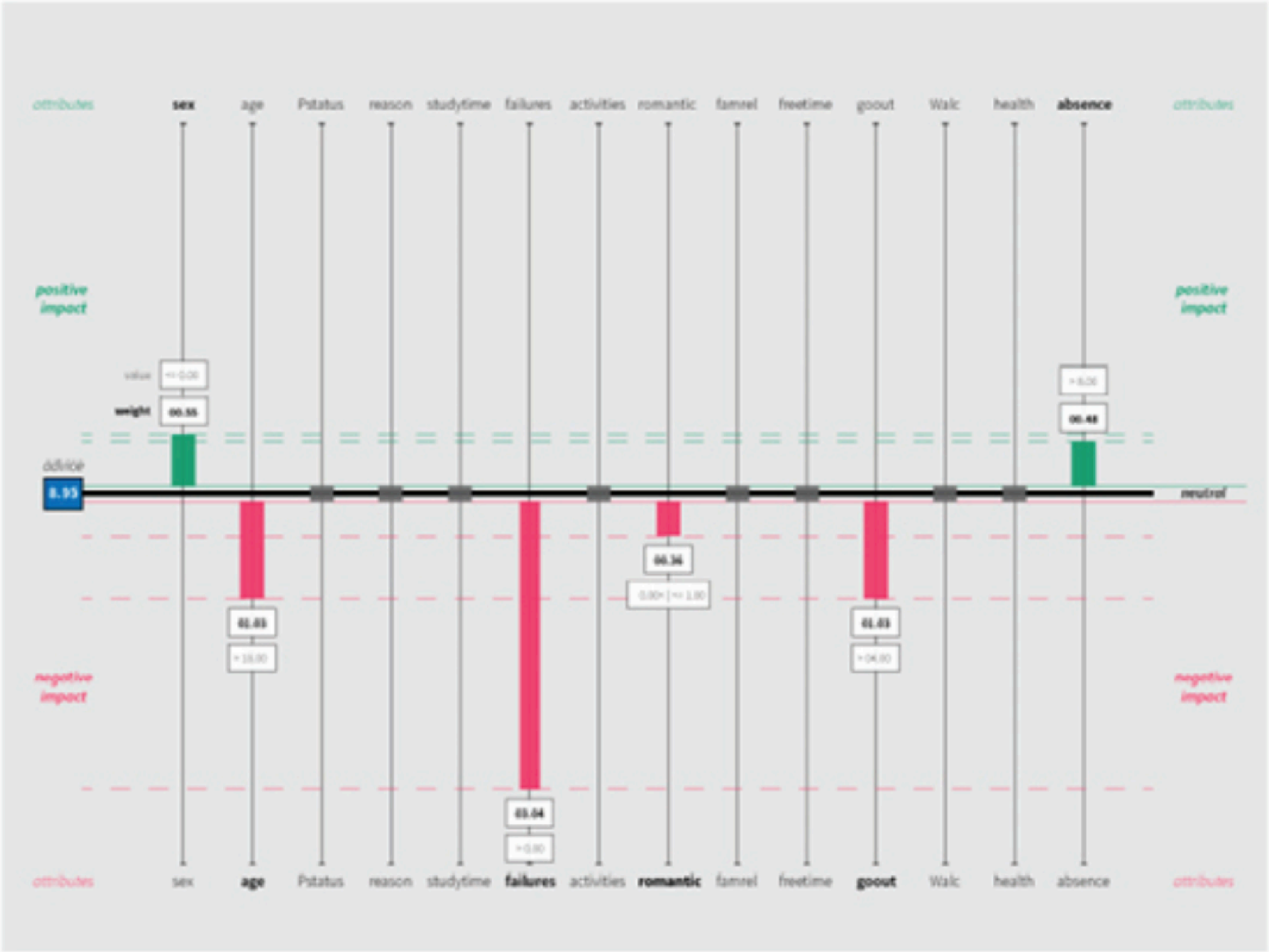
Key visual encoding channels for different kinds of information

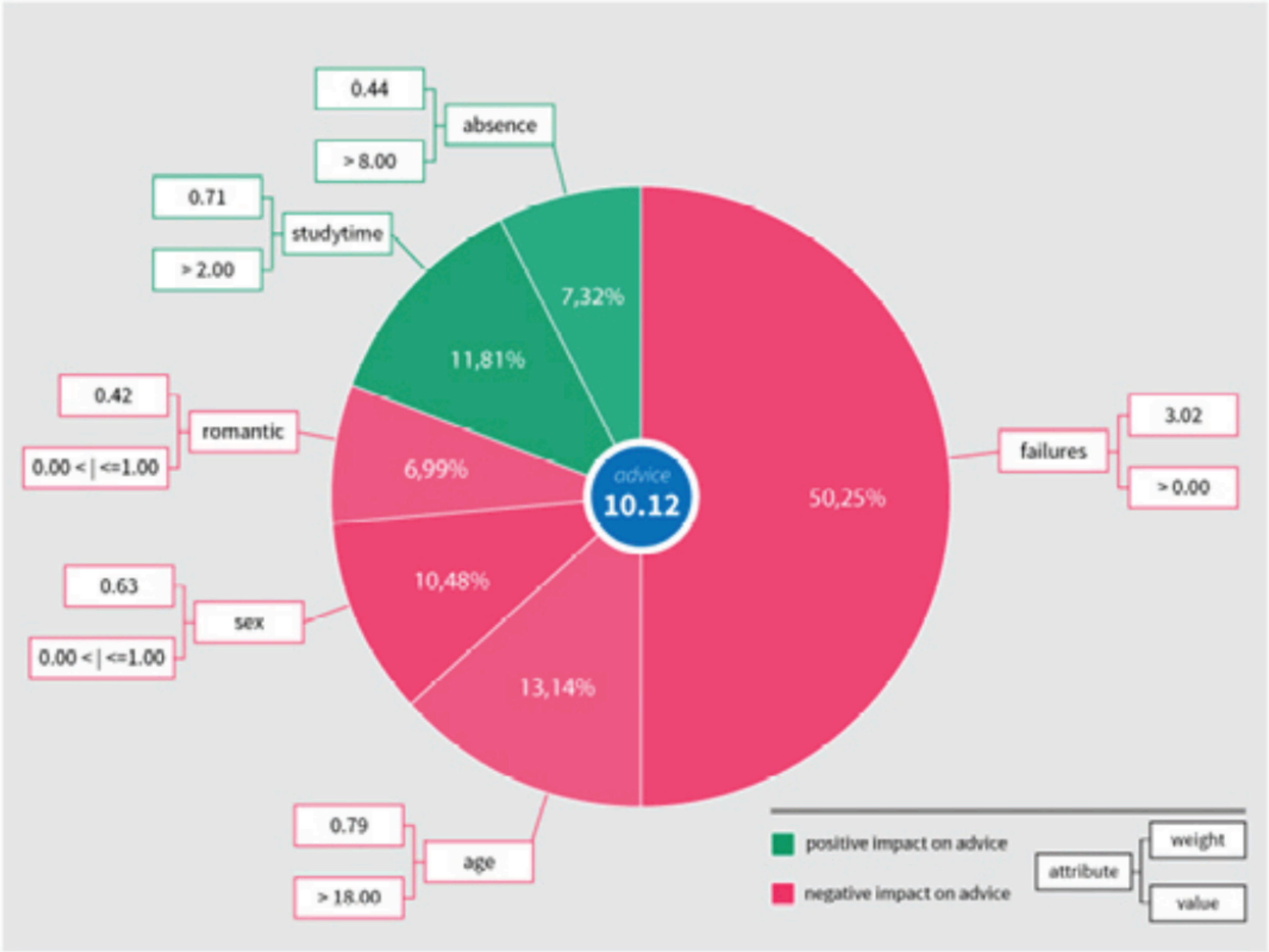
Interface and viz. is an important variable in HCNLP.

"We found there are significant effects between treatments. We conclude that the exact form of visually representing (LIME) explanations is relevant for the design of explanations in Human-AI interactions."

Why do people have a preference when it's very much the same underlying information?



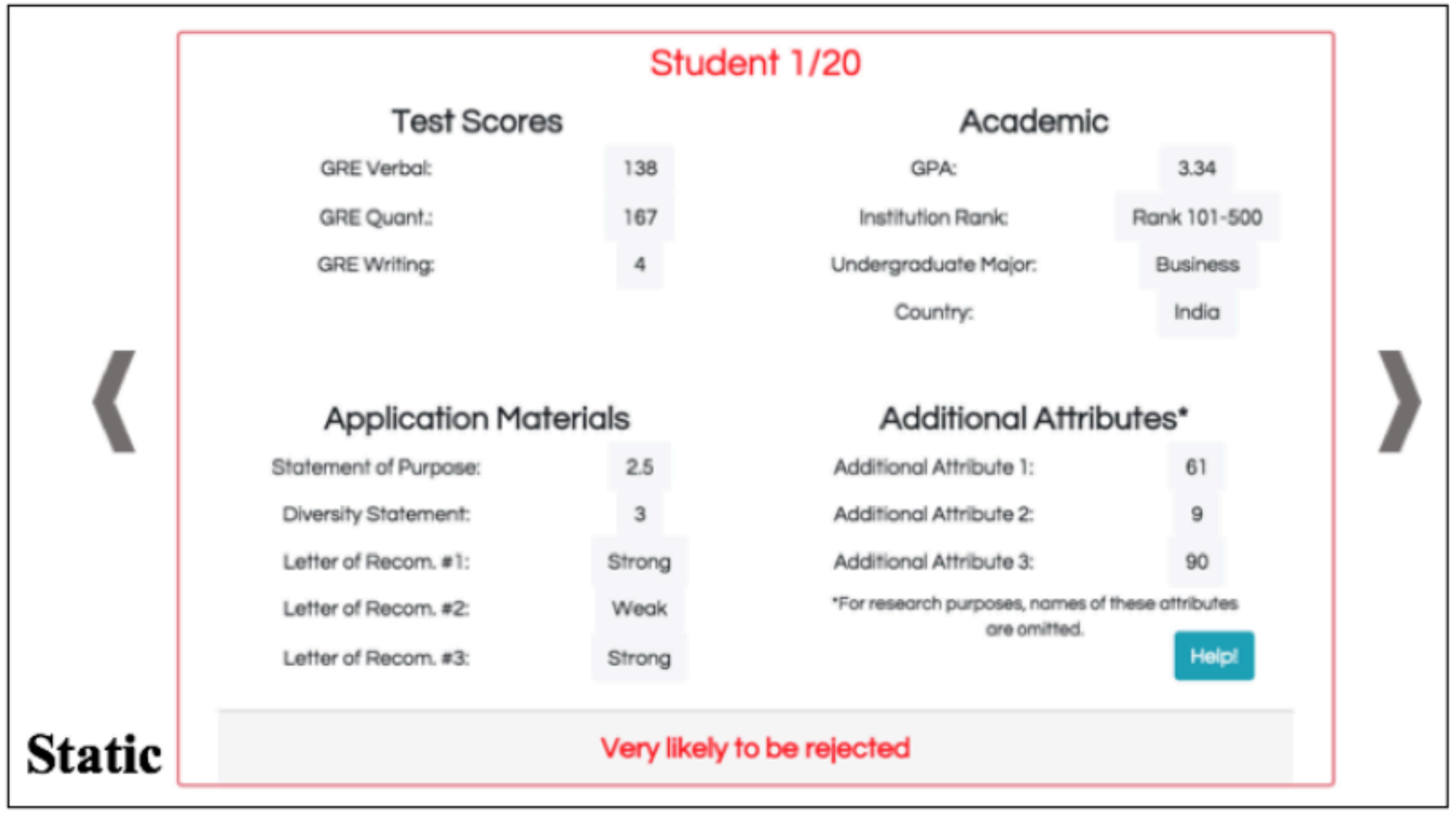


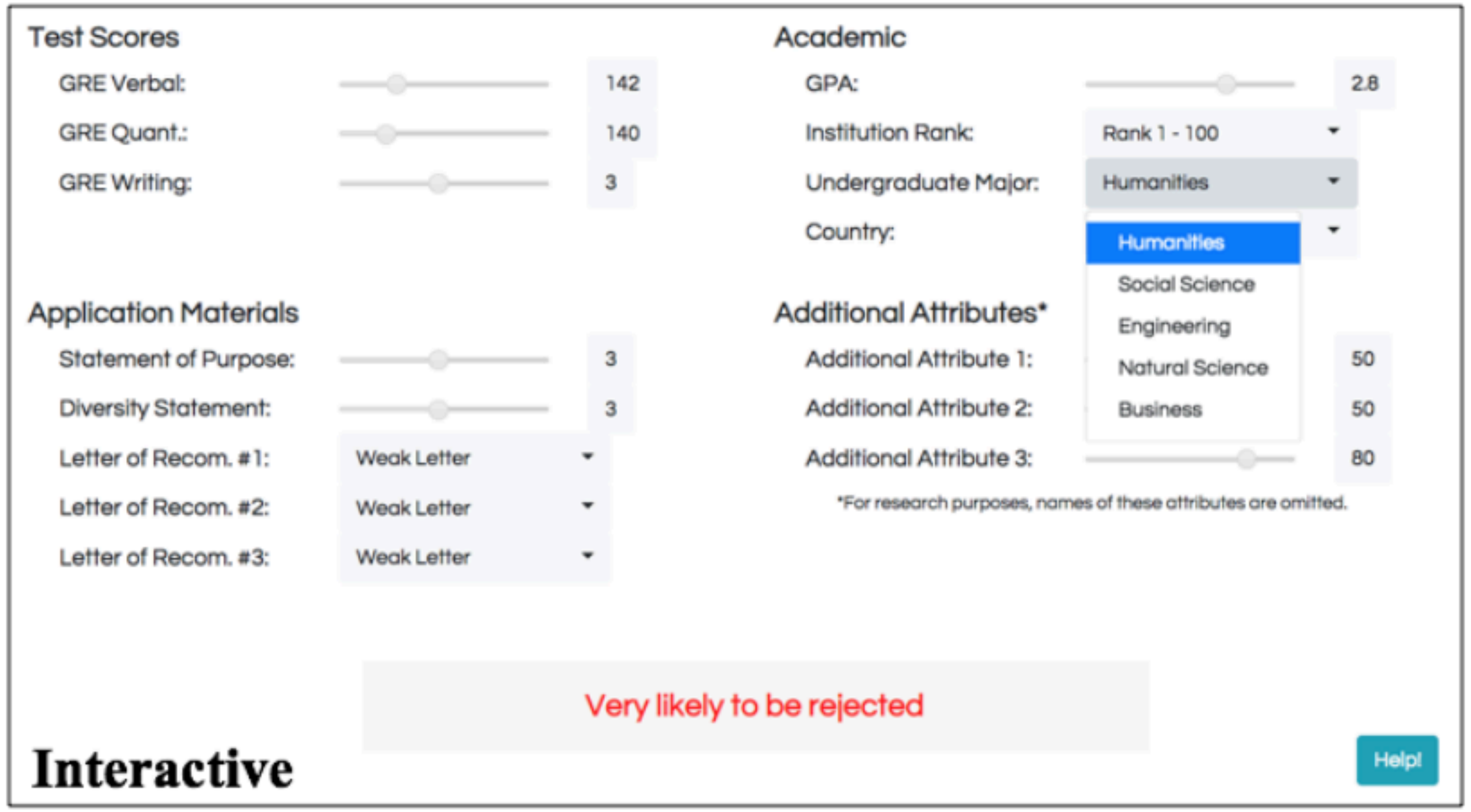


Mucha, Henrik, et al. "Interfaces for explanations in human-AI interaction: proposing a design evaluation approach." *CHI EA*. 2021.

Interface and viz. is an important variable in HCNLP.

"Although the interactive approach is more effective at improving comprehension, it comes with a trade-off of taking more time."





c. Interactiv

The Static interface (left) displays a selection of 20 unique application
interface (right) provides sliders to modify the values of attributes. The

Why does the interactive approach improve comprehension?

Cheng, Hao-Fei, et al. "Explaining decision-making algorithms through UI: Strategies to help non-expert stakeholders." *CHI 2019*

People have studied VIS x AI extensively!

Example interactive visualization articles that explain general concepts and communicate experimental insights when playing with AI models. (a) A Visual Exploration of Gaussian Processes by Görtler, Kehlbeck, and Deussen (VISxAI 2018); (b) What Have Language Models Learned? by Adam Pearce (VISxAI 2021); (c) What if we Reduce the Memory of an Artificial Doom Player? by Jaunet, Vuillemot, and Wolf (VISxAI 2019); (d) Comparing DNNs with UMAP Tour by Li and Scheidegger (VISxAI 2020); (e) The Myth of the Impartial Machine by Feng and Wu (Parametric Press); (f) FormaFluens Data Experiment by Strobelt, Phibbs, and Martino.

IEEE VIS Workshop: Visualization for AI Explainability

Why do we want to do visualization?

We use visualization to make the information more intuitive and accessible.

Local interpretability: Understand why NLP models are making their (local) predictions – which specific token is important?

Global interpretability: Get insights into what the model have learned in general.

Debugging: Identify potential problems and errors in the model.

Communication: Convey certain message (e.g., observations on models) to others.

Education: Teach intuitions and information to general audience, junior students, etc.

“Parameters” for a visualization

Goal
Why visualize

Local understand

Global understand

Communication

Education

Content
What to visualize

Input distribution

In-/out-put mapping

Activations

Attention

Postdoc explanations

Architecture

Parameter spaces

Encoding
How to visualize

Line chart

Bar chart

Scatter plot

Graph

Saliency map

Context
Assist communication

Annotations

Text integration

Aggregation

Dimension reduction

Small multiples

"Parameters" for a visualization

Goal	**Content**	**Encoding**	**Context**
Why visualize	*What to visualize*	*How to visualize*	*Assist communication*
Local understand	Input distribution	Line chart	Annotations
Global understand	In-/out-put mapping	Bar chart	Text integration
Communication	Activations	Scatter plot	Aggregation
Education	Attention	Graph	Dimension reduction
	Postdoc explanations	Saliency map	Small multiples
	Architecture		
	Parameter spaces		

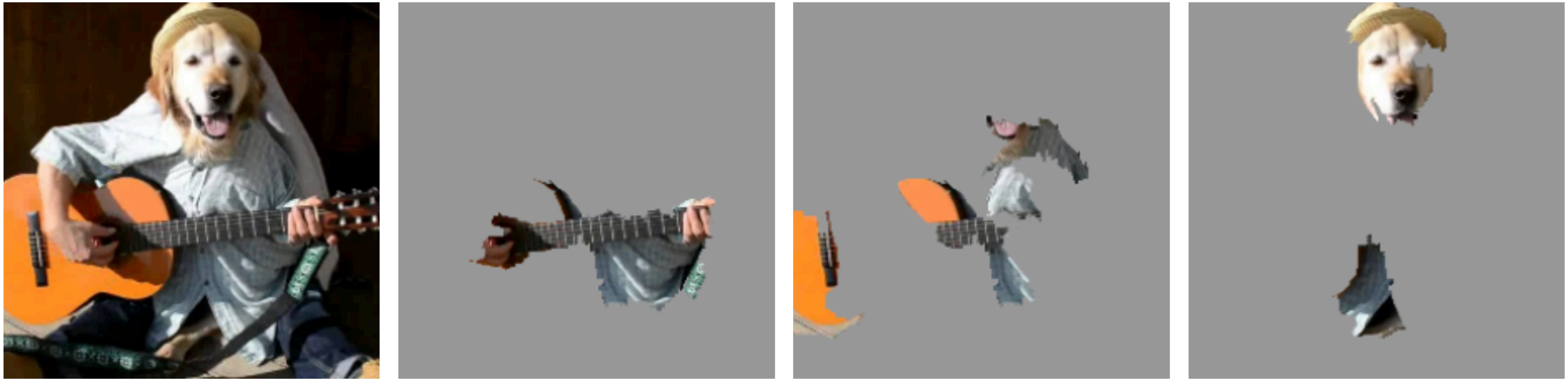
Encoding: Saliency Map

To highlight the most important or visually interesting parts of an image. Saliency maps are commonly used in CV and NLP to identify regions of interest within a document, image or video.

Jay Alammar. "Interfaces for Explaining Transformer Language Models." *2022*.

Input saliency

Similar information is available across various tasks.



(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

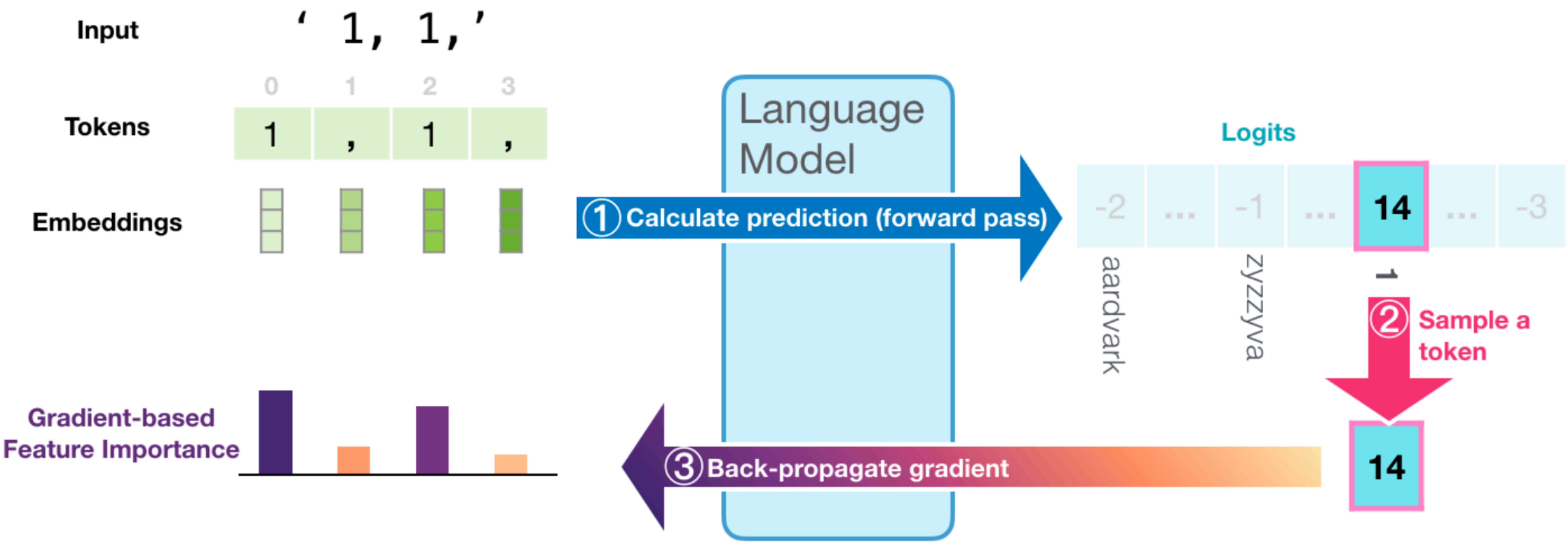
Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador" ($p = 0.21$)

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "" Why should i trust you?" Explaining the predictions of any classifier." *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016.

Content: Compute feature attribution using…

Vanilla gradient: Approximate the important of each token, using the gradient of the loss with respect to each token (computed by back-propagating to the input layer).

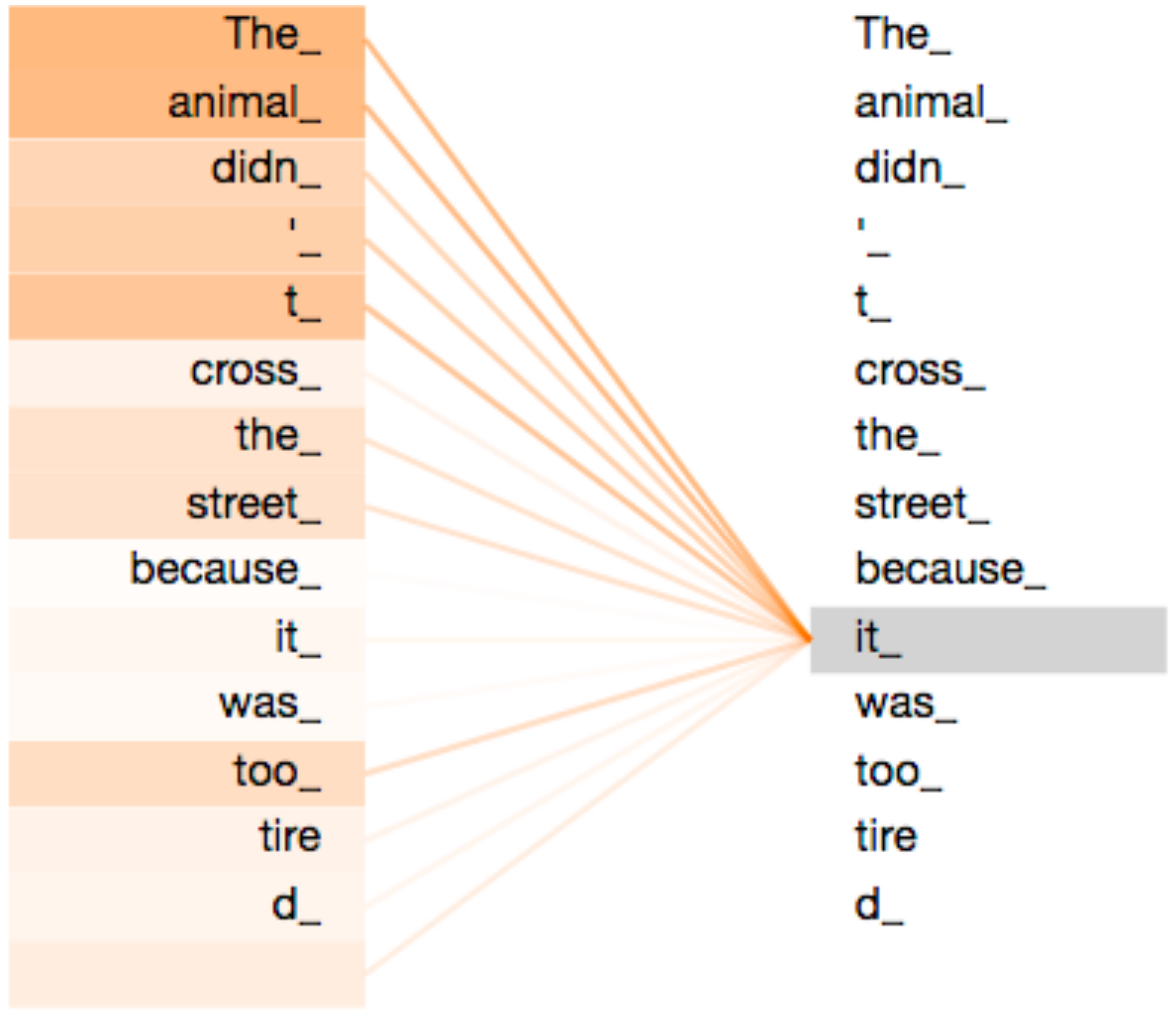
"For every amount you change this token, I change the output probability of the class/token this much"



Jay Alammar. "Interfaces for Explaining Transformer Language Models." *2022*.

Content: Compute feature attribution using…

Self-attention: In Transformers, we can directly model relationships between words in a sentence, regardless of their respective position.



Word	Value vector	Score	Value X Score
<s>		0.001	
a		0.3	
robot		0.5	
must		0.002	
obey		0.001	
the		0.0003	
orders		0.005	
given		0.002	
it		0.19	
		Sum:	

The Illustrated Transformer, Jay Alammar

Content: Compute feature attribution using…

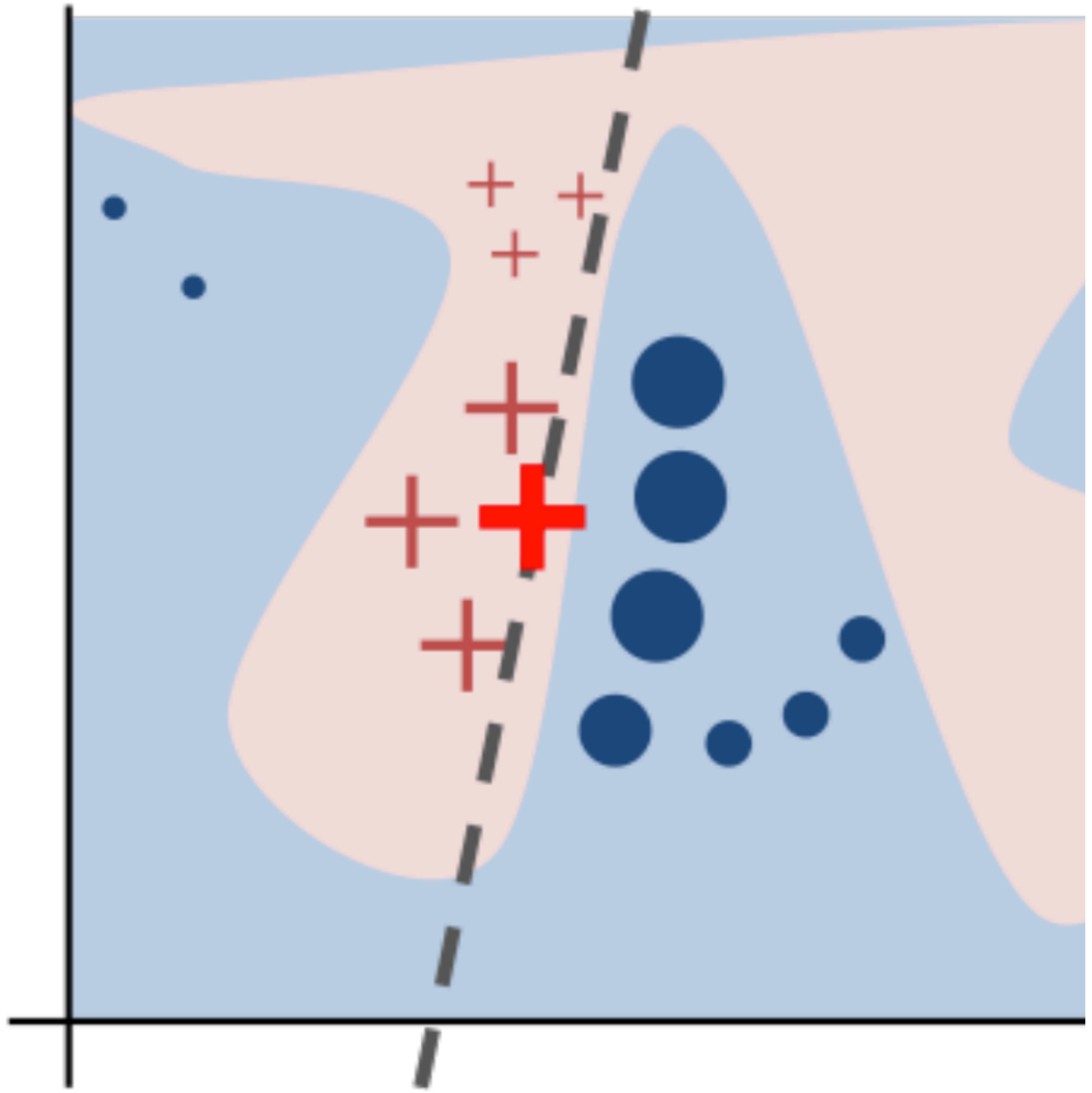
LIME: Compute local linear approximation of the model's behaviour

"While the model may be very complex globally, it is easier to approximate it around the vicinity of a particular instance. "

Look at model's predictions for a bunch of nearby inputs.
Closer points are more important than further points.
Fit a linear model. Its weights are the feature importances.

The movie is mediocre , maybe even bad .

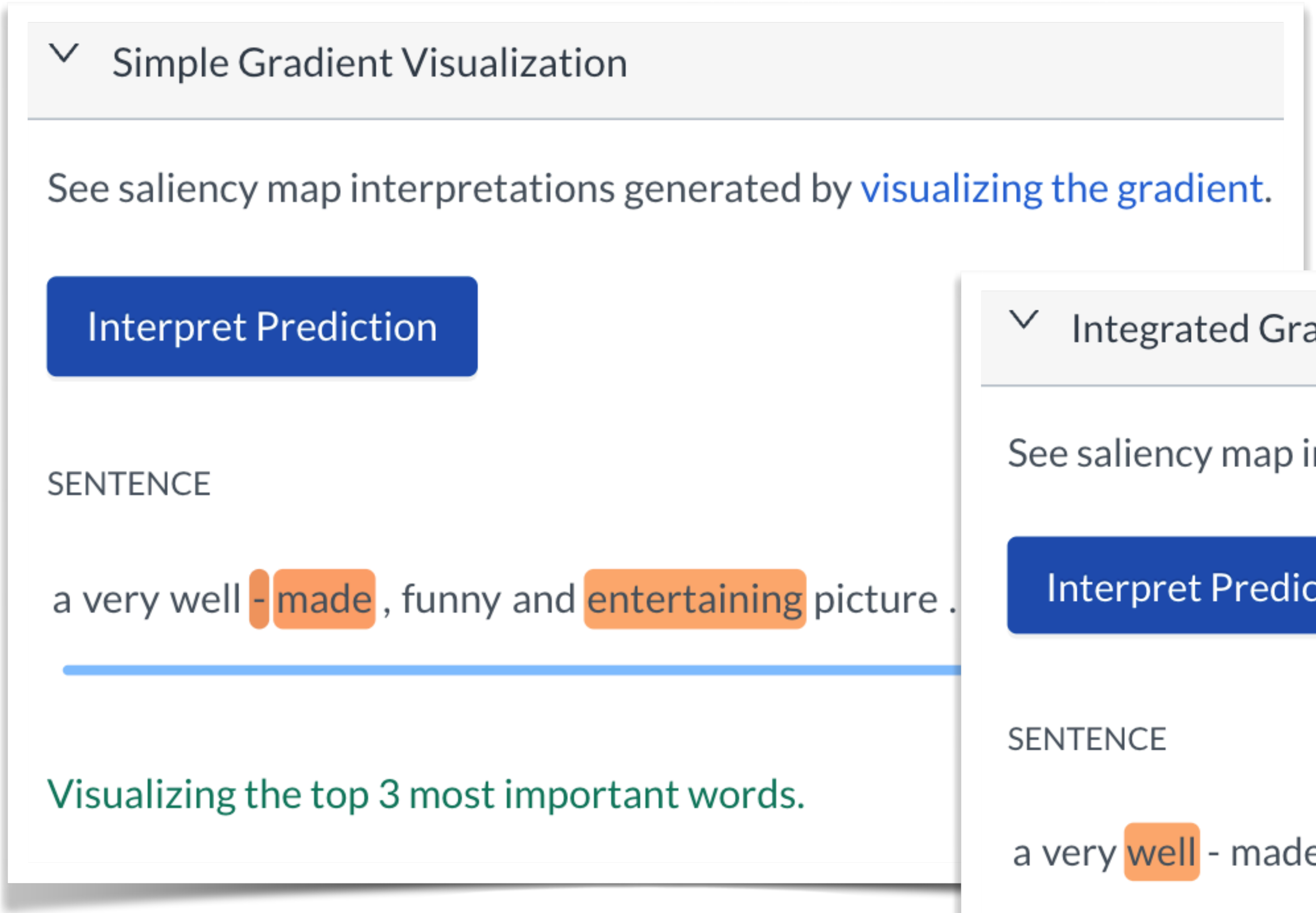
The movie is mediocre, maybe even ~~bad~~. Negative 98.0%
The movie is ~~mediocre~~, maybe even bad. Negative 98.7%
The movie is ~~mediocre~~, maybe even ~~bad~~. Positive 63.4%
The movie is ~~mediocre~~, ~~maybe~~ even bad. Positive 74.5%
The ~~movie~~ is mediocre, maybe even ~~bad~~. Negative 97.9%

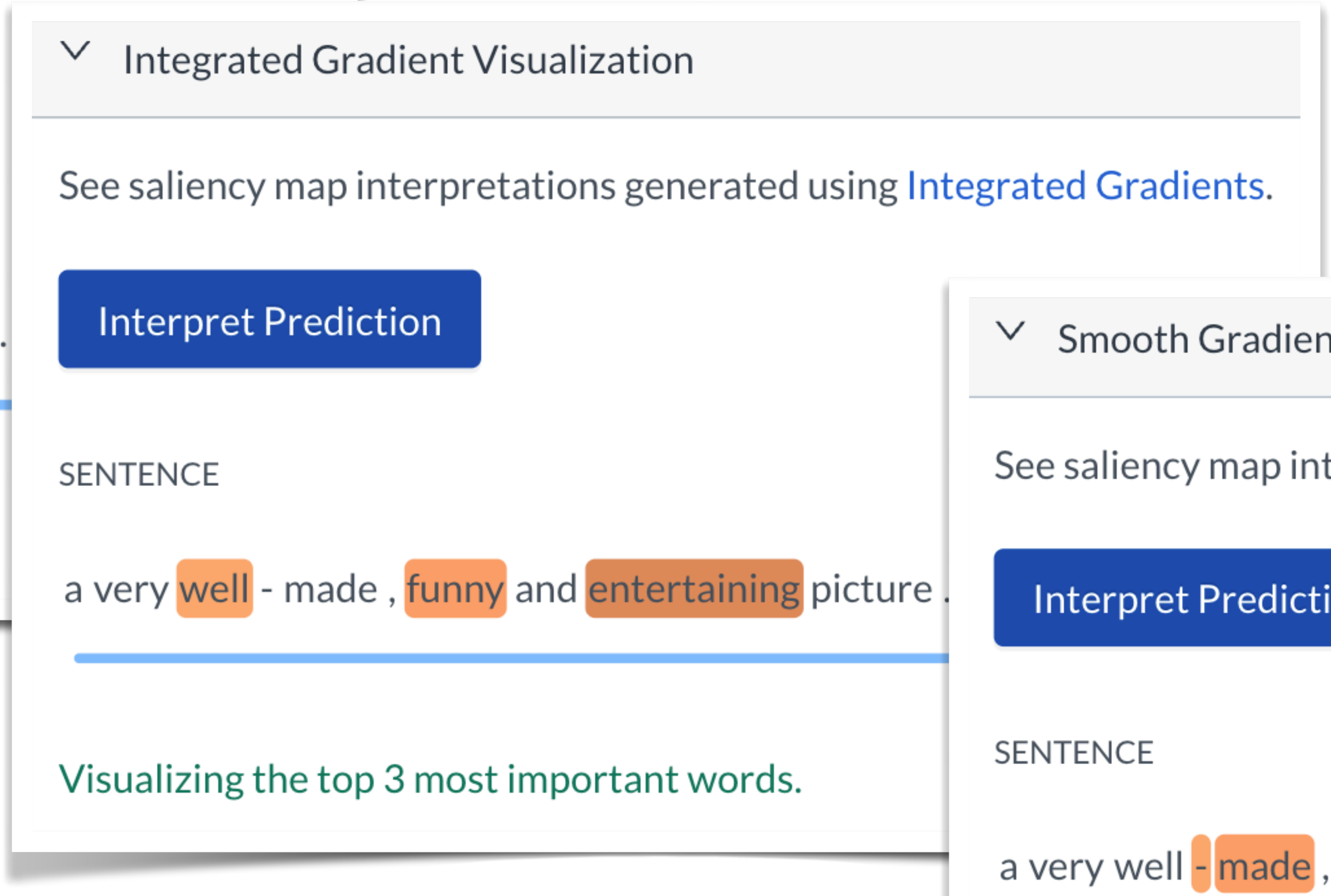


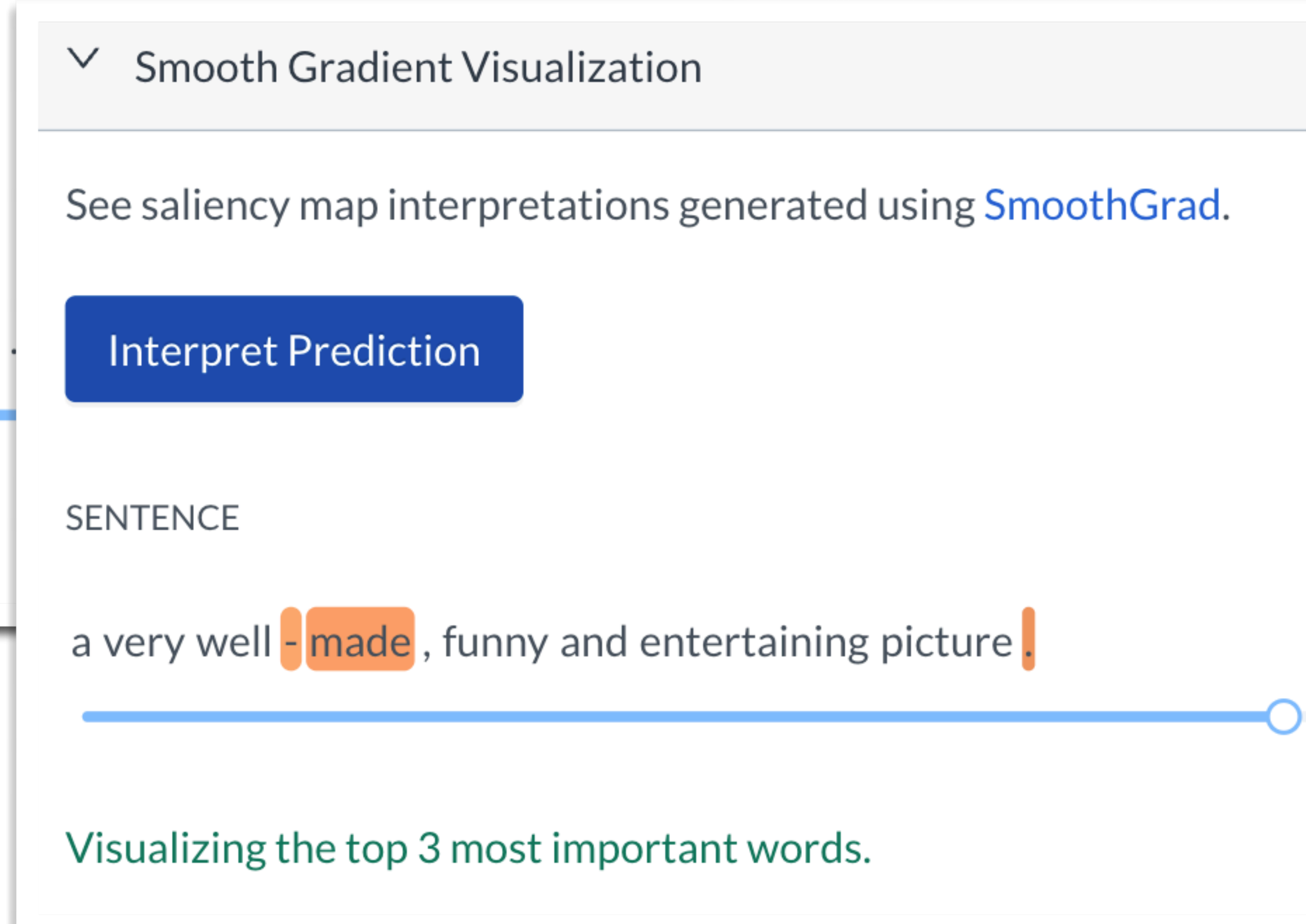
Marco Tulio, Sameer Singh, and Carlos Guestrin. "" Why should i trust you?" Explaining the predictions of any classifier." *KDD* 2016.

Reflection: Same visualization different computation

Essentially we end up with a score on each token, seems intuitive to use consistent visualization if we are comparing their algorithms.

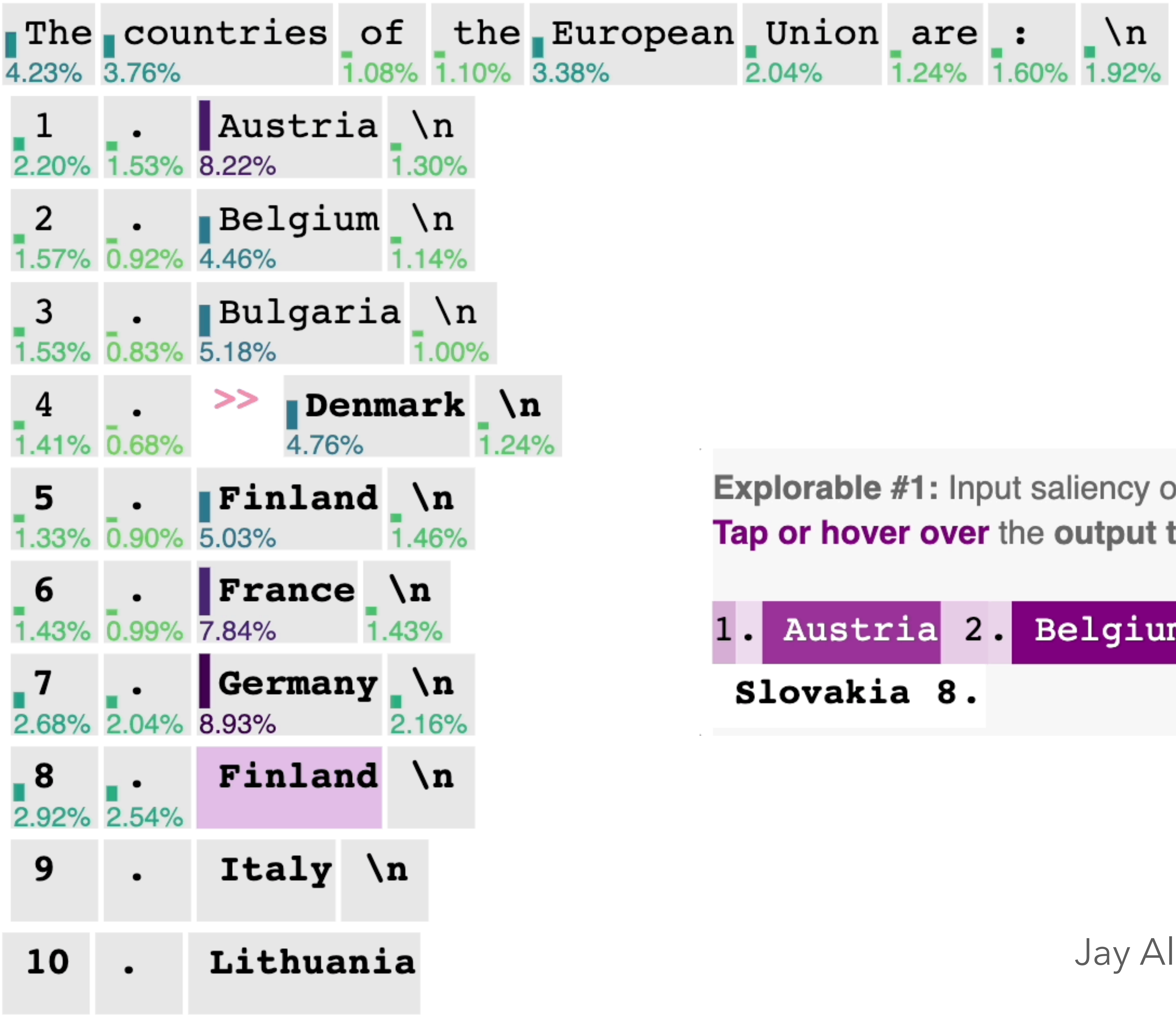






Wallace, Eric, et al. "Allennlp interpret: A framework for explaining predictions of nlp models." EMNLP 2019

Reflection: Different visualization same computation



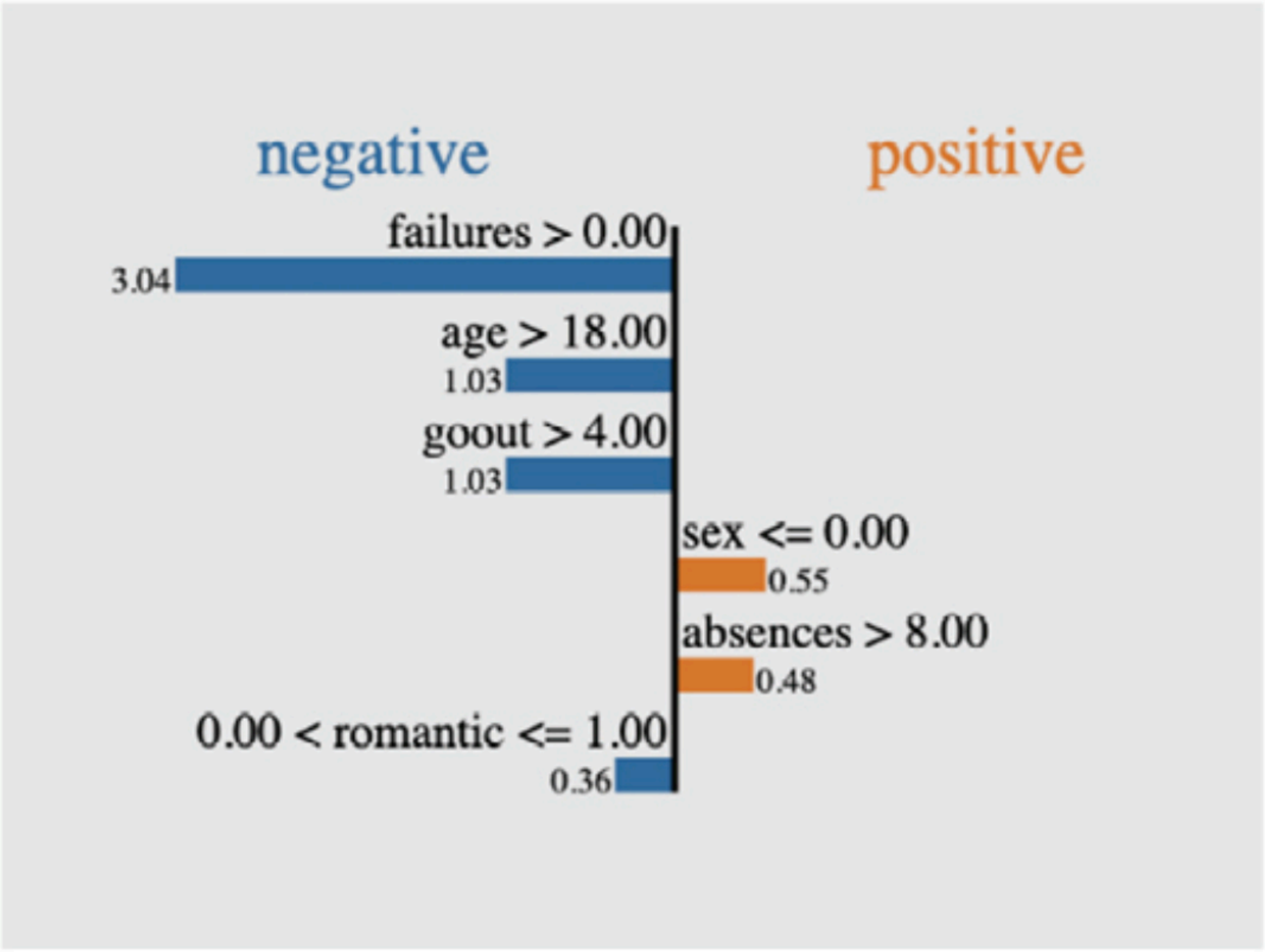
If we fix the algorithm and change the visualization, does that come with any effect?

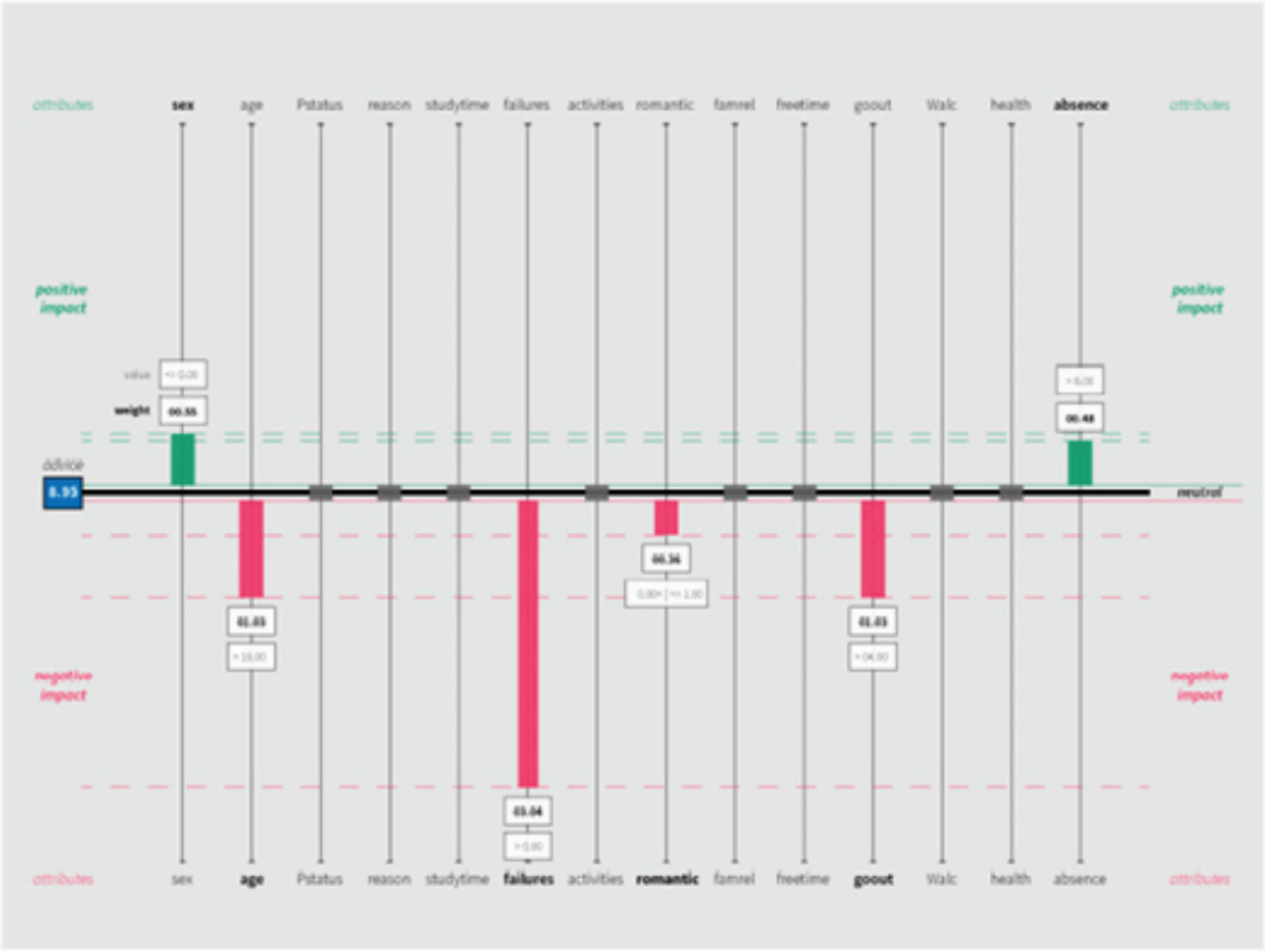
Do you have a preference and why?

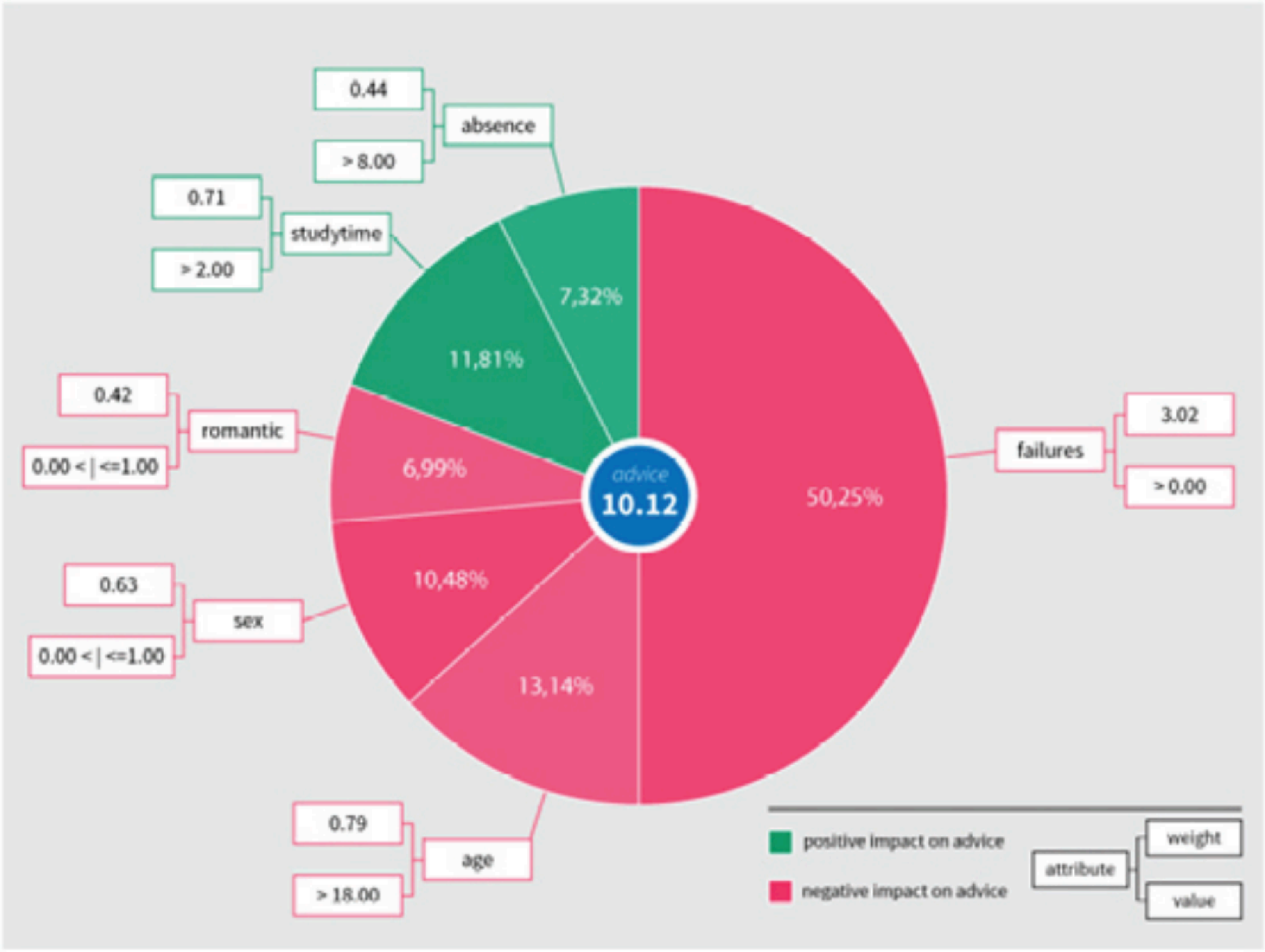
Jay Alammar. "Interfaces for Explaining Transformer Language Models." *2022*.

Prior work shows that people do have preferences.

“We found there are significant effects between treatments. We conclude that the exact form of visually representing (LIME) explanations is relevant for the design of explanations in Human-AI interactions.”







Mucha, Henrik, et al. "Interfaces for explanations in human-AI interaction: proposing a design evaluation approach." *CHI EA*. 2021.

Visual encoding has effectiveness ranking

Visual encoding: Assign **data fields** to **visual channels** (x, y, color, shape, size, …) for a chosen graphical mark type (point, bar, line, …). Also choose appropriate encoding parameters (log scale, sorting, …) and data transformations (bin, group, aggregate, …)

Data field types:

Nominal (labels or categories): Fruits: apples, oranges, …

Operations: =, ≠

Ordered: Quality of meat: Grade A, AA, AAA Q

Operations: =, ≠, >, <

Quantitative - Interval: Dates: Jan, 19, 2006; Location: (LAT 33.98, LONG -118.45)

Operations: =, ≠, >, <, -

Quantitative - Ratio (zero fixed) Physical measurement: Length, Mass, Temp, …

Operations: =, ≠, >, <, -, %

Visual encoding has effectiveness ranking

QUANTITATIVE	ORDINAL	NOMINAL
Position	Position	Position
Length	Density (Value)	Color Hue
Angle	Color Sat	Texture
Slope	Color Hue	Connection
Area (Size)	Texture	Containment
Volume	Connection	Density (Value)
Density (Value)	Containment	Color Sat
Color Sat	Length	Shape
Color Hue	Angle	Length
Texture	Slope	Angle
Connection	Area (Size)	Slope
Containment	Volume	Area
Shape	Shape	Volume

Slides: Jeffrey Heer, UW
CSE 512 Visualization

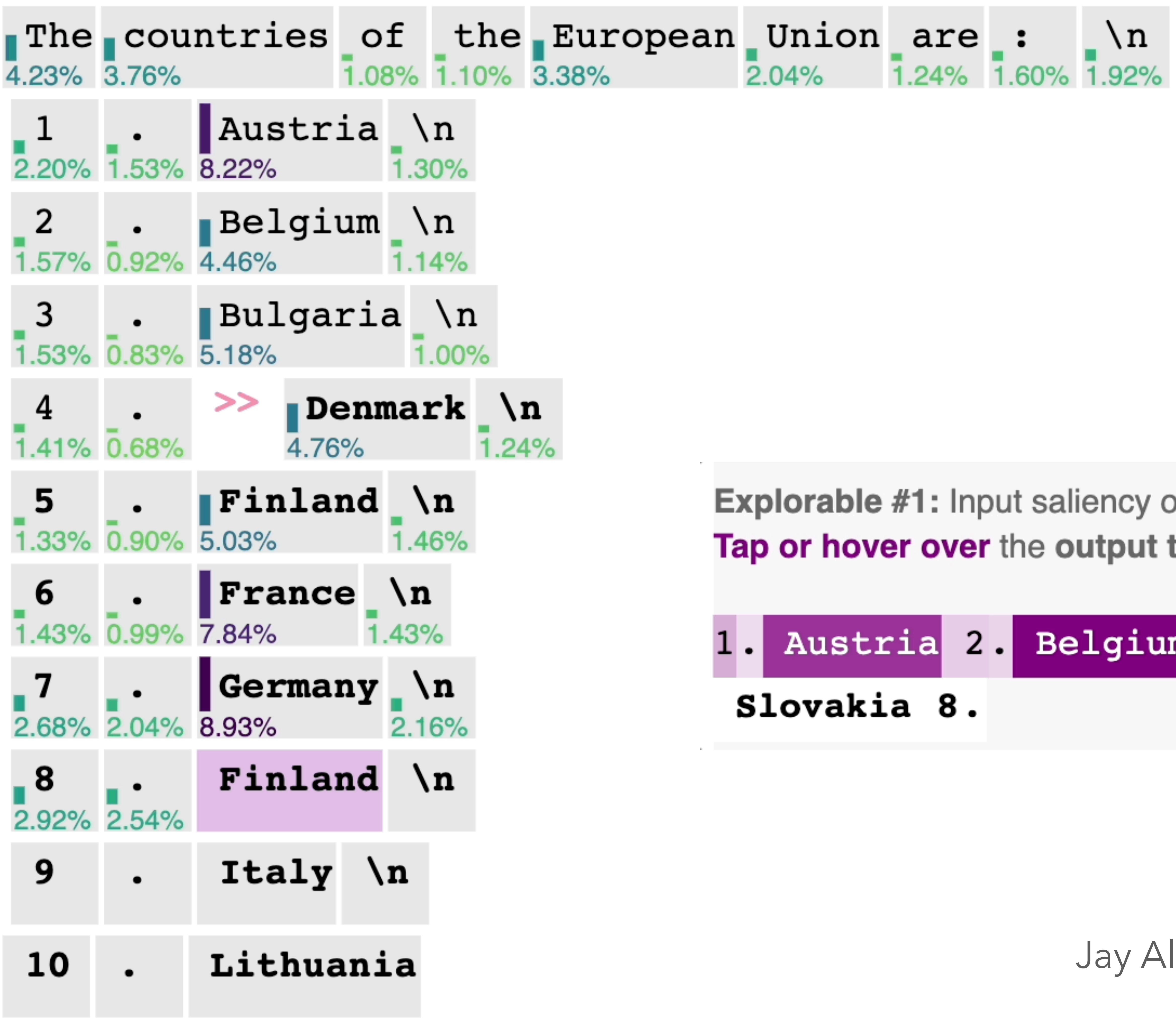
Visual encoding has effectiveness ranking

QUANTITATIVE	ORDINAL	NOMINAL
Position	**Position**	**Position**
Length	Density (Value)	Color Hue
Angle	**Color Saturation**	Texture
Slope	Color Hue	Connection
Area (Size)	Texture	Containment
Volume	Connection	Density (Value)
Density (Value)	Containment	**Color Saturation**
Color Saturation	Length	Shape
Color Hue	Angle	Length
Texture	Slope	Angle
Connection	Area (Size)	Slope
Containment	Volume	Area
Shape	Shape	Volume

Some visual encoding channels tend to be more effective across data types; Some channels are only effective in limited cases.

Slides: Jeffrey Heer, UW
CSE 512 Visualization

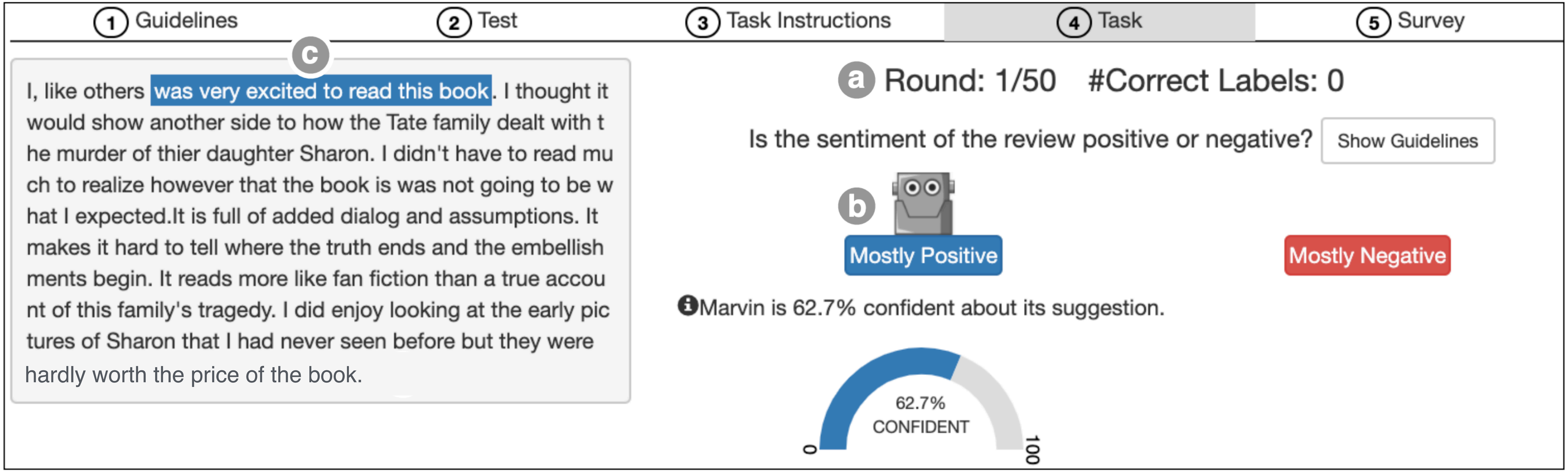
Reflection: Different visualization same computation



If you prefer exact numbers (**quantitative**), bar charts make sense (**length encoding**). But most of time we only care about the rough relative importance (for glance – **ordinal**!), which makes color more effective.

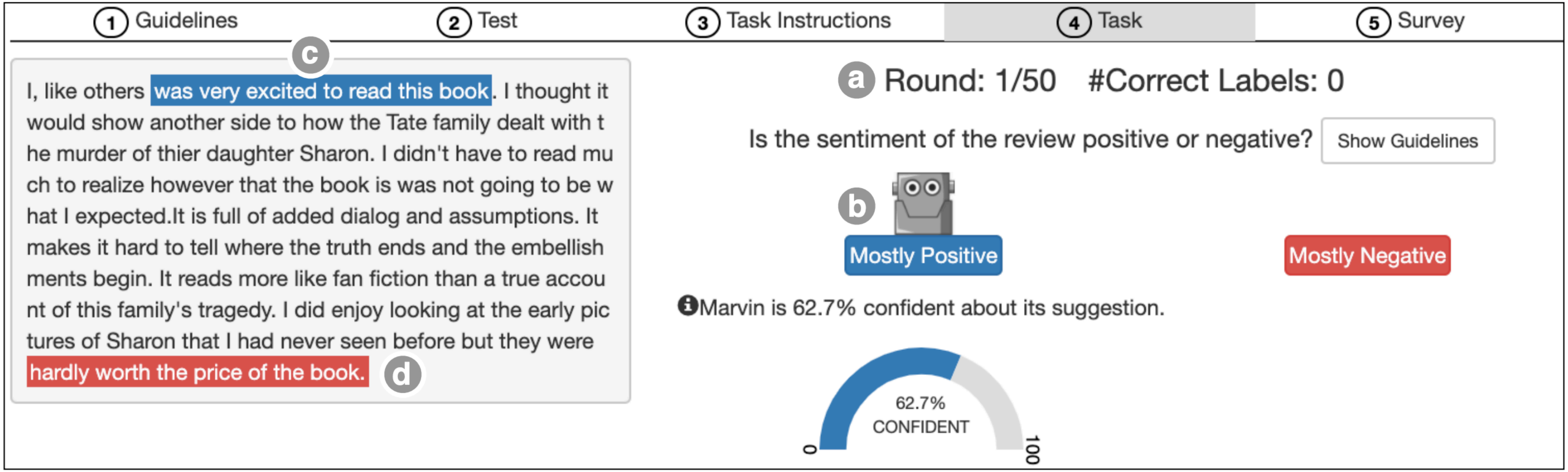
Jay Alammar. "Interfaces for Explaining Transformer Language Models." *2022*.

Cautious! Saliency map leads to cognitive bias.



Tongshuang Wu, Gagan Bansal*, Joyce Zhou+, Raymond Fok+, Besmira Nushi, Ece Kamar, Marco Tulio Ribeiro, Daniel S. Weld.* ***Does the Whole Exceed its Parts? The Effect of AI Explanations on Complementary Team Performance.*** *In the 2021 Conference on Human Factors in Computing Systems.*

Cautious! Saliency map leads to cognitive bias.



Tongshuang Wu, Gagan Bansal*, Joyce Zhou+, Raymond Fok+, Besmira Nushi, Ece Kamar, Marco Tulio Ribeiro, Daniel S. Weld. **Does the Whole Exceed its Parts? The Effect of AI Explanations on Complementary Team Performance.** In the 2021 Conference on Human Factors in Computing Systems.*

“Parameters” for a visualization

Goal	**Content**	**Encoding**	**Context**
Why visualize	*What to visualize*	*How to visualize*	*Assist communication*
Local understand	Input distribution	Line chart	Annotations
Global understand	In-/out-put mapping	Bar chart	Text integration
Communication	Activations	Scatter plot	Aggregation
Education	**Attention**	Graph	Dimension reduction
	Postdoc explanations	**Saliency map**	Small multiples
	Architecture		
	Parameter spaces		

“Parameters” for a visualization

Goal	Content	Encoding	Context
Why visualize	*What to visualize*	*How to visualize*	*Assist communication*
Local understand	Input distribution	**Line chart**	Annotations
Global understand	In-/out-put mapping	Bar chart	Text integration
Communication	**Activations**	Scatter plot	Aggregation
Education	Attention	Graph	Dimension reduction
	Postdoc explanations	**Saliency map**	Small multiples
	Architecture		
	Parameter spaces		

Neuron Activations & Factor Analysis

Inspect neuron firings inside deep neural networks can reveal the complementary and compositional roles that can be played by individual neurons, and groups of neurons.

Compared to saliency maps: Deeper understanding of the model structure.

But, more information to interpret (usually end up "forcing" meanings onto factors)

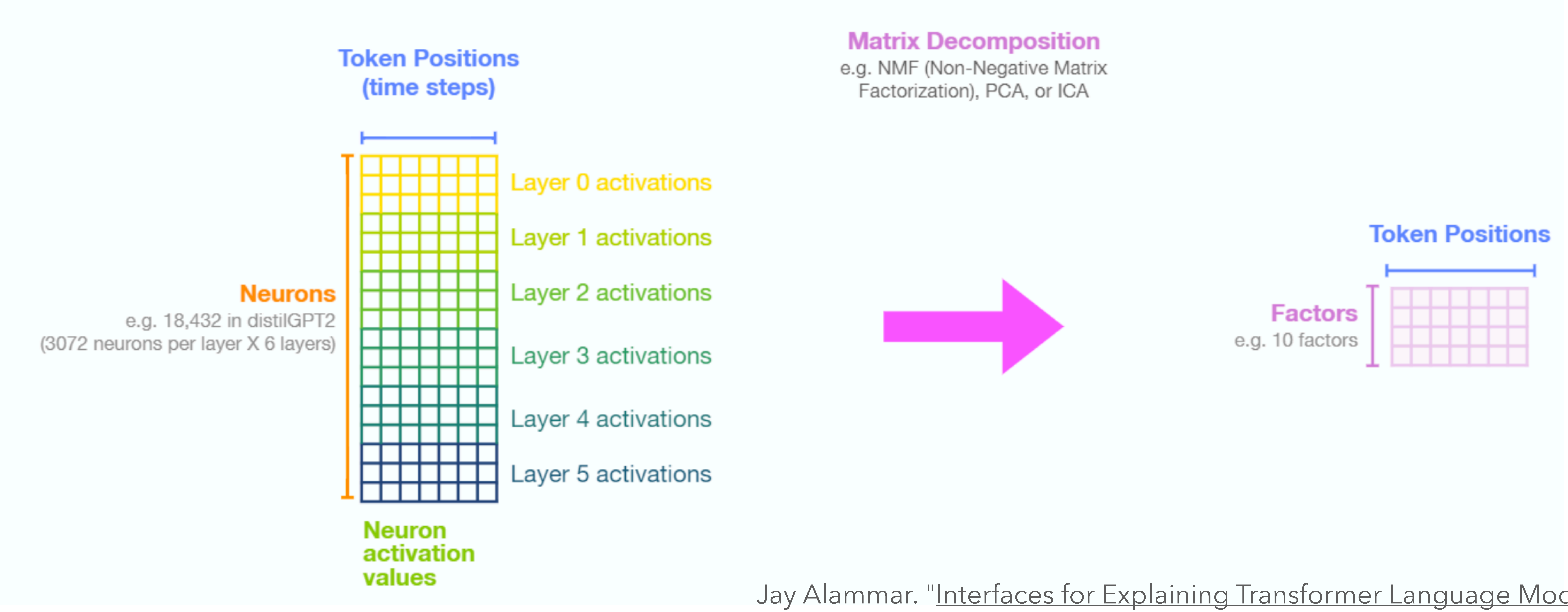
Explorable #2: Neuron activation analysis reveals four groups of neurons, each is associated with generating a certain type of token

Tap or hover over the sparklines on the left to isolate a certain factor:

Jay Alammar. "Interfaces for Explaining Transformer Language Models." *2022*.

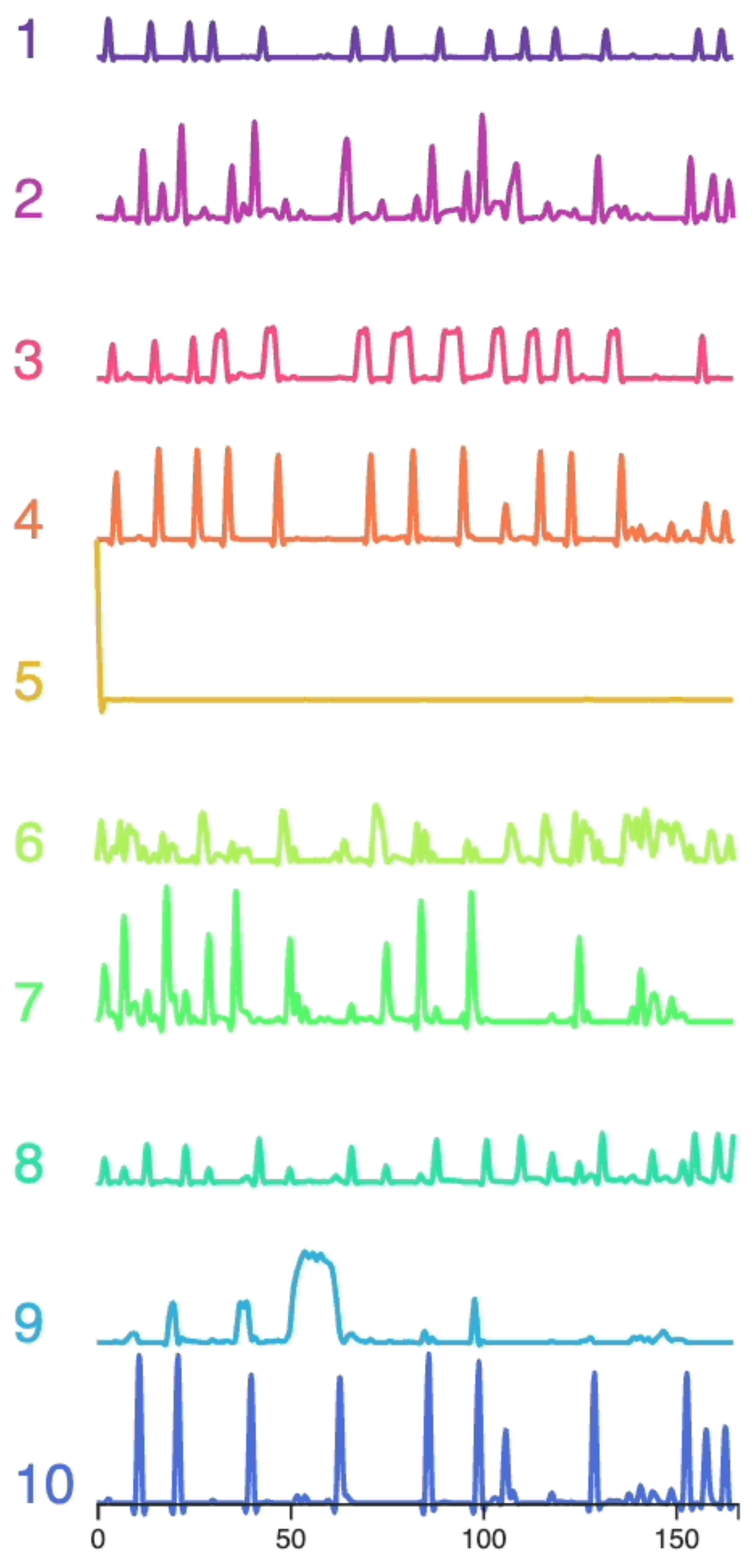
Neuron Activations & Factor Analysis

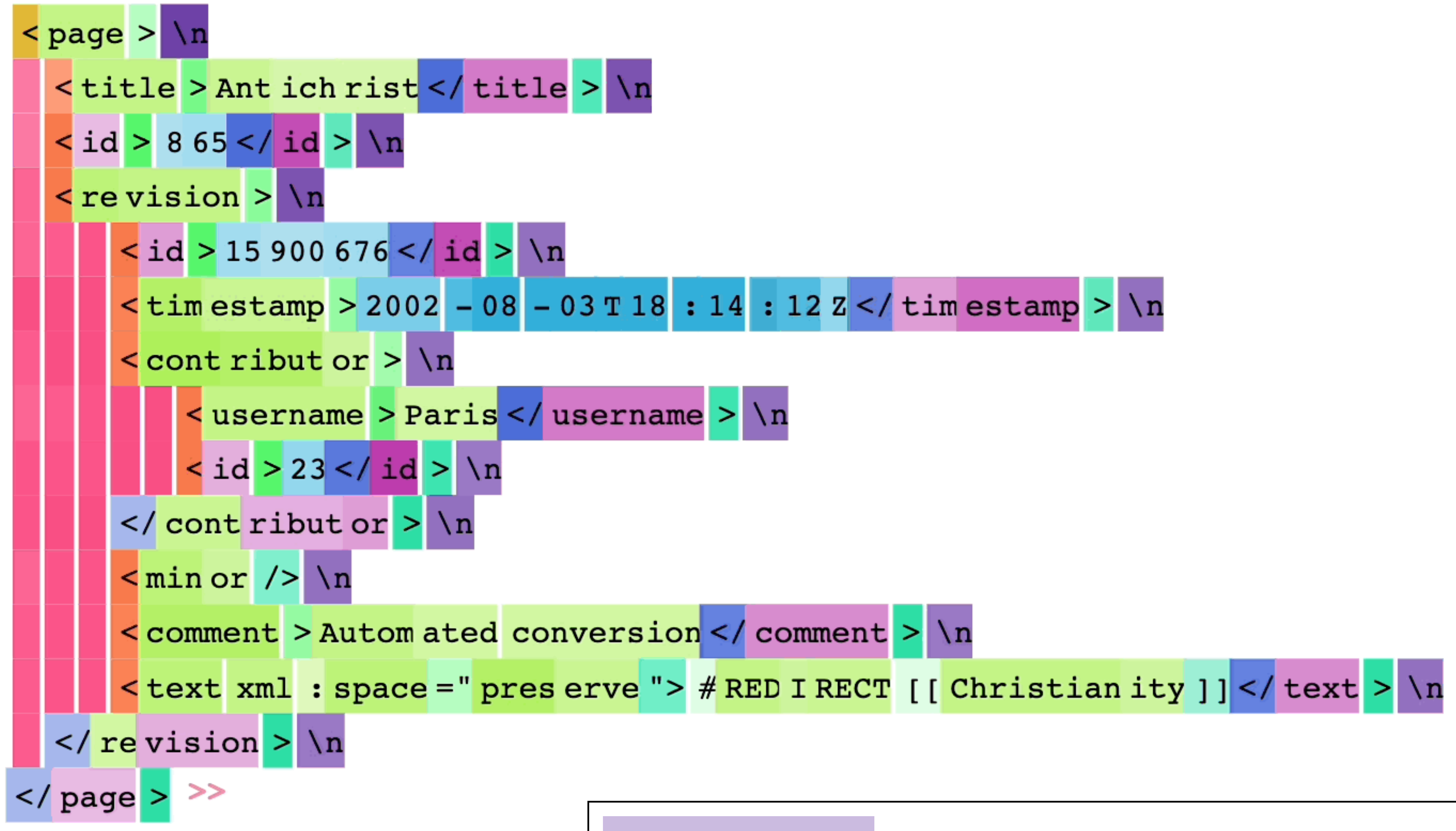
Factor analysis is done by decomposing the matrix holding the activations values of Feed-Forward Neural Network (FFNN) neurons using Non-negative Matrix Factorization. It can be used to analyze the entire network, a single layer, or groups of layers.



Jay Alammar. "Interfaces for Explaining Transformer Language Models." *2022*.

Neuron Activations & Factor Analysis





Useful demo case:
DistilGPT2 reacts to XML.
Shows a clear distinction of factors attending to different components of the syntax.

New-lines
Labels of tags, with higher activation on closing tags
Indentation spaces
The '<' (less-than) character starting XML tags
The large factor focusing on the first token. Common to GPT2 models.
Two factors tracking the '>' (greater than) character at the end of XML tags
The text inside XML tags
The '</' symbols indicating closing XML tag

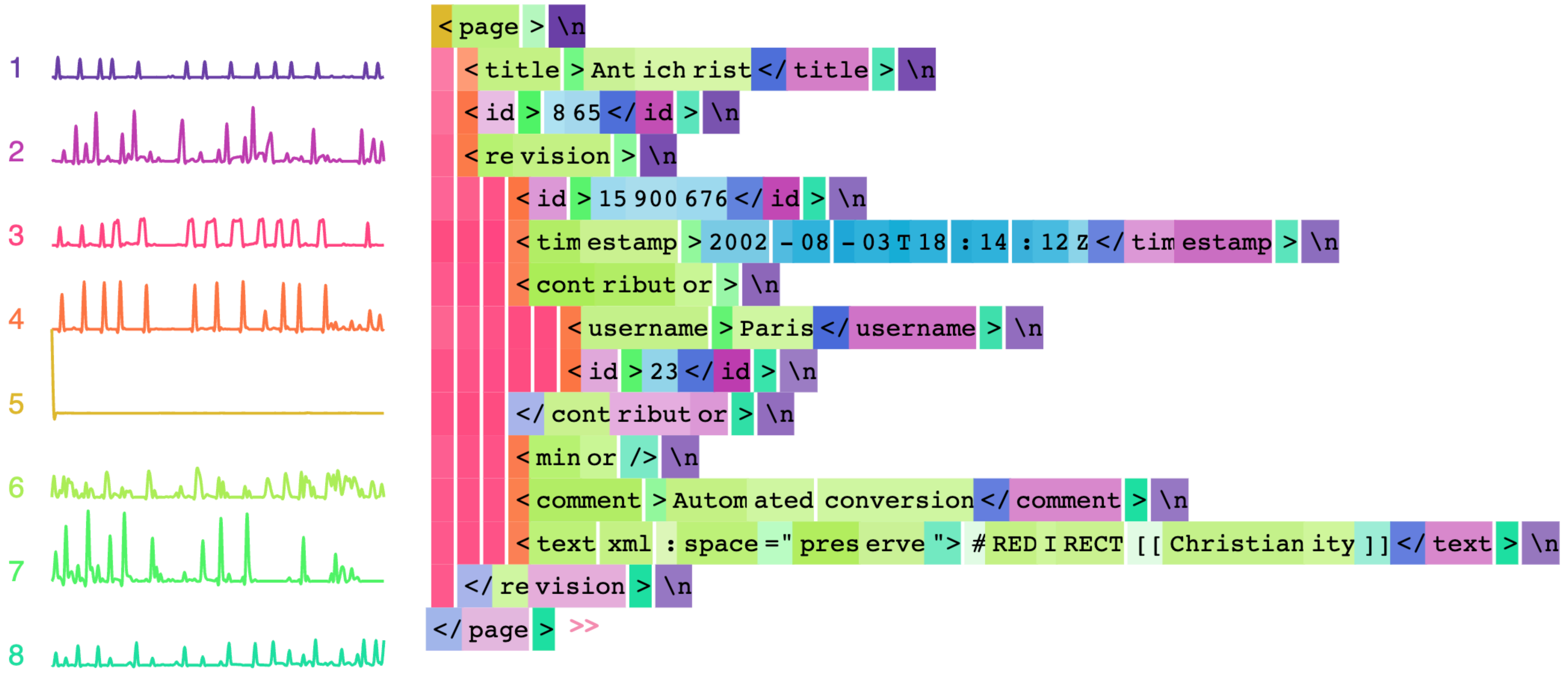
Encoding / viz.: Important techniques

"Overview first, details on-demand": People have limited attention span. They should be given a high level summary first, before they tailor the viz based on their interest and knowledge.
Overview: line charts + the max color for each token; Detail: coloring tokens using specific factors.

Small multiples: Multiple related charts that share same scale and axis, to compare faceted patterns.

Linked views: Set of coordinated visualizations that are connected such that interactions in one visualization affect the others. Help users explore and analyze data from multiple perspectives.

Text integration: When describing concepts in text, link their representations visually via e.g., thoughtful layout and consistent use of color.

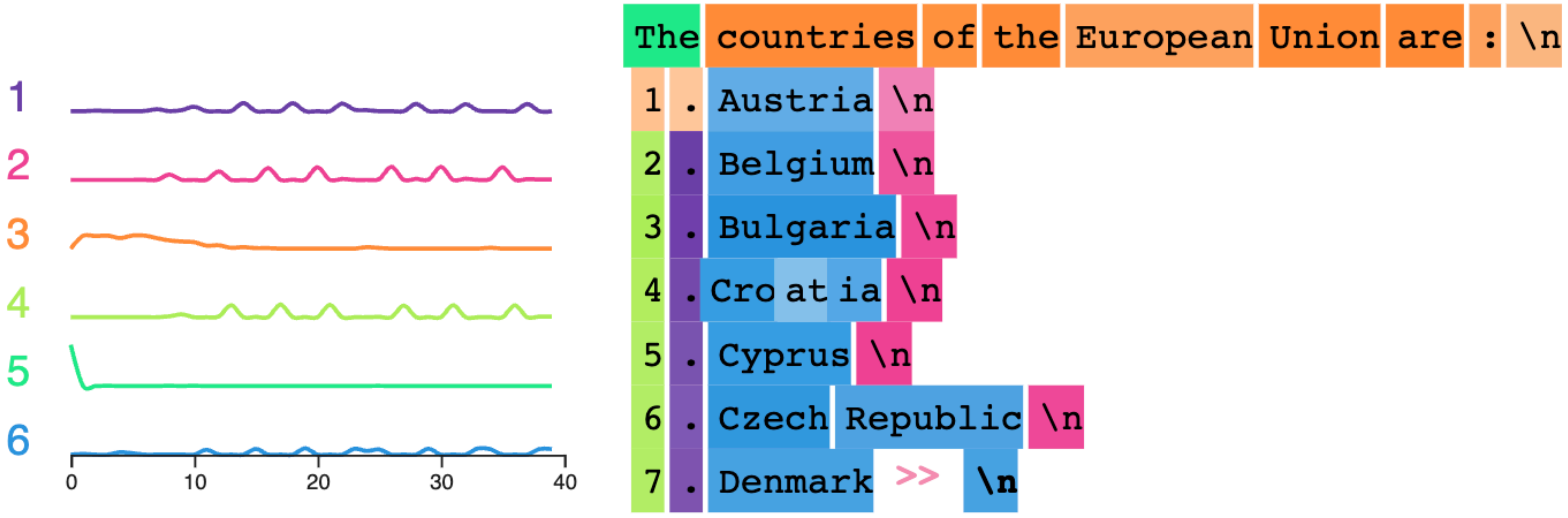


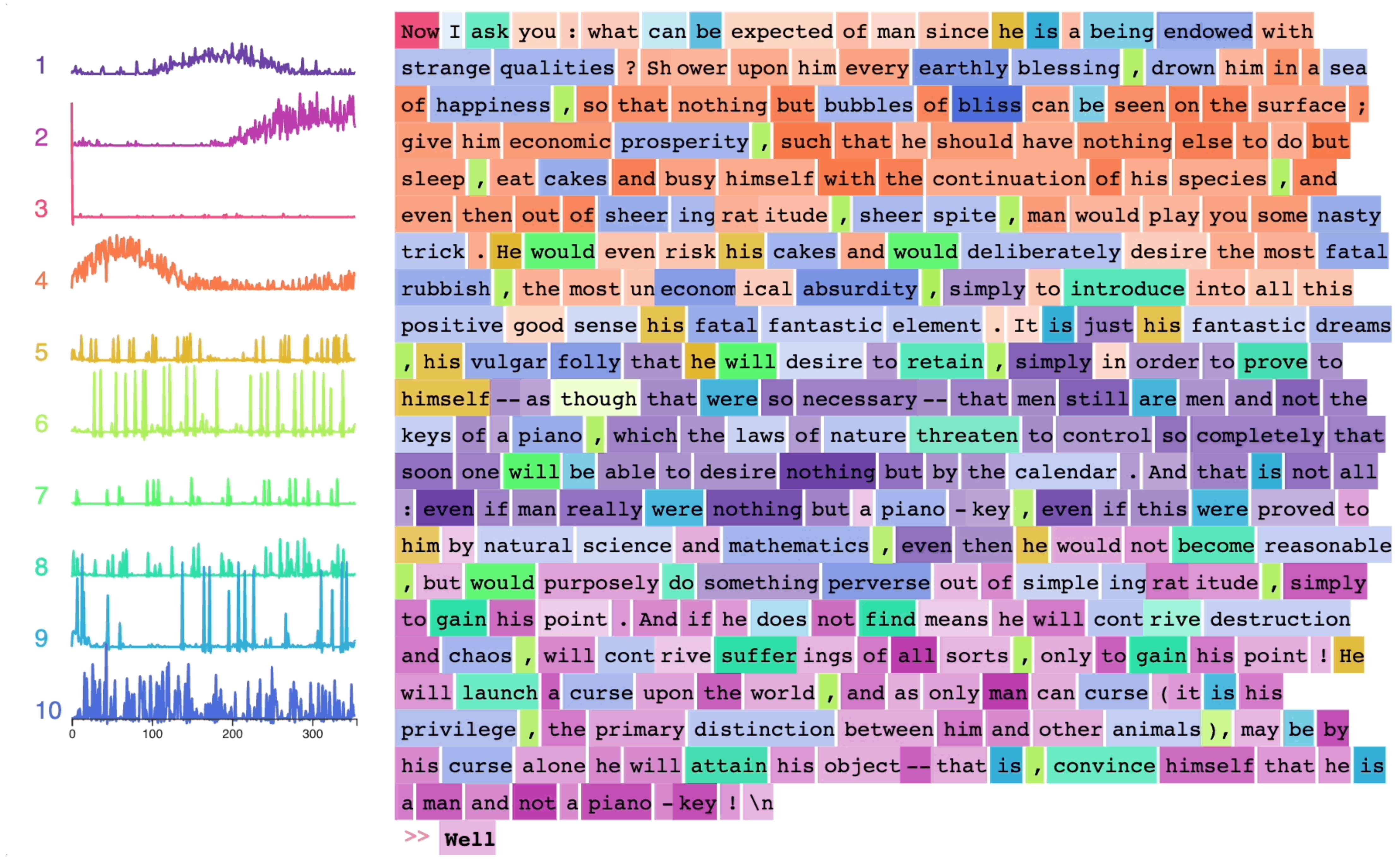
Explorable: Ten Activation Factors of XML
Tap or hover over the sparklines on the left to isolate a certain factor
Factorizing neuron activations in response to XML (that was generated by an RNN from [33]) into ten factors results in factors corresponding to:

1. New-lines
2. Labels of tags, with higher activation on closing tags
3. Indentation spaces
4. The '<' (less-than) character starting XML tags
5. The large factor focusing on the first token. Common to GPT2 models.
6. Two factors tracking the '>' (greater than) character at the end of XML tags
7. The text inside XML tags
8. The '</' symbols indicating closing XML tag

Is the same overview always useful?

Scalability challenge!
> 5 color is usually overwhelming, oscillating colors (or lines) are also overwhelming.
Some times **smoothing** is helpful.





https://jalammar.github.io/explaining-transformers/

Bonus: More encoding on text

Font size, color, overlaid shapes, etc. can all be in-situ encoding for documents.

Parra, D., et al. "Analyzing the design space for visualizing neural attention in text classification." VIS*xAI*. 2019.

"Parameters" for a visualization

Goal

Why visualize

Local understand

Global understand

Communication

Education

Content

What to visualize

Input distribution

In-/out-put mapping

Activations

Attention

Postdoc explanations

Architecture

Parameter spaces

Encoding

How to visualize

Line chart

Bar chart

Scatter plot

Graph

Saliency map

Context

Assist communication

Annotations

Text integration

Aggregation

Dimension reduction

Small multiples

“Parameters” for a visualization

Goal
Why visualize

Local understand

Global understand

Communication

Education

Content
What to visualize

Input distribution

In-/out-put mapping

Activations

Attention

Postdoc explanations

Architecture

Parameter spaces

Encoding
How to visualize

Line chart

Bar chart

Scatter plot

Graph

Saliency map

Context
Assist communication

Annotations

Text integration

Aggregation

Dimension reduction

Small multiples

“Parameters” for a visualization

Goal

Why visualize

Local understand

Global understand

Communication

Education

Content

What to visualize

Input distribution

In-/out-put mapping

Activations

Attention

Postdoc explanations

Architecture

Parameter spaces

Encoding

How to visualize

Line chart

Bar chart

Scatter plot

Graph

Saliency map

Context

Assist communication

Annotations

Text integration

Aggregation

Dimension reduction

Small multiples

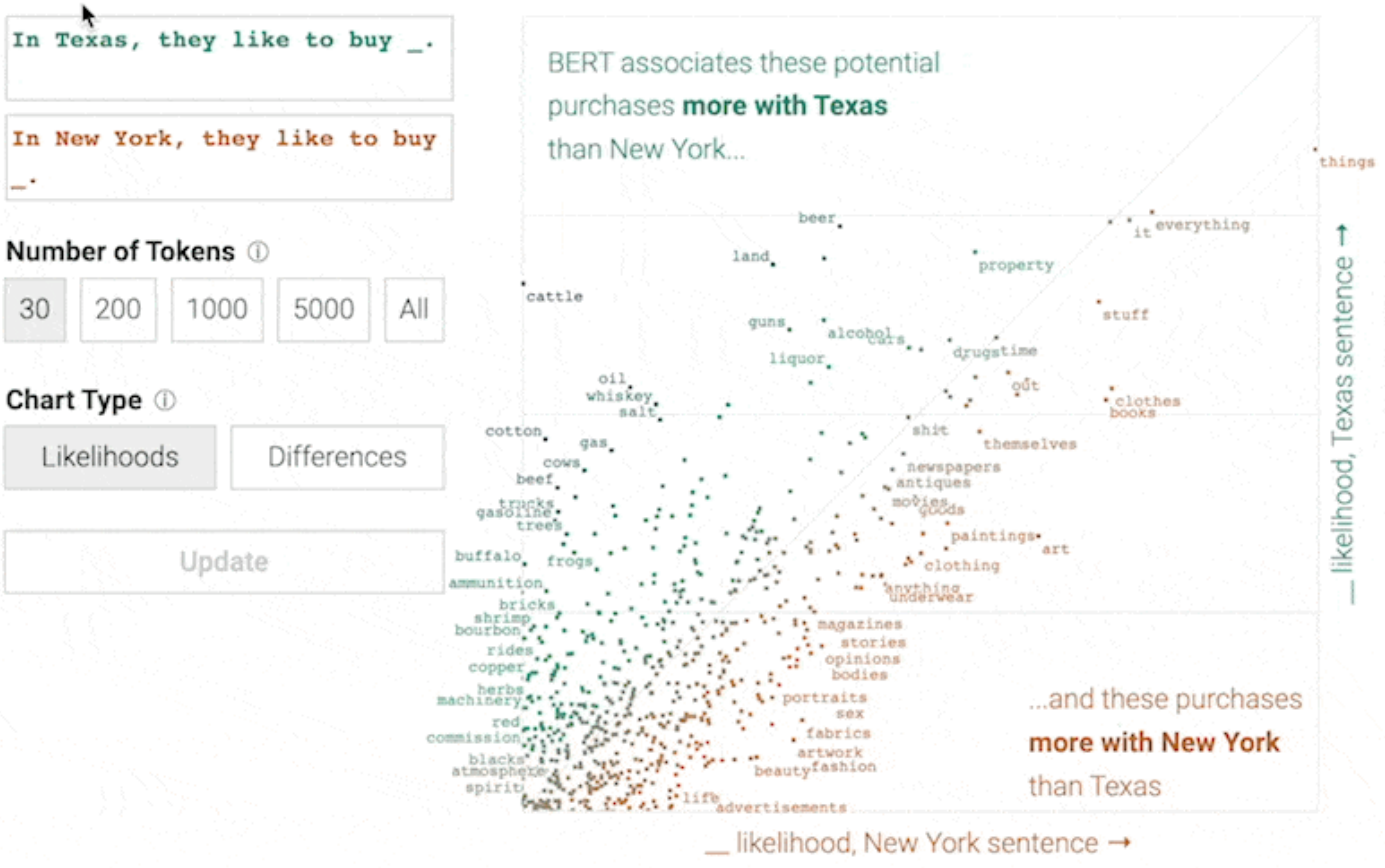
Key: Project info onto readable dimensions

Global understanding usually involves contrasting outputs with inputs.

When we have many outputs, good to **map them out on dimensions we care about**.

Color encoding helps highlight the contrast.

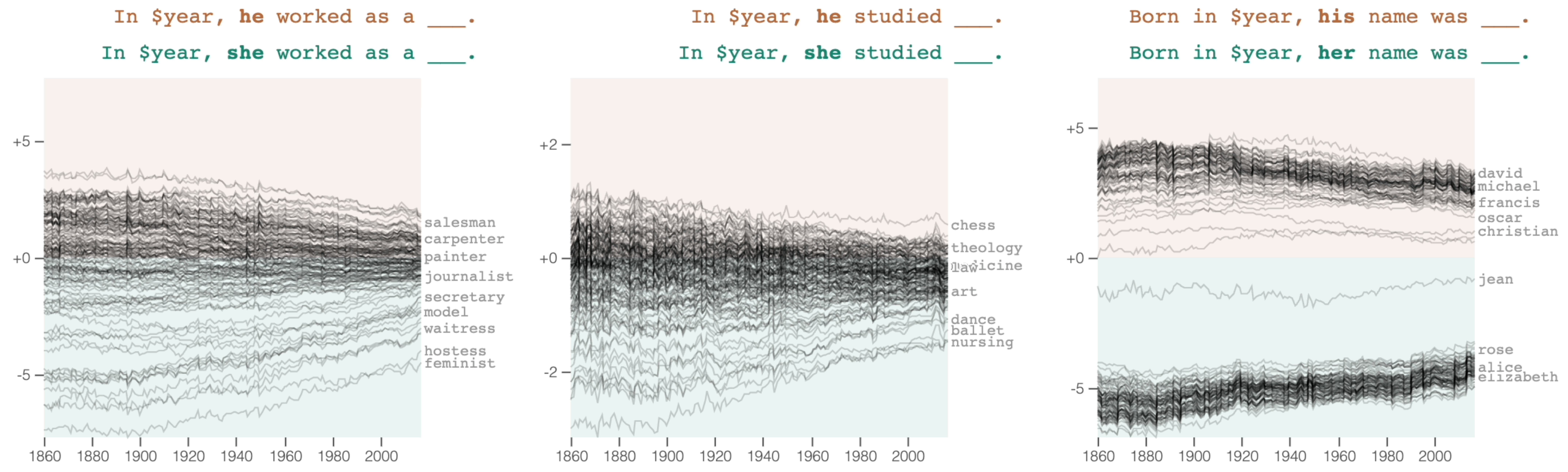
Annotations: help the reader orient by pointing out examples of patterns and important elements.



Adam Pearce. "What Have Language Models Learned?" *2021*.

Key: Project info onto readable dimensions

Some amount of **aggregation** is also important. Here we are interested in the temporal trend, which is more suitable for line chart (vs. bar chart or scatter plots). As a result, one dimension is fixed to be year, and top words are annotated on the side



The top 150 "he" and "she" completions in years from 1860-2018 are shown with the y position encoding he_logit - she_logit. Run in Colab →

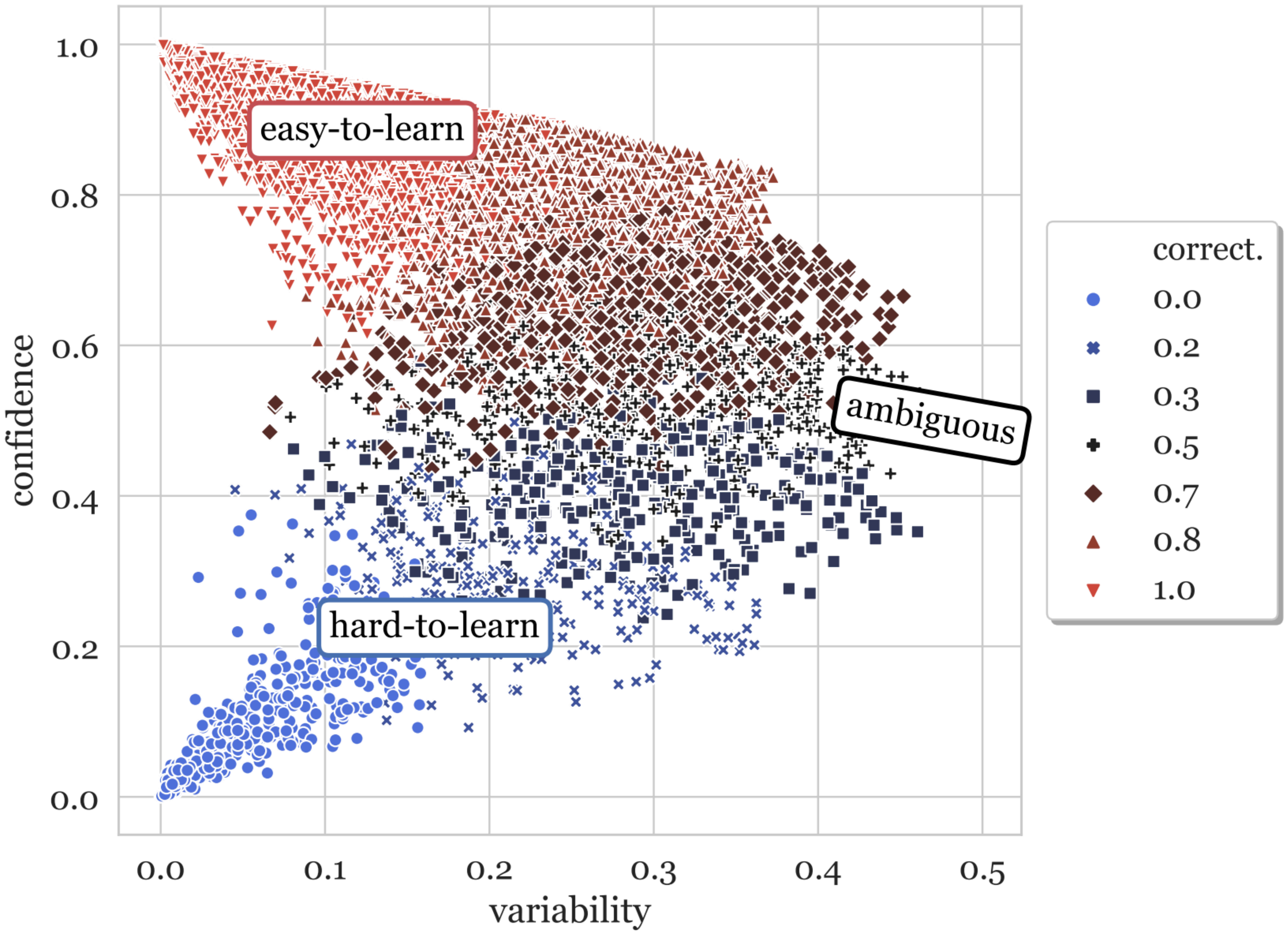
Adam Pearce. "What Have Language Models Learned?" *2021*.

Different projection: Dataset Difficulty (Data Map)

Instances that a model always predicts correctly are different from those it almost never does, or those on which it vacillates.

How to get data map?

Confidence and **Variability**: the mean and standard deviation of the gold label probabilities, predicted for each example across training epochs.

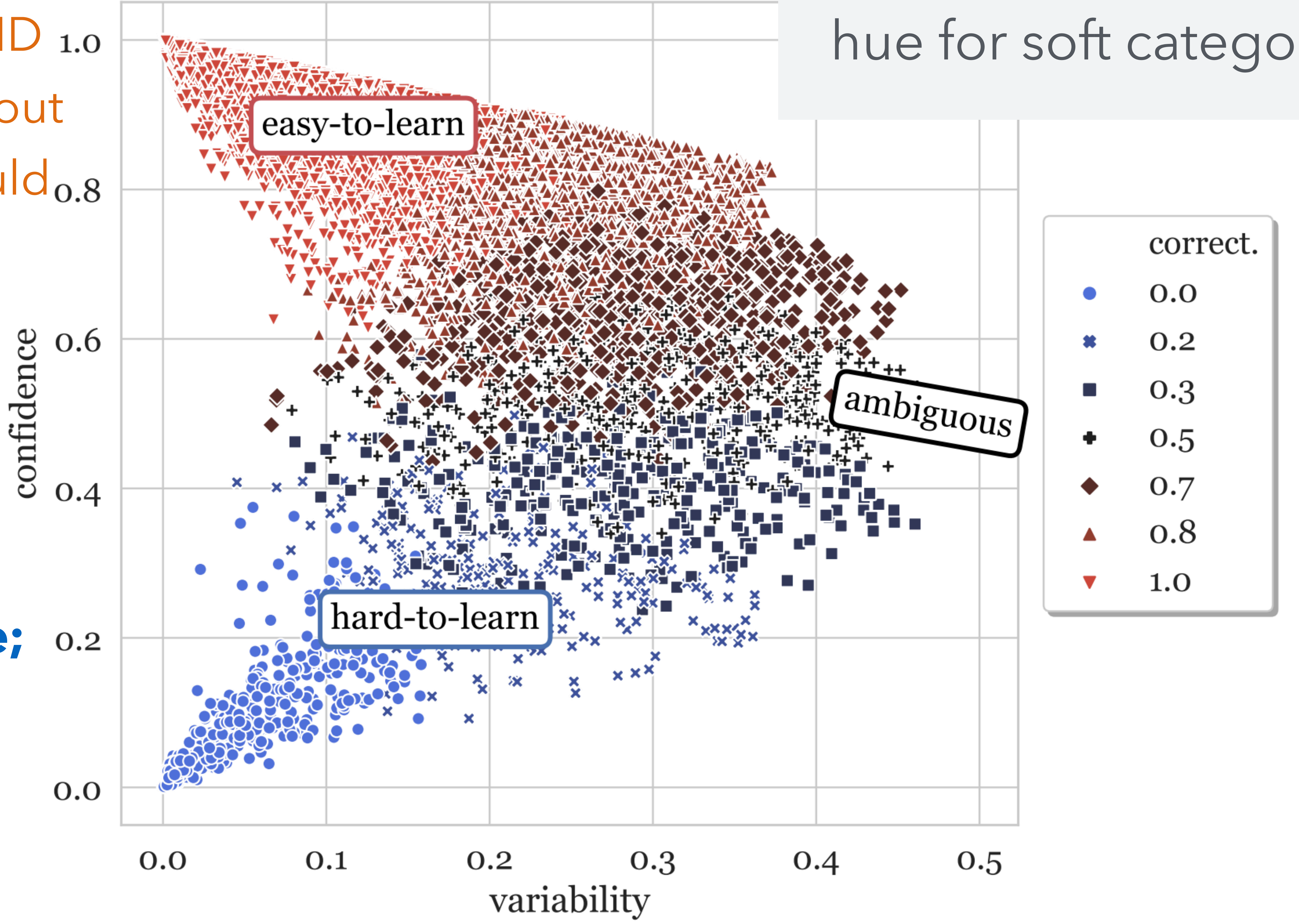


Swayamdipta, Swabha, et al. "Dataset cartography: Mapping and diagnosing datasets with training dynamics." *EMNLP 2020*

Different projection: Dataset Difficulty (Data Map)

low variability, high confidence;
play an important role in model optimization. Not as critical for ID or OOD performance, but without any such instances, training could fail to converge

Color encoding: Continuous, 2-D color hue for soft categorization.



High variability;
Promotes generalization to out-of-distribution test sets, with little or no effect on in-distribution (ID) performance.

Low variability, low confidence;
often correspond to labeling errors.

Swayamdipta, Swabha, et al. "Dataset cartography: Mapping and diagnosing datasets with training dynamics." *EMNLP 2020*

Projection through Dimensionality Reduction

When we don't have dimensions we can define clearly, we rely on **automated methods** that project nD data to (not as interpretable) 2D or 3D for viewing. Often used to interpret and sanity check high-dimensional representations fit by ML methods.

DR methods are used to aid interpretation, but are also **subject to their own interpretation issues!**

Different DR methods make different trade-offs: for example to **preserve global structure** (e.g., PCA) or **emphasize local structure** (e.g., nearest-neighbor approaches, including t-SNE and UMAP).

Slides: Jeffrey Heer, UW
CSE 512 Visualization

Dimensionality Reduction Methods

Principal Components Analysis (PCA)

Linear transformation of basis vectors, ordered by amount of data variance they explain.

t-Dist. Stochastic Neighbor Embedding (t-SNE)

Probabilistically model distance, optimize positions.

Uniform Manifold Approx. & Projection (UMAP)

Identify local manifolds, then stitch them together.

Slides: Jeffrey Heer, UW
CSE 512 Visualization

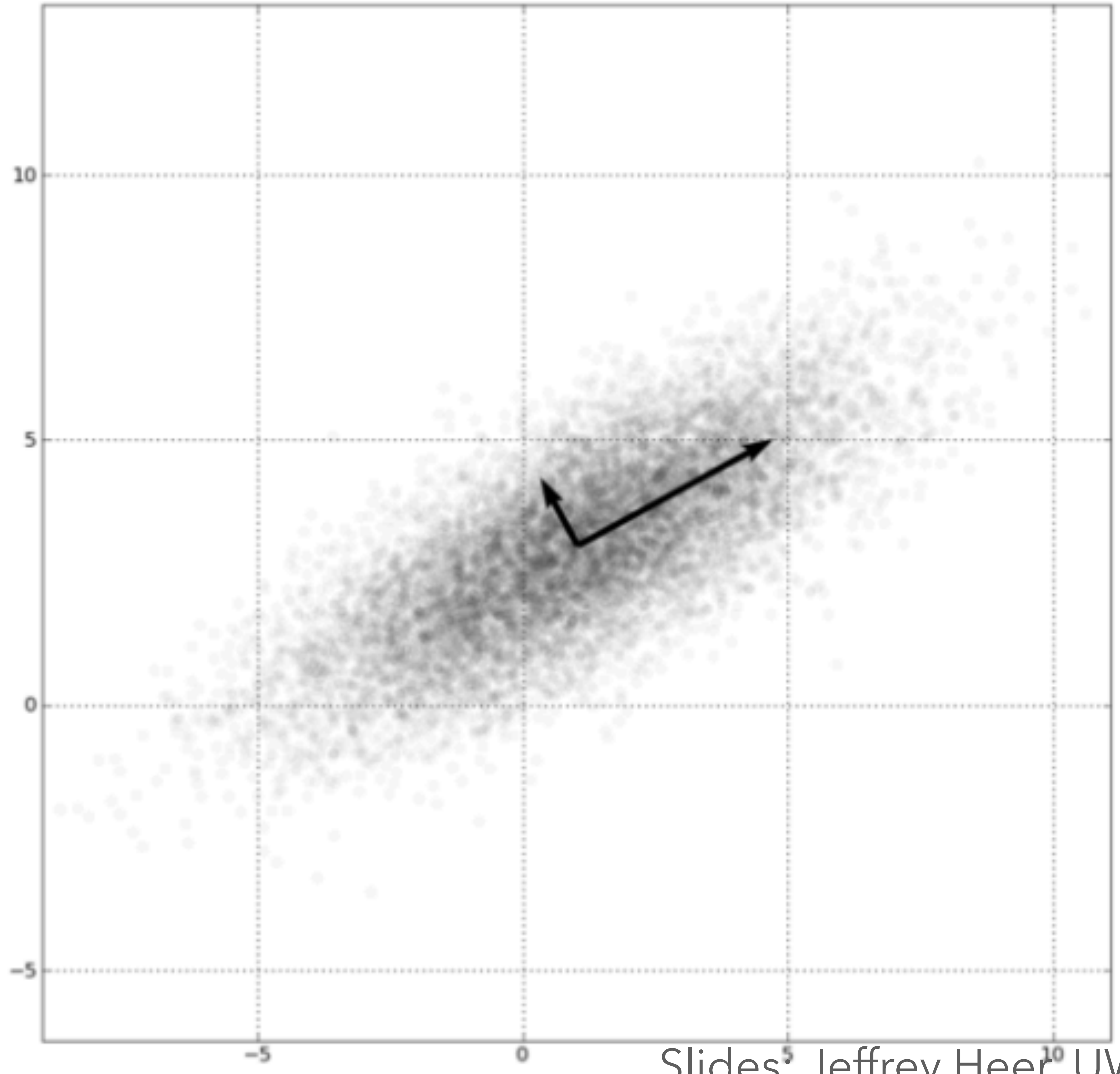
Projection (1/3): Principal Components Analysis

1. Mean-center the data.
2. Find ⊥ basis vectors that maximize the data variance.
3. Plot the data using the top vectors.

Linear transform: scale and rotate original space.

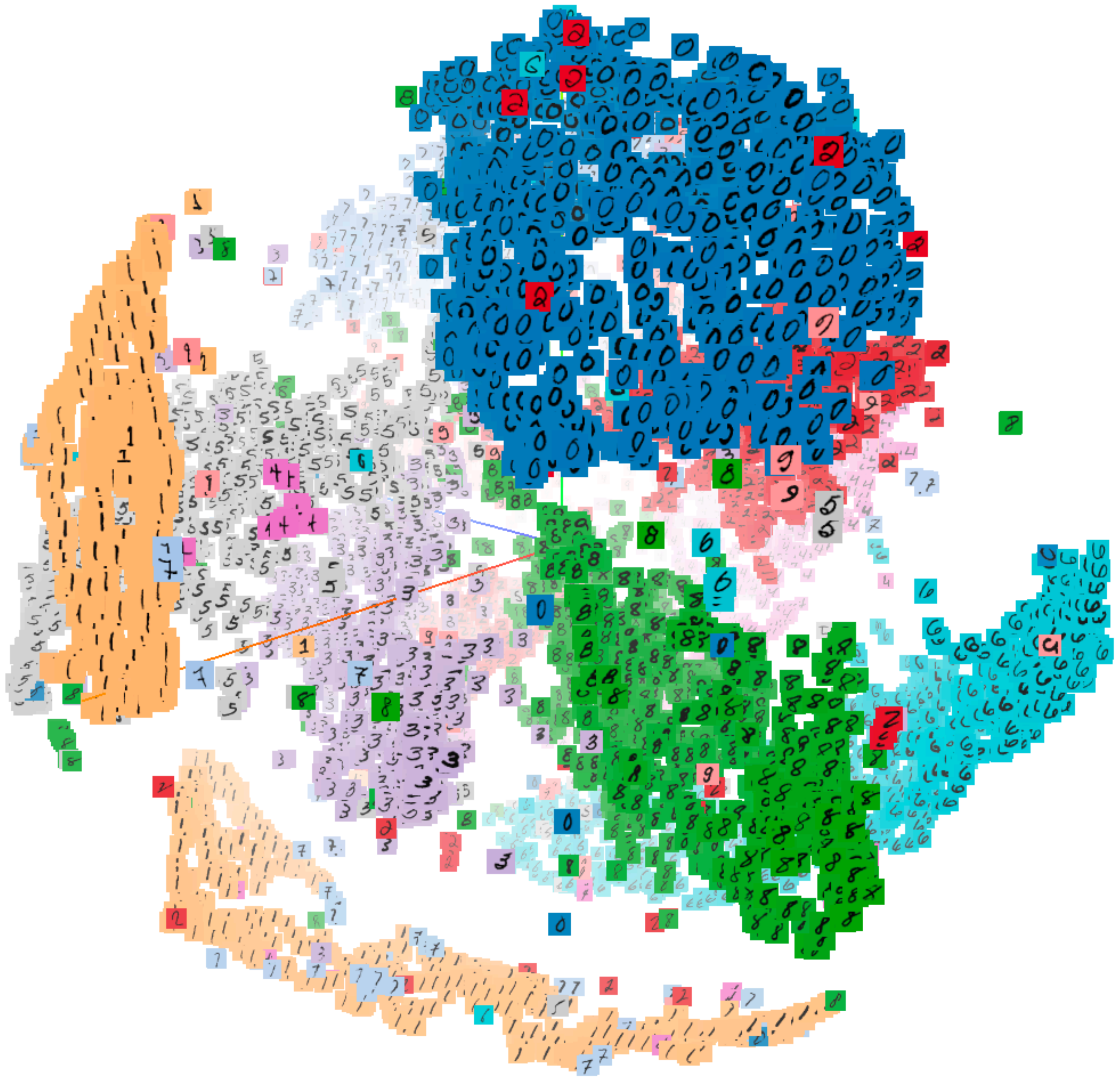
Lines (vectors) project to lines.

Preserves global distances.



Slides: Jeffrey Heer, UW
CSE 512 Visualization

Projection: Non-Linear Techniques



Distort the space, trade-off preservation of global structure to emphasize local neighborhoods. Use topological (nearest neighbor) analysis.

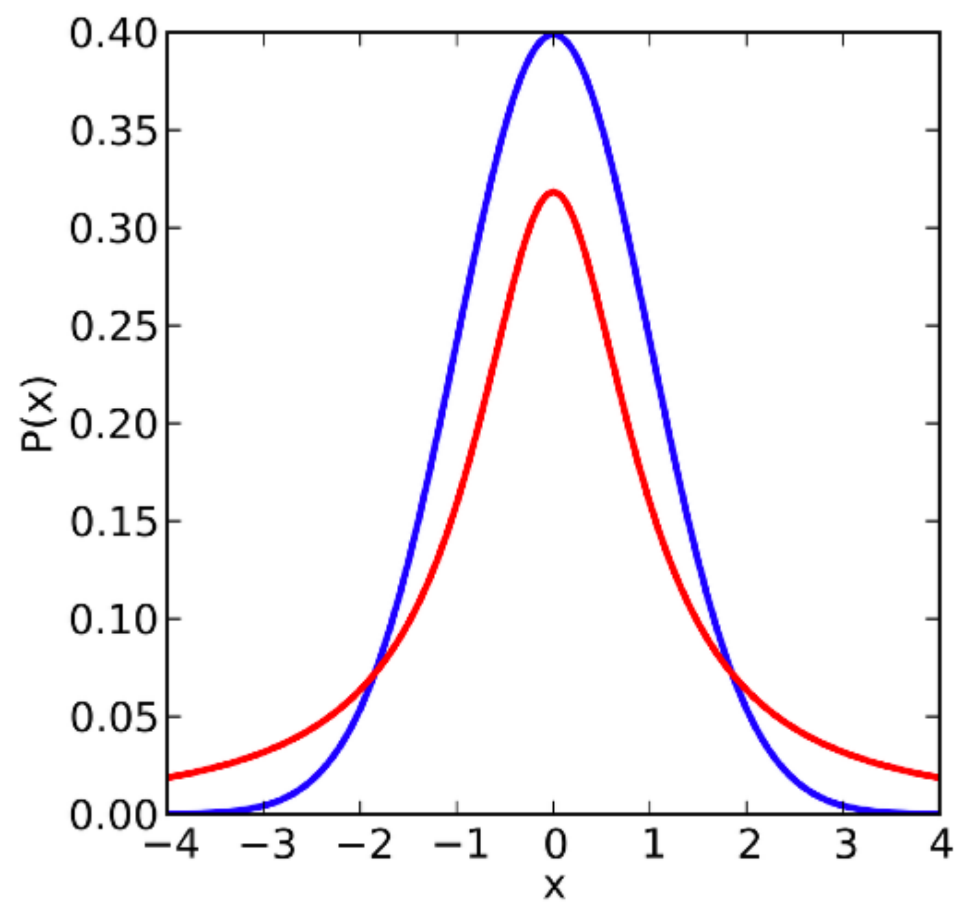
Two popular contemporary methods:
t-SNE - probabilistic interpretation of distance
UMAP - tries to balance local/global trade-off

Slides: Jeffrey Heer, UW
CSE 512 Visualization

Projection (2/3): t-SNE [Maaten & Hinton 2008]

Model probability **P** of one point "choosing" another as its neighbor in the original space, using a Gaussian distribution defined using the distance between points. Nearer points have higher probability than distant ones.

Define a similar probability **Q** in the low-dimensional (2D or 3D) embedding space, using a Student's *t* distribution *(hence the "t-" in "t-SNE"!)*. The *t*-distribution is heavy- tailed, allowing distant points to be even further apart.



Optimize to find the positions in the embedding space that minimize the Kullback-Leibler divergence between the **P** and **Q** distributions: *KL(P || Q)*

Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." *Journal of machine learning research* 9.11 (2008).
https://distill.pub/2016/misread-tsne/

Projection (2/3): t-SNE [Maaten & Hinton 2008]

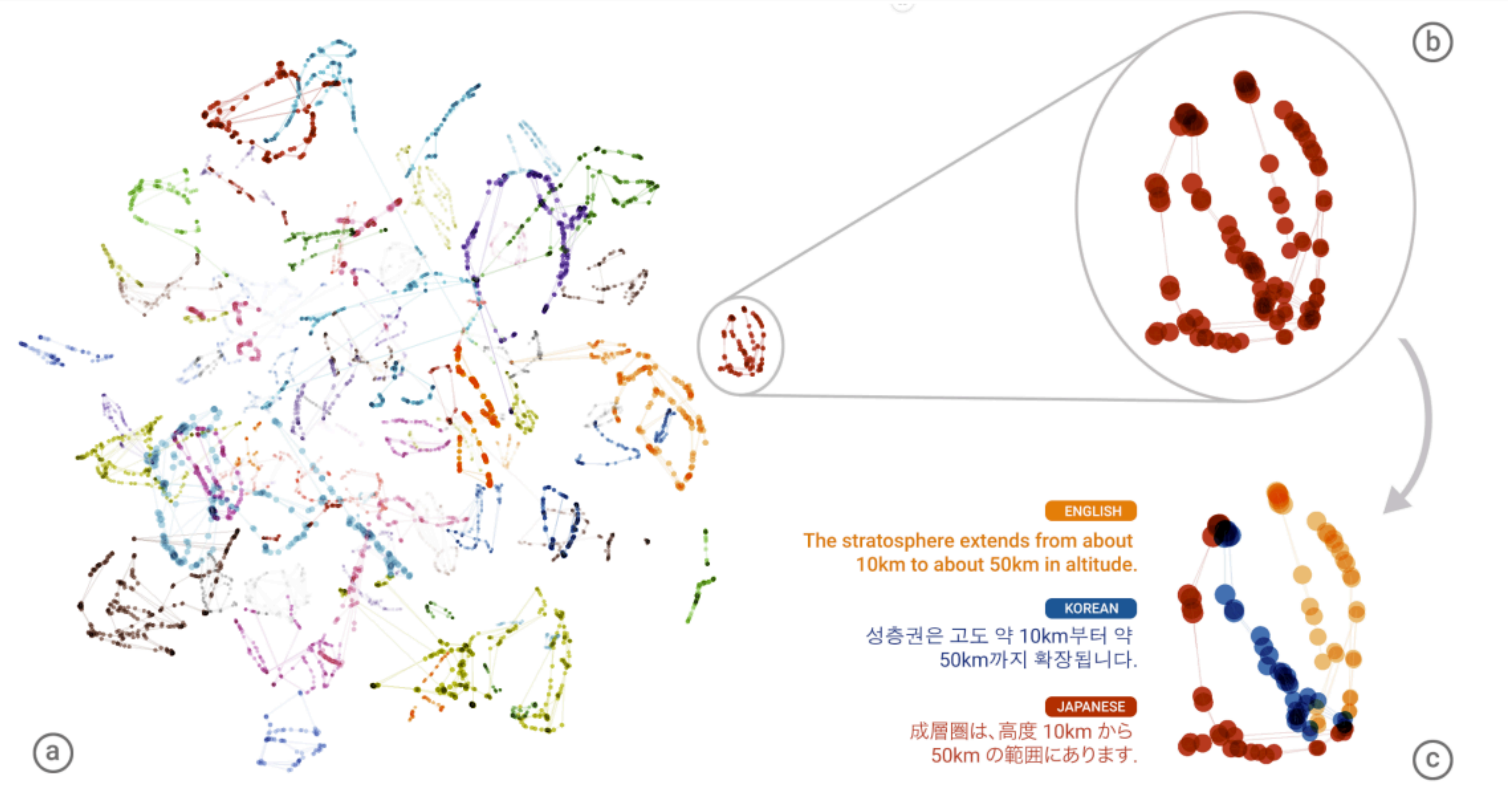


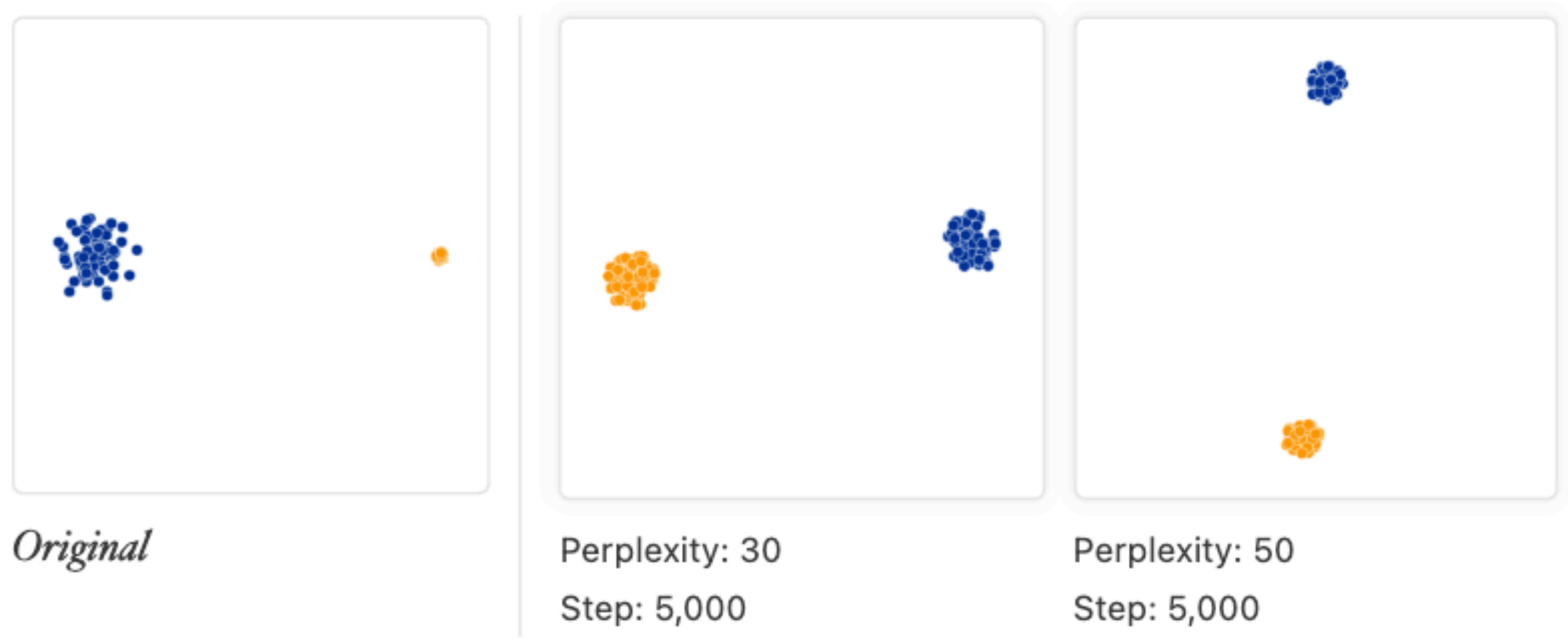
Figure 2: A t-SNE projection of the embedding of 74 semantically identical sentences translated across all 6 possible directions, yielding a total of 9,978 steps (dots in the image), from the model trained on English↔Japanese and English↔Korean examples. (a) A bird's-eye view of the embedding, coloring by the index of the semantic sentence. Well-defined clusters each having a single color are apparent. (b) A zoomed in view of one of the clusters with the same coloring. All of the sentences within this cluster are translations of "The stratosphere extends from about 10km to about 50km in altitude." (c) The same cluster colored by source language. All three source languages can be seen within this cluster.

t-SNE projection of latent space of language translation model [Johnson et al. 2018]

Johnson, Melvin, et al. "Google's multilingual neural machine translation system: Enabling zero-shot translation." *TACL 2017*

Visualization could be misleading

Cluster sizes in a t-SNE plot mean nothing



Distances between clusters might not mean anything

Non-linear projection (or really, any computation + visualization method) needs to be used with caution.

e.g., t-SNE adapts its notion of "distance" to regional density variations in the data set: it expands dense clusters, and contracts sparse ones, evening out cluster sizes. i.e., **Density equalization happens by design and is a predictable feature of t-SNE.**

As a result we cannot & should not judge relative sizes of clusters in a t-SNE plot.

!For more t-SNE pitfalls: https://distill.pub/2016/misread-tsne/

Projection (3/3): UMAP [McInnes et al. 2018]

Form weighted nearest neighbor graph, then layout the graph in a manner that balances embedding of local and global structure.

"Our algorithm is competitive with t-SNE for visualization quality and arguably preserves more of the global structure with superior run time performance." - McInnes et al. 2018

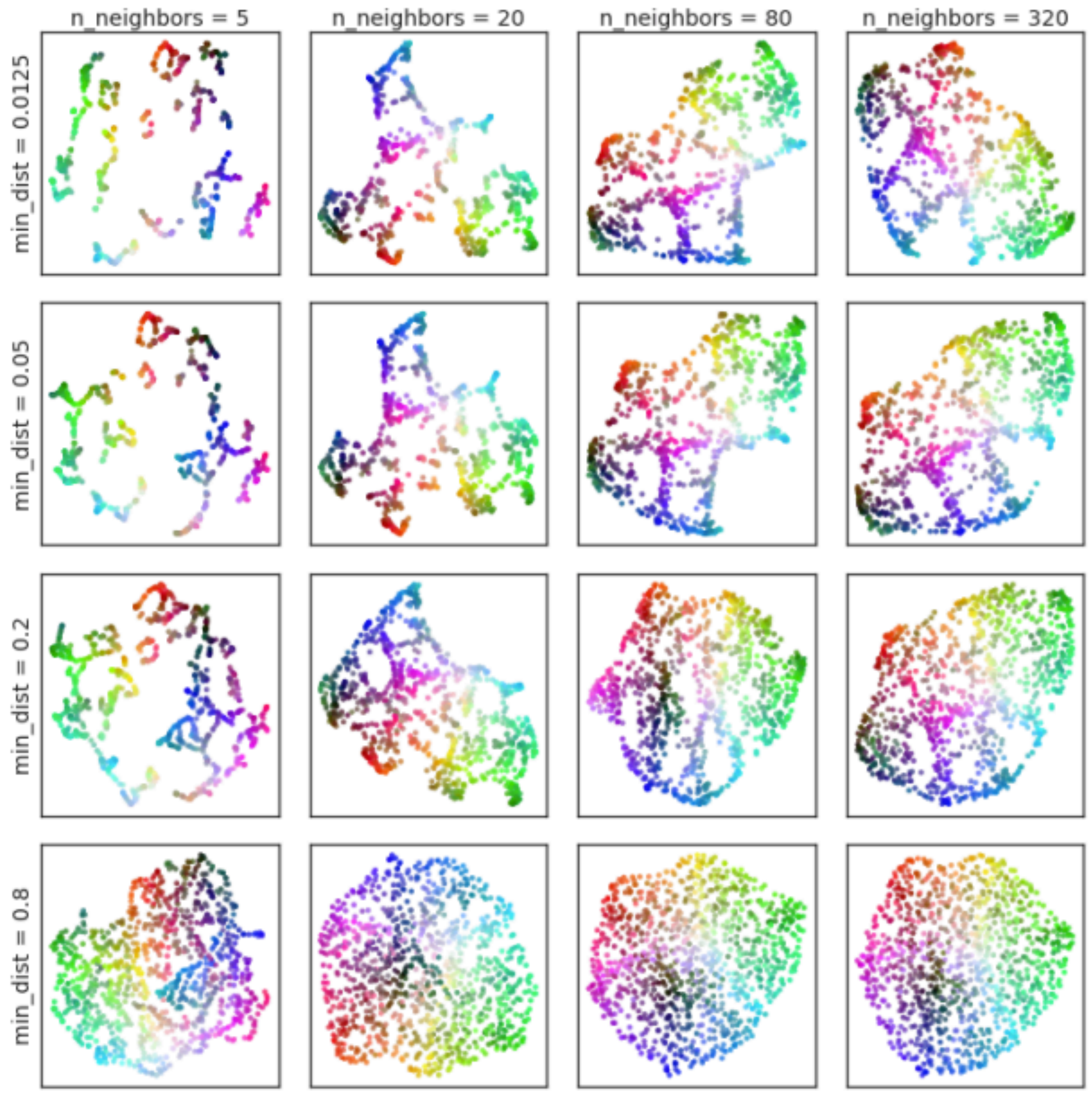
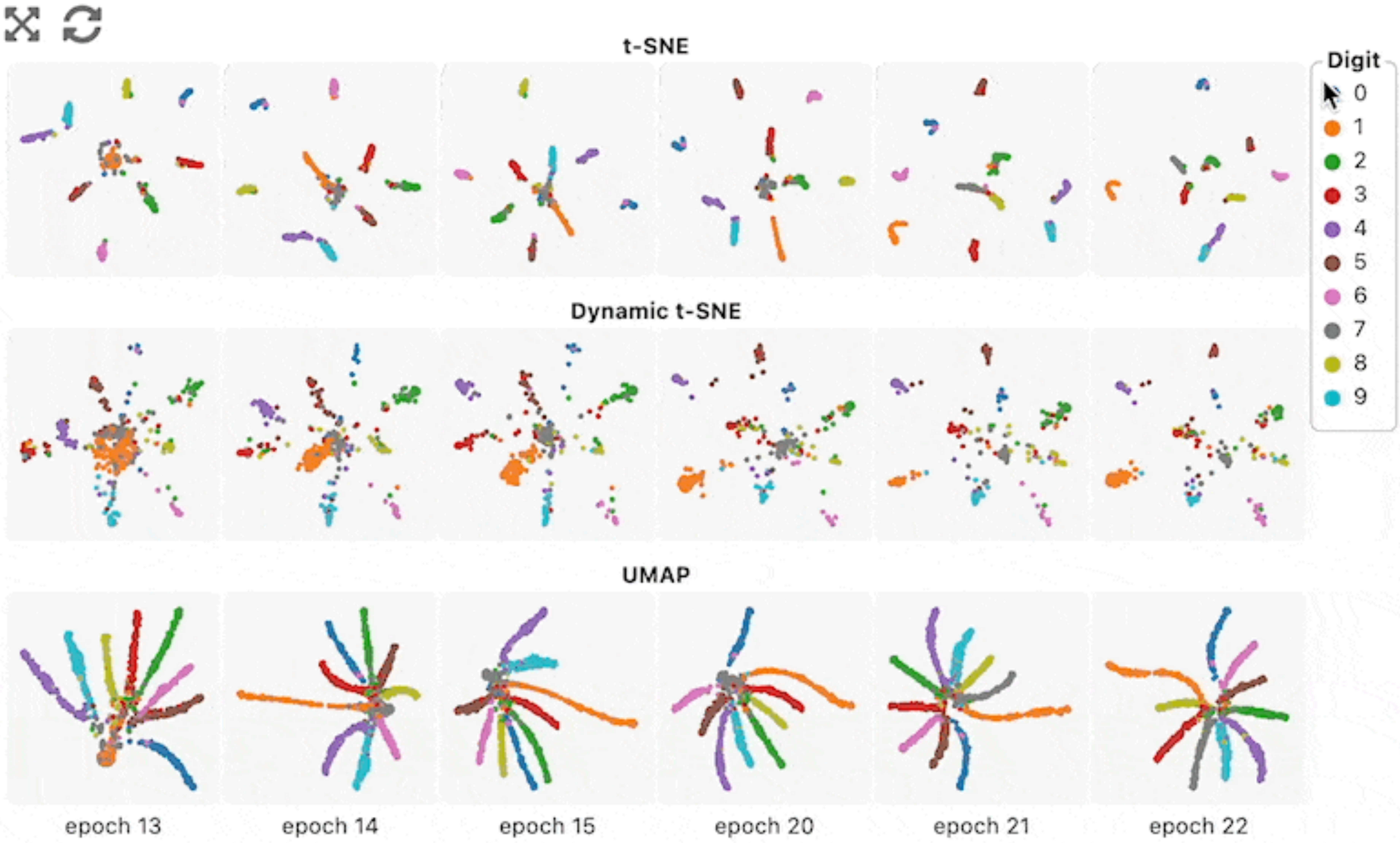


Figure 1: Variation of UMAP hyperparameters n and min-dist result in different embeddings. The data is uniform random samples from a 3-dimensional color-cube, allowing for easy visualization of the original 3-dimensional coordinates in the embedding space by using the corresponding RGB colour. Low values of n spuriously interpret structure from the random sampling noise – see Section 6 for further discussion of this phenomena.

McInnes, Leland, John Healy, and James Melville. "Umap: Uniform manifold approximation and projection for dimension reduction." *JOSS 2018*

A visual comparison between algorithms



Again, small multiples and linked views :)

Mingwei Li et al. "Visualizing Neural Networks with the Grand Tour" *2020*.

"Parameters" for a visualization

Goal *Why visualize*	**Content** *What to visualize*	**Encoding** *How to visualize*	**Context** *Assist communication*
Local understand	**Input distribution**	Line chart	**Annotations**
Global understand	**In-/out-put mapping**	Bar chart	Text integration
Communication	Activations	**Scatter plot**	**Aggregation**
Education	Attention	**Graph**	**Dimension reduction**
	Postdoc explanations	Saliency map	Small multiples
	Architecture		
	Parameter spaces		

Multi-view, interactive interfaces for understanding

We use multi-view interactions because…

Humans have a lot of inter-twined goals that cannot be embedded into a single view.

We have too much information to present at once.

Allow humans to inquiry targeted information.

Suggest information to mitigate human biases.

Best practices for these system designs

(Again) “Overview first, details on-demand”

Integrate into users’ natural developing environments

Explore intuitive interactions

Capabilities	**M**inimum **F**unctionality **T**est *failure rate % (over N tests)*	**INV**ariance Test *failure rate % (over N tests)*	**DIR**ectional Expectation Test *failure rate % (over N tests)*
+ Vocabulary	100.0% (5)	10.2% (1)	0.8% (4)
+ Robustness		11.4% (5)	
+ NER		7.6% (3)	
+ Fairness		96.4% (4)	
+ Temporal	18.8% (1)		100.0% (1)
+ Negation	99.8% (9)		
+ SRL	100.0% (5)		

Example: CheckList

Basic idea: A framework for testing models on nuanced capabilities.

An example of lightweight visual interface

"Overview first, details on-demand"

Different views can be invoked in Jupyter Notebook

Ribeiro, Marco Tulio, et al. "Beyond accuracy: Behavioral testing of NLP models with CheckList." *ACL 2020*

+	NER		7.6% (3)	
+	Fairness		96.4% (4)	
+	Temporal	18.8% (1)		100.0% (1)
-	Negation	99.8% (9)		

Minimum Functionality Test

	test name	failure rate
+	simple negations: negative	42 / 500 = 8.4%
+	simple negations: not negative	66 / 500 = 13.2%
+	simple negations: not neutral is still neutral	492 / 500 = 98.4%
+	simple negations: I thought x was positive, but it was not (should be negative)	11 / 500 = 2.2%
+	simple negations: I thought x was negative, but it was not (should be neutral or positive)	424 / 500 = 84.8%
+	simple negations: but it was not (neutral) should still be neutral	493 / 500 = 98.6%
+	Hard: Negation of positive with neutral stuff in the middle (should be negative)	370 / 500 = 74.0%
	Hard: Negation of negative with neutral stuff in the middle	[illegible]

+	simple negations: not negative	66 / 500 = 13.2%
+	simple negations: not neutral is still neutral	492 / 500 = 98.4%
+	simple negations: I thought x was positive, but it was not (should be negative)	11 / 500 = 2.2%
+	simple negations: I thought x was negative, but it was not (should be neutral or positive)	424 / 500 = 84.8%
+	simple negations: but it was not (neutral) should still be neutral	493 / 500 = 98.6%
-	Hard: Negation of positive with neutral stuff in the middle (should be negative)	370 / 500 = 74.0%

Test Summary

Test [MFT] on [NEGATION]

Hard: Negation of positive with neutral stuff in the middle (should be negative)

Result FAILURE RATE ON ALL CASES

370/500=74.0%

FILTER TEST CASES

Input free text and enter

Examples

Failed cases only

- I would n't say , given that I am from Brazil , that this food was extraordinary . — Expect: 0 | Pred: 2 (1.00)
- I would n't say , given it 's a Tuesday , that that is a beautiful aircraft . — Expect: 0 | Pred: 2 (1.00)
- I would n't say , given that I am from Brazil , that the service is wonderful . — Expect: 0 | Pred: 2 (1.00)
- I ca n't say , given all that I 've seen — Expect: 0 | Pred: 2 (1.00)

+	Hard: Negation of negative with neutral stuff in the middle	499 / 500 = 99.8%

+	simple negations: not negative	66 / 500 = 13.2%
+	simple negations: not neutral is still neutral	492 / 500 = 98.4%
+	simple negations: I thought x was positive, but it was not (should be negative)	11 / 500 = 2.2%
+	simple negations: I thought x was negative, but it was not (should be neutral or positive)	424 / 500 = 84.8%
+	simple negations: but it was not (neutral) should still be neutral	493 / 500 = 98.6%
-	Hard: Negation of positive with neutral stuff in the middle (should be negative)	370 / 500 = 74.0%

Test Summary

Test [MFT] on [NEGATION]

Hard: Negation of positive with neutral stuff in the middle (should be negative)

Result FAILURE RATE ON ALL CASES

370/500=74.0%

FILTER TEST CASES

Input free text and enter

Examples Failed cases only

I would n't say , given that I am from Brazil , that this food was extraordinary .	Expect: 0 | Pred: 2 (1.00)
I would n't say , given it 's a Tuesday , that that is a beautiful aircraft .	Expect: 0 | Pred: 2 (1.00)
I would n't say , given that I am from Brazil , that the service is wonderful .	Expect: 0 | Pred: 2 (1.00)
I ca n't say , given all that I 've seen	Expect: 0 | Pred: 2 (1.00)

+	Hard: Negation of negative with neutral stuff in the middle	499 / 500 = 99.8%

+	simple negations: not negative	66 / 500 = 13.2%
+	simple negations: not neutral is still neutral	492 / 500 = 98.4%
+	simple negations: I thought x was positive, but it was not (should be negative)	11 / 500 = 2.2%
+	simple negations: I thought x was negative, but it was not (should be neutral or positive)	424 / 500 = 84.8%
+	simple negations: but it was not (neutral) should still be neutral	493 / 500 = 98.6%
-	Hard: Negation of positive with neutral stuff in the middle (should be negative)	370 / 500 = 74.0%

Test Summary

Test [MFT] on [NEGATION]
Hard: Negation of positive with neutral stuff in the middle (should be negative)

Result FAILURE RATE ON ALL CASES
370/500=74.0%

FILTER TEST CASES
Input free text and enter

Examples

Failed cases only

- I would n't say , given that I am from Brazil , that this food was extraordinary . — Expect: 0 | Pred: 2 (1.00)
- I would n't say , given it 's a Tuesday , that that is a beautiful aircraft . — Expect: 0 | Pred: 2 (1.00)
- I would n't say , given that I am from Brazil , that the service is wonderful . — Expect: 0 | Pred: 2 (1.00)
- I ca n't say , given all that I 've seen — Expect: 0 | Pred: 2 (1.00)

+	Hard: Negation of negative with neutral stuff in the middle	499 / 500 = 99.8%

	Test	Failure rate
+	simple negations: not negative	66 / 500 = 13.2%
+	simple negations: not neutral is still neutral	492 / 500 = 98.4%
+	simple negations: I thought x was positive, but it was not (should be negative)	11 / 500 = 2.2%
+	simple negations: I thought x was negative, but it was not (should be neutral or positive)	424 / 500 = 84.8%
+	simple negations: but it was not (neutral) should still be neutral	493 / 500 = 98.6%
-	Hard: Negation of positive with neutral stuff in the middle (should be negative)	370 / 500 = 74.0%

Test Summary

Test [MFT] on [NEGATION]
Hard: Negation of positive with neutral stuff in the middle (should be negative)

Result FAILURE RATE ON ALL CASES
370/500=74.0%

FILTER TEST CASES
Input free text and enter

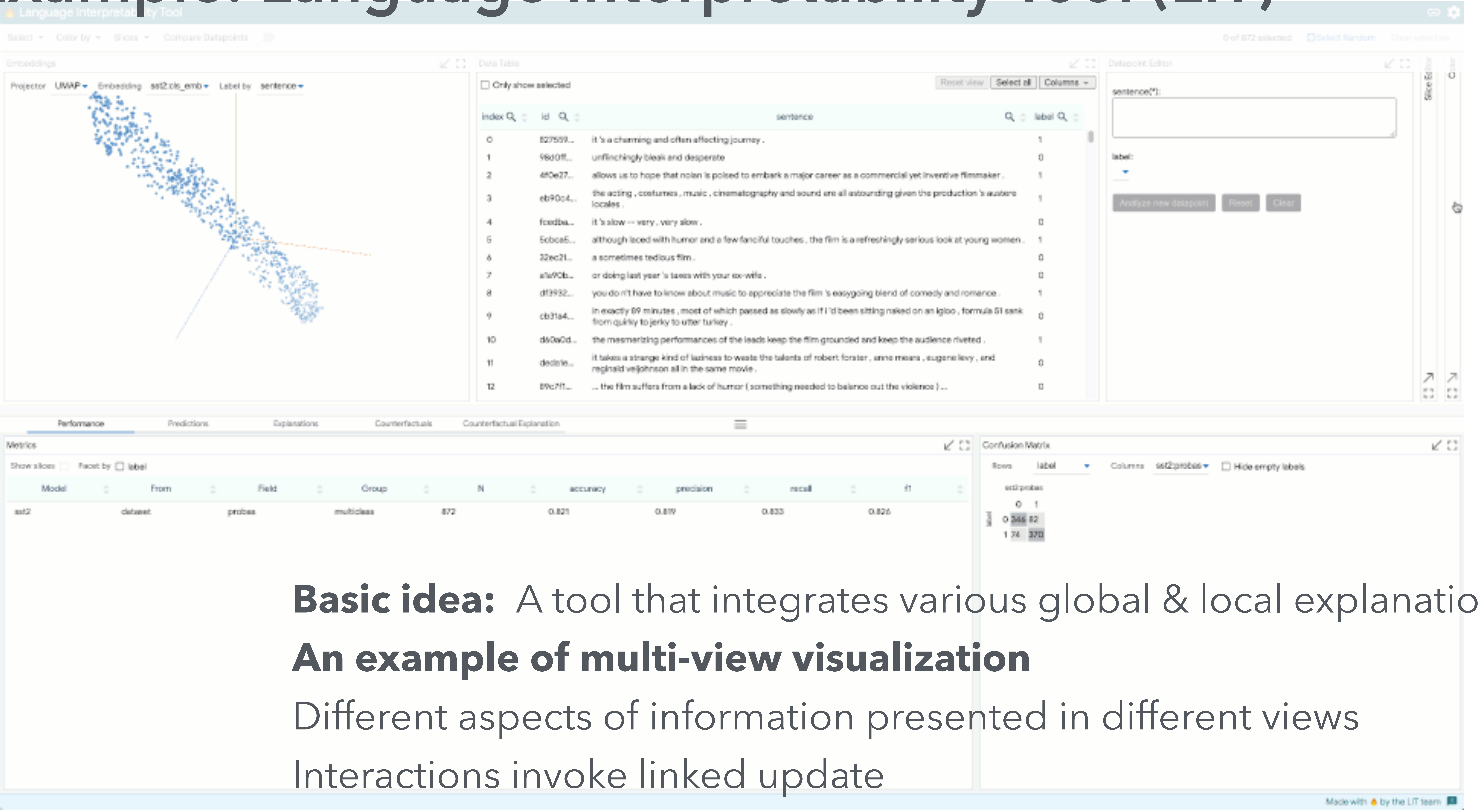
Examples

Failed cases only

- I would n't say , given that I am from Brazil , that this food was extraordinary . — Expect: 0 | Pred: 2 (1.00)
- I would n't say , given it 's a Tuesday , that that is a beautiful aircraft . — Expect: 0 | Pred: 2 (1.00)
- I would n't say , given that I am from Brazil , that the service is wonderful . — Expect: 0 | Pred: 2 (1.00)
- I ca n't say , given all that I 've seen — Expect: 0 | Pred: 2 (1.00)

+	Hard: Negation of negative with neutral stuff in the middle	

Example: Language Interpretability Tool (LIT)



Basic idea: A tool that integrates various global & local explanations.

An example of multi-view visualization

Different aspects of information presented in different views

Interactions invoke linked update

"Parameters" for a visualization

Goal
Why visualize

Local understand

Global understand

Communication

Education

Content
What to visualize

Input distribution

In-/out-put mapping

Activations

Attention

Postdoc explanations

Architecture

Parameter spaces

Encoding
How to visualize

Line chart

Bar chart

Scatter plot

Graph

Saliency map

Context
Assist communication

Annotations

Text integration

Aggregation

Dimension reduction

Small multiples

Visualization should be tied to **communication goal**

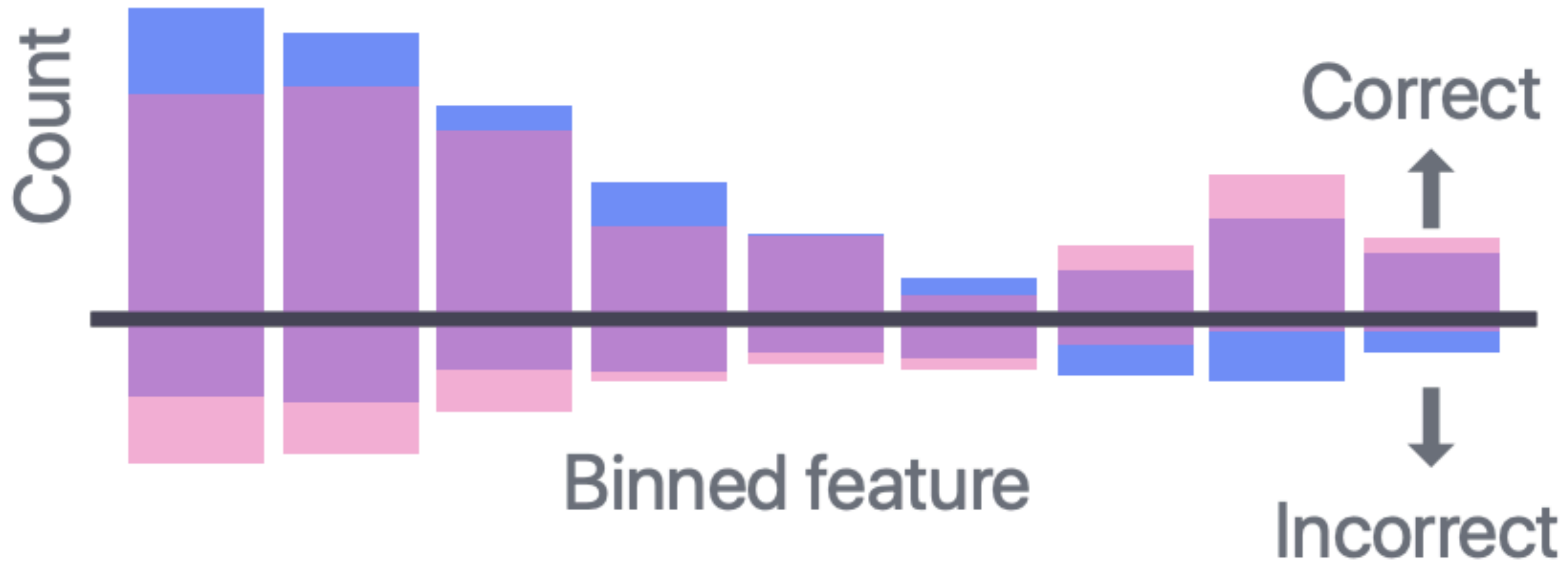


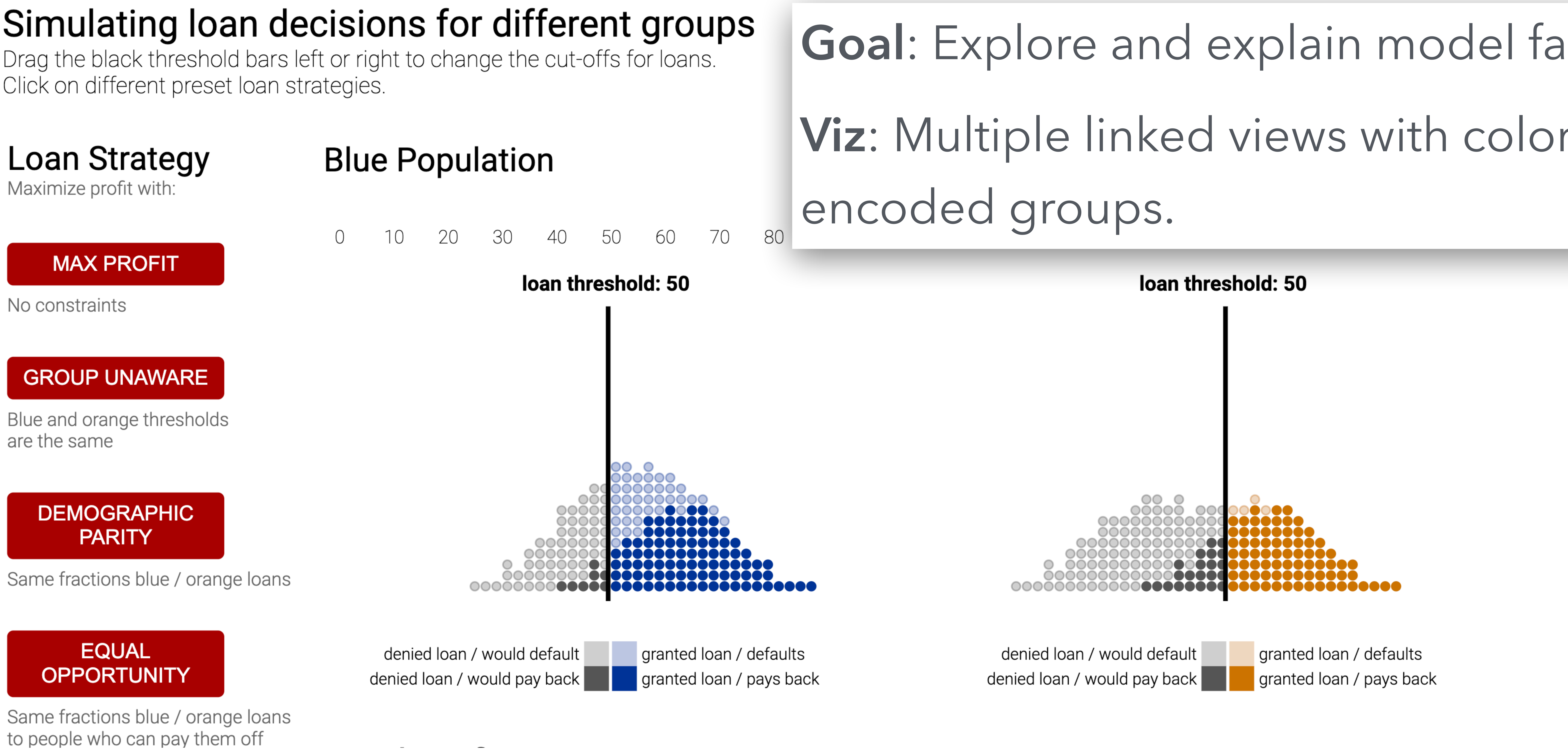
Figure 3. **The *Feature View* visualizes each feature with an *overlaid diverging histogram*. The binned feature units are placed on the x-axis and the count on the y-axis. Data instances within a bin that are correctly predicted are included above the x-axis, and instances that are incorrectly predicted are included below the x-axis. We overlay the primary and the secondary dataset versions for version comparison.**

Task: track data & model iterations.

Viz: bar chart, but data from two different iterations overlaid.

Hohman, Fred, et al. "Understanding and visualizing data iteration in machine learning." *CHI 2020*

Visualization should be tied to **communication goal**



Goal: Explore and explain model fairness.

Viz: Multiple linked views with color-encoded groups.

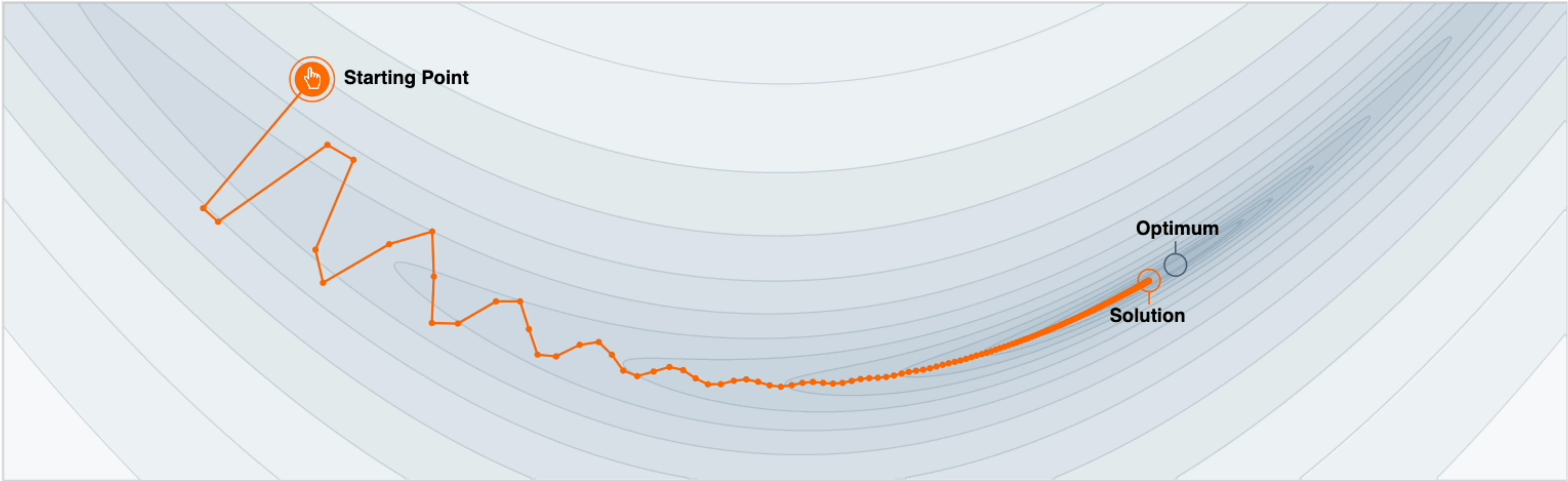
Attacking discrimination with smarter machine learning

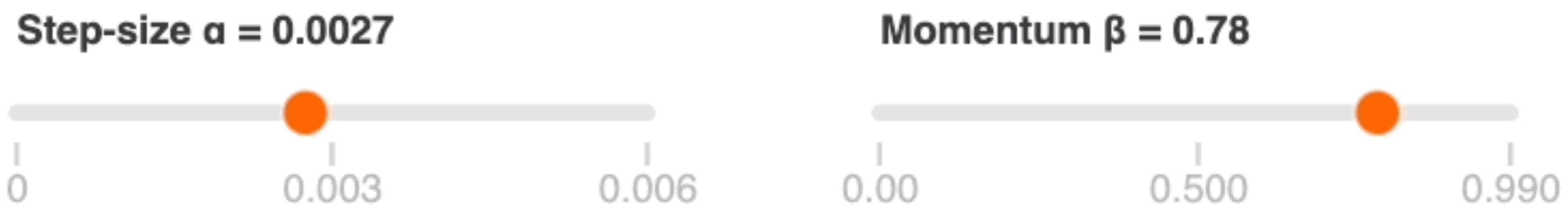
“Parameters” for a visualization

Goal	**Content**	**Encoding**	**Context**
Why visualize	*What to visualize*	*How to visualize*	*Assist communication*
Local understand	Input distribution	Line chart	Annotations
Global understand	In-/out-put mapping	Bar chart	Text integration
Communication	Activations	Scatter plot	Aggregation
Education	Attention	Graph	Dimension reduction
	Postdoc explanations	Saliency map	Small multiples
	Architecture		
	Parameter spaces		

Content: Parameter Space

Intuitively demonstrate the show the effects of certain parameters, via dynamic visualization.





We often think of Momentum as a means of dampening oscillations and speeding up the iterations, leading to faster convergence. But it has other interesting behavior. It allows a larger range of step-sizes to be used, and creates its own oscillations. What is going on?

Goh, Gabriel. "Why momentum really works." *Distill* 2.4 (2017): e6.

Content: Parameter Space

These visualizations are usually part of a larger tutorial / example set, and are closely integrated with the rest of the text.

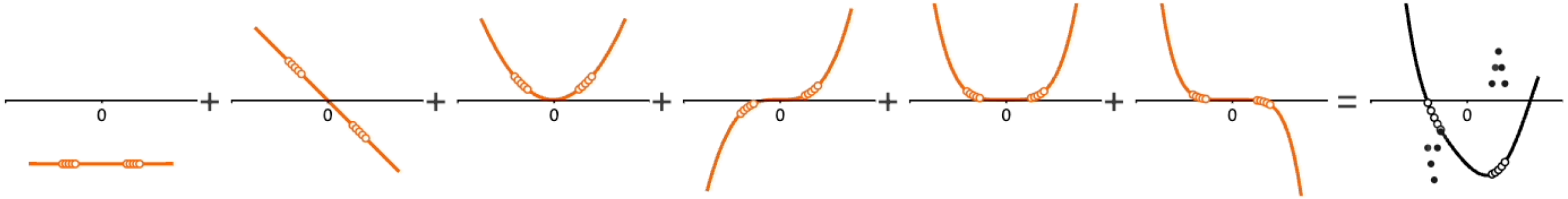
Lets see how this plays out in polynomial regression. Given 1D data, ξ_i, our problem is to fit the model

$$\text{model}(\xi) = w_1 p_1(\xi) + \cdots + w_n p_n(\xi) \qquad p_i = \xi \mapsto \xi^{i-1}$$

to our observations, d_i. This model, though nonlinear in the input ξ, is linear in the weights, and therefore we can write the model as a linear combination of monomials, like:

scrub values

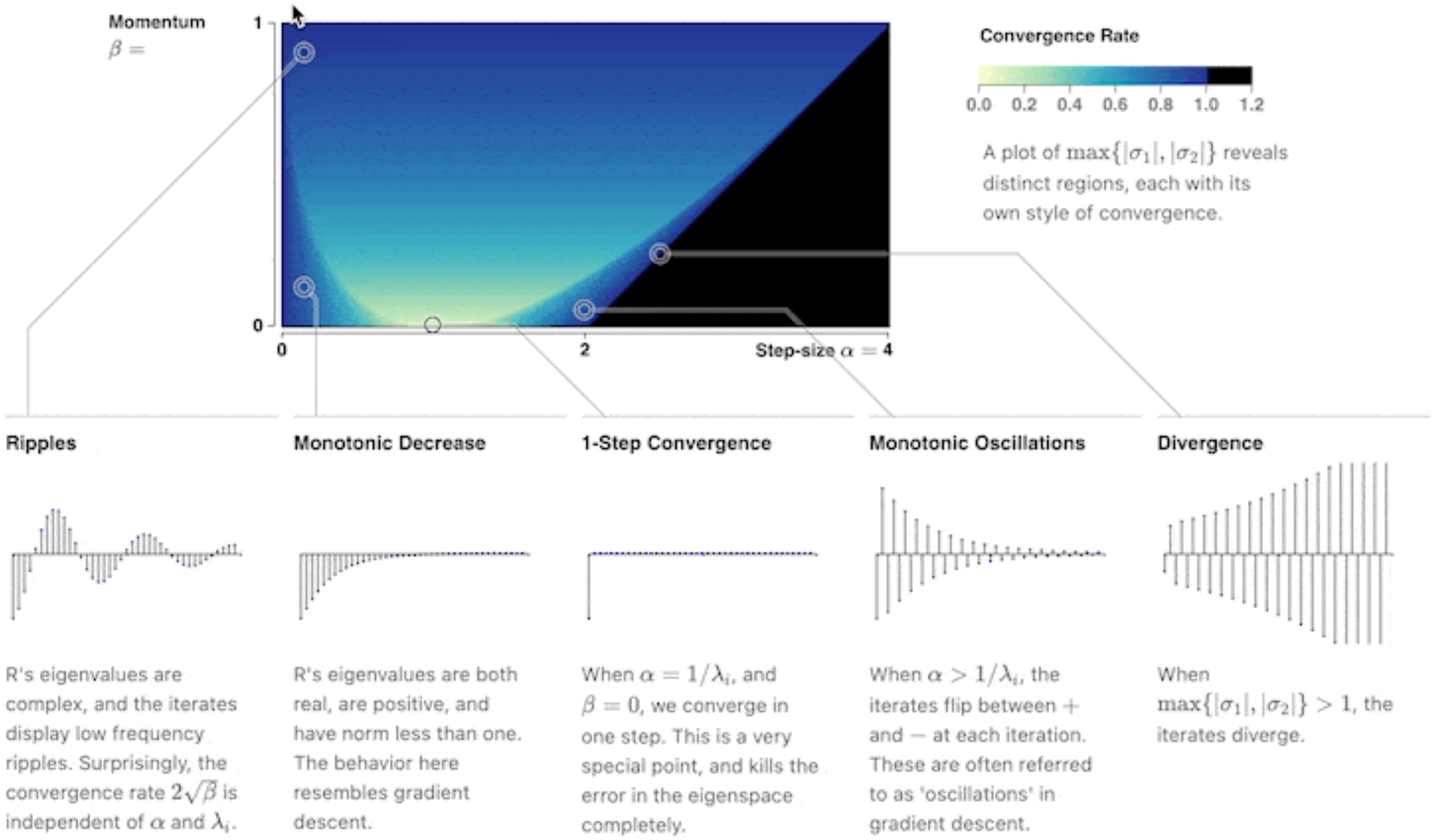
$$-2.00p_1 \quad + \quad -2.00p_2 \quad + \quad 2.00p_3 \quad + \quad 2.00p_4 \quad + \quad 2.00p_5 \quad + \quad -2.00p_6 \quad = \quad \text{model}$$



Content: Parameter Space

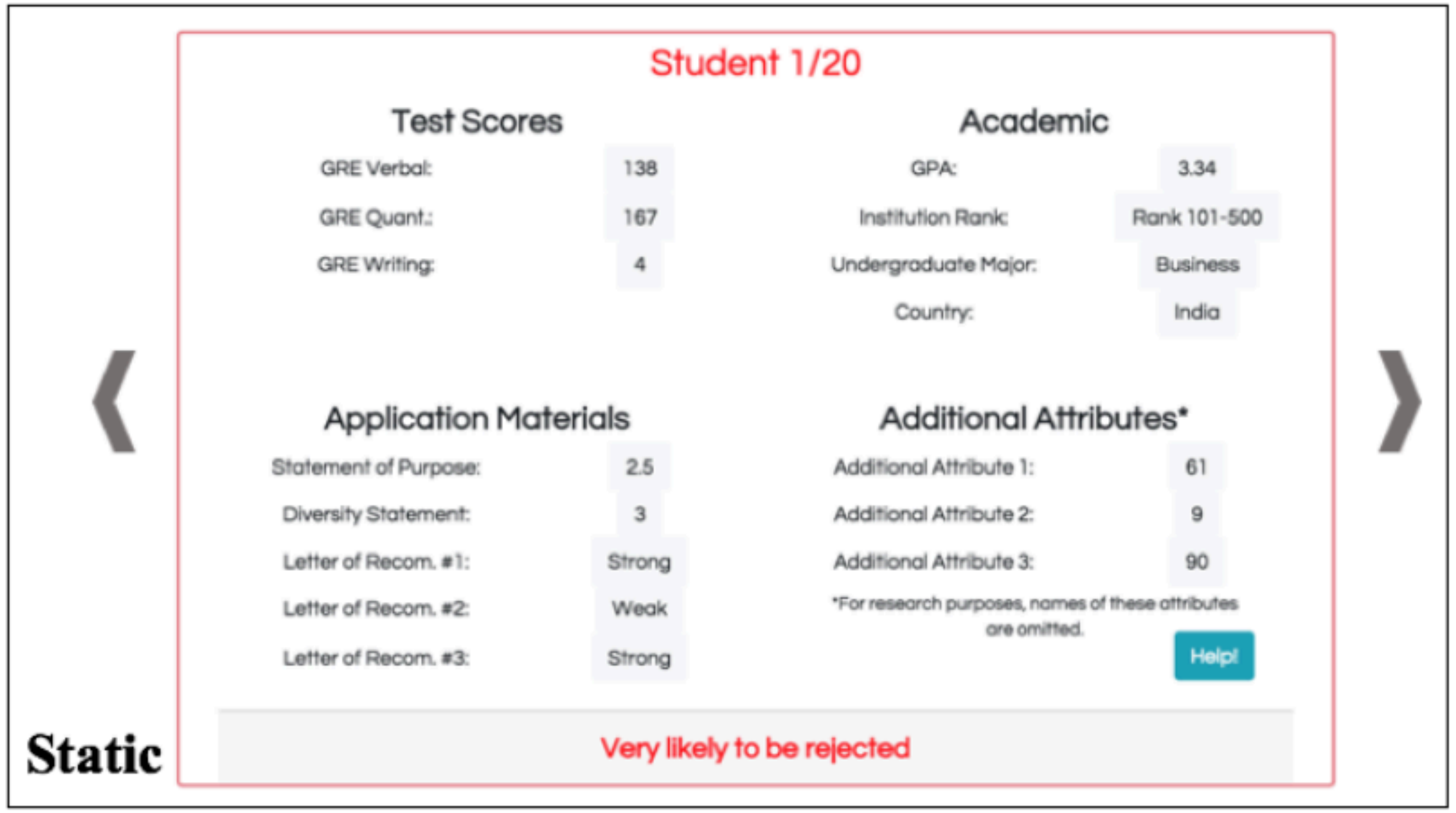
Show how parameters change together. We rarely are interested in a single parameter in isolation. When possible, allow exploration of the possibility space created by multiple parameters.

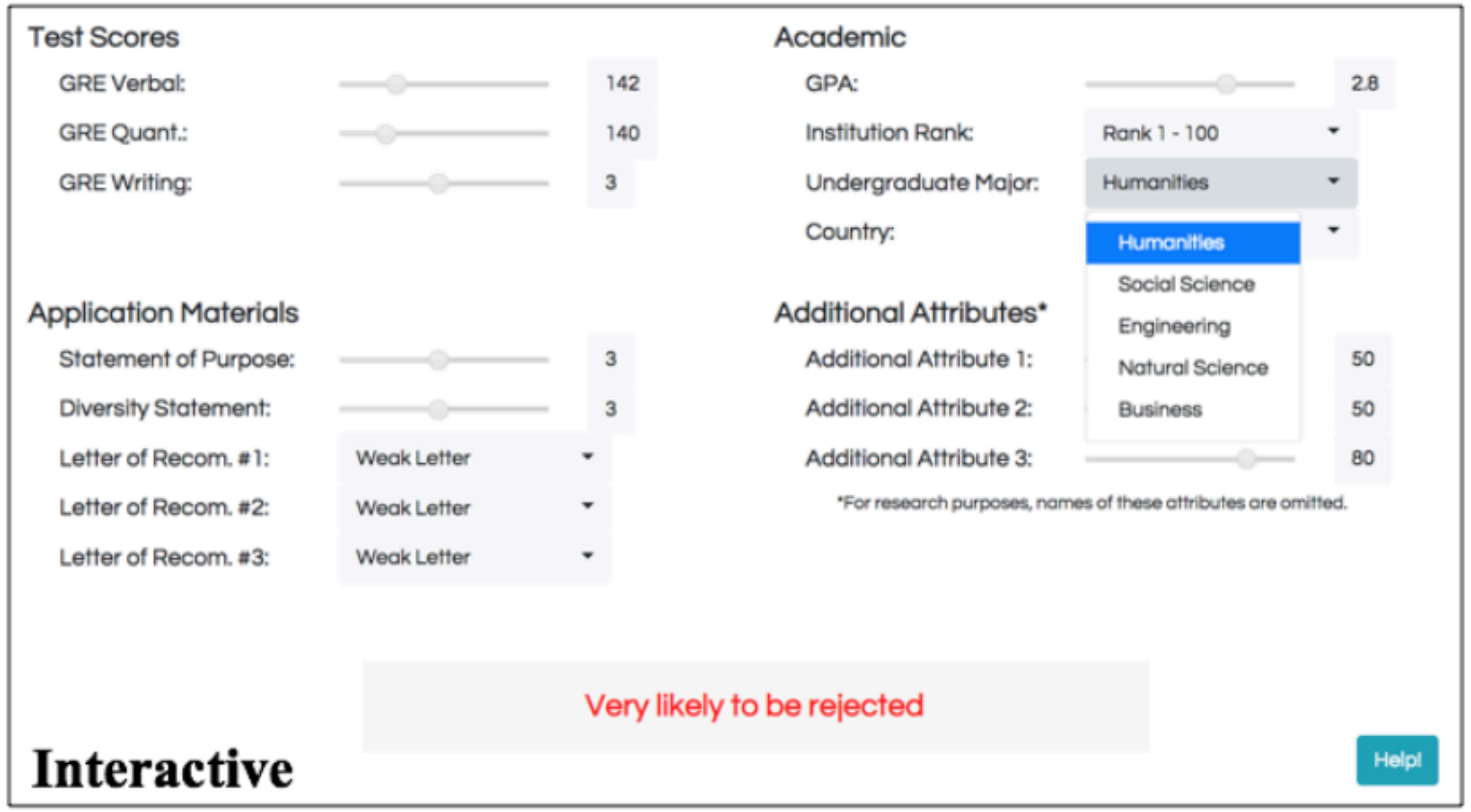
More elegantly: Map the space created by 2 parameters and link views to outputs. Annotate regions of parameter space



Why interaction improves comprehension?

"Although the interactive approach is more effective at improving comprehension, it comes with a trade-off of taking more time."



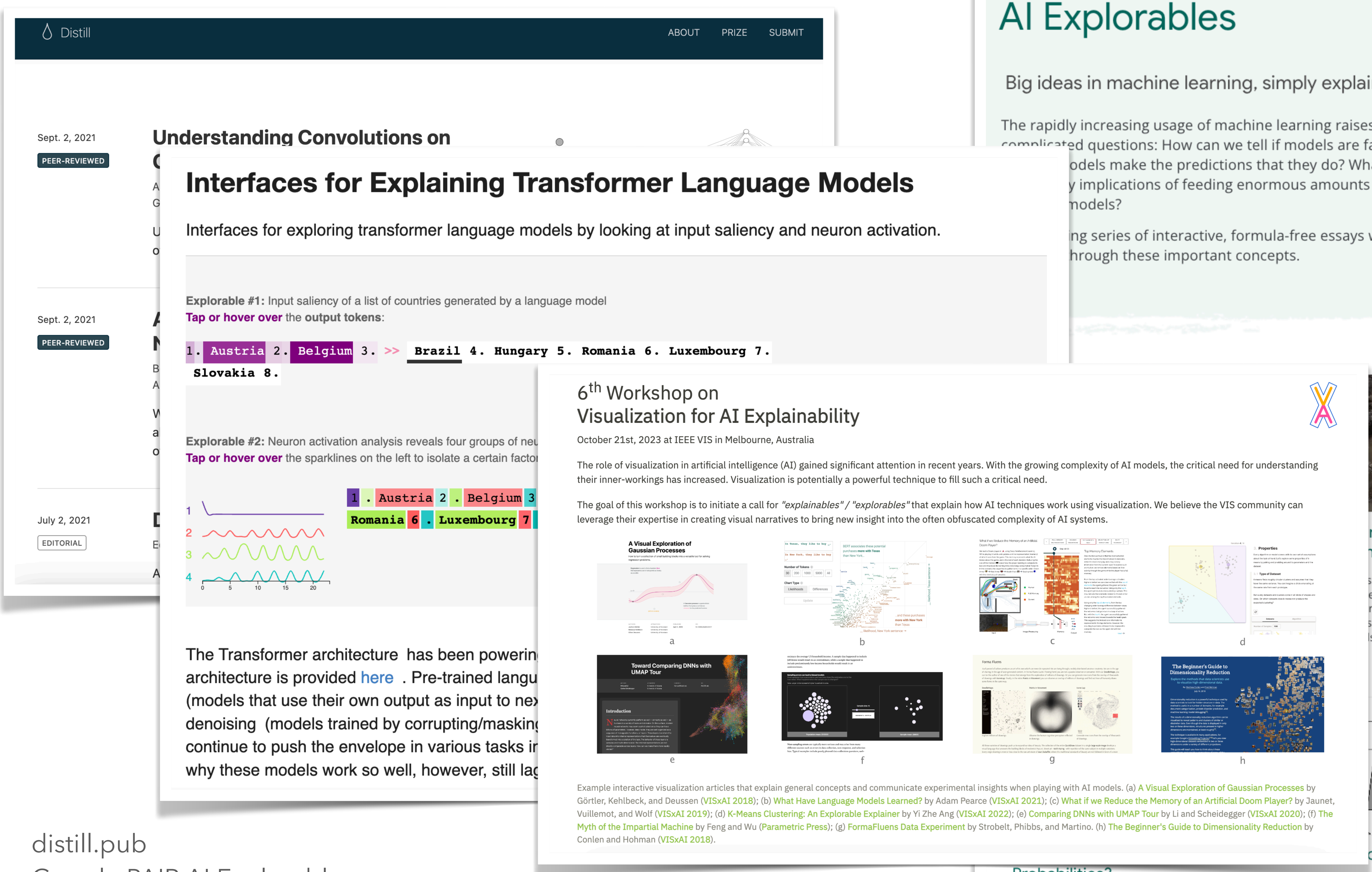


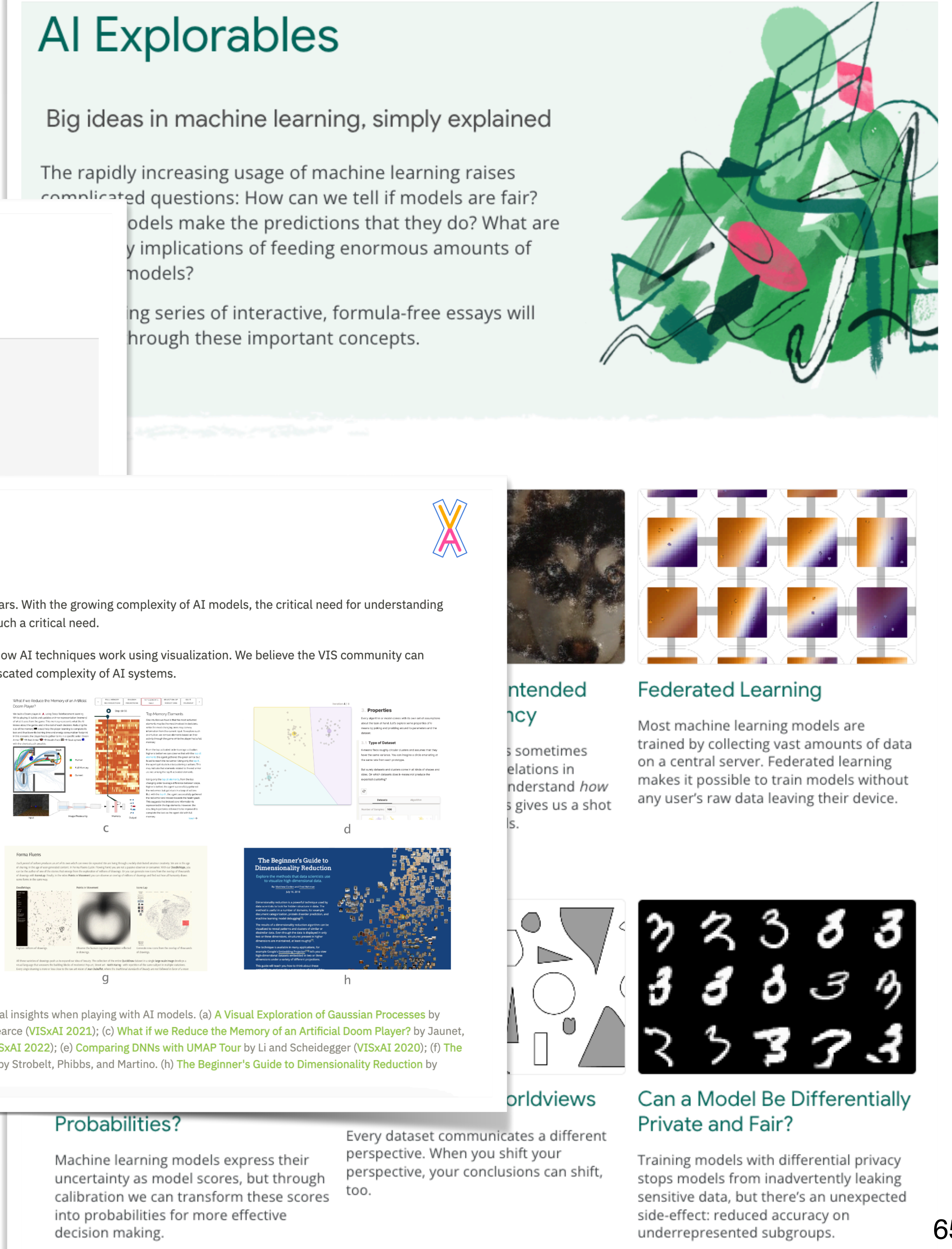
c. Interactiv

The Static interface (left) displays a selection of 20 unique application
interface (right) provides sliders to modify the values of attributes. The

Having the control to isolate/combine different variables is important.

Cheng, Hao-Fei, et al. "Explaining decision-making algorithms through UI: Strategies to help non-expert stakeholders." *CHI 2019*





distill.pub
Google PAIR AI Explorable
Jay Alammar
VISxAI

Some reflection on the parameters, and VIZ x NLP

Goal
Why visualize

Content
What to visualize

Encoding
How to visualize

Context
Assist communication

The most important thing of visualization is you want to achieve some **goal**, using certain **content**.

There has been 20+ years of study on effective visualization (e.g. line chart better for trend, must be for quantitative values; bar chart better for comparison). Usually once you know your goal, it's not too hard to find optimal **visualization encodings**.

Clear legend & textual annotation is essential.

Importantly, **content** can really be **any information** you can compute and obtain around your model – input, output, all sorts of scores. viz. is a shared topic across data collection/curation, model training and debugging, deployment, and knowledge sharing, and probably shouldn't be taken for granted :)

Practical Visualization Tools

Altair (in my opinion) the best visualization package with various encoding options.

Ecco a viz library for Language Model feature attribution and neuron activations.

Jupyter Widget + React Most typical way to build Notebook-embedded plug-ins.

CohereAI / Jay Alammar has (in my opinion) the most useful visualization for NLP beginners

Distill.pub and PAIR explorable has interactive articles you can play with.

Draco or VizLinter has some quick overview on visualization constraint 101.

Recap

Model visualization can happen at any stage in model development and deployment.

Visualization encoding changes based on what patterns we are trying to convey, based on what data.

Most common visualizations overlay information on top of dimensions we are familiar of (token-wise saliency map); Others reduce the uninterpretable dimension to some interpretable number (dimensionality reduction).

More holistic linked views give you more holistic understanding, but require more effort (to build, and to interact with).