EXPLORING THE LIME ALGORITHM FOR INTERPRETABLE DEEP LEARNING ON TEXT AND IMAGES

Anna Katharina Granberg Mortensen, Filip Sawicki, Javier García Ciudad, Thomas Theis Petersen

Index

1. Introduction
2. Algorithm
3. Simple explainer
4. Samples and distance
5. Neighborhood generation
6. Discussion and Conclusion

Introduction

LIME (Local interpretable model-agnostic explanations)

Trust and Transparency

Hyperparameters

Image and Text data

Evaluation Metrics

Coefficient of variation (CoV)

- Mean of the standard deviations divided by the mean of all the coefficients

Intersection over union (IoU)

- Intersection of two feature sets divided by the union

Visual inspection

Introduction

LIME (Local Interpretable Model-agnostic Explanations)

Image and Text data

Hyperparameters

Evaluation

- Coefficient of variation (CoV) $\bar{\sigma}/\overline{\square}$
- Intersection over union (IoU)
- Visual inspection

Introduction

LIME (Local Interpretable Model-agnostic Explanations)

Image and Text data

Hyperparameters

Performance and robustness

Evaluation

- Coefficient of variation (CoV)
- Intersection over union (IoU)
- Visual inspection

The Lime Algorithm

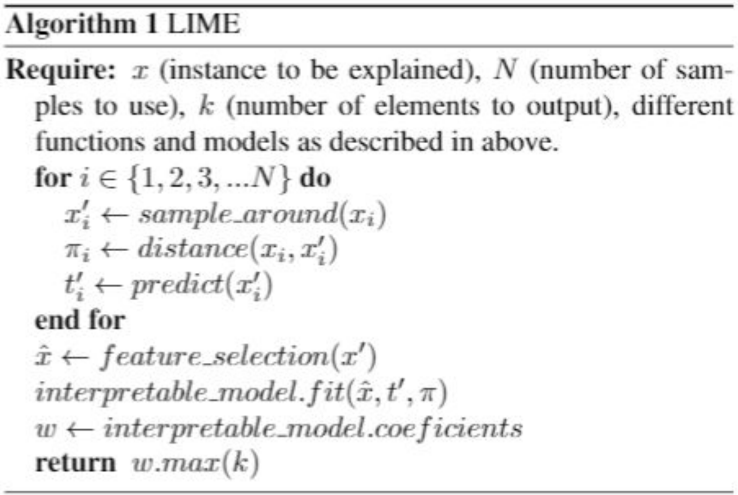
Sample_around “**x**” - neighborhood data “**x’** ”, by creating active segments “**a**”

Distance “**π**” - weights from distance between **x** and **x’**

Predict “**t’** ” (probabilities) - a black-box model, to classify on **x’**

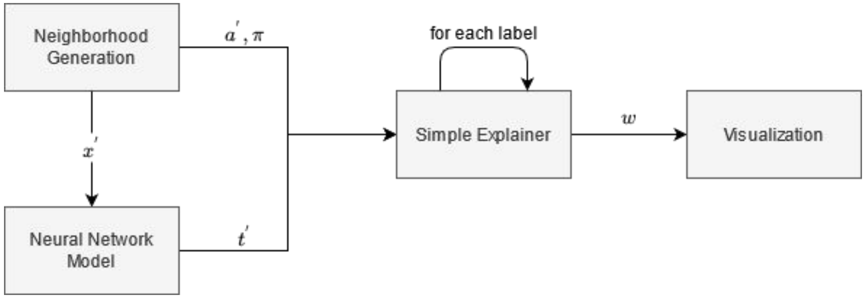
Active segments “**α**” are arrays of binary values which correspond to "turned on" features

Fit interpretable models given instance, black-box probabilities and distance weights to get coefficients “***w***”



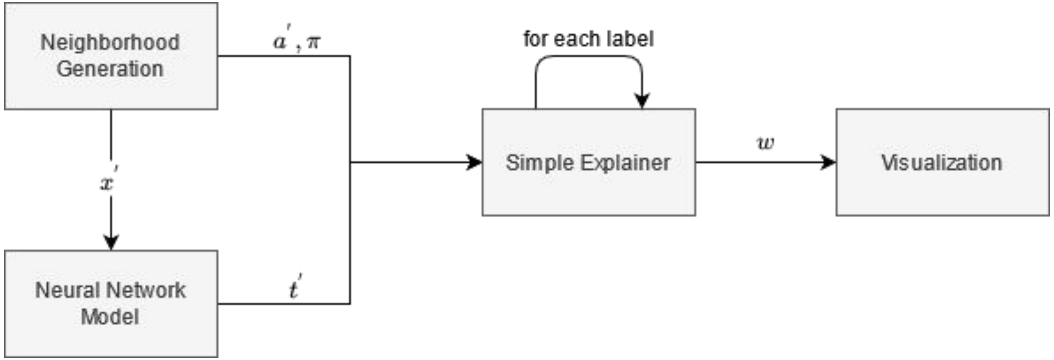
Algorithm 1 LIME

```
Require: $x$ (instance to be explained), $N$ (number of samples to use), $k$ (number of elements to output), different functions and models as described in above.
  for $i \in \{1, 2, 3, ...N\}$ do
    $x'_i \leftarrow sample\_around(x_i)$
    $\pi_i \leftarrow distance(x_i, x'_i)$
    $t'_i \leftarrow predict(x'_i)$
  end for
  $\hat{x} \leftarrow feature\_selection(x')$
  $interpretable\_model.fit(\hat{x}, t', \pi)$
  $w \leftarrow interpretable\_model.coeficients$
  return $w.max(k)$
```



Simple Explainer - Method

- Trains regression model on weighted active segments “**α, π**” from neighbourhood generation fitting output probabilities from neural network model “**t**”
- Coefficients of a simple model are used for explanation assessment “***w***”
- Any regression model can be used as long as its coefficients have the same dimensionality as input data



Simple Explainer - Feature & Model Selection

Simple Explainer

Feature Selection

- Limit dimensionality issues
- Improve training speed especially for image data

Methods

- Highest Weights
- Forward Selection
  - Found to be 20x slower
  - As good as highest weights

features for simple model

Model Selection

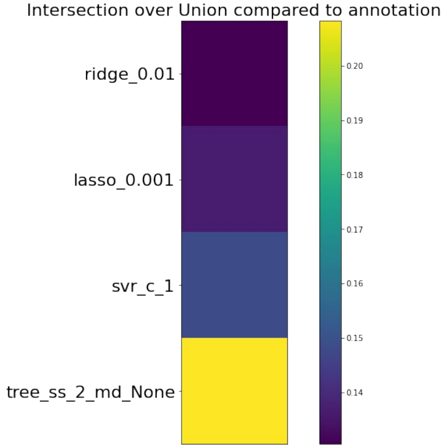
- Model that is easy to interpret
- Returns some measure of feature importance
- Quick to fit on input features

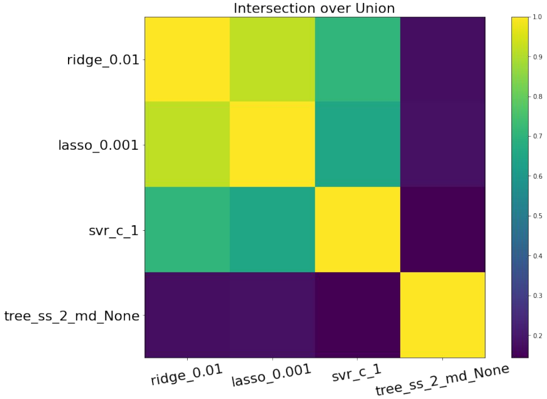
Methods

- Ridge
- Lasso
- Linear SVM
- Decision Tree

Simple Explainer - Comparison

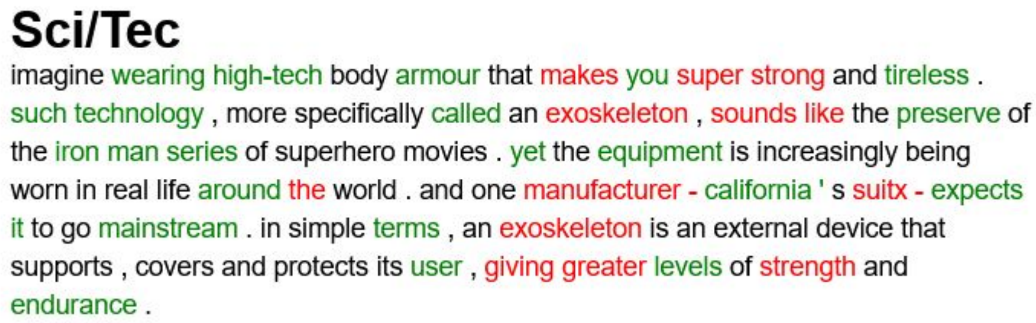
	Ridge	Lasso	Linear SVM	Decision Tree
Interpretation of model's features	Coefficients	Coefficients	Coefficients	**Feature Importance**
Number of selected significant features	More	Less	More	Less
Domain of features	Positive and Negative	Positive and Negative	Positive and Negative	**Only Positive**
Explanation characteristics	**Good choice** for any type of data, quite robust	**Heavily penalizes** features with smaller coefficients	**Ambiguous results** with a lot of variation	**Inherent nonlinearity** that can find **different features** compared to other methods


Intersection over Union compared to annotation


Intersection over Union

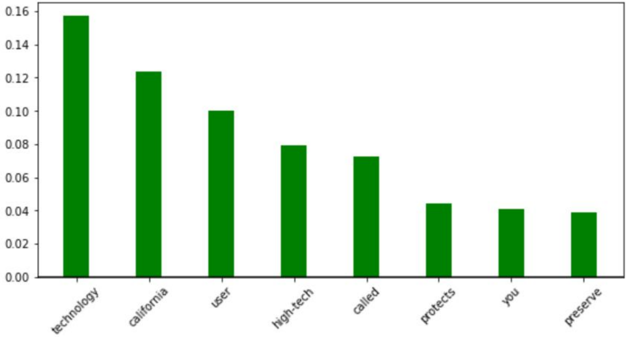
Ridge vs Decision Tree - Text

Ridge

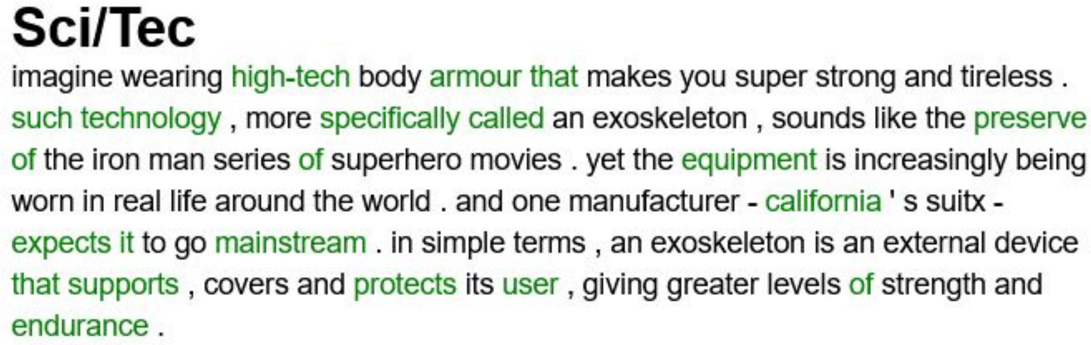


Sci/Tec

imagine wearing high-tech body armour that makes you super strong and tireless . such technology , more specifically called an exoskeleton , sounds like the preserve of the iron man series of superhero movies . yet the equipment is increasingly being worn in real life around the world . and one manufacturer - california ' s suitx - expects it to go mainstream . in simple terms , an exoskeleton is an external device that supports , covers and protects its user , giving greater levels of strength and endurance .

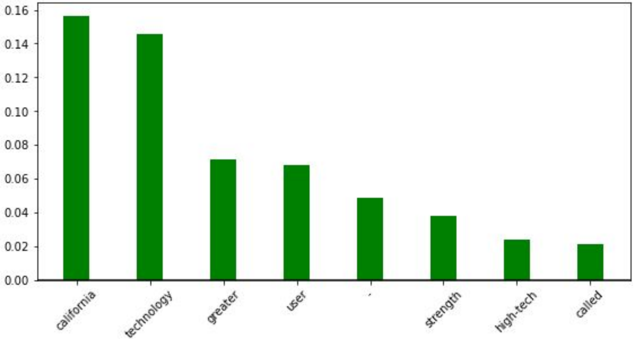


Decision Tree



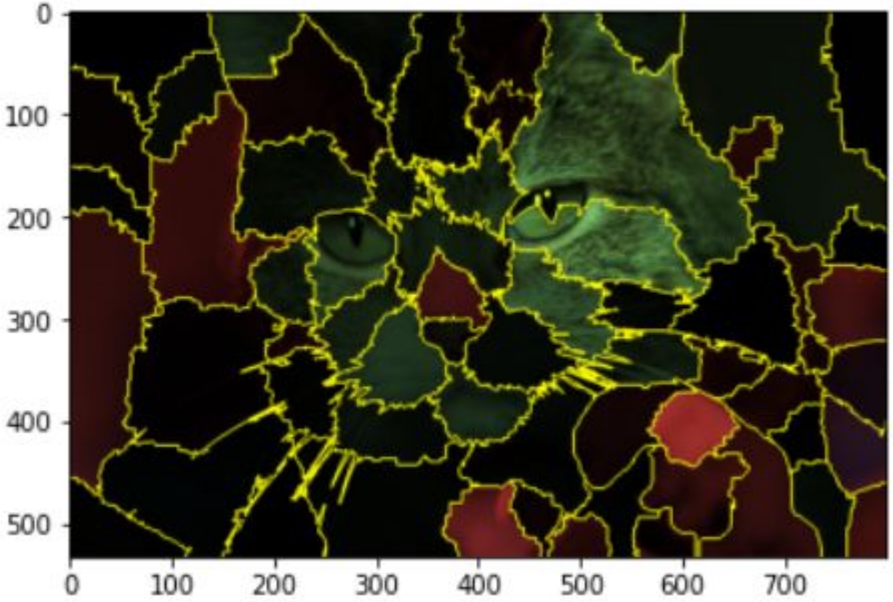
Sci/Tec

imagine wearing high-tech body armour that makes you super strong and tireless . such technology , more specifically called an exoskeleton , sounds like the preserve of the iron man series of superhero movies . yet the equipment is increasingly being worn in real life around the world . and one manufacturer - california ' s suitx - expects it to go mainstream . in simple terms , an exoskeleton is an external device that supports , covers and protects its user , giving greater levels of strength and endurance .

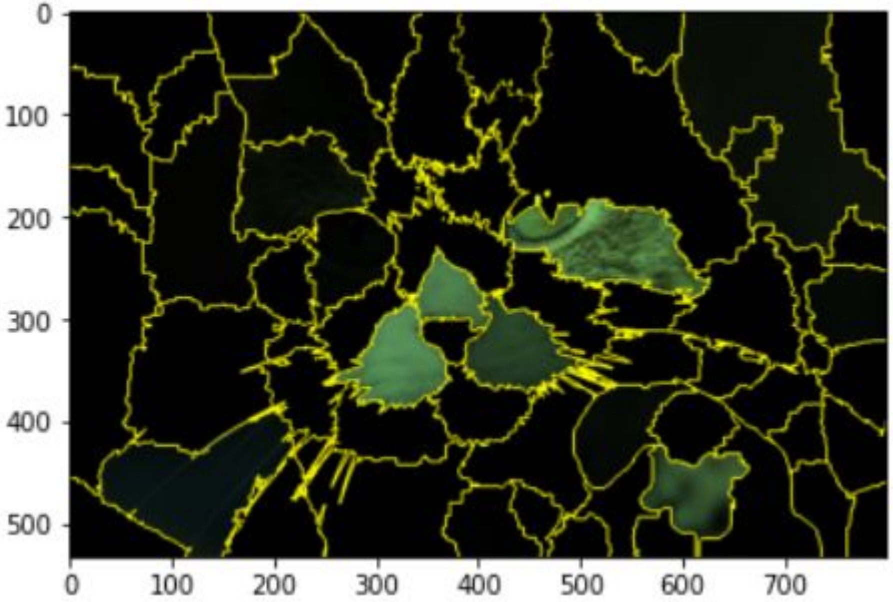


Ridge vs Decision Tree - Image

Ridge Model



Tree Model



Number of sample and distance weight

Experiments on number of samples

- Found to perform better with increasing number of samples

Experiments on the distance between original and sample data

- Found have little effect on the interpretable outcome
- Especially with more samples

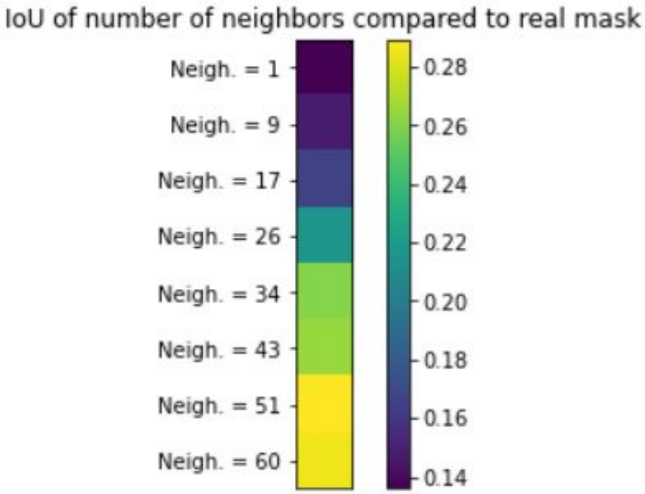


Image data

Neighborhood generation - Text

Methods

- Random_uniform (baseline)
- Random_normal
- One_on
- One_off
- Consecutive

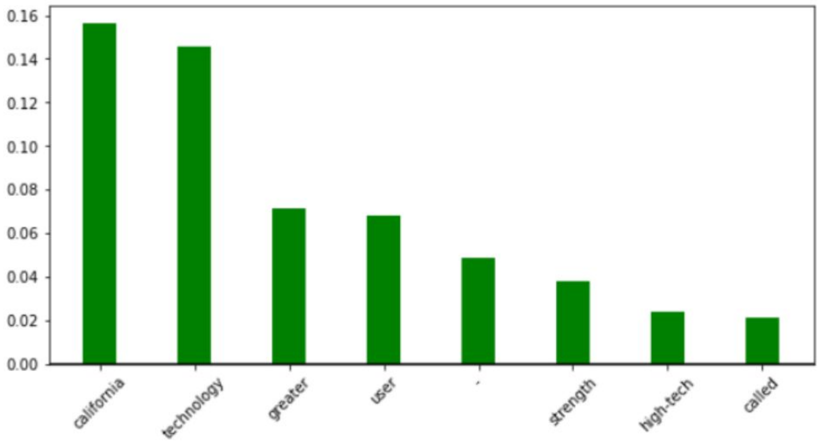
Results

- IoU
  - Best: ru and rn in both models
  - In tree, all methods have similar performance
- Visual inspection
  - Ridge: similar results
  - Tree: rn not performing as well

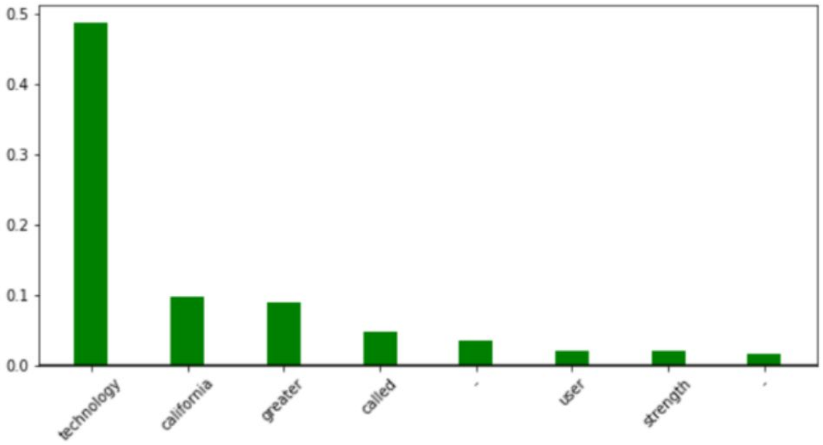
	ru	rn	one_on	one_off	con
IoU Ridge	0.24	0.24	0.18	0.13	0.19
IoU Tree	0.25	0.22	0.21	0.22	0.18

Neighborhood generation - Text

Annotation: **high-tech**, strong, **technology**, equipment, real, life, manufacturer, expects, external, device, protects, **user**, levels, **strength**

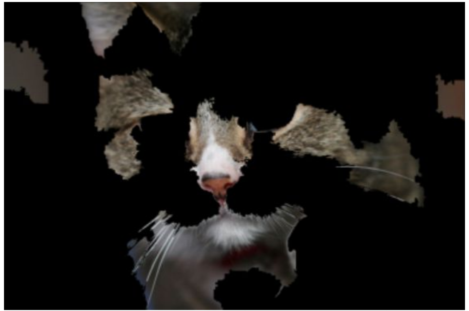


Random uniform - Ridge

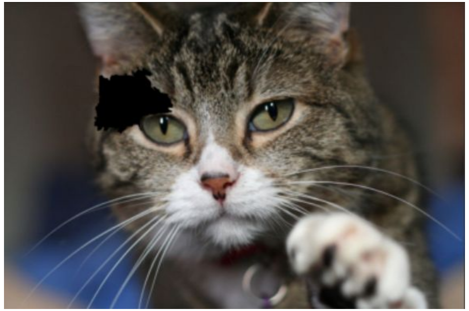


Random normal - Tree

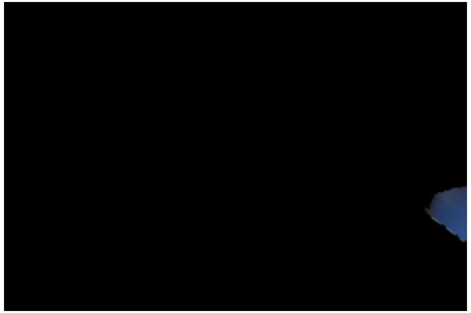
Neighborhood generation - Image



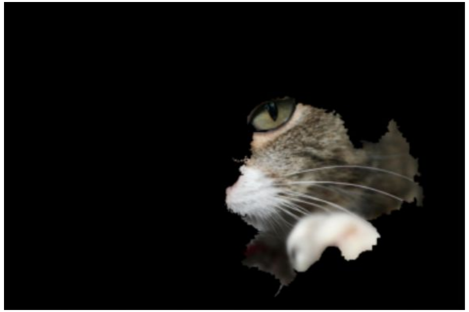
Random (baseline)



One_off

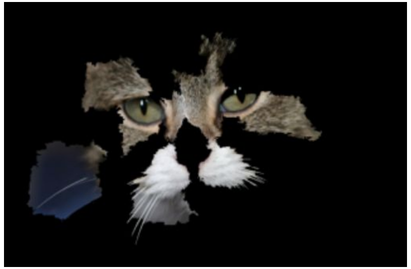


One_on

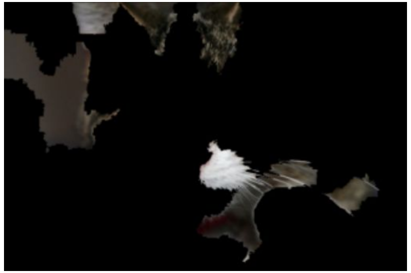


Radio

Neighborhood generation - Image



Random



One_off

- Best: Random and Radio
- Size and meaning of the superpixels is very influencing

	Random	One_on	One_off	Radio
IoU	0.38	0.16	0.11	0.32

Discussion

Intersection over Union and annotations

Only 2 instances of data

Impact of randomness

Conclusion

Best models:	Decision Tree (IoU) and Ridge (Visual)
Best neighborhood generation:	Random methods
Most important parameters (CoV):	Simple model and neighborhood generation

Discussion

- There is no good objective metric to evaluate explanations
- Experiments carried out in only 2 instances of data
- Randomness in each execution

Conclusion

- Best models: Decision Tree (IoU) and Ridge (subjective)
- Best neighborhood generation: Random, but depends on the data
- Most important parameters (CoV): simple model and neighborhood generation