cse 512 - Data Visualization Model Interpretability



Jeffrey Heer University of Washington

What do we mean by "interpretable"?

Varied notions of "interpretable"

Causal (why): the degree to which a person can understand the cause of a result.

Predictive (what): the degree to which a person can predict the model's result.

By whom? For what purpose?

Why "interpretable" models?

Why "interpretable" models? **Fairness**: assess for bias / discrimination **Privacy**: protect sensitive information **Reliability:** sensitivity to input changes **Causality**: explanatory, not just predictive **Trust:** make informed deployment choices

Approaches to Interpretability

- "Inherently" interpretable models (?)
- Decision trees, decision lists
- Linear models
- Generalized additive models (GAMs)

Approaches to Interpretability

- "Inherently" interpretable models (?)
- Decision trees, decision lists
- Linear models
- Generalized additive models (GAMs)
- Inspection / analysis of existing models
- Visualize model features, activations
- "Model-agnostic" analysis of behavior

Model Assessment

Transforming Data

How well does the curve fit the data?



Plot the Residuals

Plot vertical distance from best fit curve Residual graph shows goodness of fit



[Cleveland 85]

Plot the Residuals

Plot vertical distance from best fit curve Residual graph shows goodness of fit



[Cleveland 85]

Heteroscedasticity!

Multiple Plotting Options

Plot model in data space

Plot data in model space



[Cleveland 85]

Model Tracker [Amershi et al. 2015]



Assessing Fairness [Wattenberg et al. 2016]

Loan Strategy

Maximize profit with:

0 10 70 80 90 100 80 90 100 MAX PROFIT loan threshold: 50 loan threshold: 50 No constraints **GROUP UNAWARE** Blue and orange thresholds are the same DEMOGRAPHIC PARITY Same fractions blue / orange loans EQUAL OPPORTUNITY denied loan / would default denied loan / would default granted loan / defaults granted loan / defaults Same fractions blue / orange loans denied loan / would pay back granted loan / pays back to people who can pay them off denied loan / would pay back granted loan / pays back

Total profit = 19600

Correct 76%

loans granted to paying applicants and denied to defaulters

Blue Population



Incorrect 24% loans denied to paying applicants and granted to defaulters



Correct 87%

loans granted to paying applicants and denied to defaulters

Orange Population



Incorrect 13%

loans denied to paying applicants and granted to defaulters



Project nD data to 2D or 3D for viewing. Often used to interpret and sanity check high-dimensional representations fit by machine learning methods.

Project nD data to 2D or 3D for viewing. Often used to interpret and sanity check high-dimensional representations fit by machine learning methods.

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

Project nD data to 2D or 3D for viewing. Often used to interpret and sanity check high-dimensional representations fit by machine learning 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).

Reduction Techniques

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.

Principal Components Analysis



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

Principal Components Analysis



Linear transform: scale and rotate original space.

Lines (vectors) project to lines.

Preserves global distances.

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

How to Use t-SNE Effectively

Although extremely useful for visualizing high-dimensional data, t-SNE plots can sometimes be mysterious or misleading. By exploring how it behaves in simple cases, we can learn to use it more effectively.



Visualizing t-SNE [Wattenberg et al. '16]



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.

t-SNE [Maaten & Hinton 2008]

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.



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.
- 2. 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.
- 3. Optimize to find the positions in the embedding space that minimize the Kullback-Leibler divergence between the **P** and **Q** distributions: *KL(P* || *Q)*

Multiplicity [Stefaner 2018]



t-SNE projection of photos taken in Paris, France

MTEmbedding [Johnson et al. 2018]



t-SNE projection of latent space of language translation model.

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 colorcube, 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.

Reader Behavior [Conlen et al. 2019]



UMAP projection of reader activity for an interactive article.

Visualization of "Deep" Neural Network Models

TensorFlow Graph [Wongsuphasawat et al. 2018]



Fig. 1. The TensorFlow Graph Visualizer shows a convolutional network for classifying images (tf_cifar). (a) An overview displays a dataflow between groups of operations, with *auxiliary nodes* extracted to the side. (b) Expanding a group shows its nested structure.

ActiVis [Kahng et al. 2017]

A Model Architecture



1. Susan starts exploring the model overview. She selects a data node (yellow).



4. Inspecting instance #120's activations reveals it activates neurons in ways different from correctly classified ones (#38, #47) and from its class (NUM).

Clicking an instance in instance selection view adds it to neuron activation view **3.** Susan explores classification results for instances (questions). She wonders why question #120, asking about **num**eric values, is misclassified.

Fig. 1. ACTIVIS integrates several coordinated views to support exploration of complex deep neural network models, at both instanceand subset-level. **1.** Our user Susan starts exploring the model architecture, through its *computation graph* overview (at A). Selecting a *data node* (in yellow) displays its *neuron activations* (at B). **2.** The *neuron activation matrix view* shows the activations for instances and instance subsets; the *projected view* displays the 2-D projection of instance activations. **3.** From the *instance selection* panel (at C), she explores individual instances and their classification results. **4.** Adding instances to the matrix view enables comparison of activation patterns across instances, subsets, and classes, revealing causes for misclassification.

Seq2Seq-Vis [Strobelt et al. 2018]



Fig. 1. Example of Seq2Seq-Vis. In the translation view (left), the source sequence *"our tool helps to find errors in seq2seq models using visual analysis methods."* is translated into a German sentence. The word *"seq2seq"* has correct attention between encoder and decoder (red highlight) but is not part of the language dictionary. When investigating the encoder neighborhoods (right), the user sees that *"seq2seq"* is close to other unknown words *"(unk)"*. The buttons enable user interactions for deeper analysis.
Local Explanations (Model Specific)

Convolutional Neural Nets

CNNs for Image Processing



Prototypical CNN Architecture

GoogLeNet - 22 layers!



Feature Visualization [Olah et al. 2017]

Convolutional Neural Network (CNN) for Images

Basic Idea:

Select one or more "neurons" in a network layer Optimize to find input that maximizes excitation

Feature Visualization for CNNs



Starting from random noise, we optimize an

noise, we optimize an image to activate a particular neuron (layer mixed4a, unit 11).



Step 0



Step 4



Step 48



Step 2048

 \rightarrow

Single Unit Visualizations

Dataset Examples show us what neurons respond to in practice



Baseball—or stripes? *mixed4a, Unit 6*

Animal faces—or snouts? *mixed4a, Unit 240*

Clouds—or fluffiness? *mixed4a, Unit 453*

Buildings—or sky? *mixed4a, Unit 492*

Optimization isolates the causes of behavior from mere correlations. A neuron may not be detecting what you initially thought.

Single Unit Visualizations





Negative optimized



Minimum activation examples

Slightly negative activation examples



Slightly positive activation examples



Maximum activation examples



Positive optimized

Layer mixed 4a, unit 492

Optimizing for Diversity



Simple Optimization



Optimization with diversity reveals four different, curvy facets. *Layer mixed4a, Unit 97*



Simple Optimization



Optimization with diversity reveals multiple types of balls. Layer mixed5a, Unit 9

Multi-Unit Visualization

Dataset examples and optimized examples of **random directions** in activation space. The directions shown here were hand-picked for interpretability.

REPRODUCE IN A



mixed3a, random direction

mixed4c, random direction

mixed4d, random direction

mixed5a, random direction

Multi-Unit Visualization

By jointly optimizing two neurons we can get a sense of how they interact.

REPRODUCE IN A CO NOTEBOOK











Jointly optimized

Feature Visualization for CNNs

Convolutional Neural Network (CNN) for Images

Basic Idea:

Select one or more "neurons" in a network layer Optimize to find input that maximizes excitation

Challenges:

Choice of optimization? What dimensions to inspect? How to constrain or regularize? Unconstrained approach leads to model artifacts. Applicability to non-image data?

Local Explanations (Model Agnostic)

LIME [Ribeiro et al. 2016]

Local Interpretable Model-Agnostic Explanations

Model-agnostic: take any classifier as input

Model-agnostic: take any classifier as input

For a given prediction: Identify aspects meaningful to a person Perturb those aspects around the prediction (e.g., remove words or image regions) Fit local "interpretable" model to the results (e.g., locally-weighted linear model)

LIME Intuition



Despite complex global structure, a locally weighted linear model may suffice to explain.





Original Image

Interpretable Components

Why is this predicted to be a "tree frog"?



Original Image P(tree frog) = 0.54





Original Image P(tree frog) = 0.54



Regions sufficient for "frog" detection

Model-agnostic: take any classifier as input

For a given prediction: Identify aspects meaningful to a person Perturb those aspects around the prediction (e.g., remove words or image regions) Fit local "interpretable" model to the results (e.g., locally-weighted linear model)

For an entire model:

Optimize for a set of representative examples



(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)





(b) Explanation

Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

	Before	After
Trusted the bad model	10 out of 27	3 out of 27
Snow as a potential feature	12 out of 27	25 out of 27

Table 2: "Husky vs Wolf" experiment results.





(b) Explanation

Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

	Before	After
Trusted the bad model	10 out of 27	3 out of 27
Snow as a potential feature	12 out of 27	25 out of 27

Table 2: "Husky vs Wolf" experiment results.

Detects snow, not wolves!

Latent Space Cartography: Visual Analysis of Vector Space Embeddings

Yang Liu, Eunice Jun, Qisheng Li, Jeffrey Heer University of Washington Interactive Data Lab

https://github.com/uwdata/latent-space-cartography

graphy

Unlabeled Data

ML model

Vector Space Representation



Vector Space: Example

Word embeddings

represent a word as a vector of numbers

dragon





Vector Space: Example

Word embeddings

represent a word as a vector of numbers

dragon



Latent Spaces in Generative Models





Word embeddings

represent a word as a vector of numbers

dragon



Latent Spaces in Generative Models





Why are latent spaces important?



Serve as features for downstream ML applications





Why are latent spaces important?



2





How the meaning of words change over time

Serve as features for downstream ML applications



Biologically meaningful latent spaces





Human judgement is essential in interpreting latent spaces





Latent Space Cartography

Mapping meaningful dimensions of latent spaces









Built a visual analysis system, also named Latent Space



Latent Space Cartography

Visual Analysis of Vector Space Embeddings

Introduction Background and motivations

System Walkthrough

Workflow and system features via a scenario on emojis

Case Study

Two analysis scenarios for word embeddings

Conclusion

Our contributions and future work

Background: Variational Auto-encoder (VAE)



Background: Variational Auto-encoder (VAE)



Variational Auto-encoder (VAE)



Background: Variational Auto-encoder (VAE)

Variational Auto-encoder (VAE)






Background: Variational Auto-encoder (VAE)



Loss = Reconstruction loss + KL divergence



Latent Space in VAE

RTER S INIELSENIM - Using artific

CARTER S., NIELSEN M.: Using artificial intelligence to augment human intelligence. Distill (2017). https://distill.pub/2017/aia.

ABCDE... ABCDE...





Latent Space in VAE



CARTER S., NIELSEN M.: Using artificial intelligence to augment human intelligence. Distill (2017). https://distill.pub/2017/aia.

Attribute vector

 \geq



VAEs might help me understand emojis!









Latent spaces might help me understand emojis!





Latent spaces might help me understand emojis!

• Control 5 and Particle 1 and	Q (Searc	h Emojipedia		
Since See See See See See See See See See S	😃 Smileys & People 🛛 👪 Animals & N	ature 🗧 Food & Drink 📀 Ac	tivity 📓 Travel & Place	as 💡 Objects 🔢 Symbols 😷 Flags	
Employed for annipy, people, familes, hand gestures, clothing and accessories. Granning Face Granning Face Granning Face With Smiling Fyes Beaming Face With Smiling Fyes Granning Face With Smiling Fyes Beaming Face With Smiling Fyes Sing Face With Smiling Face With Applied Sing Face With Smiling Fyes Sing Face With Smiling	😃 Smileys & People			Categories	
Grinning Face Grinning Face WIN Binling Eyes Barming Face WIN Smilling Eyes Barming Face WIN Smilling Eyes Grinning Face WIN Smilling Eyes Grinning Face WIN Smilling Face Special Down Face Specia	Emojis for smileys, people, families, h	and gestures, clothing and acc	cessories.	Catogonoo	
Face With Teams of Jay Most Popular Sighthy Smilling Face Upside Down Face Upside Down Face Pace Heat Smilling Face With Heat System Face With Teams of Jay Smilling Face With Heat System Smilling Face With Heat System Start Should Smilling Face With Heat System	Grinning Face Grinning Face With Big Eyes Grinning Face With Smiling Eyes Beaming Face With Smiling Eyes Grinning Squinting Face Grinning Squinting Face Grinning Face With Sweat Folling on the Floor Laughing			Similey's A People Animals Nature Food & Drink Codd & Drink Achivity Travel & Places Objects Symbols Flags	
Mer (type) Output Shape Param Connected to wr (type) Output Shape Param Connected to wr (type) Output Shape Param Connected to wr (town 201) (None, 64, 64, 4) 0 Imput(0)[0] wr (town 201) (None, 52, 32, 64) 1886 conr_100[10] wr (Conv201) (None, 52, 32, 64) 18928 conr_200[10] wr (Conv201) (None, 52, 52, 64) 18928 conr_200[10] mr (Conv201) (None, 52) 22588 faitBill01 mr (Conv201) (None, 52) 22588 faitBill01	Face With Tears of Joy Silphthy Smiling Face Silphthy Smiling Face Wriking Face Smiling Face With Smiling Eyes Smiling Face With Heart Smiling Face With Heart-Eyes Smiling Face With Heart-Eyes Smiling Face With Heart-Eyes Smiling Face Sing Face Source Face Smiling Face Sm			Most Popular ♥ Red Heart ₩ Face With Tears of Joy ₩ Similing Face With Hearts ♥ Similing Face With Hearts ♥ Similing Face With Similing Eyes ♥ Tinking Face ♥ Heav(Heck Mark ♥ Pleading Face	
er (type) Output Shape Parcm # Connected to ut (InputLeyer) (None, 64, 64, 4) 6 w_1 (Conv2D) (None, 54, 64, 4) 6 w_2 (Conv2D) (None, 52, 52, 64) 1886 conv_1[0][0] w_2 (Conv2D) (None, 52, 52, 64) 36928 conv_2[0][0] w_4 (Conv2D) (None, 52, 52, 64) 36928 conv_2[0][0] m_4 (Conv2D) (None, 52, 52, 64) 36928 conv_2[0][0] t (Flatten) (None, 5558) 6 conv_2[0][0] dm (Conse) (None, 1824) 275898 flat[0][0]	o Sming Face With Haut-Eyea Star-Struck Face Blowing a Kiss Kissing Face Smilling Face			 Similay Face With Similay Eyes Therking Face Heavy Check Mark Pleading Face 	
er (type) Cotyput Shape Parks Connected to ut (Inputtayer) (None, 64, 64, 4) 0 v_1 (Conv2D) (None, 54, 64, 4) 68 input(0)[0] v_2 (Conv2D) (None, 32, 32, 64) 1088 conv_1[0][0] v_3 (Conv2D) (None, 32, 32, 64) 30728 conv_2[0][0] v_4 (Conv2D) (None, 32, 32, 64) 30728 conv_2[0][0] t (Flatten) (None, 52, 32, 64) 30728 conv_2[0][0] t (Shatten) (None, 52, 52, 54, 64) 30728 conv_2[0][0] t (Shatten) (S					
Nut (Inputlayer) (None, 64, 64, 4) 0 w_l (Conv2D) (None, 64, 64, 4) 68 input[0](0] w_l (Conv2D) (None, 32, 32, 64) 1086 conv_l[0](0] w_d (Conv2D) (None, 32, 32, 64) 36928 conv_c10](0] w_d (Conv2D) (None, 32, 32, 64) 36928 conv_c10][0] tr (Filter) (None, 52, 32, 64) 36928 conv_c10][0] tr (Filter) (None, 55580 0 conv_c10[0] dán (Conse) (None, 1024) 67169888 filt0] ean (Conse) (None, 22) 2556 hidden[0]	ver (type)	Output Shape	Param # Co	nnected to	
Ny 1 (Conv 2D) (None, 54, 64, 4) 68 input[0]0] Ny 2 (Conv 2D) (None, 32, 32, 64) 1086 conv_100[0] Ny 3 (Conv 2D) (None, 32, 32, 64) 36728 conv_200[0] Ny 4 (Conv 2D) (None, 32, 32, 64) 36728 conv_200[0] nx 4 (Conv 2D) (None, 32, 32, 64) 36720 conv_200[0] nx (Filten) (None, 56586) e conv_40[0] ddm (Donse) (None, 1824) 67120980 file[10] man (Conse) (None, 22) 22568 hidden[0]	put (InputLayer)	(None, 64, 64, 4)	8		
w_2 (Cenv2D) (None, 32, 52, 64) 1886 conv_1[0][0] w_3 (Cenv2D) (None, 32, 32, 64) 39728 conv_2[0][0] w_4 (Cenv2D) (None, 32, 52, 64) 39728 conv_2[0][0] tt (Filter) (None, 5558) e conv_4[0][0] dem (Dense) (None, 1624) 67569868 fle[16][0] ean (Dense) (None, 22) 2566 hidden[0][0]	w_1 (Conv2D)	(None, 64, 64, 4)	68 in	put(8)[8]	-
N_3 (Conv2D) (Mone, 32, 32, 54) 30728 conv_2(E(E)) v_4 (Conv2D) (Mone, 32, 32, 64) 30720 conv_2(E(E)) rt (Filten) (Mone, 32, 52, 64) 30720 conv_4(E(E)) dem (Donse) (Mone, 1624) 07160980 flat(E)(E) ean (Donse) (Mone, 1624) 25560 hidden(E(E))	iv_2 (Conv2D)	(None, 32, 32, 64)	1036 co	nv_1[0][0]	
n 4 (Conv2D) (Mone, 32, 32, 64) 30928 conv_3010[6] t (Flatten) (Mone, 56536) é conv_4(40[16]) Men (Dense) (Mone, 1824) 67169886 flat[6][6] (Mone, 22) 22568 hidden(40[6])	rv_3 (Conv2D)	(None, 32, 32, 64)	36928 co	nv_2[0][0]	
rt (Flatten) (Mone, 5553e) e conv_4(0)[0] Men (Dense) (None, 1024) 67109088 flat(0)[0] wan (Dense) (None, 22) 22550 hidsen(0)[0]	nv_4 (Conv2D)	(None, 32, 32, 64)	36928 co	nv_3[0][0]	
Hen (Dense) (None, 1824) 67169883 flat0 ean (Dense) (None, 22) 22550 hidden(0)(0)	rt (Flatten)	(None, 65536)	9 co	nv_4[0][0]	
rean (Dense) (None, 22) 22550 hidden[0](0]	iden (Dense)	(None, 1024)	67109883 F1	at[0][0]	
	nean (Dense)	(None, 22)	22550 hi	dden[8][8]	

conv_1 (Conv2D)	(None,	64, 64, 4)	68	input[0][0]
conv_2 (Conv2D)	(None,	32, 32, 64)	1036	conv_1[0][0]
conv_3 (Conv2D)	(None,	32, 32, 64)	36928	conv_2[0][0]
conv_4 (Conv2D)	(None,	32, 32, 64)	36928	conv_3[0][0]
flat (Flatten)	(None,	65536)	8	conv_4[0][0]
hidden (Dense)	(None,	1824)	67109888	flet[0][0]
z_mean (Dense)	(None,	22)	22550	hidden[8][8]
z_log_var (Dense)	(None,	22)	22558	hidden[0][0]
lambda_1 (Lambda)	(None,	22)	8	z_mean(0)(0) z_log_var[0][0]
decoder_hid (Dense)	(None,	1024)	23552	lambde_1[0][0]
decoder_upsample (Dense)	(None,	65536)	67174488	decoder_hid[0][0]
decoder_reshape (Reshape)	(None,	32, 32, 64)	9	dccoder_upsample[0][0]
decoder_deconv_1 (Conv2DTranspo	(None,	32, 32, 64)	36928	decoder_reshape[0][0]
decoder_deconv_2 (Conv2DTranspo	(None,	32, 32, 64)	36928	decoder_deconv_1(0)(0)
decoder_deconv_3_upsamp (Conv2D	(None,	65, 65, 64)	36928	decoder_deconv_2[0][0]
decoder_mean_squash (Conv2D)	(None,	64, 64, 4)	1028	decoder_deconv_3_upsamp[0][0]
Total parans: 134,539,764 Trainable parans: 134,539,764				

Crawl ~24,000 emojis

Train 6 variational auto-encoders (VAEs) with varying latent dimensions

Load latent spaces into LSC!







I would like to gain initial familiarity with the latent space ...





Examining an overview distribution













PCA













Android version 9 adopts a distinct style compared to its earlier versions









Does the latent space capture this trend?















Does the latent space capture this trend?













Q

Defining an attribute vector

Interactively group samples







Defining an attribute vector









How do emojis fall along the attribute vector spectrum?





Viewing data distribution relative to the attribute vector



X-axis: attribute vector direction





Viewing data distribution relative to the attribute vector



Viewing data distribution relative to the attribute vector







What emojis are considered by the model to be more similar to Android 9 than Android 7?









a salient relationship?

Assess interpolation	Asse salier	ess ncy
	Δεερεε	Compare
nip	analogies	relationships

Does the attribute vector reliably represent





overview



a salient relationship?



	Assess interpolation		Assess saliency	
Examine relationship		Assess analogies		Compare relationships

Does the attribute vector reliably represent







overview



a salient relationship?



int	Assess terpolation		Assess salienc	5 X Y
Examine relationship	a	Assess nalogies		Compare relationships

Does the attribute vector reliably represent





overview



Does the attribute vector reliably represent a salient relationship?



	Assess interpolation		Assess saliency	
Examine relationship		Assess analogies		Compare relationships





















OTHER VECTORS

-0.02 Microsoft Smileys - Twitter Smileys



Pair alignment in the original space



$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|}$$











Problem: as dimensionality increases, random vectors are more likely to be orthogonal!









Cosine similarity divided by pooled standard deviation

$$Sp = \sqrt{\frac{(n_1 - 1)}{(n_1 - 1)}}$$

 $(1)s_1^2 + (n_2 - 1)s_2^2$ $(n_1 + n_2 - 2)$





Might our attribute vector transform an arbitrary emoji into Android 7 style?





Performing attribute vector arithmetic







Performing attribute vector arithmetic







Performing attribute vector arithmetic

0

° ⊂, ▼

Example: man - woman













(... investigated more attribute vectors ...)







How do multiple attribute vectors relate?





Examining how multiple attribute vectors relate

Visualizing attribute vectors in a global view





UMAP

PCA

Attribute vector projection












t-SNE



UMAP

Leg Down Leg Up Samsung Samsung Dark Skil Orange Food Green Food Clocks)without:Tick ndroid 9:0:Face Twitter Smileys Microsoft Smileys DS 4.0 Faces FB clocks Cry group Attribute vector projection













t-SNE











t-SNE





Step 2: map control points to 2D

Step 3: render a Catmull-Rom spline







t-SNE

Step 1: sample at regular intervals



Step 2: map control points to 2D

- Find k nearest neighbors
- Map neighbors to 2D
- Compute a weighted average

Step 3: render a Catmull-Rom spline













Orthogonal vectors represent independent dimensions ...

Semantic axes to re-orient the latent space?

* - - >



COSINE LABEL 0.17 Multiple People - Single Person

-0.19 Woman - Man

OTHER VECTORS (

- 0.09 Apple Skin Yellow Apple Skin Light
- -0.02 Microsoft Smileys Twitter Smileys
- -0.20 Leg Up Leg Down
- -0.03 Green Food Orange Food
- -0.16 Laugh group Cry group
- 1.00 Android 4-7 [new] Android 9 [new]
- 0.01 Microsoft empty circle Twitter circle
- 0.11 Twitter clock Twitter circle



Visual Analysis of Vector Space Embeddings

Introduction Background and motivations

System Walkthrough

Workflow and system features via a scenario on emojis

Case Study

Two analysis scenarios for word embeddings

Conclusion

Our contributions and future work



Gender Biases in Word Embeddings ... with a few simple interactions



Gender Biases in Word Embeddings



BOLUKBASI T., CHANG K.-W., ZOU J. Y., SALIGRAMA V., KALAI A. T.: Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In Advances in Neural Information Processing Systems (2016), pp. 4349-4357

Bolukubasi et al. quantify which words are closer to he versus she in word embedding to reveal gender stereotypes

We'll quickly replicate the findings in LSC





Groups		Vector
←	fam	nily
Start: male		End: female
15 total		15 total

Select a point (click, or search) to apply this attribute vector.



OTHER VECTORS Ø

COSINE LABEL

-0.02	country - capital
0.07	participle - present
-0.19	nationality adjective - nationality
-0.19	past tense - participle 2
0.01	plural - singular
1.00	female - male
0.12	state - city
0.13	adverb - adjective
0.07	negative - positive
0.17	comparative - adjective 2
-0.01	superlative - adjective 3











Brushed

season

director

player

victory

mark

joined

ball

mayor

w.

owner

brothers

bowl

founder

rugby

gov.

•

hero

singh

longtime

regiment

jacques

Q

businessman

critic



÷

÷

÷

÷

+

t

+

+

t

+

÷

+

+



-





Groups	Vectors
← far	nily
Start: male	End: female
15 total	15 total
apply this at	tribute vector.
apply this at	or search, to tribute vector.
PAIRS WITHIN THE VECT	OR CAVerage: 0.65
PAIRS WITHIN THE VECT -0.8 -0.6 -0.4 -0.2 0 Pairwise Cos	OR C Average: 0.65 0.0 0.2 0.4 0.6 0.8 sine Similarity
apply this att PAIRS WITHIN THE VECT -0.8 -0.6 -0.4 -0.2 0 Pairwise Cos OTHER VECTORS <i>\$</i>	Tribute vector.



COSINE	LABEL
-0.02 0.07 -0.19 -0.19 0.01 1.00	country - capital participle - present nationality adjective - nationali past tense - participle 2 plural - singular female - male





Analysis of Analogy Benchmark



Google's Analogy Benchmark

Family

• • •

king:queen son:daughter uncle:aunt

MIKOLOV T., YIH W.-T., ZWEIG G.: Linguistic regularities in continuous space word representations. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2013), pp. 746-751

• • •

Comparative

- bad:worse
- bright:brighter high:higher





Google's Analogy Benchmark

Family

• • •

king:queen son:daughter uncle:aunt

MIKOLOV T., YIH W.-T., ZWEIG G.: Linguistic regularities in continuous space word representations. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2013), pp. 746-751



Google's Analogy Benchmark

Family

• • •

king:queen son:daughter uncle:aunt

We use words in these analogy groups to define attribute vectors in LSC

MIKOLOV T., YIH W.-T., ZWEIG G.: Linguistic regularities in continuous space word representations. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2013), pp. 746-751

son:daughter king:queen

v = vec(king) - vec(son) + vec(queen)

Is v the nearest neighbor of vec(daughter)?





Dimension = 50

Google's analogy test score: 52.8% (317/600)

Google's analogy test score: 78.0% (468/600)





Dimension = 100

Dimension = 300

Google's analogy test score: 78.7% (472/600)



Dimension = 50

Google's analogy test score: 52.8% (317/600)

Dimension = 100

Google's analogy test score: 78.0% (468/600)





Dimension = 300

Google's analogy test score: 78.7% (472/600)

71



Google's analogy test score: 52.8% (317/600)

Dimension = 100

Google's analogy test score: 78.0% (468/600)





Saliency assessments agree with analogy test scores

Google's analogy test score:



Dimension = 300

78.7% (472/600)





Case Study: Cancer Transcriptomes



Case Study: Cancer Transcriptomes

Biological latent space





Disagree with prior work!

Their list Agreement

G I agree that vector subtraction makes the most

Proliferative	38 -	16%	
Differentiated	38 +	3%	
Differentiated	79 -	0%	

WAY G. P., GREENE C. S.: Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders. In Proceedings of Pacific Symposium on Biocomputing (2018), vol. 23, pp. 80-91.



Visual Analysis of Vector Space Embeddings

Introduction Background and motivations

System Walkthrough

Workflow and system features via a scenario on emojis

Case Study

Two analysis scenarios for word embeddings

Conclusion

Our contributions and future work





Mapping meaningful dimensions of latent spaces

Compare relationships A workflow of interpretation tasks



Mapping meaningful dimensions of latent spaces



A workflow of interpretation tasks

A visual analysis system for supporting this workflow



Mapping meaningful dimensions of latent spaces



X-axis: attribute vector direction

A workflow of interpretation tasks



A **visual analysis system** for supporting this workflow

- Linear projection to provide context



Mapping meaningful dimensions of latent spaces





A workflow of interpretation tasks

A **visual analysis system** for supporting this workflow

- Linear projection to provide context

- Methods to assess vector saliency



Mapping meaningful dimensions of latent spaces



A workflow of interpretation tasks

A **visual analysis system** for supporting this workflow

- Linear projection to provide context
- Methods to assess vector saliency
- Methods to compare multiple vectors







- Mapping meaningful dimensions of latent spaces
 - A workflow of interpretation tasks

A visual analysis system for supporting this workflow

- **Case studies** across multiple domains
 - Scientific findings on cancer gene expression

weddin	g	
pink		
mom		
nurse		
bedroo	m	

Gender stereotypes in word embeddings



81



Mapping meaningful dimensions of latent spaces



Available at: https://github.com/uwdata/latent-space-cartography

A workflow of interpretation tasks

A visual analysis system for supporting this workflow

Case studies across multiple domains





Latent Space Cartography: Visual Analysis of Vector Space Embeddings

Yang Liu, Eunice Jun, Qisheng Li, Jeffrey Heer University of Washington Interactive Data Lab

https://github.com/uwdata/latent-space-cartography

graphy

