Human-Centered NLP

CSE 538

NLP, The Course

Overall NLP Concept

I. Syntax

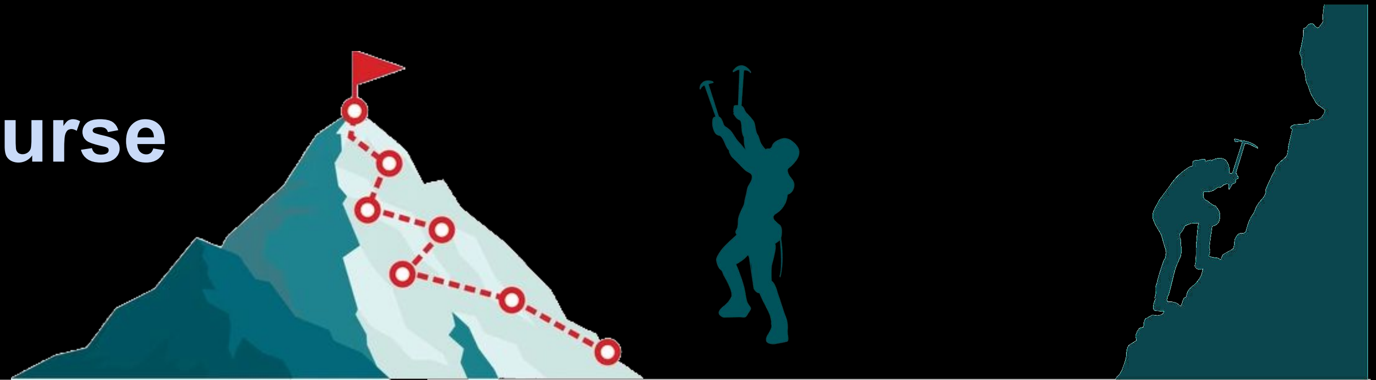
II. Semantics

Overall NLP Concept

III. Language Modeling

IV. Applications

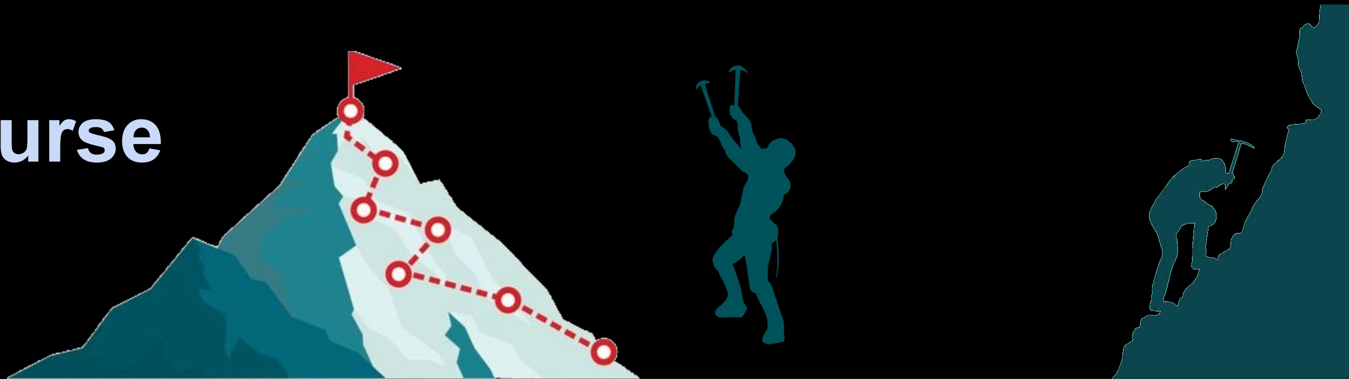
NLP, The Course



Overall NLP Concept
I. Syntax
Introduction to NLP; Tokenization; Words Corpora
One-hot, and Multi-hot encoding. Parts-of-Speech; Named Entities;
Parsing; Verbal Predicates;Dependency Parsing
II. Semantics
Dependency Parsing; Word Sense Disambiguation
Vector Semantics (Embeddings), Word2vec
Probabilistic Language Models Ngram Classifier, Topic Modeling

Overall NLP Concept
III. Language Modeling
IV. Applications

NLP, The Course



Overall NLP Concept

I. Syntax

Introduction to NLP; Tokenization; Words Corpora

One-hot, and Multi-hot encoding.
Parts-of-Speech; Named Entities;

Parsing; Verbal Predicates;Dependency Parsing

II. Semantics

Dependency Parsing; Word Sense Disambiguation

Vector Semantics (Embeddings), Word2vec

Probabilistic Language Models
Ngram Classifier, Topic Modeling

Overall NLP Concept

III. Language Modeling

Ethical Considerations

Masked Language Modeling (autoencoding)

Generative Language Modeling (autoregressive)

Applying LMs

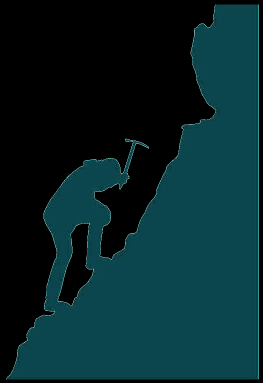
IV. Applications

Language and Psychology
(advanced sentiment)

Speech and Audio Processing, Dialog (chatbots)

Question Answering, Translation

NLP The Course



Overall NLP Concept

I. Syntax

Introduction to NLP; Tokenization; Words Corpora

One-hot, and Multi-hot encoding. Parts-of-Speech; Named Entities;

Parsing; Verbal Predicates;Dependency Parsing

II. Semantics

Dependency Parsing; Word Sense Disambiguation

Vector Semantics (Embeddings), Word2vec

Probabilistic Language Models Ngram Classifier, Topic Modeling

Overall NLP Concept

III. Language Modeling

Ethical Considerations

Masked Language Modeling (autoencoding)

Generative Language Modeling (autoregressive)

Applying LMs

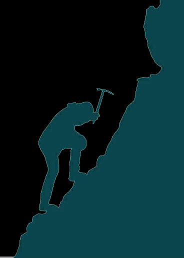
IV. Applications

Language and Psychology (advanced sentiment)

Speech and Audio Processing, Dialog (chatbots)

Question Answering, Translation

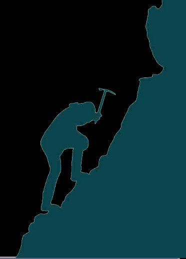
NLP The Course



Overall NLP Concept	Computation or ML
I. Syntax | Classification	
Introduction to NLP; Tokenization; Words Corpora	
One-hot, and Multi-hot encoding. Parts-of-Speech; Named Entities;	
Parsing; Verbal Predicates;Dependency Parsing	
II. Semantics | Probabilistic Models	
Dependency Parsing; Word Sense Disambiguation	
Vector Semantics (Embeddings), Word2vec	
Probabilistic Language Models Ngram Classifier, Topic Modeling	

Overall NLP Concept	Computation or ML
III. Language Modeling | Transformers	
Ethical Considerations	
Masked Language Modeling (autoencoding)	
Generative Language Modeling (autoregressive)	
Applying LMs	
IV. Applications | Custom Statistical or Symbolic	
Language and Psychology (advanced sentiment)	
Speech and Audio Processing, Dialog (chatbots)	
Question Answering, Translation	

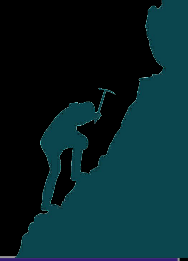
NLP The Course



Overall NLP Concept	Computation or ML
I. Syntax | Classification	
Introduction to NLP; Tokenization; Words Corpora	Regular Expressions; Edit Distance
One-hot, and Multi-hot encoding. Parts-of-Speech; Named Entities;	Maximum Entropy Classifier (LogReg), Gradient Descent,
Parsing; Verbal Predicates;Dependency Parsing	Cross Validation; Regularization Accuracy Metrics; Shift Reduce
II. Semantics | Probabilistic Models	
Dependency Parsing; Word Sense Disambiguation	Term Probabilities; N-d Vectors
Vector Semantics (Embeddings), Word2vec	LDA, Skipgram Model
Probabilistic Language Models Ngram Classifier, Topic Modeling	markov assumption, chain rule, smoothing

Overall NLP Concept	Computation or ML
III. Language Modeling | Transformers	
Ethical Considerations	Model cards, Pred Bias Frmwrk
Masked Language Modeling (autoencoding)	Neural Networks; Backprop Cross-Entropy Loss Self-Attention,
Generative Language Modeling (autoregressive)	Positional encodings The Transformer: Beam Search
Applying LMs	Fine-Tuning, zero-/few-shot, Instruction tuning
IV. Applications | Custom Statistical or Symbolic	
Language and Psychology (advanced sentiment)	Differential Language Analysis; Adaptive Modeling; Human LMing
Speech and Audio Processing, Dialog (chatbots)	Wave Transforms; RNNs
Question Answering, Translation	Multihop Reasoning

NLP The Course



Overall NLP Concept	Computation or ML
I. Syntax | Classification	
Introduction to NLP; Tokenization; Words Corpora	Regular Expressions; Edit Distance
One-hot, and Multi-hot encoding. Parts-of-Speech; Named Entities;	Maximum Entropy Classifier (LogReg), Gradient Descent,
Parsing; Verbal Predicates;Dependency Parsing	Cross Validation; Regularization Accuracy Metrics; Shift Reduce
II. Semantics | Probabilistic Models	
Dependency Parsing; Word Sense Disambiguation	Term Probabilities; N-d Vectors
Vector Semantics (Embeddings), Word2vec	LDA, Skipgram Model
Probabilistic Language Models Ngram Classifier, Topic Modeling	markov assumption, chain rule, smoothing

Overall NLP Concept	Computation or ML
III. Language Modeling | Transformers	
Ethical Considerations	Model cards, Pred Bias Frmwrk
Masked Language Modeling (autoencoding)	Neural Networks; Backprop Cross-Entropy Loss Self-Attention,
Generative Language Modeling (autoregressive)	Positional encodings The Transformer: Beam Search
Applying LMs	Fine-Tuning, zero-/few-shot, Instruction tuning
IV. Applications | Custom Statistical or Symbolic	
Language and Psychology (advanced sentiment)	**Differential Language Analysis; Adaptive Modeling; Human LMing**
Speech and Audio Processing, Dialog (chatbots)	Wave Transforms; RNNs
Question Answering, Translation	Multihop Reasoning

Natural Language Processing

Psychological & Health Sciences

Extraversion

Schwartz, H. A., Eichstaedt, ... & Ungar. (2013). Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one, 8(9)*.

Introversion

Schwartz, H. A., Eichstaedt, ... & Ungar. (2013). Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one, 8(9)*.

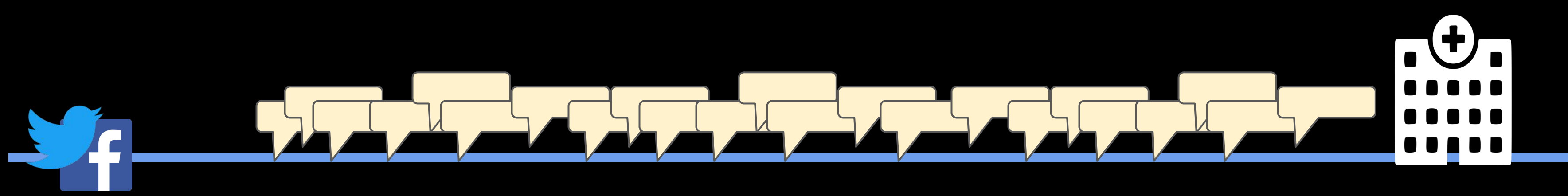
Natural Language Processing

Psychological & Health Sciences

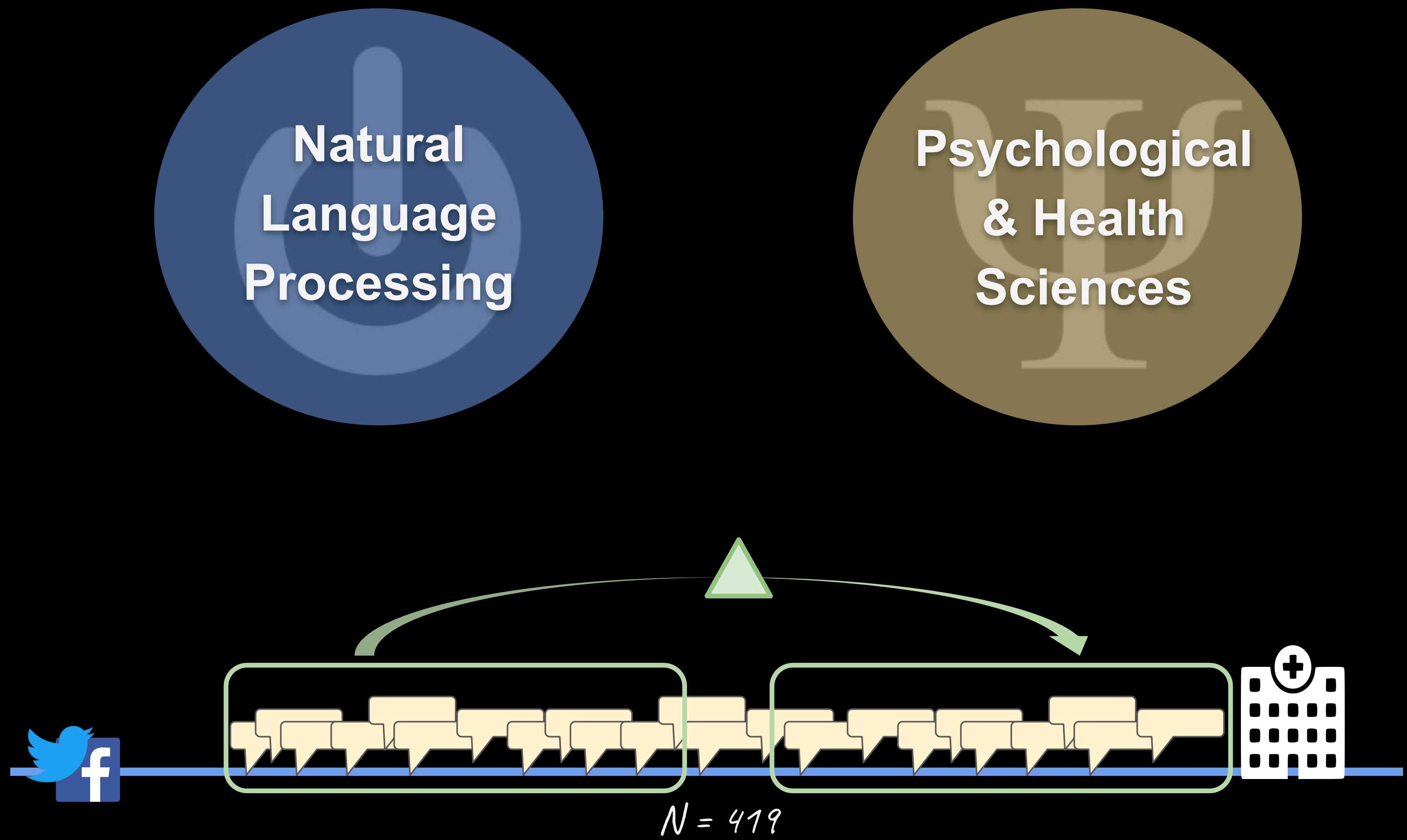
Schwartz, H. A., Eichstaedt, ... & Ungar. (2013). Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one, 8(9)*.

Natural Language Processing

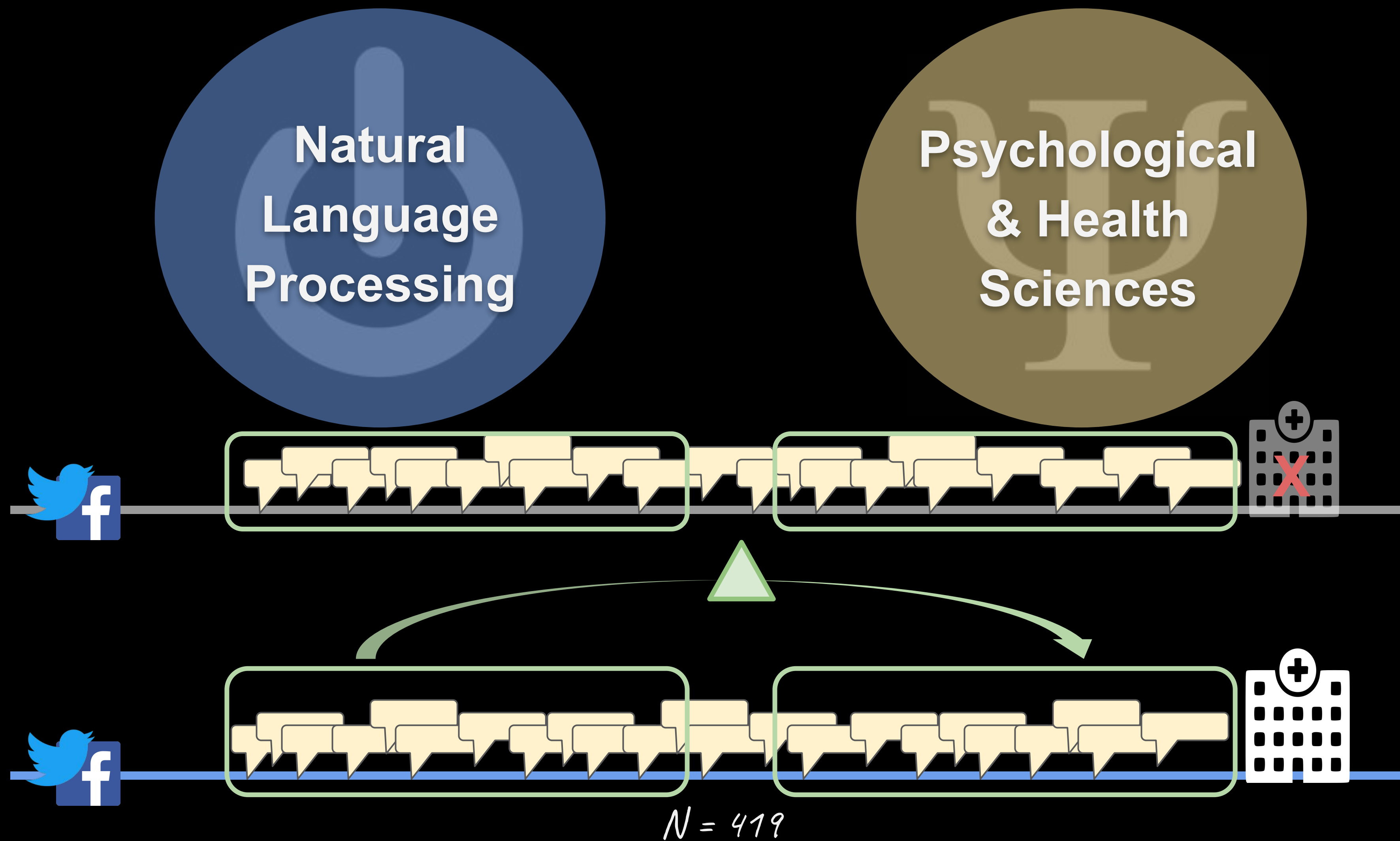
Psychological & Health Sciences



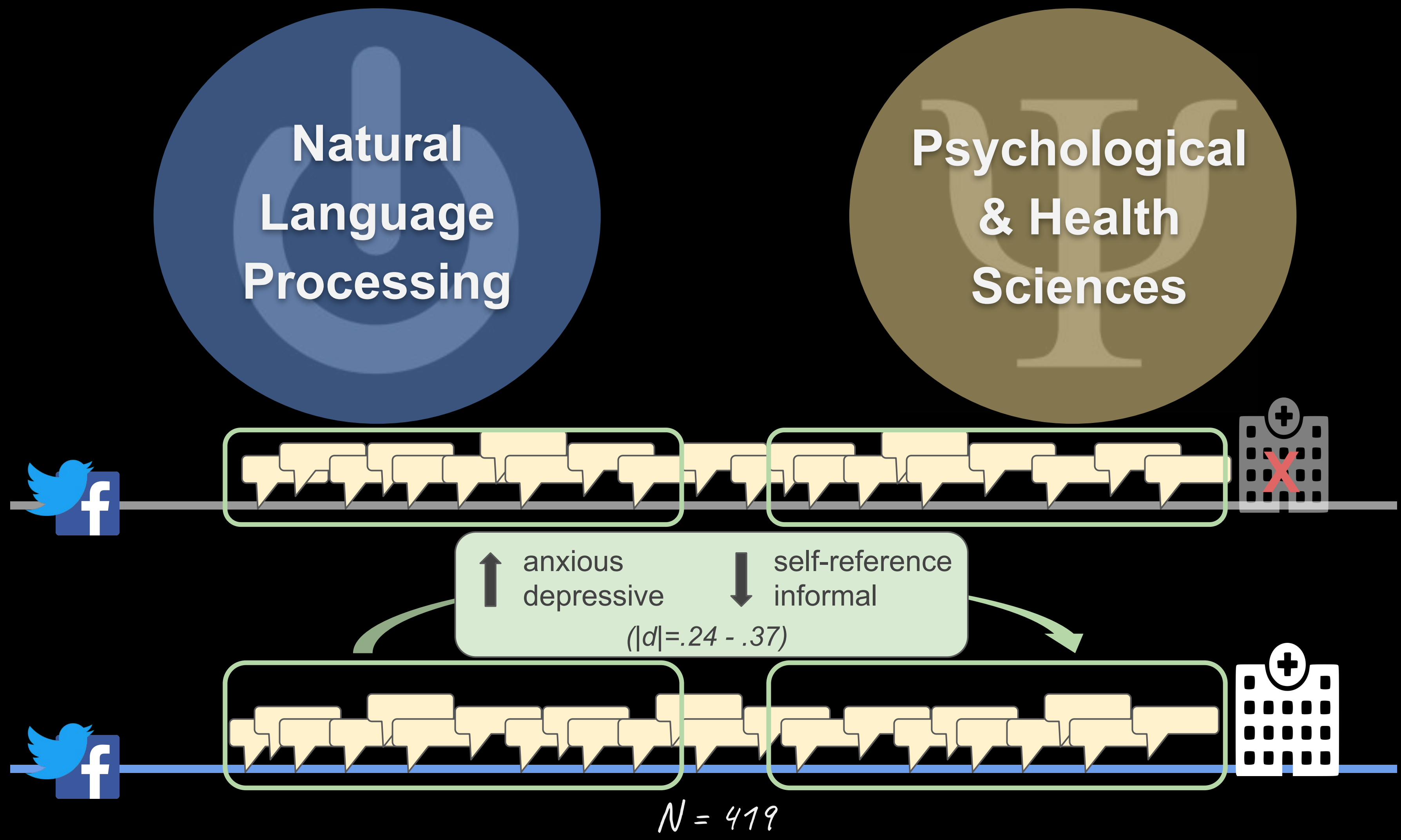
Guntuku, S. C., Schwartz, H. A., Kashyap, A., Gaulton, J. S., Stokes, D. C., Asch, D. A., ... & Merchant, R. M. (2020). Variability in Language used on Social Media prior to Hospital Visits. *Nature - Scientific Reports*, *10(1),* 1-9.



Guntuku, S. C., Schwartz, H. A., Kashyap, A., Gaulton, J. S., Stokes, D. C., Asch, D. A., ... & Merchant, R. M. (2020). Variability in Language used on Social Media prior to Hospital Visits. *Nature - Scientific Reports*, *10(1),* 1-9.



Guntuku, S. C., Schwartz, H. A., Kashyap, A., Gaulton, J. S., Stokes, D. C., Asch, D. A., ... & Merchant, R. M. (2020). Variability in Language used on Social Media prior to Hospital Visits. *Nature - Scientific Reports*, *10(1),* 1-9.



Guntuku, S. C., Schwartz, H. A., Kashyap, A., Gaulton, J. S., Stokes, D. C., Asch, D. A., ... & Merchant, R. M. (2020). Variability in Language used on Social Media prior to Hospital Visits. *Nature - Scientific Reports*, *10(1),* 1-9.

Natural Language Processing

Psychological & Health Sciences

Overly Simplified Problem-Statement:

Natural language is written by

Overly Simplified Problem-Statement:

Natural language is written by **people.**

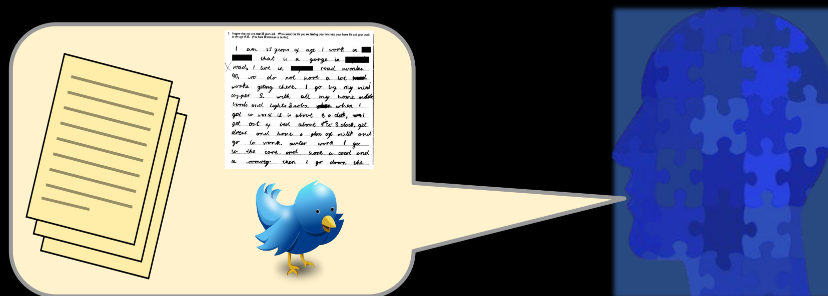
Overly Simplified Problem-Statement:

Natural language is written by **people.**

Problem

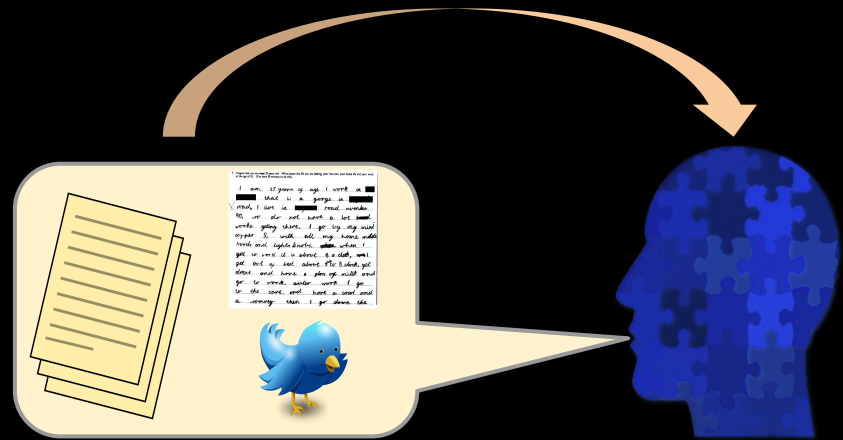
Natural language is written by **people.**

Natural language is generated by people.



People have different beliefs, backgrounds, styles, vocabularies, preferences, knowledge, personalities, …

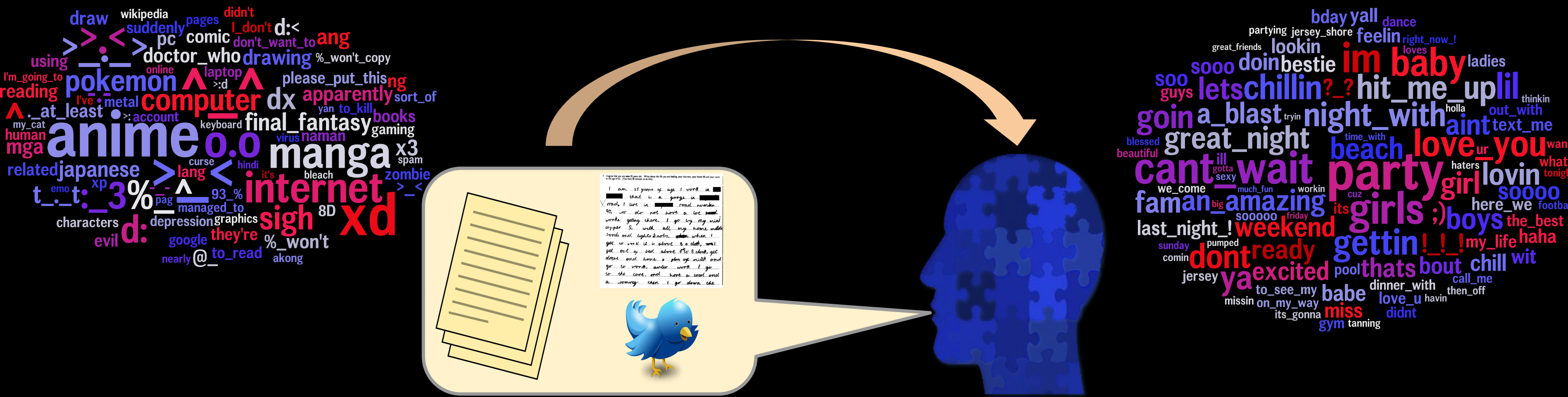
Natural language is generated by people.



People have different beliefs, backgrounds, styles, vocabularies, preferences, knowledge, personalities, …,

and our language reflects these differences.

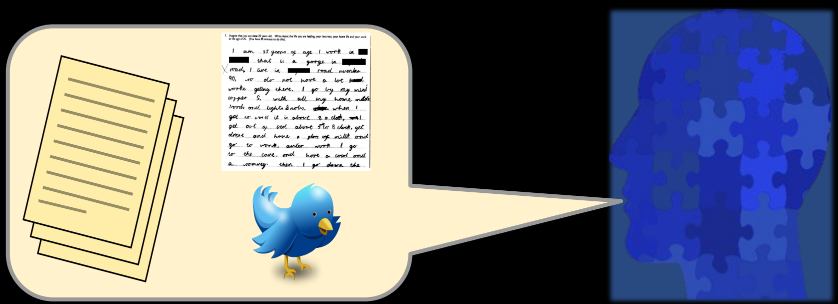
Natural language is generated by people.



People have different beliefs, backgrounds, styles, vocabularies, preferences, knowledge, personalities, …,

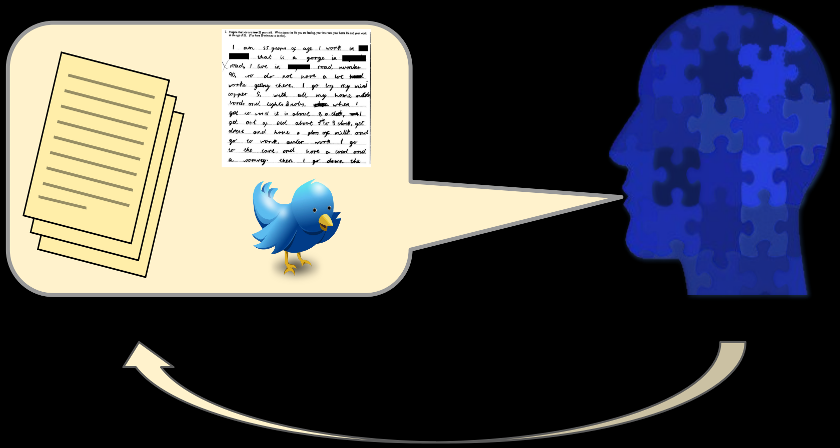
and our language reflects these differences.

Human Centered NLP:



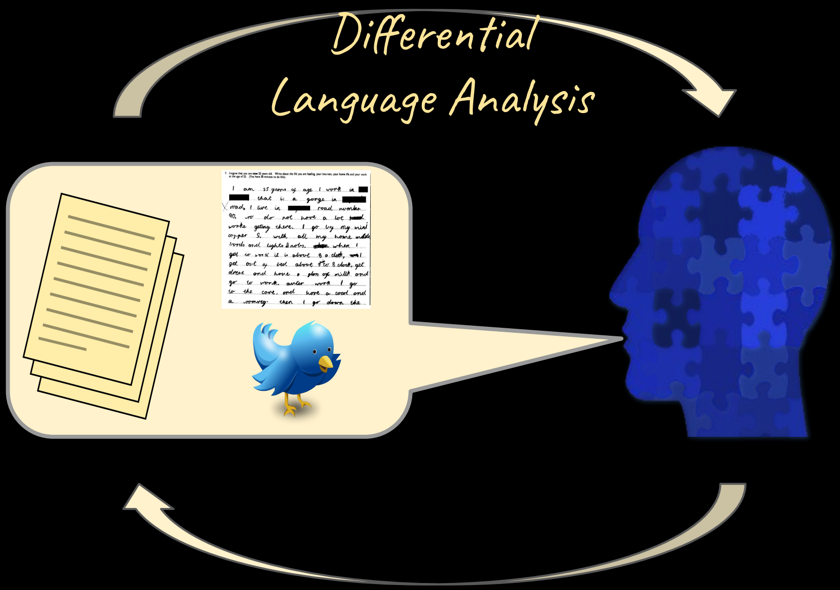
Human Centered NLP:

1. Model language as a human process



Human Centered NLP:

1. Model language as a human process
2. Use language to better understand humans.



Human-Centered NLP – We will cover:

1. Differential Language Analysis
2. Human Factor Adaptation
3. Human Language Modeling

Differential Language Analysis

Input:

Linguistic features

Human or community attribute

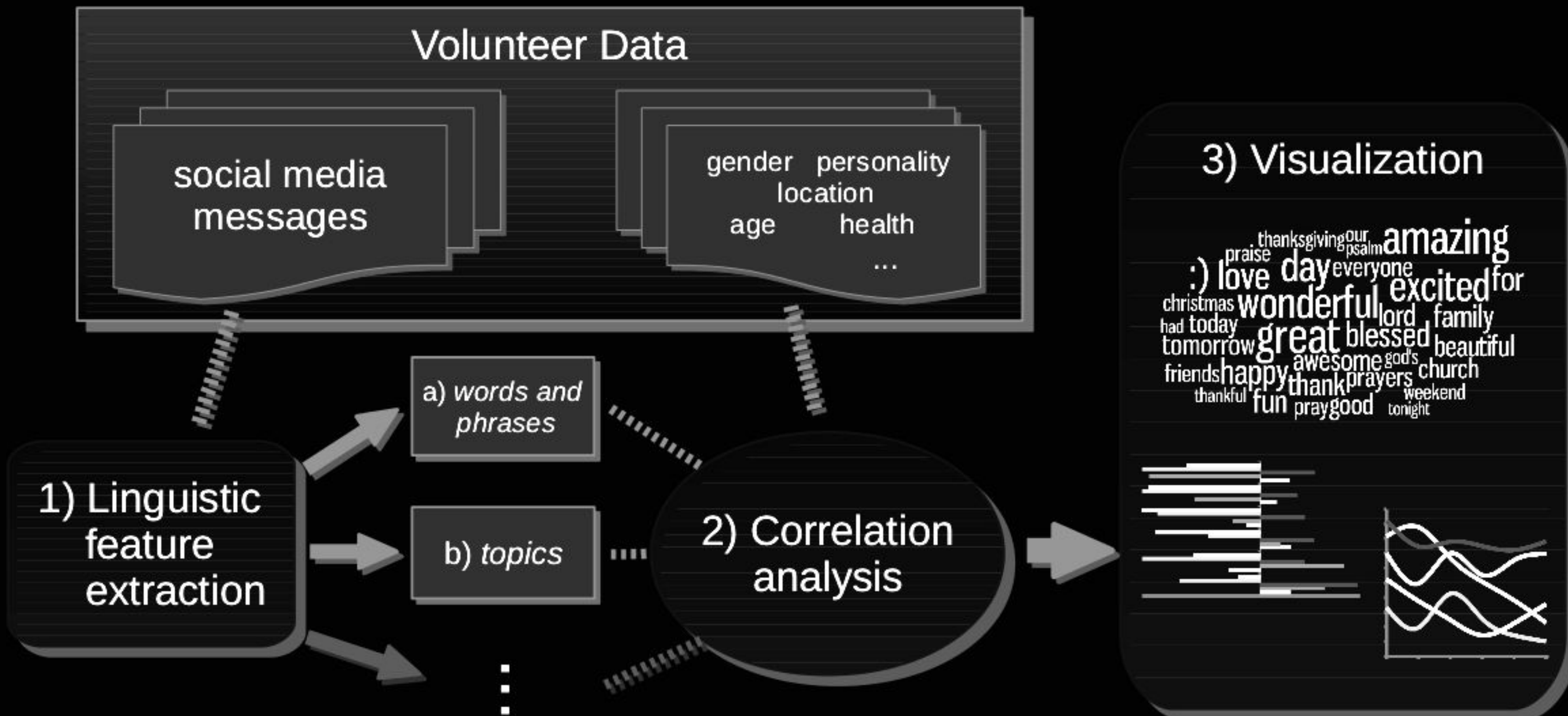
Output:

Features distinguishing attribute

Goal: Data-driven insights about an attribute

E.g. Words distinguishing communities with increases in real estate prices.

Differential Language Analysis



Differential Language Analysis

Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation

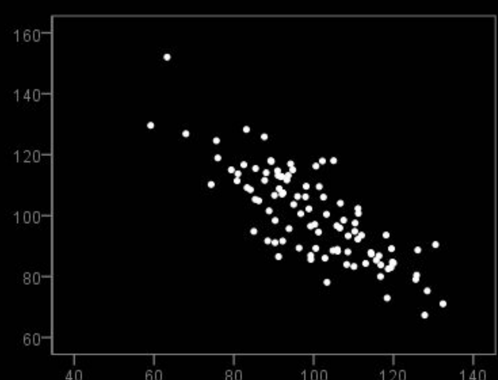
    Limitation: Doesn’t handle controls

Differential Language Analysis

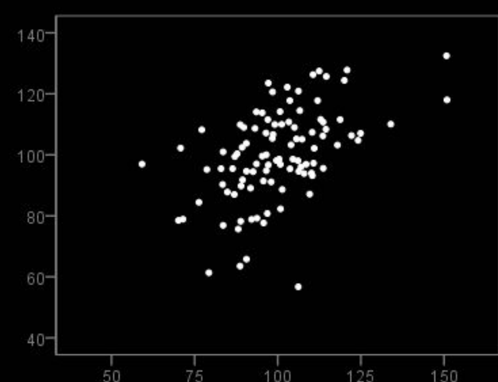
Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

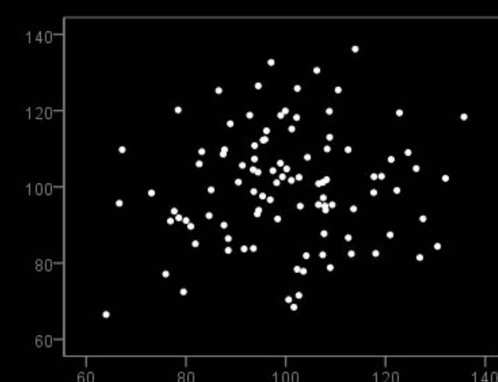
- Pearson Product-Moment Correlation
  - Limitation: Doesn’t handle controls



r = -0.8



r = 0.5



r = 0.1

Differential Language Analysis

Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation

  Limitation: Doesn't handle controls

- Standardized Multivariate Linear Regression

  Fit the model:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

Differential Language Analysis

Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation
  Limitation: Doesn’t handle controls

- **Standardized** Multivariate Linear Regression
  Fit the model:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

  Adjust all variables to have “mean center” and “unit variance”:

Differential Language Analysis

Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation

  Limitation: Doesn't handle controls

- **<u>Standardized</u>** Multivariate Linear Regression

  Fit the model:

  $$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

  Adjust all variables to have "mean center" and "unit variance":

$$z = \frac{x - \mu}{\sigma}$$

$\mu =$ Mean

$\sigma =$ Standard Deviation

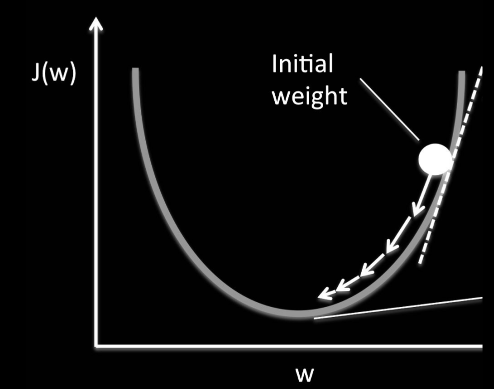
Differential Language Analysis

Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation
  Limitation: Doesn't handle controls

- Standardized Multivariate Linear Regression
  Fit the model:
  $$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$
  Option 1: Gradient Descent:
  $J = \sum (y - \hat{y})^2$ -- "Sum of Squares" Error



Differential Language Analysis

Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation

  Limitation: Doesn’t handle controls

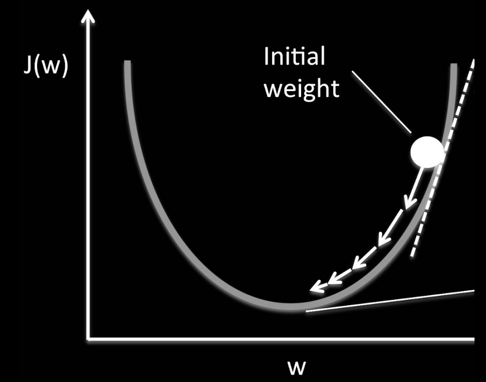
- Standardized Multivariate Linear Regression

  Fit the model: $$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$

  Option 1: Gradient Descent:

  $J = \sum (y - \hat{y})^2$ -- “Sum of Squares” Error

  Option 2: Matrix model: $$Y = X\beta + \epsilon$$



Differential Language Analysis

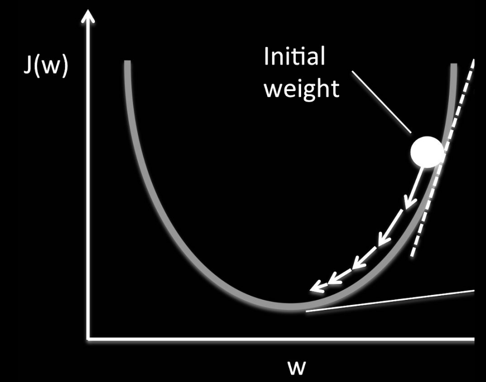
Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation
    Limitation: Doesn't handle controls

- Standardized Multivariate Linear Regression
  Fit the model: $$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$$
  Option 1: Gradient Descent:
    $J = \sum (y - \hat{y})^2$ -- "Sum of Squares" Error
  Option 2: Matrix model: $Y = X\beta + \epsilon$
  Matrix Computation Solution:

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$



Differential Language Analysis

Methods of Correlation Analysis:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- Pearson Product-Moment Correlation

  Limitation: Doesn't handle controls

- Standardized Multivariate Linear Regression

  Fit the model: $Y_i = \beta_0 + \boxed{\beta_1} X_{i1} + \beta_2 X_{i2} + \ldots + \beta_m X_{m1} + \epsilon_i$

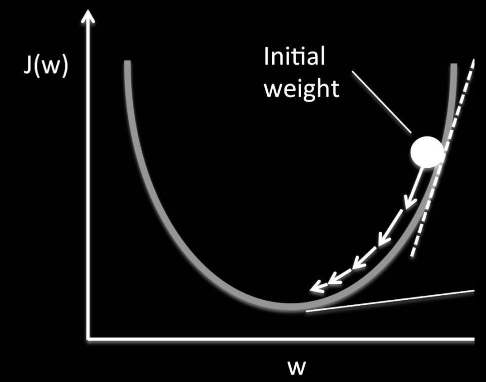
  Option 1: Gradient Descent:

  $J = \sum (y - \hat{y})^2$ -- "Sum of Squares" Error

  Option 2: Matrix model: $Y = X\beta + \epsilon$

  Matrix Computation Solution:

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$



Differential Language Analysis

Methods of "Correlation" Analysis for binary outcomes:

- Logistic Regression over Standardized variables
- Odds Ratio

$$\frac{\dfrac{\dfrac{countA(\text{"horrible"})}{NA}}{1-\dfrac{countA(\text{"horrible"})}{NA}}}{\dfrac{\dfrac{countB(\text{"horrible"})}{NB}}{1-\dfrac{countB(\text{"horrible"})}{NB}}}$$

Differential Language Analysis

Methods of "Correlation" Analysis for binary outcomes:

- Logistic Regression over Standardized variables
- Odds Ratio

$$\frac{\dfrac{\frac{countA(\text{"horrible"})}{NA}}{1-\frac{countA(\text{"horrible"})}{NA}}}{\dfrac{\frac{countB(\text{"horrible"})}{NB}}{1-\frac{countB(\text{"horrible"})}{NB}}} \propto log\left(\frac{\frac{countA(\text{"horrible"})}{NA}}{1-\frac{countA(\text{"horrible"})}{NA}}\right) - log\left(\frac{\frac{countB(\text{"horrible"})}{NB}}{1-\frac{countB(\text{"horrible"})}{NB}}\right)$$

Differential Language Analysis

Methods of "Correlation" Analysis for binary outcomes:

- Logistic Regression over Standardized variables

- Odds Ratio

$$\frac{\dfrac{\frac{countA(\text{"horrible"})}{NA}}{1-\frac{countA(\text{"horrible"})}{NA}}}{\dfrac{\frac{countB(\text{"horrible"})}{NB}}{1-\frac{countB(\text{"horrible"})}{NB}}} \propto log\left(\frac{\frac{countA(\text{"horrible"})}{NA}}{1-\frac{countA(\text{"horrible"})}{NA}}\right) - log\left(\frac{\frac{countB(\text{"horrible"})}{NB}}{1-\frac{countB(\text{"horrible"})}{NB}}\right)$$

$$= log\left(\frac{countA(\text{"horrible"})}{NA-countA(\text{"horrible"})}\right) - log\left(\frac{countB(\text{"horrible"})}{NB-countB(\text{"horrible"})}\right)$$

Differential Language Analysis

$$log\left(\frac{countA(\text{"horrible"})}{NA-countA(\text{"horrible"})}\right) - log\left(\frac{countB(\text{"horrible"})}{NB-countB(\text{"horrible"})}\right)$$

- Odds Ratio using Informative Dirichlet Prior

$$\delta_w^{(i-j)} = \log\left(\frac{f_w^i + \alpha_w}{n^i + \alpha_0 - (f_w^i + \alpha_w)}\right) - \log\left(\frac{f_w^j + \alpha_w}{n^j + \alpha_0 - (f_w^j + \alpha_w)}\right) \qquad (20.9)$$

(Monroe et al., 2010; Jurafsky, 2017)

Differential Language Analysis

$$log\left(\frac{countA(\text{“horrible”})}{NA-countA(\text{“horrible”})}\right)-log\left(\frac{countB(\text{“horrible”})}{NB-countB(\text{“horrible”})}\right)$$

- Odds Ratio using **Informative Dirichlet Prior**

$$\delta_w^{(i-j)}=\log\left(\frac{f_w^i+\boxed{\alpha_w}}{n^i+\boxed{\alpha_0}-(f_w^i+\boxed{\alpha_w})}\right)-\log\left(\frac{f_w^j+\boxed{\alpha_w}}{n^j+\boxed{\alpha_0}-(f_w^j+\boxed{\alpha_w})}\right) \quad (20.9)$$

(where n^i is the size of corpus i, n^j is the size of corpus j, f_w^i is the count of word w in corpus i, f_w^j is the count of word w in corpus j, α_0 is the size of the background corpus, and α_w is the count of word w in the background corpus.)

(Monroe et al., 2010; Jurafsky, 2017)

Differential Language Analysis

$$log\left(\frac{countA(\text{"horrible"})}{NA-countA(\text{"horrible"})}\right) - log\left(\frac{countB(\text{"horrible"})}{NB-countB(\text{"horrible"})}\right)$$

- Odds Ratio using **Informative Dirichlet Prior**

$$\delta_w^{(i-j)} = \log\left(\frac{f_w^i + \boxed{\alpha_w}}{n^i + \boxed{\alpha_0} - (f_w^i + \boxed{\alpha_w})}\right) \quad \text{og}\left(\frac{f_w^j + \boxed{\alpha_w}}{n^j + \boxed{\alpha_0} - (f_w^j + \boxed{\alpha_w})}\right) \qquad (20.9)$$

(where n^i is the size of corpus i, n^j is the s pus j, f_w^i is the count of word w in corpus i, f_w^j is the count of word w in is the size of the background corpus, and α_w is the count of word w in corpus.)

Bayesian term for "smoothing": accounts for uncertainty as a function of event frequency (i.e. words observed less) by integrating "prior" beliefs mathematically.

(Mo

Differential Language Analysis

$$log\left(\frac{countA(\text{"horrible"})}{NA-countA(\text{"horrible"})}\right)-log\left(\frac{countB(\text{"horrible"})}{NB-countB(\text{"horrible"})}\right)$$

- Odds Ratio using **Informative Dirichlet Prior**

$$\delta_w^{(i-j)}=\log\left(\frac{f_w^i+\boxed{\alpha_w}}{n^i+\boxed{\alpha_0}-(f_w^i+\boxed{\alpha_w})}\right)-\log\left(\frac{f_w^j+\boxed{\alpha_w}}{n^j+\boxed{\alpha_0}-(f_w^j+\boxed{\alpha_w})}\right)\qquad(20.9)$$

(where n^i is the size of corpus i, n^j is the s … pus j, f_w^i is the count of word w in corpus i, f_w^j is the count of word w in … is the size of the background corpus, and α_w is the count of word w in … corpus.)

Bayesian term for "smoothing": accounts for uncertainty as a function of event frequency (i.e. words observed less) by integrating "prior" beliefs mathematically.
"Informative": the prior is based on past evidence. Here, the total frequency of the word.

(Mo

Differential Language Analysis

$$log\left(\frac{countA(\text{“horrible”})}{NA - countA(\text{“horrible”})}\right) - log\left(\frac{countB(\text{“horrible”})}{NB - countB(\text{“horrible”})}\right)$$

- Odds Ratio using Informative Dirichlet Prior

$$\delta_w^{(i-j)} = \log\left(\frac{f_w^i + \alpha_w}{n^i + \alpha_0 - (f_w^i + \alpha_w)}\right) - \log\left(\frac{f_w^j + \alpha_w}{n^j + \alpha_0 - (f_w^j + \alpha_w)}\right) \qquad (20.9)$$

(where n^i is the size of corpus i, n^j is the size of corpus j, f_w^i is the count of word w in corpus i, f_w^j is the count of word w in corpus j, α_0 is the size of the background corpus, and α_w is the count of word w in the background corpus.)

(Monroe et al., 2010; Jurafsky, 2017)

Differential Language Analysis

$$log\left(\frac{countA(\text{"horrible"})}{NA-countA(\text{"horrible"})}\right) - log\left(\frac{countB(\text{"horrible"})}{NB-countB(\text{"horrible"})}\right)$$

- Odds Ratio using Informative Dirichlet Prior

$$\delta_w^{(i-j)} = \log\left(\frac{f_w^i + \alpha_w}{n^i + \alpha_0 - (f_w^i + \alpha_w)}\right) - \log\left(\frac{f_w^j + \alpha_w}{n^j + \alpha_0 - (f_w^j + \alpha_w)}\right) \quad (20.9)$$

(where n^i is the size of corpus i, n^j is the size of corpus j, f_w^i is the count of word w in corpus i, f_w^j is the count of word w in corpus j, α_0 is the size of the background corpus, and α_w is the count of word w in the background corpus.)

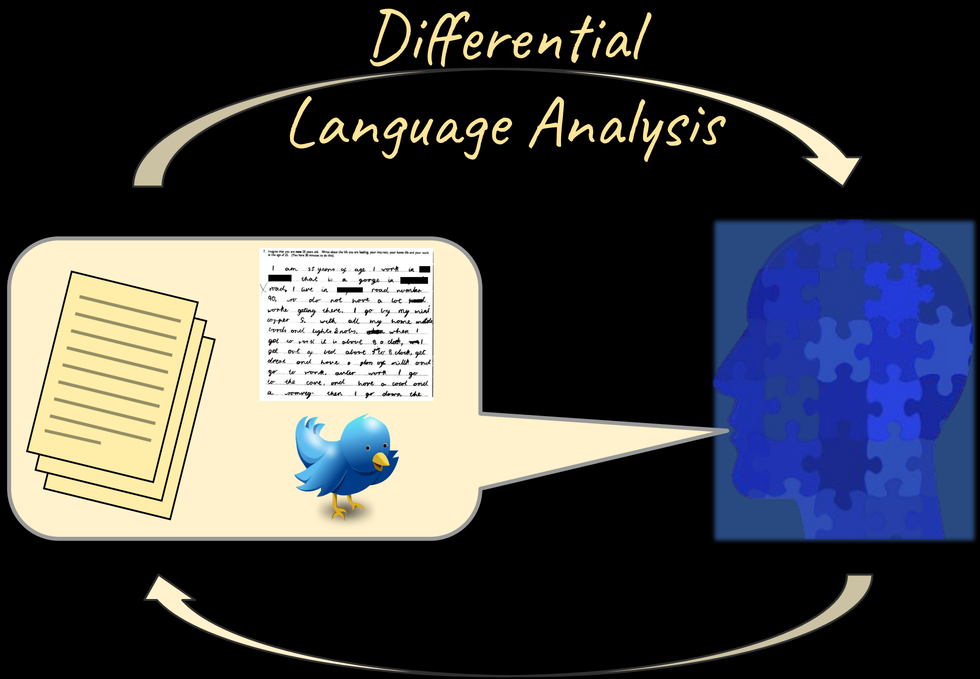
Final score is standardized (z-scored): $\dfrac{\hat{\delta}_w^{(i-j)}}{\sqrt{\sigma^2\left(\hat{\delta}_w^{(i-j)}\right)}}$, where

$$\sigma^2\left(\hat{\delta}_w^{(i-j)}\right) \approx \frac{1}{f_w^i + \alpha_w} + \frac{1}{f_w^j + \alpha_w}$$

(Monroe et al., 2010; Jurafsky, 2017)

DLATK

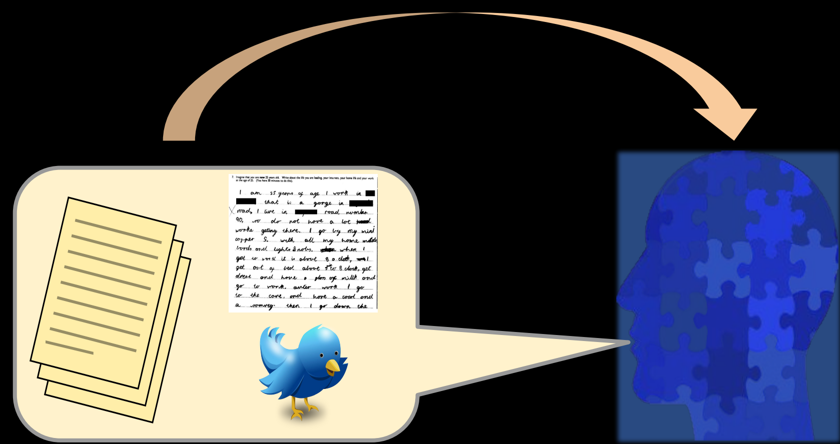
Python Library, CLI, and Colab for DLA



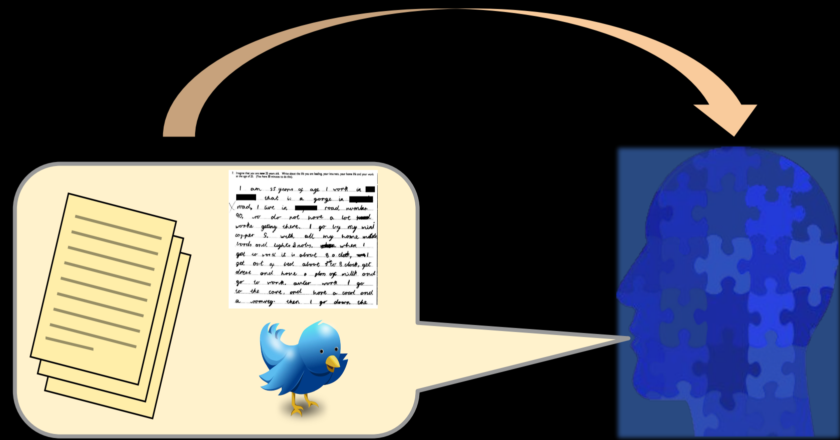
https://dlatk.github.io/

Getting Started in Colab

Natural language is generated by people.



Natural language is generated by people.



Shannon, 1948

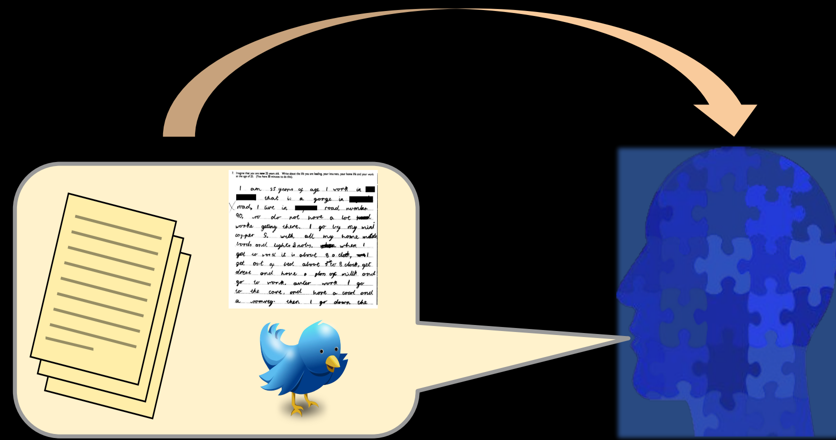
Mosteller & Wallace 1963

Clark & Schober, 1992

Mairesse, Walker, et al., 2007

Hovy & Soogaard, 2015

Natural language is generated by *people*.



"The common misconception is that language has got to do with words and what they mean. It does not. It has to do with people and what they mean."

Shannon, 1948

Mosteller & Wallace 1963

Clark & Schober, 1992

Mairesse, Walker, et al., 2007

Hovy & Soogaard, 2015

Human-Centered NLP – We will cover:

1. Differential Language Analysis
2. Human Factor Adaptation
3. Human Language Modeling

Human-Centered NLP – We will cover:

1. Differential Language Analysis
2. **Human Factor Adaptation**
3. Human Language Modeling

Approaches to Human Factor Inclusion

1. **Bias Mitigation:** Optimize so as not to pick up on unwanted relationships.

   (e.g. image captioner label pictures of men in kitchen as women)

Approaches to Human Factor Inclusion

1. **Bias Mitigation:** Optimize so as not to pick up on unwanted relationships.

   (e.g. image captioner label pictures of men in kitchen as women)

2. **Additive:** Include direct effect of human factor on outcome.

   (e.g. age and distinguishing PTSD from Depression; covariate in regression)

Approaches to Human Factor Inclusion

1. **Bias Mitigation:** Optimize so as not to pick up on unwanted relationships.
   (e.g. image captioner label pictures of men in kitchen as women)

2. **Additive:** Include direct effect of human factor on outcome.
   (e.g. age and distinguishing PTSD from Depression)

3. **Adaptive:** Allow meaning if language to change depending on human context. (also called “compositional”)
   (e.g. “sick” said from a young individual versus old individual)

Approaches to **Human Factor** Inclusion

1. **Bias Mitigation:** Opti… so as not to pick up on

   What are human “factors”?

   (e.g. image captioner label pictures of men in kitchen as women)

2. **Additive:** Include direct effect of human factor on outcome.
   (e.g. age and distinguishing PTSD from Depression)

3. **Adaptive:** Allow meaning if language to change depending on human context. (also called “compositional”)
   (e.g. “sick” said from a young individual versus old individual)

Human Factors

--- Any attribute, represented as a continuous or discrete variable, of the humans generating the natural language.

E.g.

- Gender
- Age
- Personality
- Ethnicity
- Socio-economic status

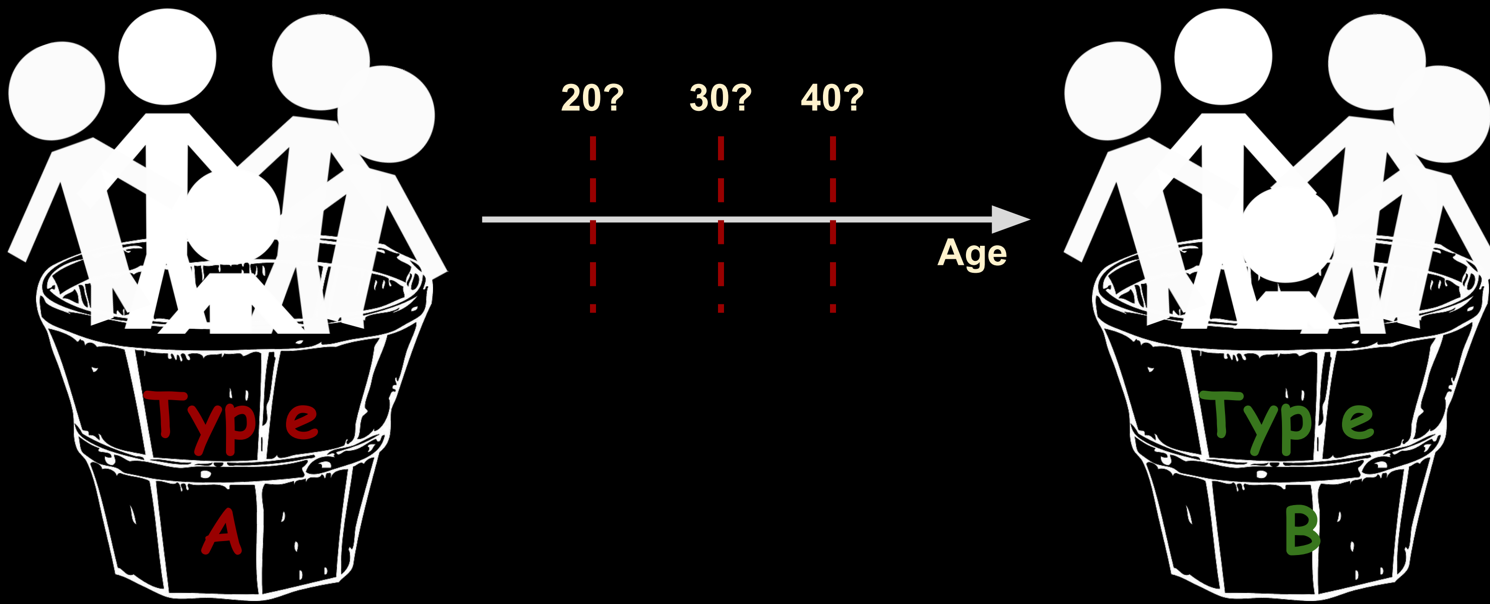
Human Factors

typically requires putting people into discrete bins *"most latent variables of interest to psychiatrists and personality and clinical psychologists are dimensional [continuous]"*

(Haslam et al., 2012)

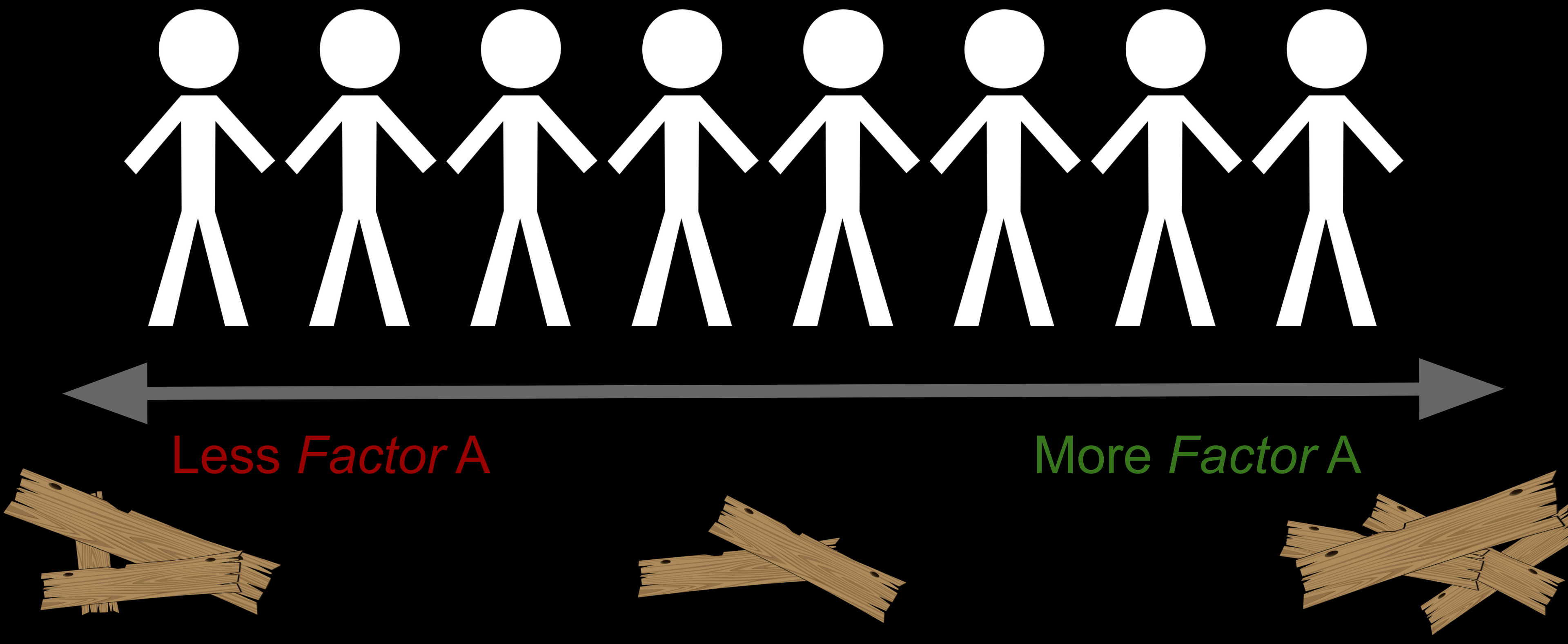
"most latent variables of interest to psychiatrists and personality and clinical psychologists are dimensional [continuous]"

(Haslam et al., 2012)



“most latent variables of interest to psychiatrists and personality and clinical psychologists are dimensional [continuous]”

(Haslam et al., 2012)



Adaptation Approach: Domain Adaptation

Features for: source target

$$\Phi^s(x) = \langle x, x, \mathbf{0} \rangle, \quad \Phi^t(x) = \langle x, \mathbf{0}, x \rangle$$

Frustratingly Easy Domain Adaptation

Hal Daumé III
School of Computing
University of Utah
Salt Lake City, Utah 84112
me@hal3.name

Abstract

We describe an approach to domain adaptation that is appropriate exactly in the case

supervised case. The fully supervised case models the following scenario. We have access to a large, annotated corpus of data fr

Adaptation Approach: Domain Adaptation

Features for: source | target

$$\Phi^s(\boldsymbol{x}) = \langle \boldsymbol{x}, \boldsymbol{x}, \mathbf{0} \rangle, \quad \Phi^t(\boldsymbol{x}) = \langle \boldsymbol{x}, \mathbf{0}, \boldsymbol{x} \rangle$$

```
newX = []
for all x in source_x:
  newX.append(x + x + [0]*len(x))
for all x in target_x:
  newX.append(x + [0]*len(x), x)
```

Frustratingly Easy Domain Adaptation

Hal Daumé III
School of Computing
University of Utah
Salt Lake City, Utah 84112
me@hal3.name

Abstract

We describe an approach to domain adapta-
tion that is appropriate exactly in the case

supervised case. The fully supervised case mod-
els the following scenario. We have access to a
large, annotated corpus of data from

Adaptation Approach: Domain Adaptation

Features for: source target

$$\Phi^s(\boldsymbol{x}) = \langle \boldsymbol{x}, \boldsymbol{x}, \mathbf{0} \rangle, \quad \Phi^t(\boldsymbol{x}) = \langle \boldsymbol{x}, \mathbf{0}, \boldsymbol{x} \rangle$$

```
newX = []
for all x in source_x:
  newX.append(x + x + [0]*len(x))
for all x in target_x
  newX.append(x + [0]*len(x), x)

newY = source_y + target_y

model = model.train(newX,newY)
```

Adaptation Approach: Factor Adaptation

Human Centered NLP with User-Factor Adaptation

Veronica E. Lynn, Youngseo Son, Vivek Kulkarni
Niranjan Balasubramanian and **H. Andrew Schwartz**
Stony Brook University
Stony Brook, NY
{velynn, yson, vvkulkarni, niranjan, has}@cs.stonybrook.edu

Abstract

We pose the general task of *user-factor adaptation* — adapting supervised learning models to real-valued user factors inferred from a background of their lan-

and Costa Jr., 1989; Ruscio and Ruscio, 2000; Widiger and Samuel, 2005).

Here, we ask how one can adapt NLP models to real-valued human *factors* – continuous valued attributes that capture fine-grained differences be-

Residualized Factor Adaptation for Community Social Media Prediction Tasks

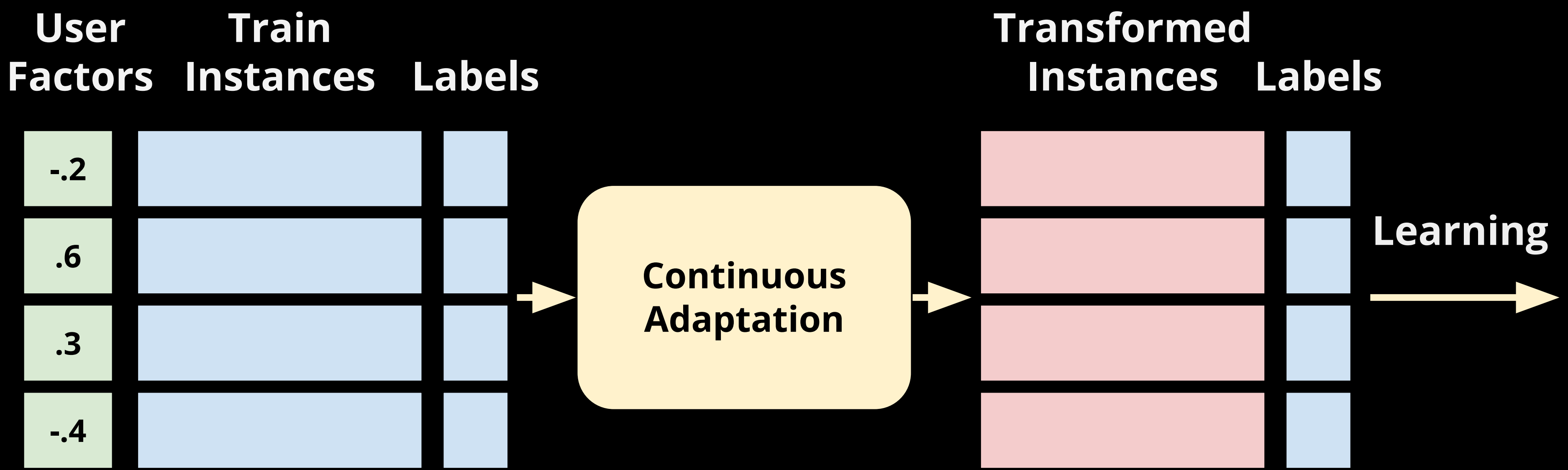
Mohammadzaman Zamani,[1] **H. Andrew Schwartz,**[1] **Veronica E. Lynn,**[1]
Salvatore Giorgi,[2] and **Niranjan Balasubramanian**[1]
[1] Computer Science Department, Stony Brook University
[2]Department of Psychology, University of Pennsylvania
mzamani@cs.stonybrook.edu

Abstract

Predictive models over social media language ... promise in capturing community

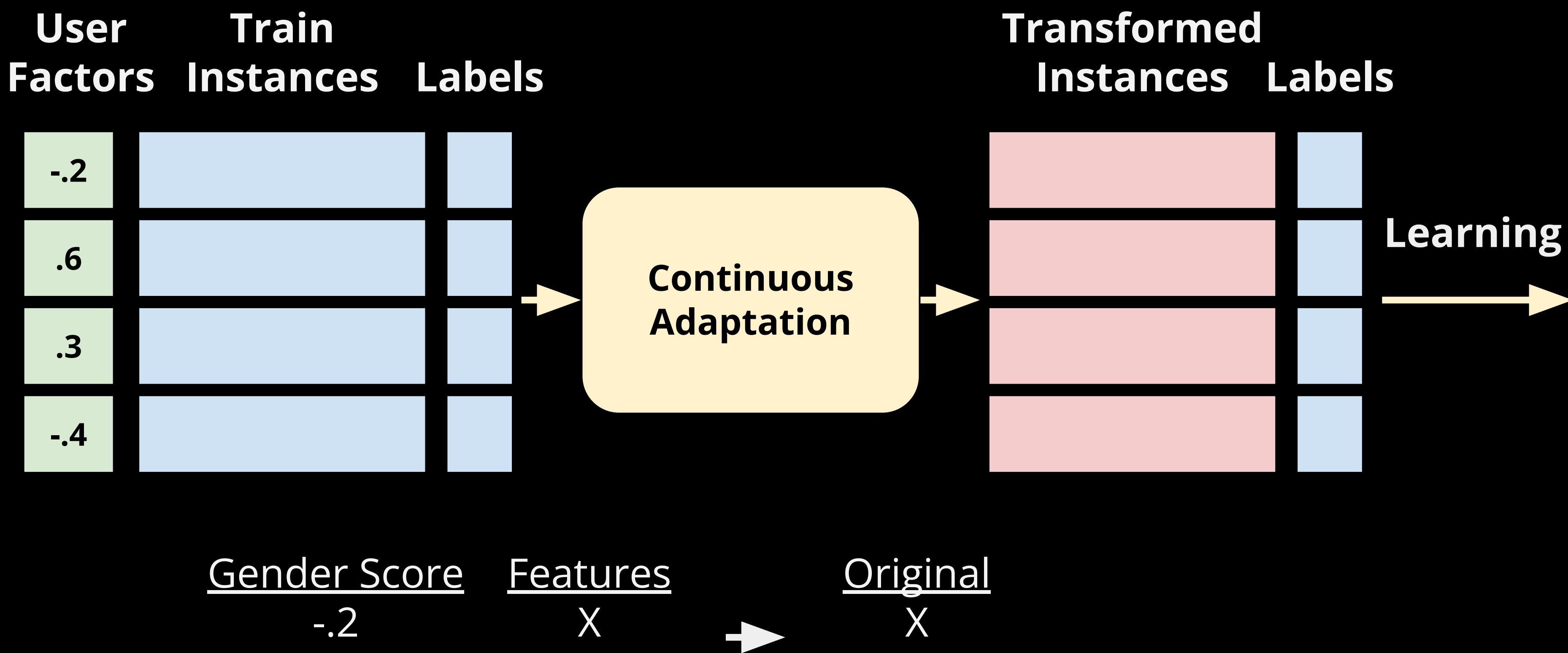
linked to socio-demographic factors (age, gender, race, education, income levels) with many social scientific studies supporting their predictive

Our Method: Continuous Adaptation



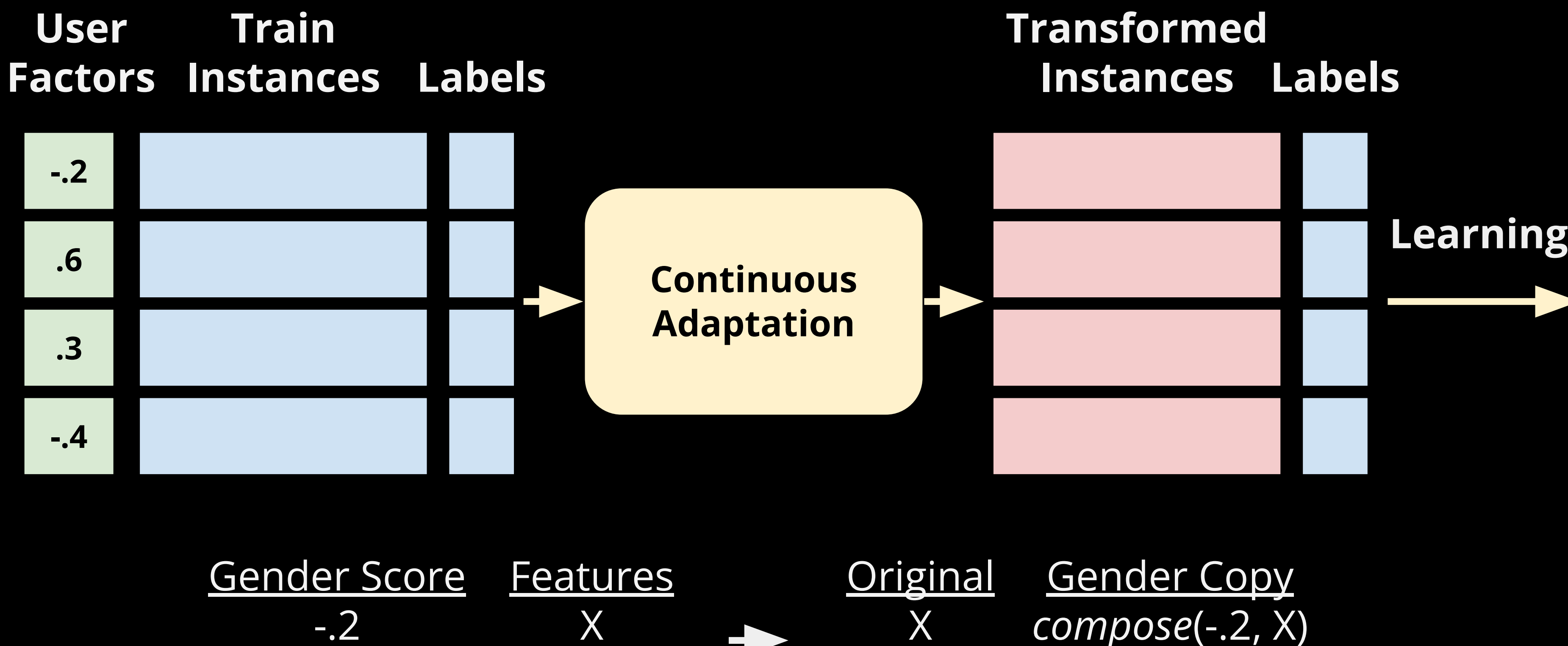
(Lynn et al., 2017)

Our Method: Continuous Adaptation



(Lynn et al., 2017)

Our Method: Continuous Adaptation



(Lynn et al., 2017)

User Factor Adaptation: Handling multiple factors

Replicate features for each factor:

A compositional function c combines d user factor scores $f_{u,d}$ with original feature values $\mathbf{x}$:

$$\Phi(\mathbf{x}, u) = \langle \mathbf{x}, c(f_{u,1}, \mathbf{x}), c(f_{u,2}, \mathbf{x}), \cdots, c(f_{u,d}, \mathbf{x}) \rangle$$

(Lynn et al., 2017)

User Factor Adaptation: Handling multiple factors

Replicate features for each factor:

A compositional function c combines d user factor scores $f_{u,d}$ with original feature values $\mathbf{x}$:

$$\Phi(\mathbf{x}, u) = \langle \mathbf{x}, c(f_{u,1}, \mathbf{x}), c(f_{u,2}, \mathbf{x}), \cdots, c(f_{u,d}, \mathbf{x}) \rangle$$

User	Factor Classes	Augmented Instance $\Phi(\mathbf{x}, u)$
User 1	F_1	$\langle \mathbf{x}, \mathbf{x}, \mathbf{0}, \mathbf{0}, \cdots, 0 \rangle$
User 2	F_2	$\langle \mathbf{x}, \mathbf{0}, \mathbf{x}, \mathbf{0}, \cdots, 0 \rangle$
User 3	F_1, F_3	$\langle \mathbf{x}, \mathbf{x}, \mathbf{0}, \mathbf{x}, \cdots, 0 \rangle$
User 4	F_k	$\langle \mathbf{x}, \mathbf{0}, \mathbf{0}, \cdots, 0, \mathrm{x} \rangle$

Table 1: Discrete Factor Adaptation: Augmentations of an original instance vector $\mathbf{x}$ under different factor class mappings. With k domains the augmented feature vector is of length $n(k+1)$.

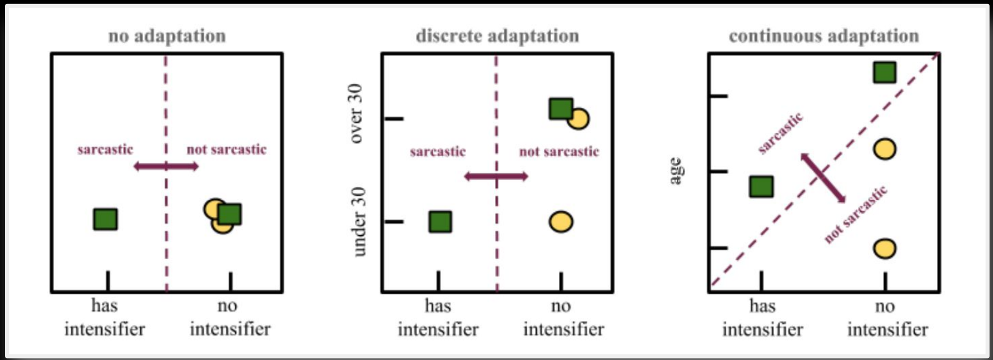
(Lynn et al., 2017)

User Factor Adaptation: Handling multiple factors

Replicate features for each factor:

A compositional function c combines d user factor scores $f_{u,d}$ with original feature values $\mathbf{x}$:

$$\Phi(\mathbf{x}, u) = \langle \mathbf{x}, c(f_{u,1}, \mathbf{x}), c(f_{u,2}, \mathbf{x}), \cdots, c(f_{u,d}, \mathbf{x}) \rangle$$



User	Factor Classes	Augmented Instance $\Phi(\mathbf{x}, u)$
User 1	F_1	$\langle \mathbf{x}, \mathbf{x}, \mathbf{0}, \mathbf{0}, \cdots, \mathbf{0} \rangle$
User 2	F_2	$\langle \mathbf{x}, \mathbf{0}, \mathbf{x}, \mathbf{0}, \cdots, \mathbf{0} \rangle$
User 3	F_1, F_3	$\langle \mathbf{x}, \mathbf{x}, \mathbf{0}, \mathbf{x}, \cdots, \mathbf{0} \rangle$
User 4	F_k	$\langle \mathbf{x}, \mathbf{0}, \mathbf{0}, \cdots, \mathbf{0}, \mathrm{x} \rangle$

Table 1: Discrete Factor Adaptation: Augmentations of an original instance vector $\mathbf{x}$ under different factor class mappings. With k domains the augmented feature vector is of length $n(k+1)$.

(Lynn et al., 2017)

Main Results

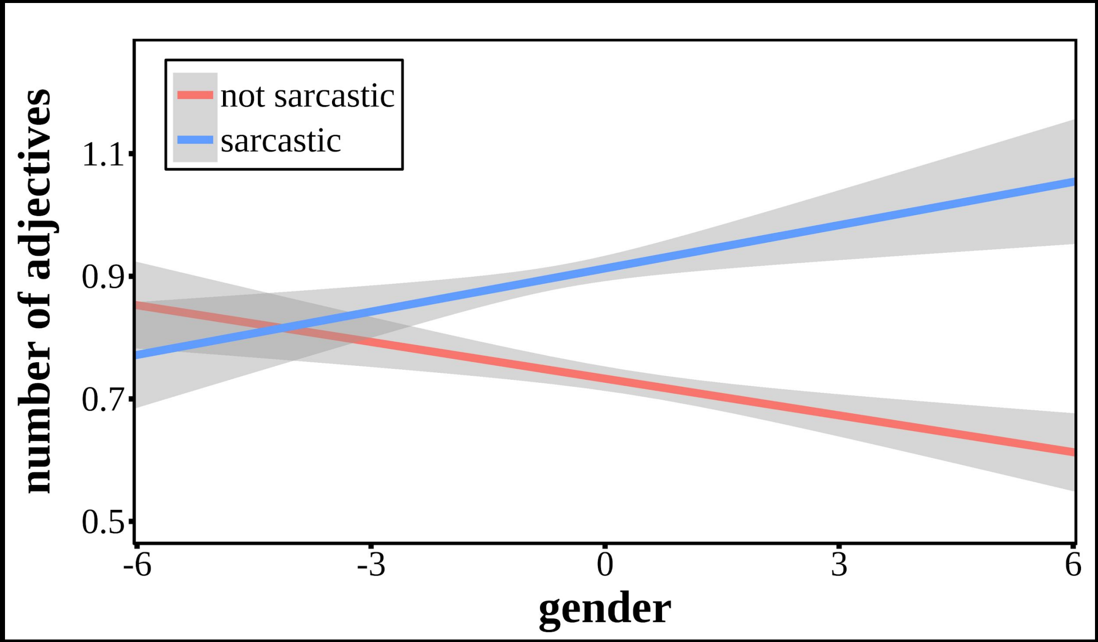
Adaptation improves over unadapted baselines (Lynn et al., 2017)

Task	**Metric**	**No Adaptation**	**Gender**	**Personality**	**Latent (User Embed)**
Stance	F1	64.9	**65.1 (+0.2)**	**66.3 (+1.4)**	**67.9 (+3.0)**
Sarcasm	F1	73.9	**75.1 (+1.2)**	**75.6 (+1.7)**	**77.3 (+3.4)**
Sentiment	Acc.	60.6	**61.0 (+0.4)**	**61.2 (+0.6)**	**60.7 (+0.1)**
PP-Attach	Acc.	71.0	70.7 (-0.3)	70.2 (-0.8)	70.8 (-0.2)
POS	Acc.	91.7	**91.9 (+0.2)**	91.2 (-0.5)	90.9 (-0.8)

Example: How Adaptation Helps

<u>Women</u>
more adjectives→sarcasm

<u>Men</u>
more adjectives→no sarcasm



more "male" **more "female"**

Problem

User factors are not always available.

Solution: User Factor Inference

<u>**past tweets**</u>

inferred factors

<u>**Known**</u>

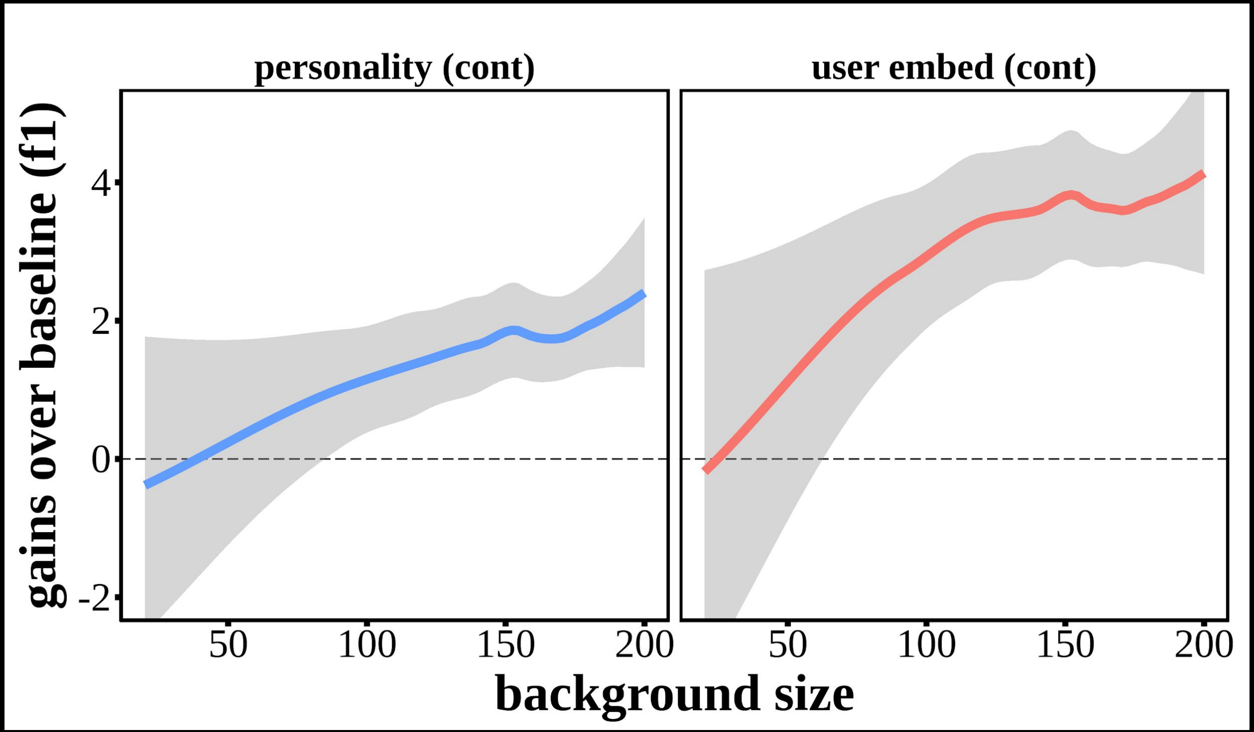
Age (Sap et al. 2014)
Gender (Sap et al. 2014)
Personality (Park et al. 2015)

<u>**Latent**</u>

User Embeddings
(Kulkarni et al. 2017)
Word2Vec
TF-IDF

Background Size

Using more background tweets to infer factors produces larger gains

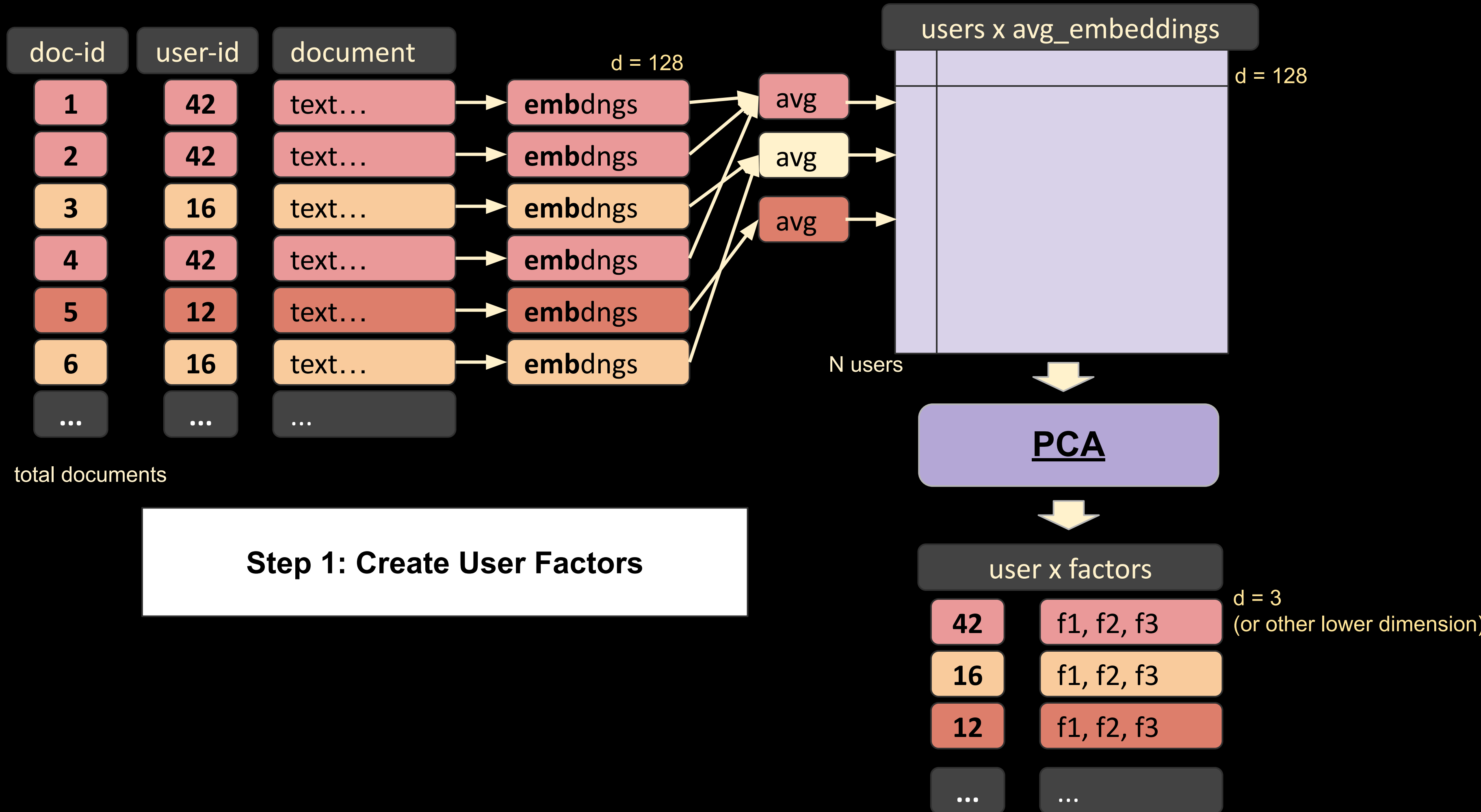


Full User Factors Adaptation Pipeline: with latent factors from training

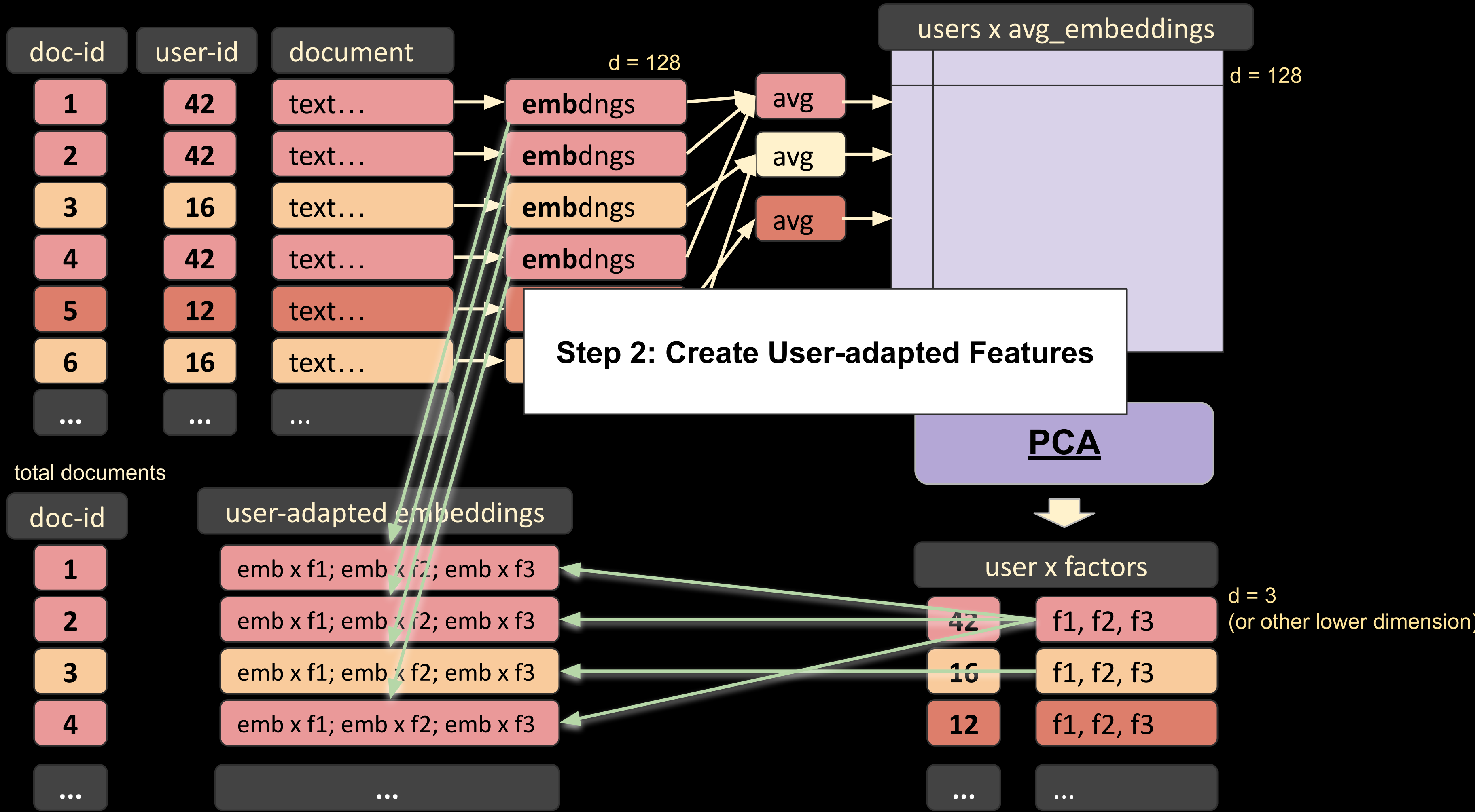
doc-id	user-id	document	d = 128
1	42	text…	embdngs
2	42	text…	embdngs
3	16	text…	embdngs
4	42	text…	embdngs
5	12	text…	embdngs
6	16	text…	embdngs
…	…	…	

total documents

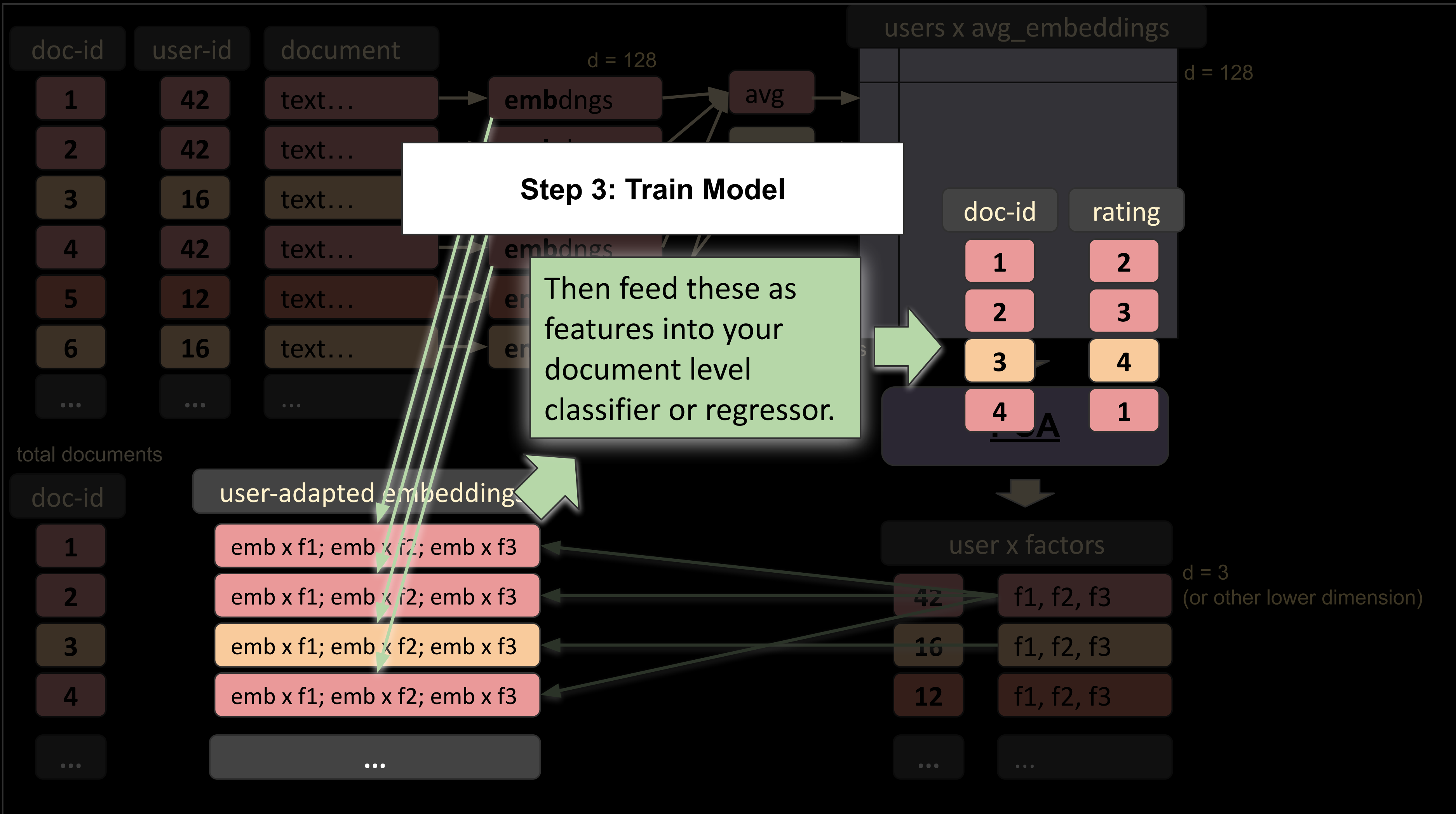
Full User Factors Adaptation Pipeline: with latent factors from training



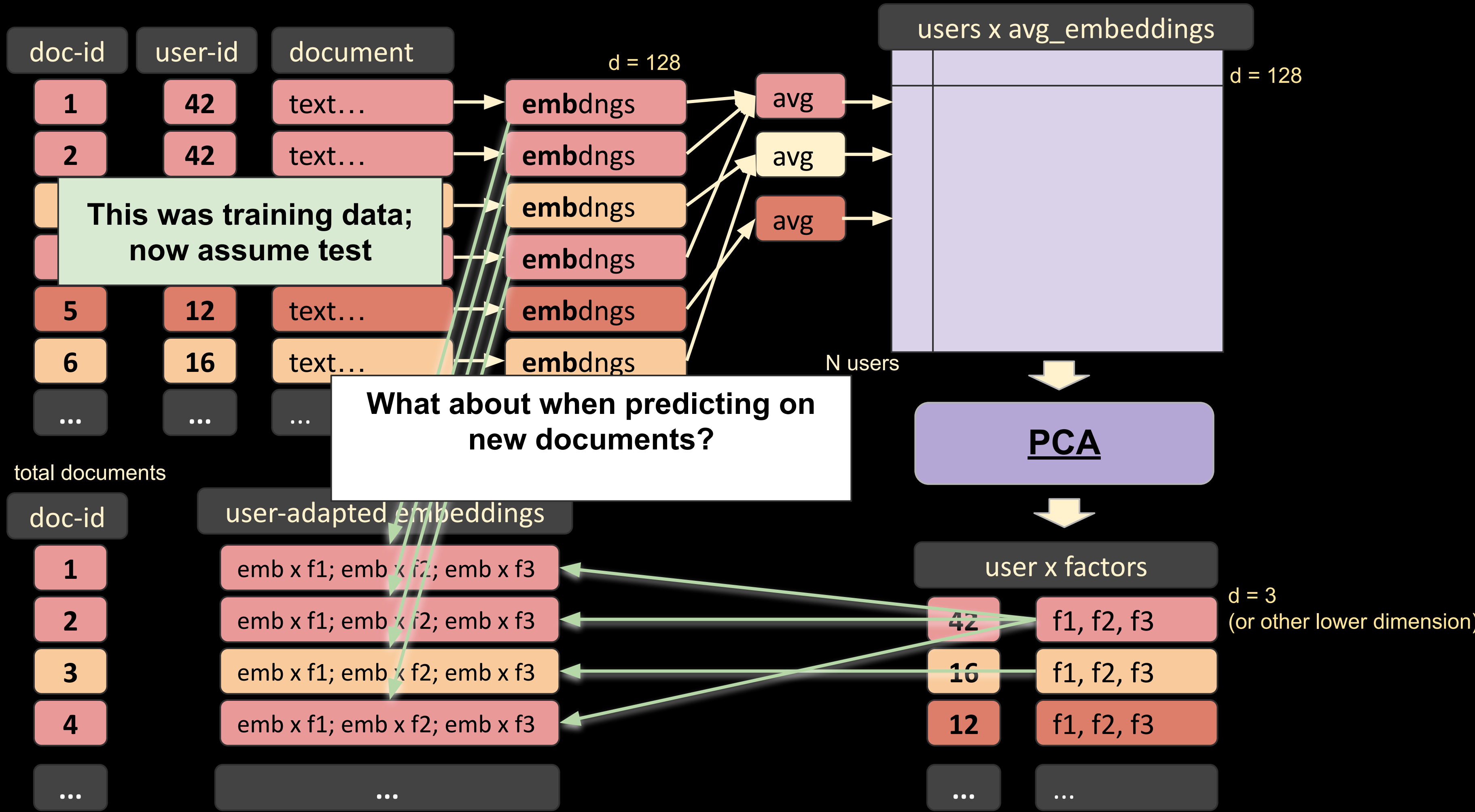
Full User Factors Adaptation Pipeline: with latent factors from training



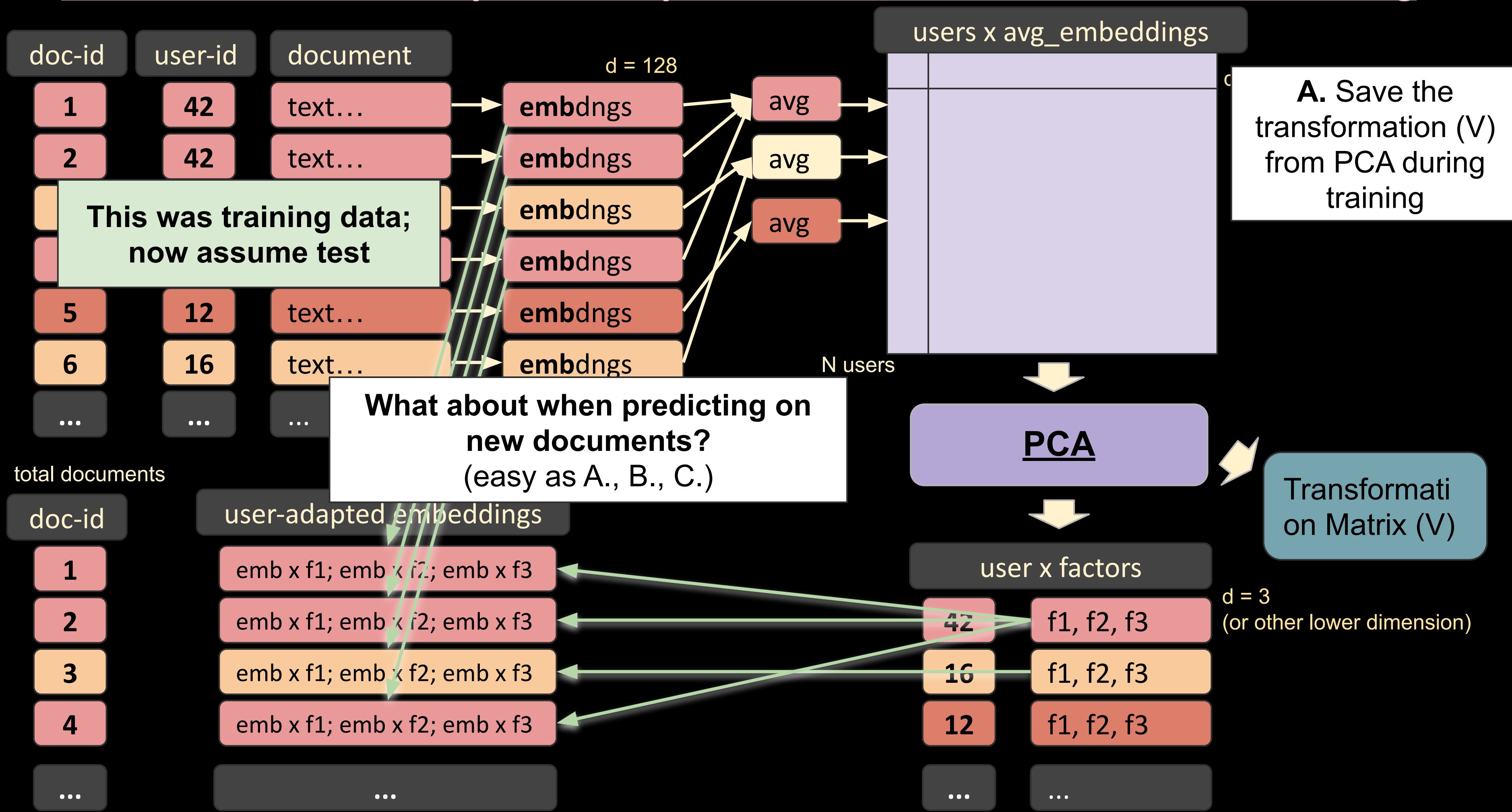
Full User Factors Adaptation Pipeline: with latent factors from training



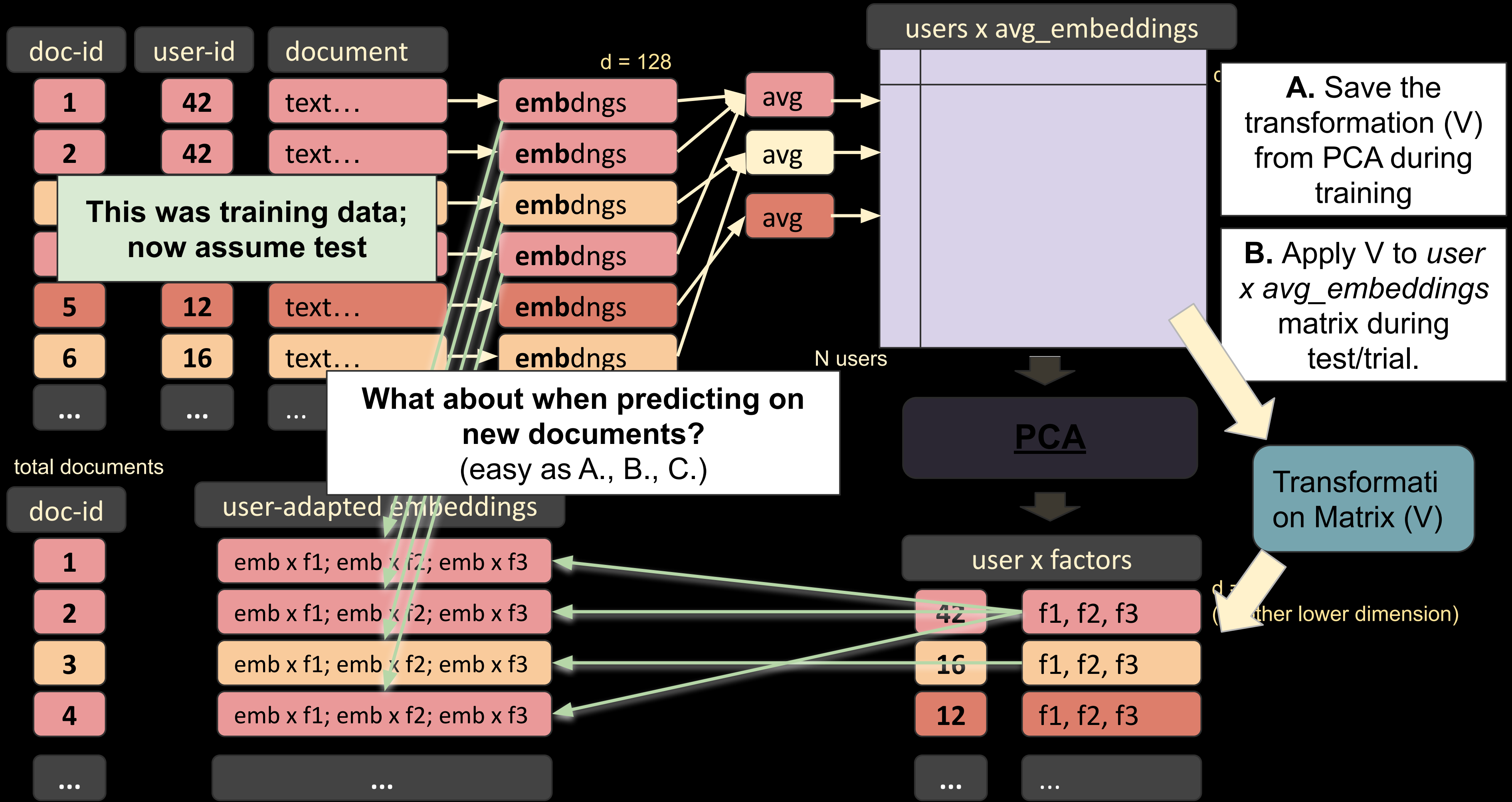
Full User Factors Adaptation Pipeline: with latent factors from training



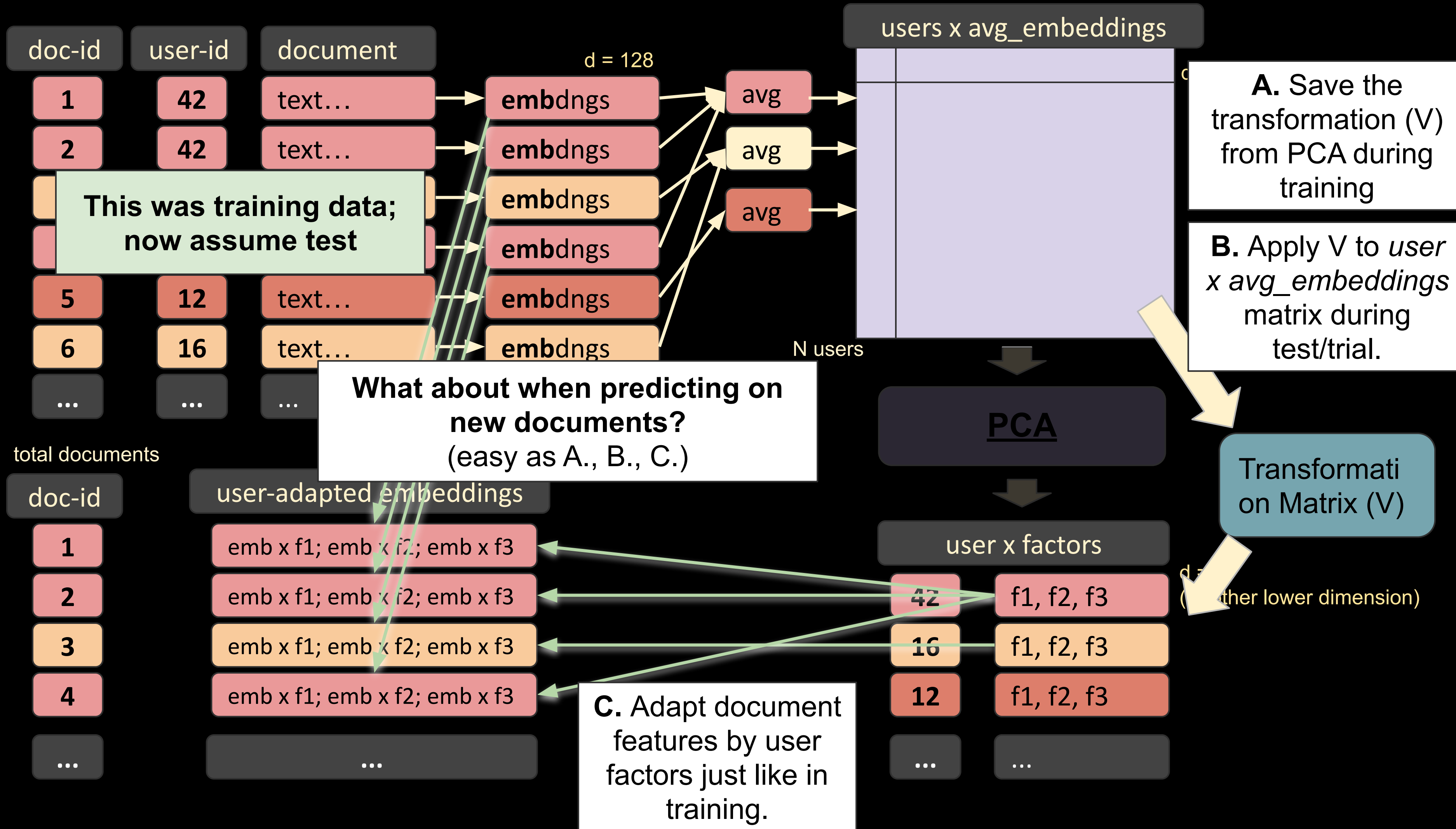
Full User Factors Adaptation Pipeline: with latent factors from training



Full User Factors Adaptation Pipeline: with latent factors from training



Full User Factors Adaptation Pipeline: with latent factors from training



Approaches to Human Factor Inclusion

1. **Bias Mitigation:** Optimize so as not to pick up on unwanted relationships.

   (e.g. image captioner label pictures of men in kitchen as women)

2. **Additive:** Include direct effect of human factor on outcome.

   (e.g. age and distinguishing PTSD from Depression)

3. **Adaptive:** Allow meaning if language to change depending on human context. (also called “compositional”)

   ?

   (e.g. “sick” said from a young individual versus old individual)

Approaches to Human Factor Inclusion

1. **Bias Mitigation:** Optimize so as not to pick up on unwanted relationships.
   (e.g. image captioner label pictures of men in kitchen as women)

2. **Additive:** Include direct effect of human factor on outcome.
   (e.g. age and distinguishing PTSD from Depression)

3. **Adaptive:** Allow meaning if language to change depending on human context. (also called “compositional”)
   (e.g. “sick” said from a young individual versus old individual)

Human-Centered NLP – We will cover:

1. Differential Language Analysis
2. **Human Factor Adaptation**
3. Human Language Modeling

Human-Centered NLP – We will cover:

1. Differential Language Analysis
2. Human Factor Adaptation
3. **Human Language Modeling**

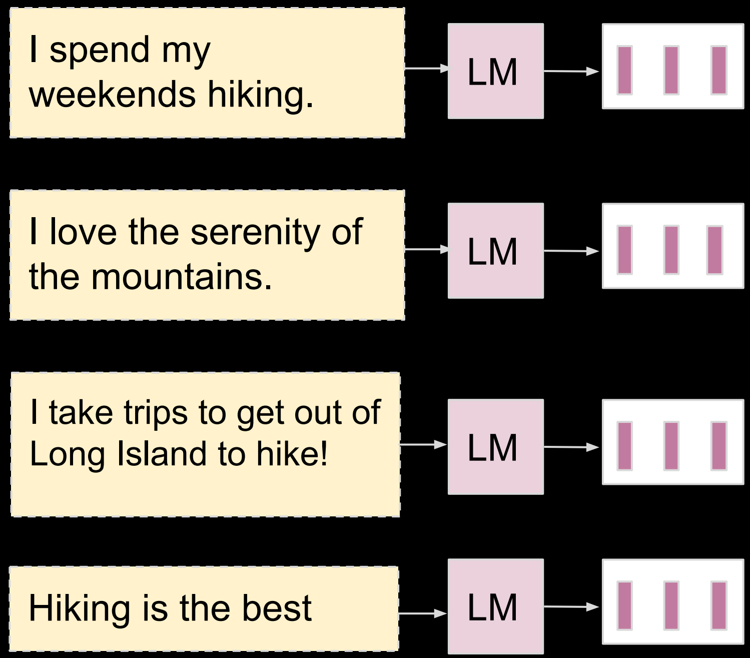
Language Modeling

probability of a token sequence

$$Pr(\mathbf{W}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1})$$

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.**
In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Language Modeling



Language Modeling: What's Missing?

1. Addressing *Ecological Fallacy:* Treating dependent phenomena as if independent. (Piantadosi et al., 1988; Steel and Holt, 1996)
2. Modeling the higher order structure.

Language Modeling

probability of a token sequence

$$Pr(\mathbf{W}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1})$$

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.**
In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Human Language Modeling (HuLM)

LM - probability of a token sequence

$$Pr(\mathbf{W}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1})$$

HuLM - probability of a token sequence,
in the context of the human that generated it.

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.**
In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Human Language Modeling (HuLM)

LM

- probability of a token sequence

$$Pr(\mathbf{W}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1})$$

$$Pr(\mathbf{W}|\mathbf{U}_{static}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1}, \mathbf{U}_{static})$$

HuLM

- probability of a token sequence, in the context of the human that generated it.

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Human Language Modeling (HuLM)

LM

- probability of a token sequence

$$Pr(\mathbf{W}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1})$$

$$Pr(\mathbf{W}|\mathbf{U}_{static}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1}, \mathbf{U}_{static})$$

static user representation

HuLM

- probability of a token sequence, in the context of the human that generated it.

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Human Language Modeling (HuLM)

LM

- probability of a token sequence

$$Pr(\mathbf{W}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1})$$

$$Pr(\mathbf{W}|\mathbf{U}_{static}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1}, \mathbf{U}_{static})$$

static user representation

HuLM

$$Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i} | w_{t,1:i-1}, \mathbf{U}_{1:t-1})$$

- probability of a token sequence, in the context of the human that generated it.

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.**
In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Human Language Modeling (HuLM)

LM - probability of a token sequence

$$Pr(\mathbf{W}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1})$$

$$Pr(\mathbf{W} | \mathbf{U}_{static}) = \prod_{i=1}^{n} Pr(w_i | w_{1:i-1}, \mathbf{U}_{static})$$

static user representation

HuLM

$$Pr(\mathbf{W}_t | \mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i} | w_{t,1:i-1}, \mathbf{U}_{1:t-1})$$

"user state" representation

- probability of a token sequence, in the context of the human that generated it.

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

User State Representation, U

$$\boxed{Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i}|w_{t,1:i-1}, \mathbf{U}_{1:t-1})}$$

no history ⟷ all data

$U_{1:t-1} = \varnothing$

(reduces to a standard LM: $Pr(w_i|w_{1:i-1})$)

- *doesn't capture the person*

$U_{1:t-1} = w_{1,1:n_1}, w_{2,1:n_2}, \ldots, w_{t-1,1:n_{t-1}}$

(all previous docs and tokens by the person)

- *huge*
- *no generalizations*

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

User State Representation, U

$$\boxed{Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i}|w_{t,1:i-1}, \mathbf{U}_{1:t-1})}$$

no history ⟷ all data

$U_{1:t-1} = \varnothing$

(reduces to a standard LM: $Pr(w_i|w_{1:i-1})$)

- *doesn't capture the person*

history of user states

$U_{1:t-1} = w_{1,1:n_1}, w_{2,1:n_2}, \ldots, w_{t-1,1:n_{t-1}}$

(all previous docs and tokens by the person)

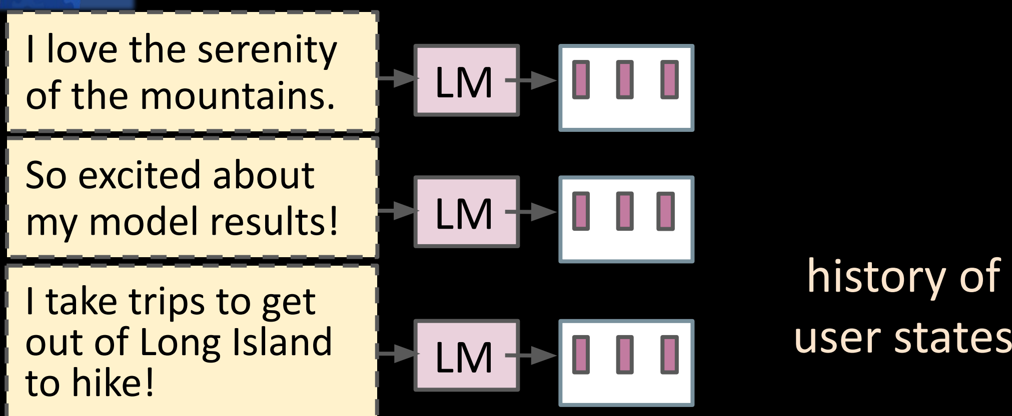
- *huge*
- *no generalizations*

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

User State Representation, U

$$\boxed{Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i}|w_{t,1:i-1}, \mathbf{U}_{1:t-1})}$$

State and Trait Theory from Psychology: ***Traits*** – the stable characteristics of "who someone is" – define a distribution of potential ***states*** of being that moderate human behavior (i.e. language).

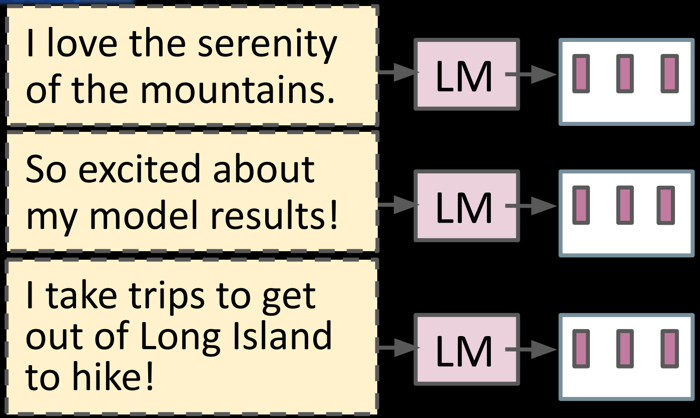


Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

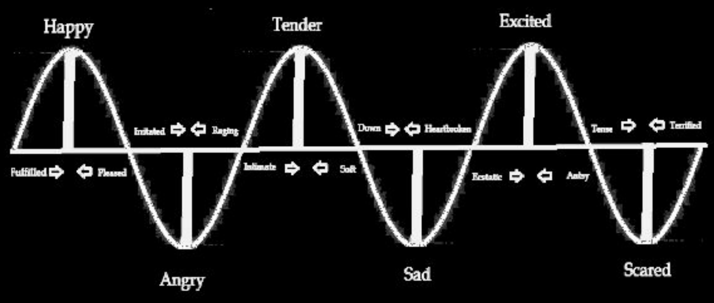
User State Representation, U

$$\boxed{Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i}|w_{t,1:i-1}, \mathbf{U}_{1:t-1})}$$

State and Trait Theory from Psychology: ***Traits*** – the stable characteristics of "who someone is" – define a distribution of potential ***states*** of being that moderate human behavior (i.e. language).



$U_{1:t-1}$ = [a sequence of *states*]

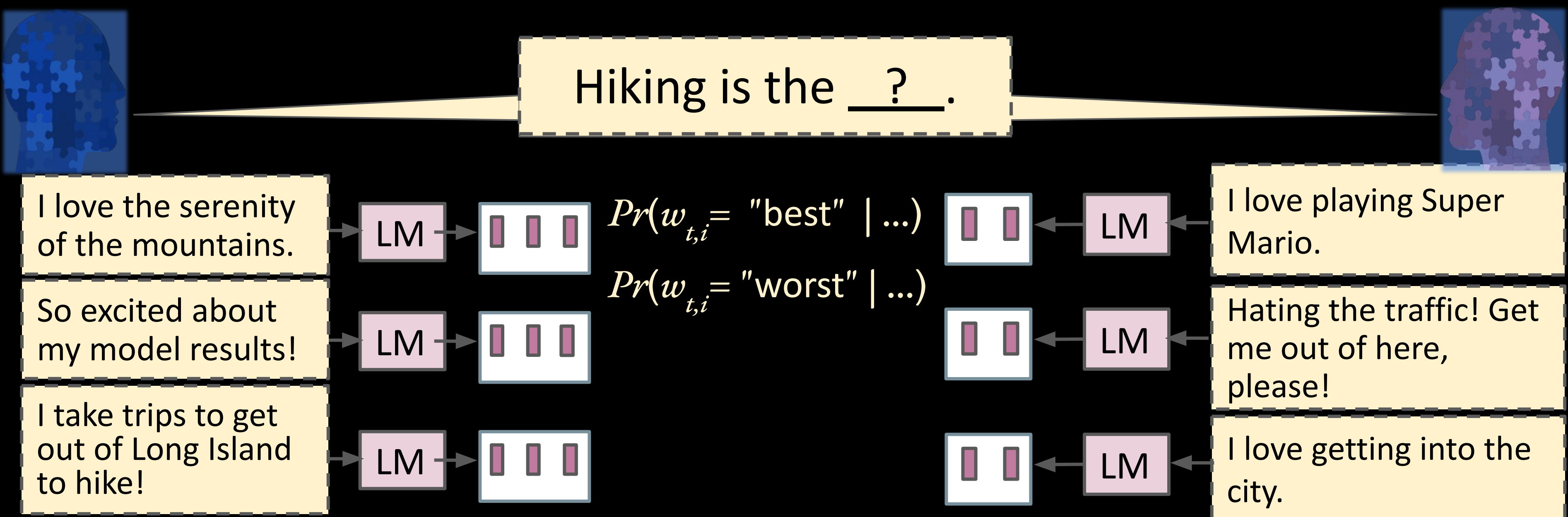


(Washington Outsider, 2014)

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

User State Representation, U

$$Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i}|w_{t,1:i-1}, \mathbf{U}_{1:t-1})$$



Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

User State Representation: Motivation

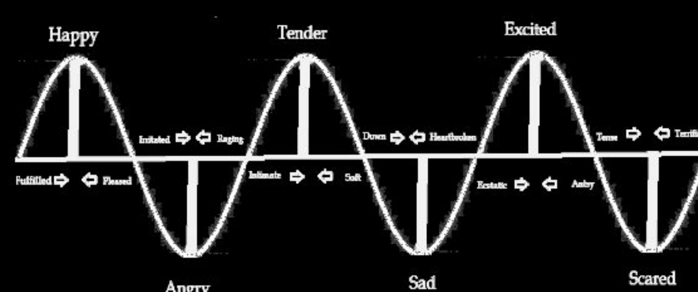
- **Addressing *Ecological Fallacy:*** Treating dependent phenomena (i.e. sequences from the same person) as if independent. (Piantadosi et al., 1988; Steel and Holt, 1996)
- **Modeling the higher order structure.**
- Building on **ideas from human factor inclusion**/adaptation (Lynn et al., 2017; Huang & Paul, 2019; Hovy & Yang, 2021) and **personalized modeling.** (King & Cook, 2020; Jaech & Ostendorf, 2018)

I love the serenity of the mountains. — U_j

So excited about my model results! — U_{j+1}

I take trips to get out of Long Island to hike! — U_{j+2}

$U_{1:t-1}$ = [a sequence of *states*]



(Washington Outsider, 2014)

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Human Language Modeling (HuLM)

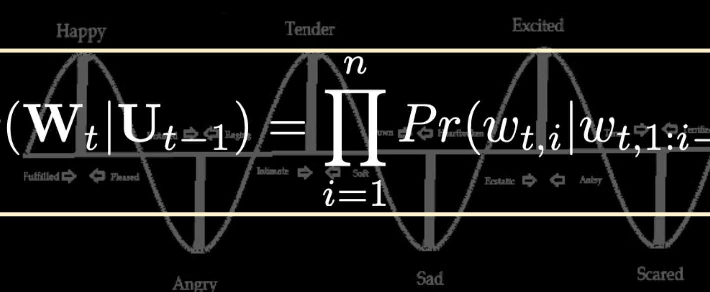
- **Addressing *Ecological Fallacy:*** Treating dependent phenomena (i.e. sequences from the same person) as if independent. (Piantadosi et al., 1988; Steel and Holt, 1996)
- **Modeling the higher order structure.**
- Building on **ideas from human factor inclusion**/adaptation (Lynn et al., 2017; Huang & Paul, 2019; Hovy & Yang, 2021) and **personalized modeling.** (King & Cook, 2020; Jaech & Ostendorf, 2018)

I love the serenity of the mountains. — U_j

So excited about my model results! — U_{j+1}

I take trips to get out of Long Island to hike! — U_{j+2}

$U_{1:t-1}$ = [a sequence of *states*]



$$Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i}|w_{t,1:i-1}, \mathbf{U}_{1:t-1})$$

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Human Language Modeling (HuLM)

Goal: Language modeling as a task grounded in the "natural" generators of language, people.

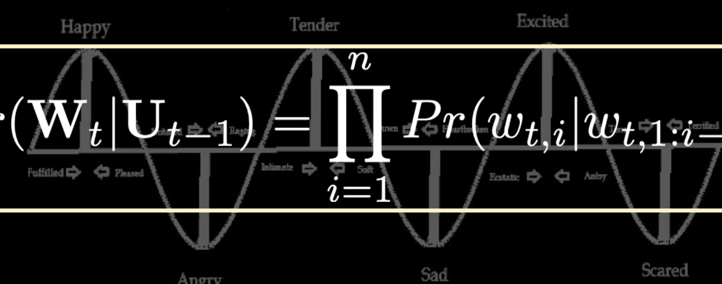
The *HuLM* task definition: Estimate the probability of a sequence of tokens, $w_{t,1:i}$, conditioned on a higher-order representation, U_t, constituting the human state of being just before the sequence generation.

I love the serenity of the mountains. — U_j

So excited about my model results! — U_{j+1}

I take trips to get out of Long Island to hike! — U_{j+2}

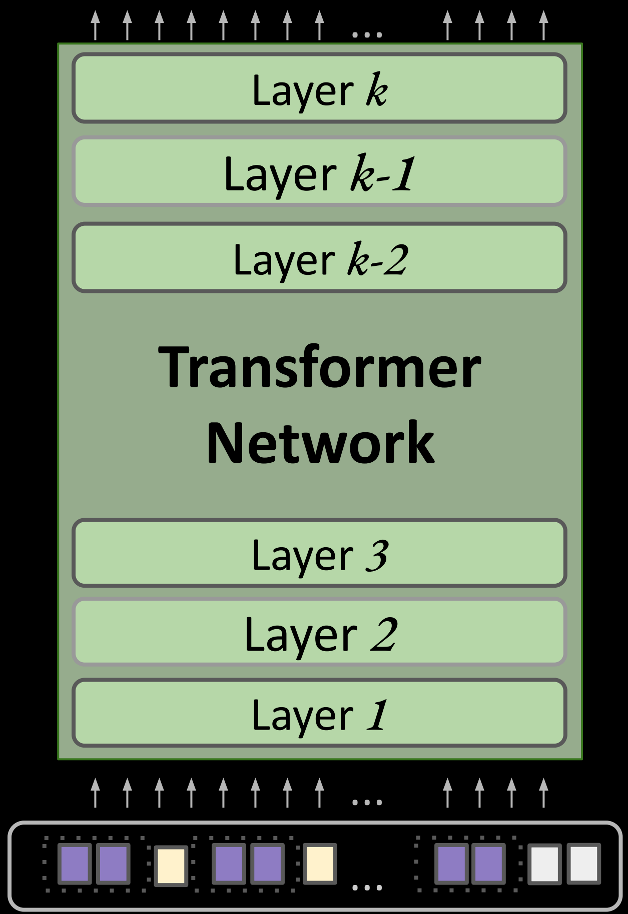
$U_{1:t-1}$ = [a sequence of *states*]



$$Pr(\mathbf{W}_t|\mathbf{U}_{t-1}) = \prod_{i=1}^{n} Pr(w_{t,i}|w_{t,1:i-1}, \mathbf{U}_{1:t-1})$$

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

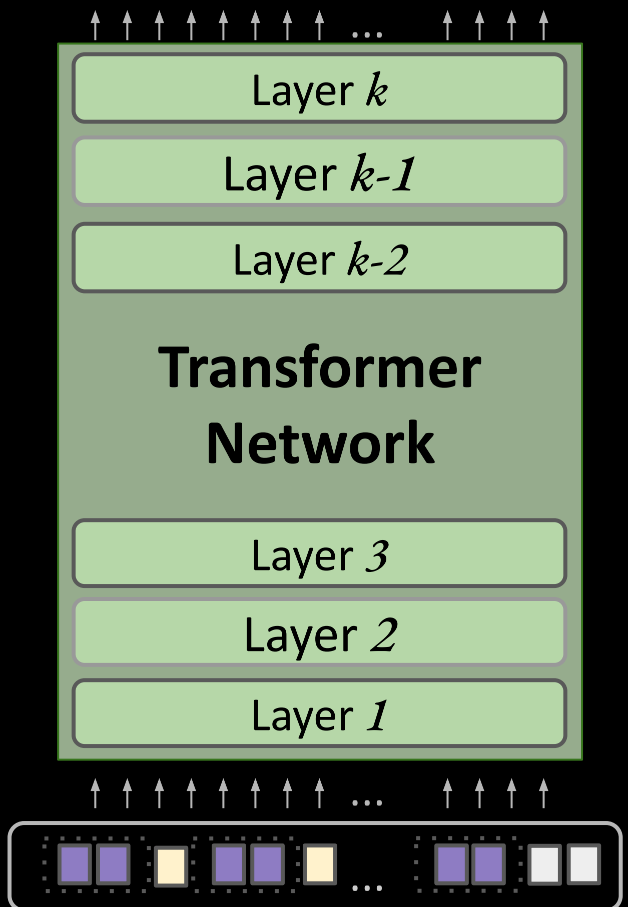
How to Adapt the Transformer?



Input: A Block of Temporally Ordered User Messages

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

How to Adapt the Transformer?

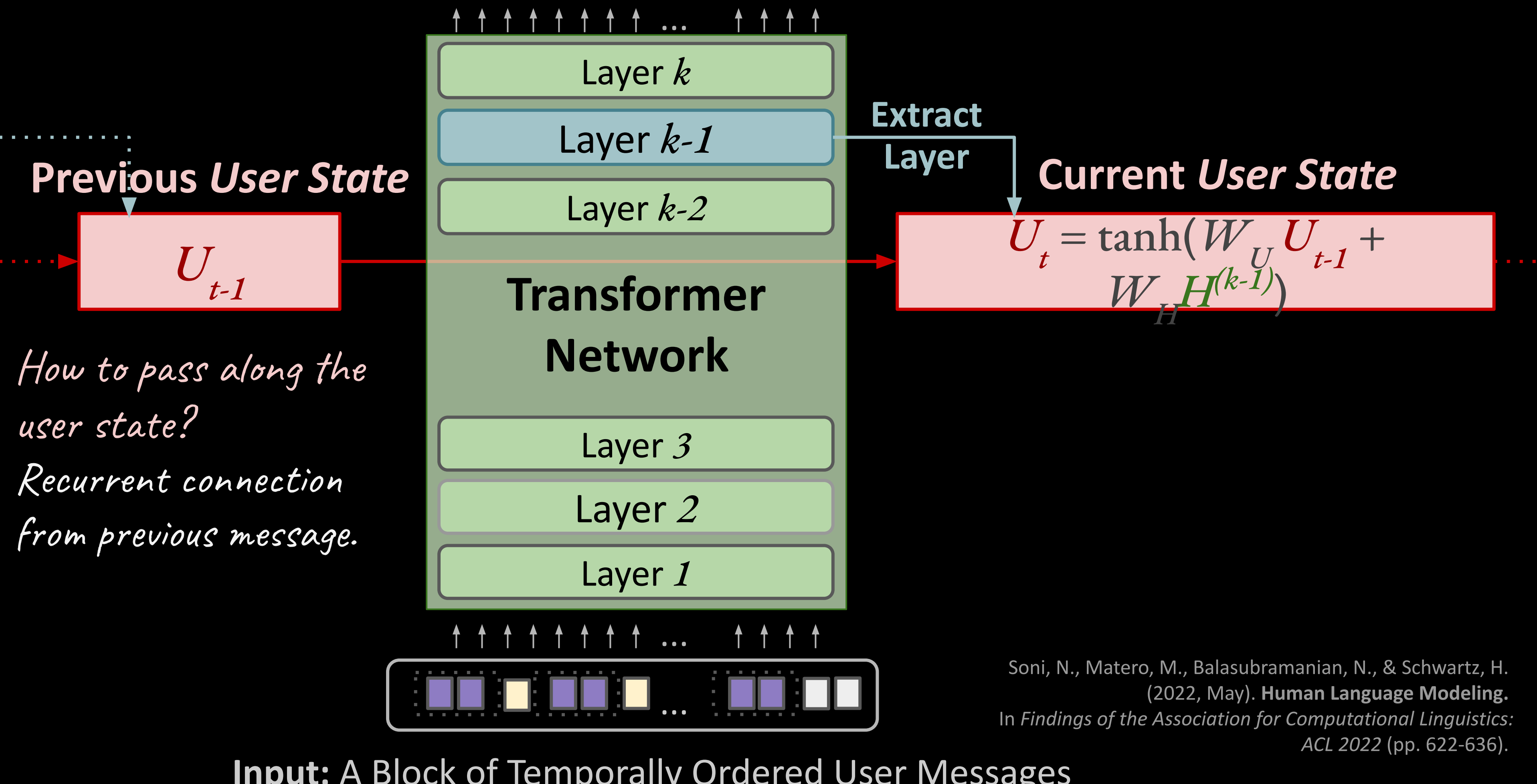


How to pass along the user state?

Input: A Block of Temporally Ordered User Messages

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

How to Adapt the Transformer?

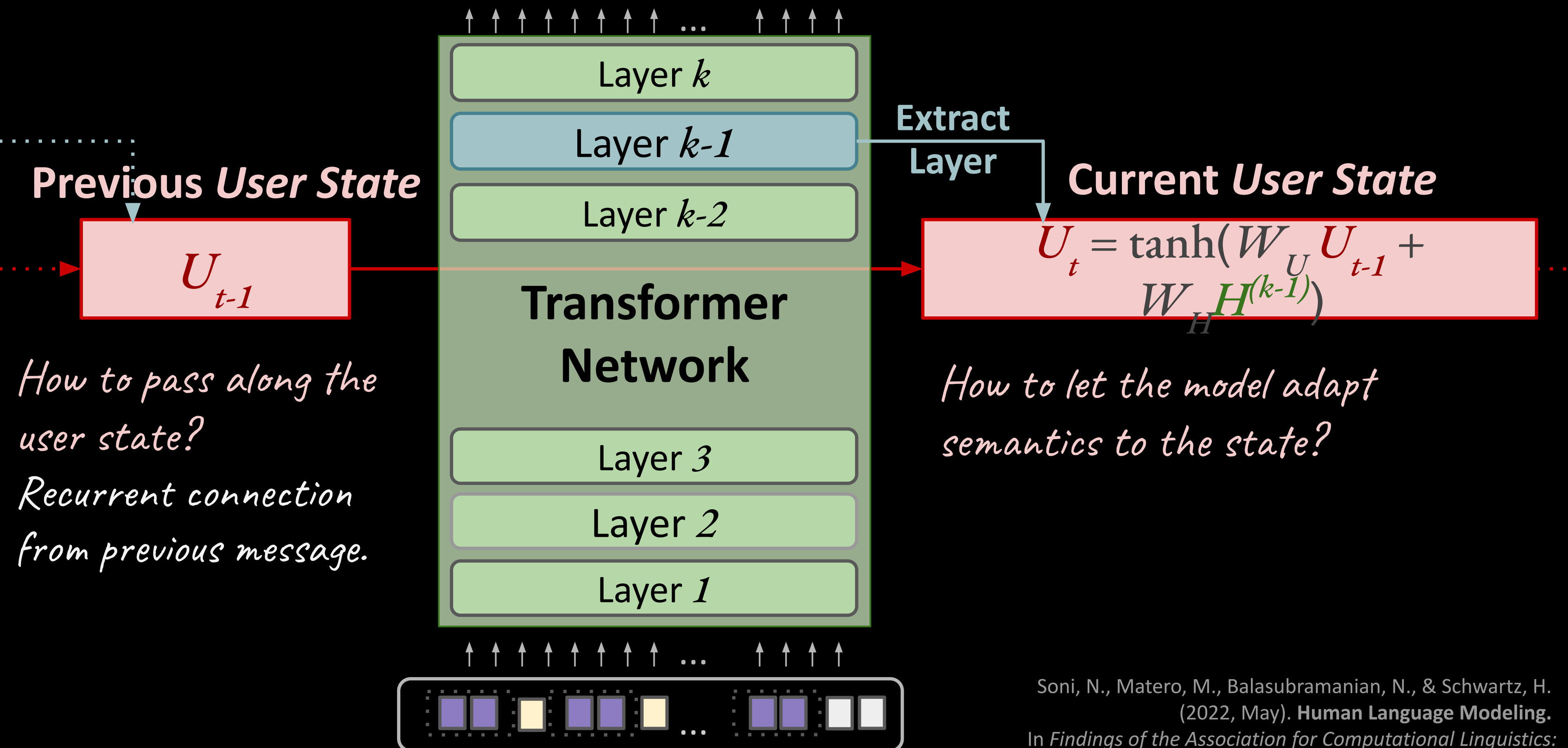


How to pass along the user state?

Recurrent connection from previous message.

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

How to Adapt the Transformer?



How to pass along the user state?

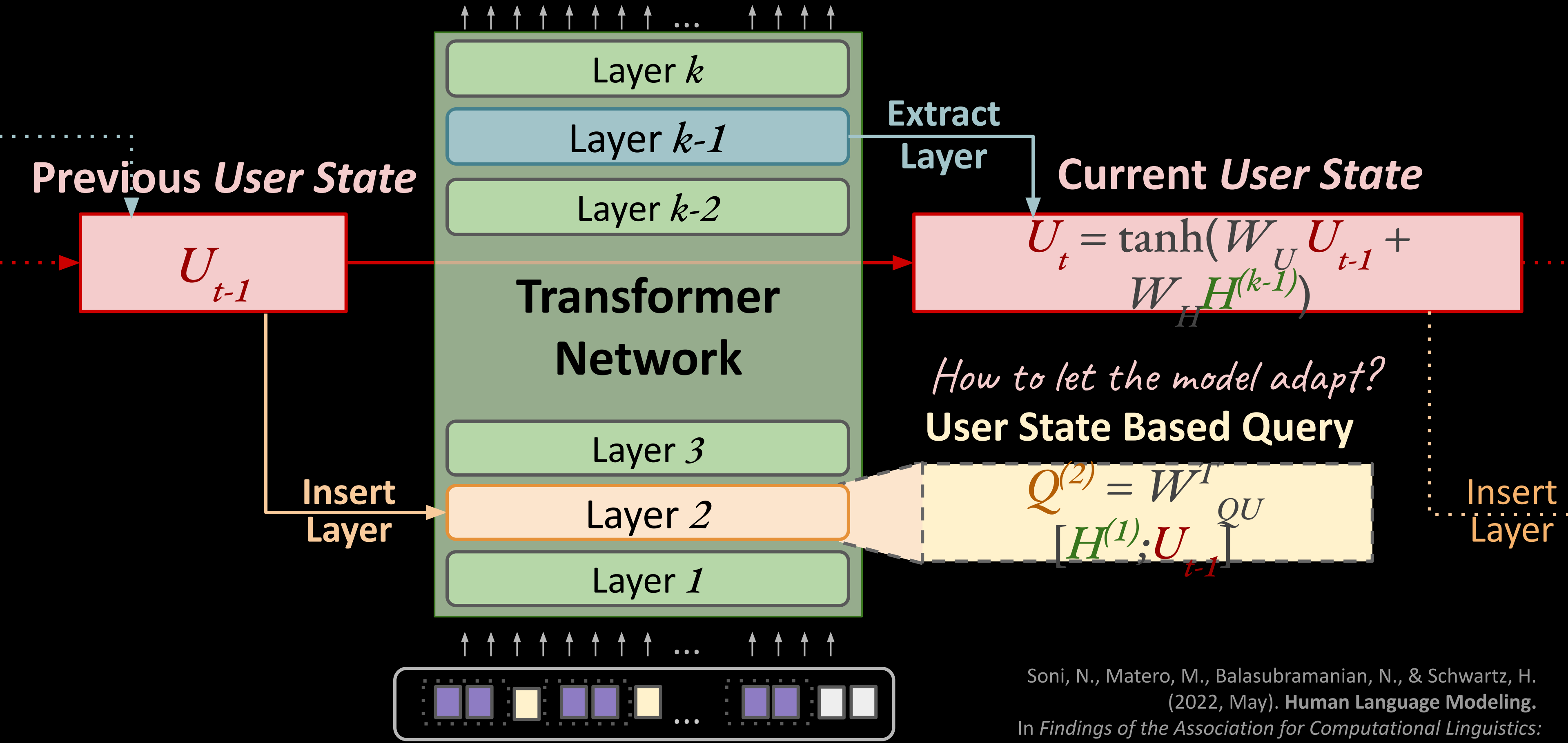
Recurrent connection from previous message.

How to let the model adapt semantics to the state?

Input: A Block of Temporally Ordered User Messages

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

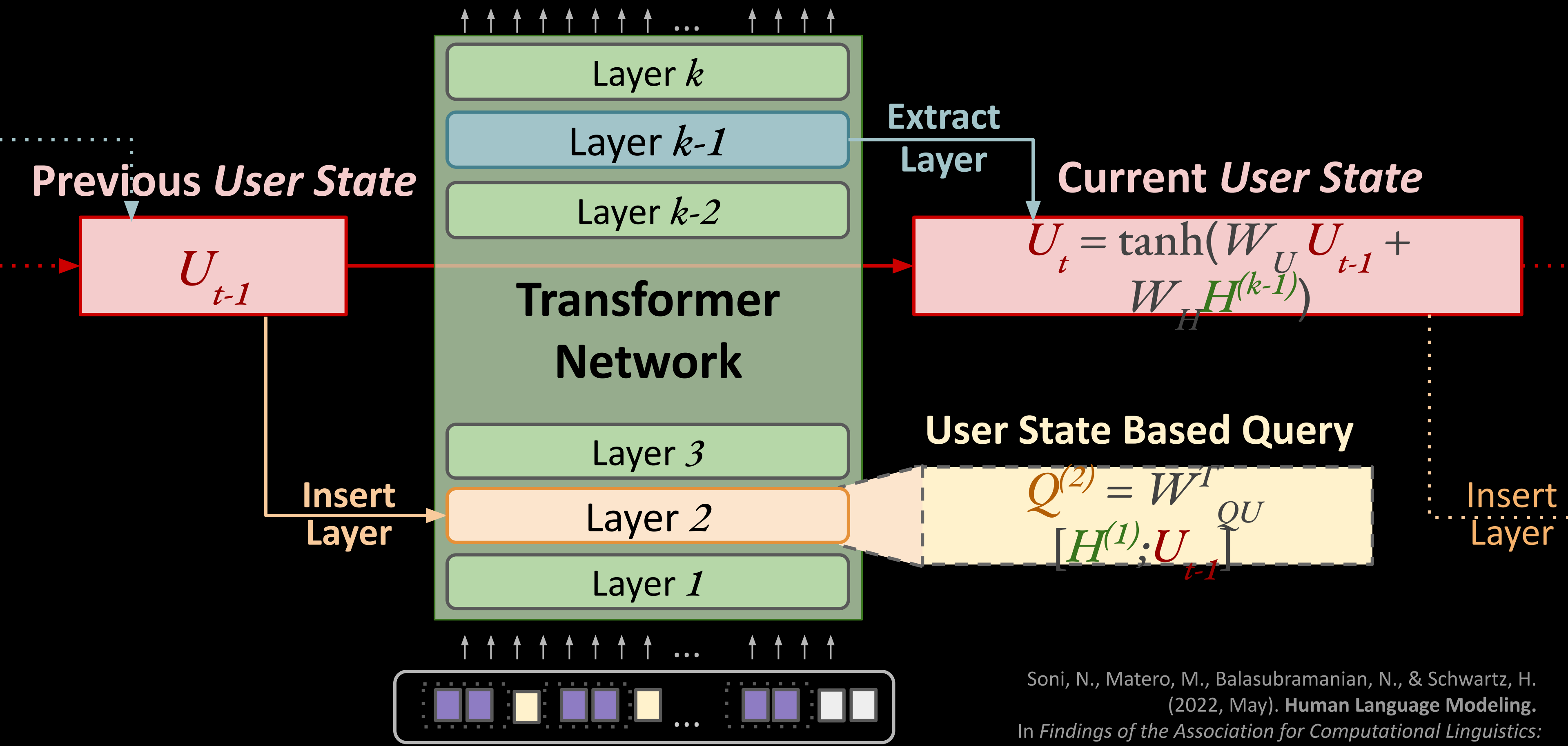
How to Adapt the Transformer?



Input: A Block of Temporally Ordered User Messages

Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

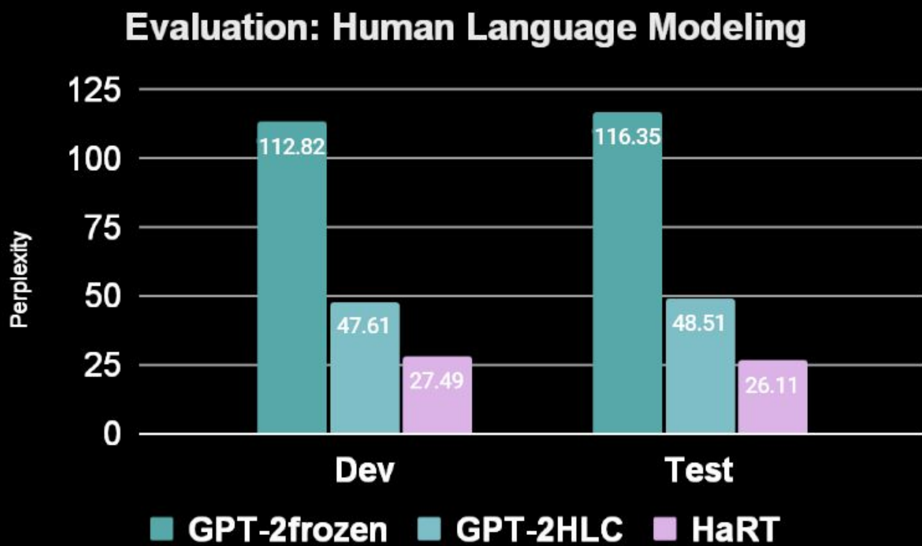
Human-aware Recurrent Transformer (HaRT)

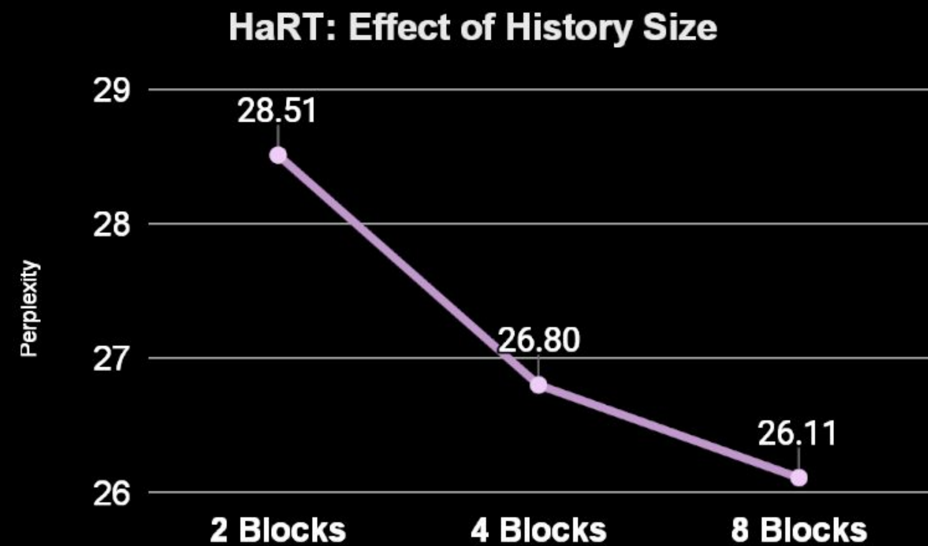


Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). **Human Language Modeling.** In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

Input: A Block of Temporally Ordered User Messages

Human Language Modeling





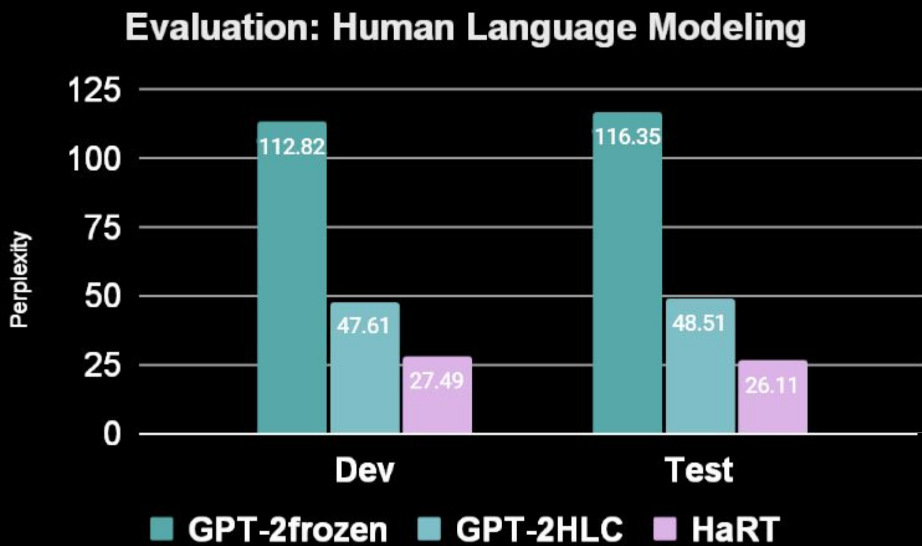
Dataset: Human Language Corpus (HLC)

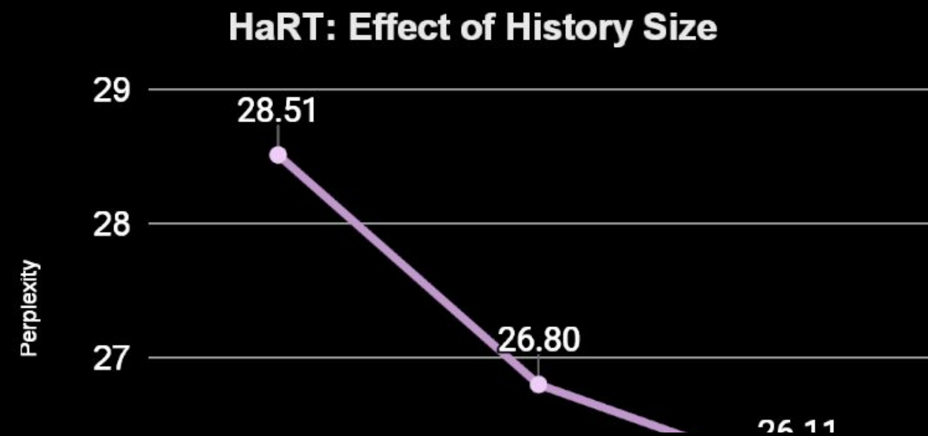
Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

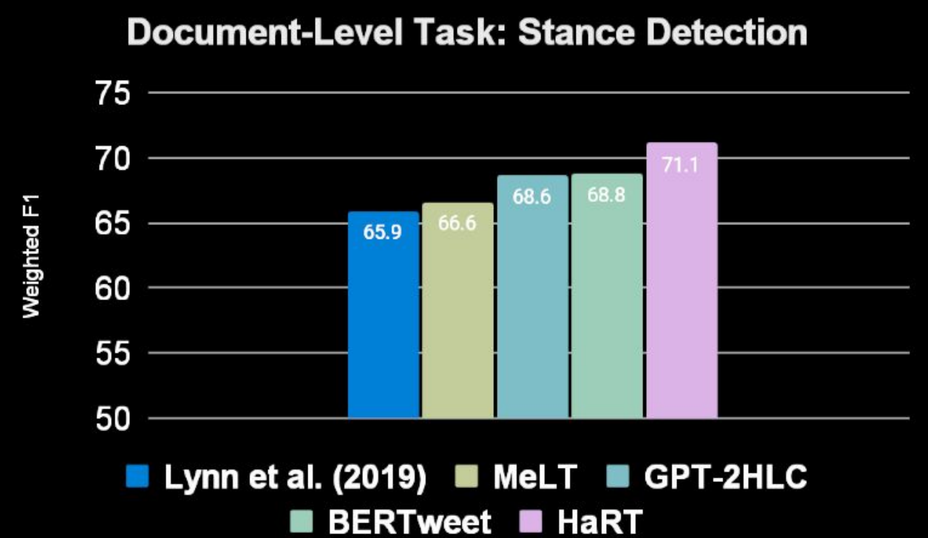
+

Train	Dev	Test
users = 96k msgs = 36m (8 blocks= ~17m)	users = 2k msgs = 830k + **seen** users: 2.5k msgs: 230k	users = 2k msgs = 690k + **seen** users: 2.5K msgs: 240k

Human Language Modeling

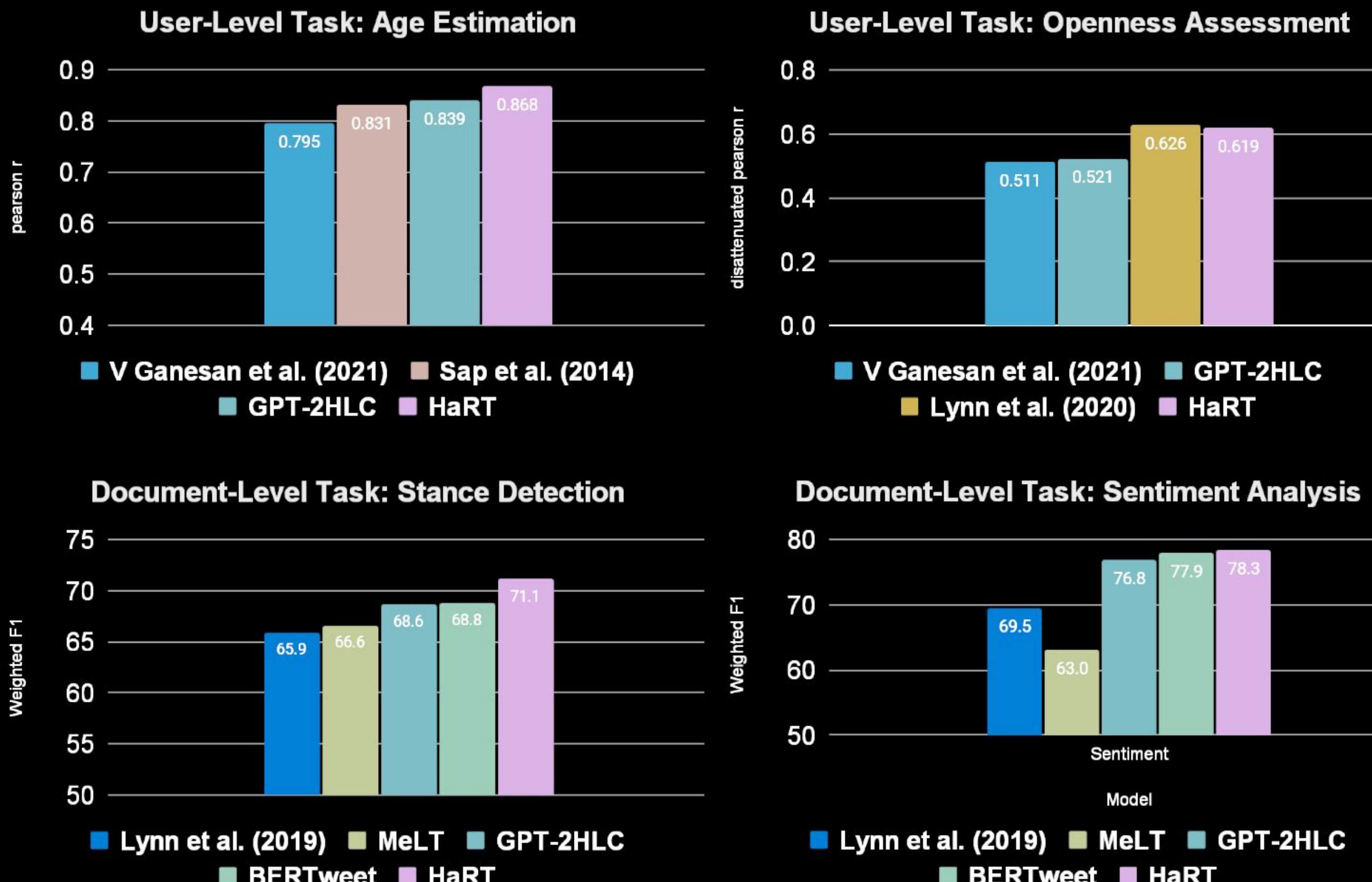






Dataset: Human Language Corpus (

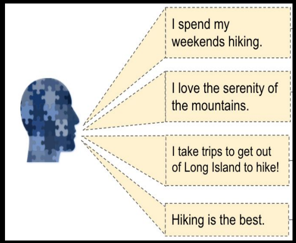
Soni, N., Matero, M., Balasubramanian, N., & Schwartz, H. (2022, May). Human Language Modeling. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 622-636).

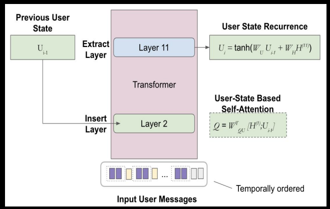


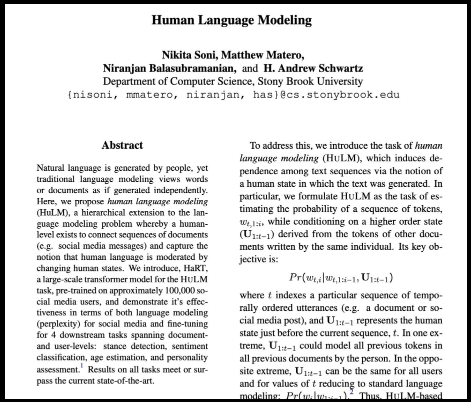
User-Level Task: Age Estimation
pearson r
0.9
0.8
0.7
0.6
0.5
0.4
0.795
0.831
0.839
0.868
V Ganesan et al. (2021)
Sap et al. (2014)
GPT-2HLC
HaRT
User-Level Task: Openness Assessment
disattenuated pearson r
0.8
0.6
0.4
0.2
0.0
0.511
0.521
0.626
0.619
V Ganesan et al. (2021)
GPT-2HLC
Lynn et al. (2020)
HaRT
Document-Level Task: Stance Detection
Weighted F1
75
70
65
60
55
50
65.9
66.6
68.6
68.8
71.1
Lynn et al. (2019)
MeLT
GPT-2HLC
BERTweet
HaRT
Document-Level Task: Sentiment Analysis
Weighted F1
80
70
60
50
69.5
63.0
76.8
77.9
78.3
Sentiment
Model
Lynn et al. (2019)
MeLT
GPT-2HLC
BERTweet
HaRT

HuLM/HaRT Takeaways

- **HuLM:** Extension of language modeling with notion of user.
- **HaRT:** First step toward large *human* language models.
- Progress for large LMs grounded in language's "natural" generators, people.
- [GitHub Repository]







Human Language Modeling

Nikita Soni, Matthew Matero,
Niranjan Balasubramanian, and **H. Andrew Schwartz**
Department of Computer Science, Stony Brook University
{nisoni, mmatero, niranjan, has}@cs.stonybrook.edu

Abstract

Natural language is generated by people, yet traditional language modeling views words or documents as if generated independently. Here, we propose *human language modeling* (HuLM), a hierarchical extension to the language modeling problem whereby a human-level exists to connect sequences of documents (e.g. social media messages) and capture the notion that human language is moderated by changing human states. We introduce, HaRT, a large-scale transformer model for the HuLM task, pre-trained on approximately 100,000 social media users, and demonstrate it's effectiveness in terms of both language modeling (perplexity) for social media and fine-tuning for 4 downstream tasks spanning document- and user-levels: stance detection, sentiment classification, age estimation, and personality assessment.[1] Results on all tasks meet or surpass the current state-of-the-art.

To address this, we introduce the task of *human language modeling* (HuLM), which induces dependence among text sequences via the notion of a human state in which the text was generated. In particular, we formulate HuLM as the task of estimating the probability of a sequence of tokens, $w_{t,1:i}$, while conditioning on a higher order state ($\mathbf{U}_{1:t-1}$) derived from the tokens of other documents written by the same individual. Its key objective is:

$$Pr(w_{t,i} | w_{t,1:i-1}, \mathbf{U}_{1:t-1})$$

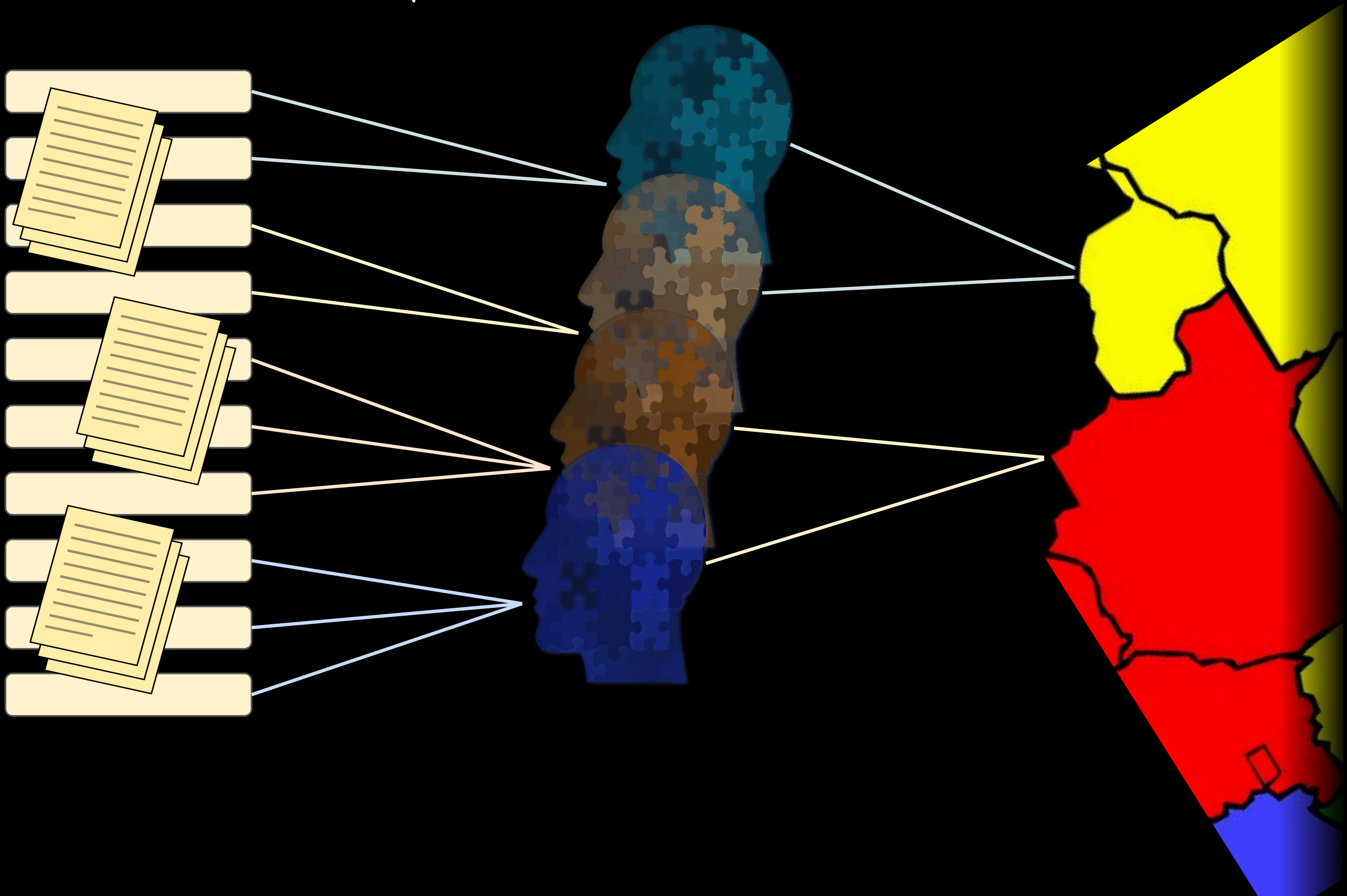
where t indexes a particular sequence of temporally ordered utterances (e.g. a document or social media post), and $\mathbf{U}_{1:t-1}$ represents the human state just before the current sequence, t. In one extreme, $\mathbf{U}_{1:t-1}$ could model all previous tokens in all previous documents by the person. In the opposite extreme, $\mathbf{U}_{1:t-1}$ can be the same for all users and for values of t reducing to standard language modeling: $Pr(w_i | w_{1:i-1})$.[2] Thus, HuLM-based

Human-Centered NLP – Review:

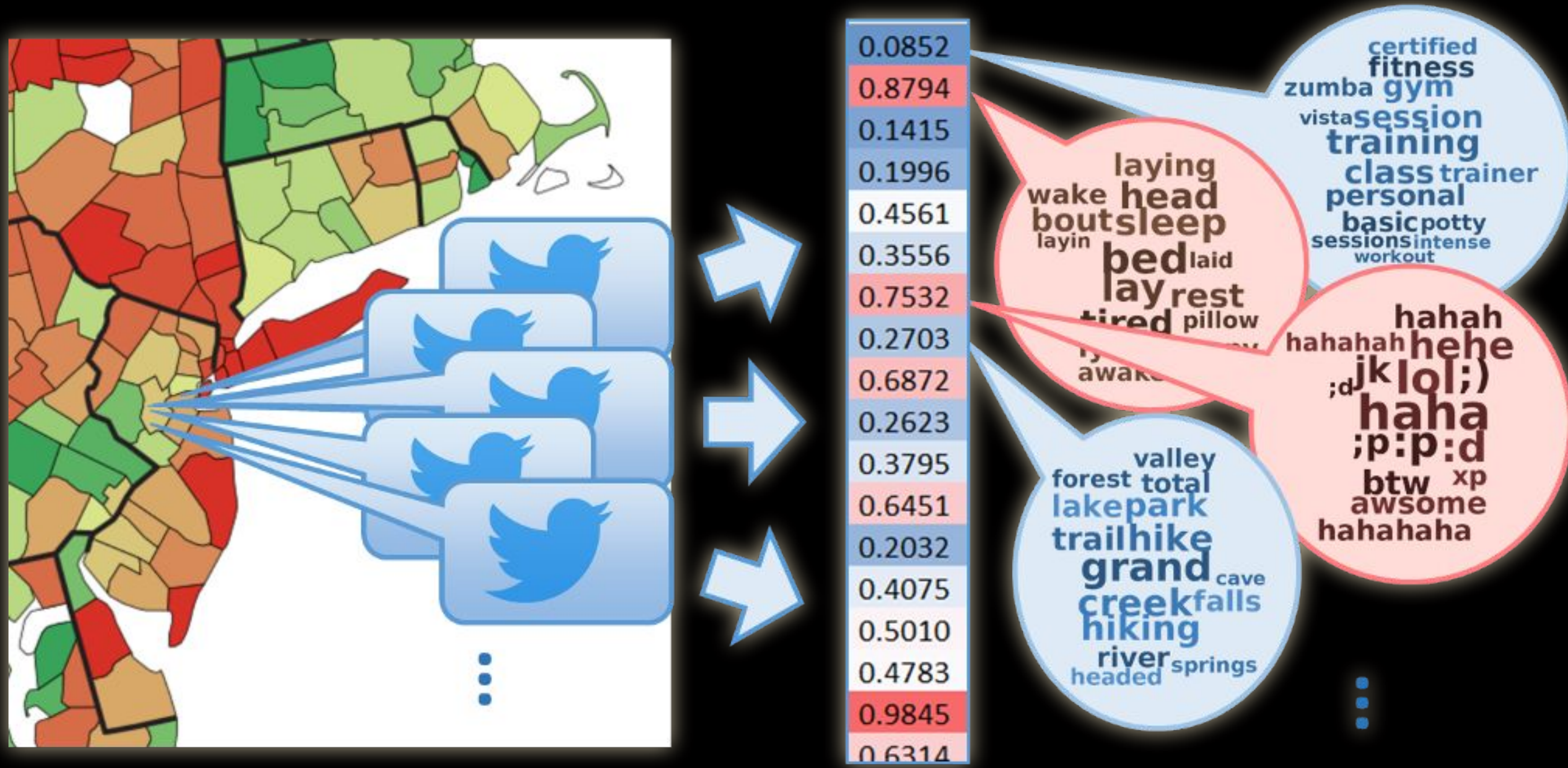
1. Differential Language Analysis
2. Human Factor Adaptation
3. Human Language Modeling

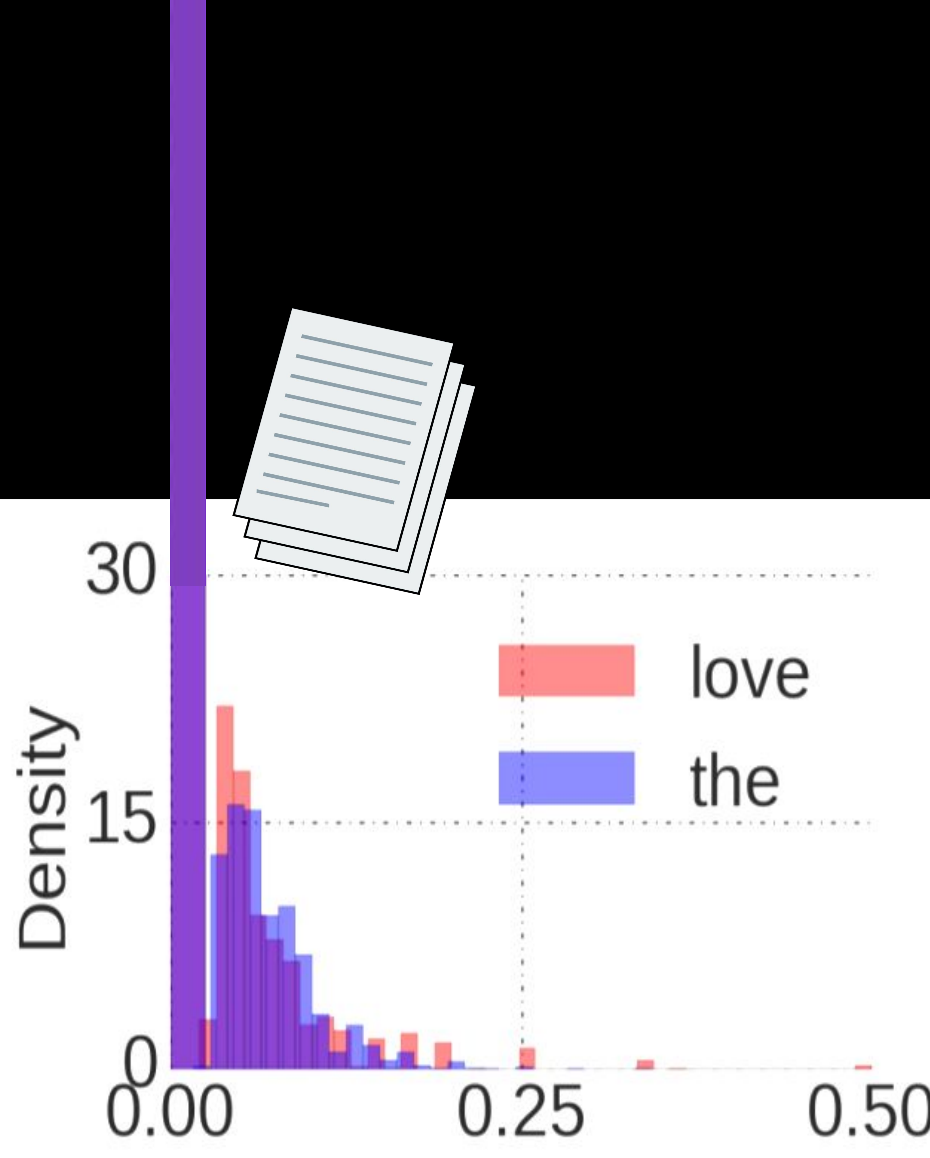
Supplement: On the multi-level nature of words:

Data are inherently multi-level.

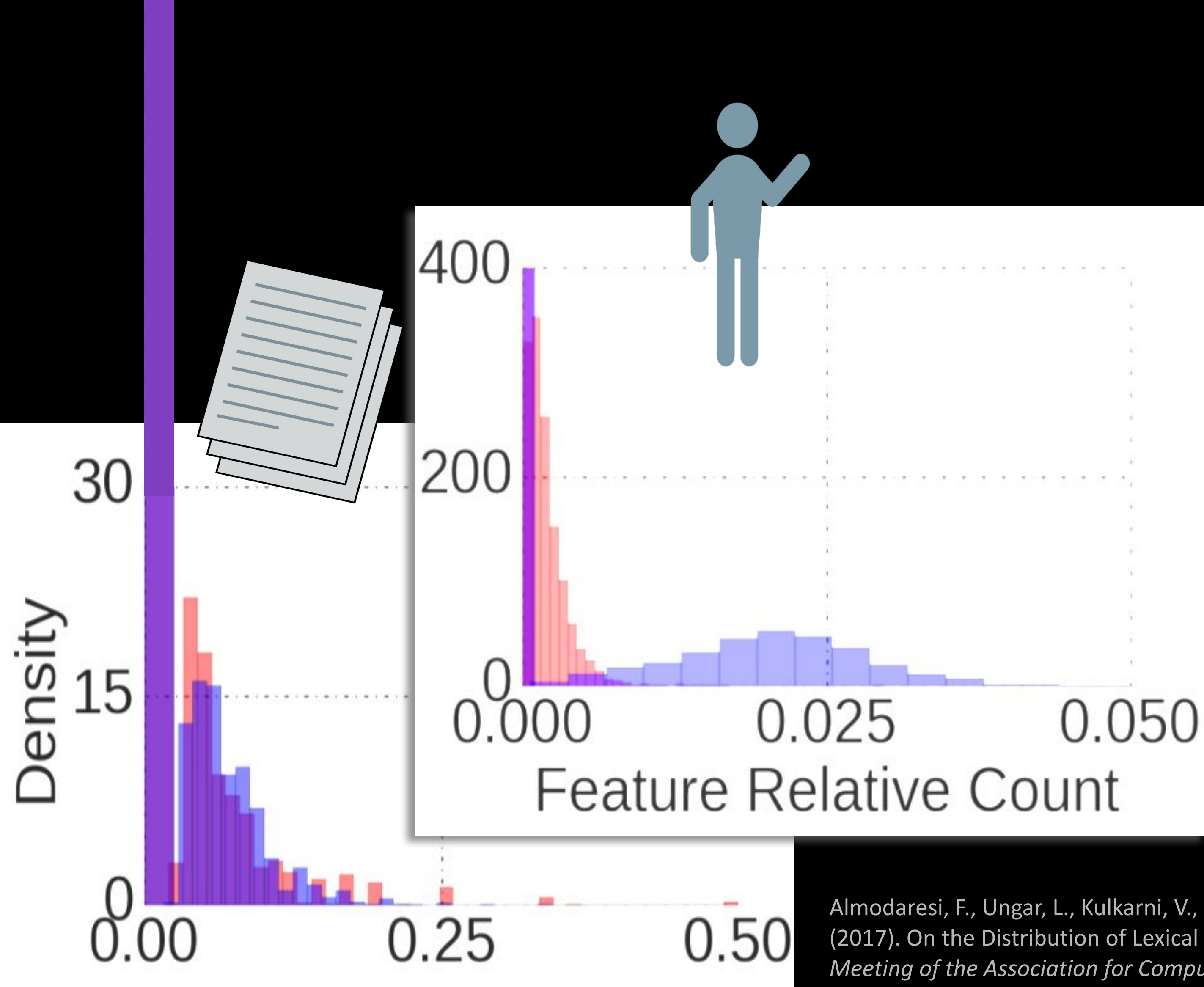


1,639,750 tweets from 5,226 users in 420 counties

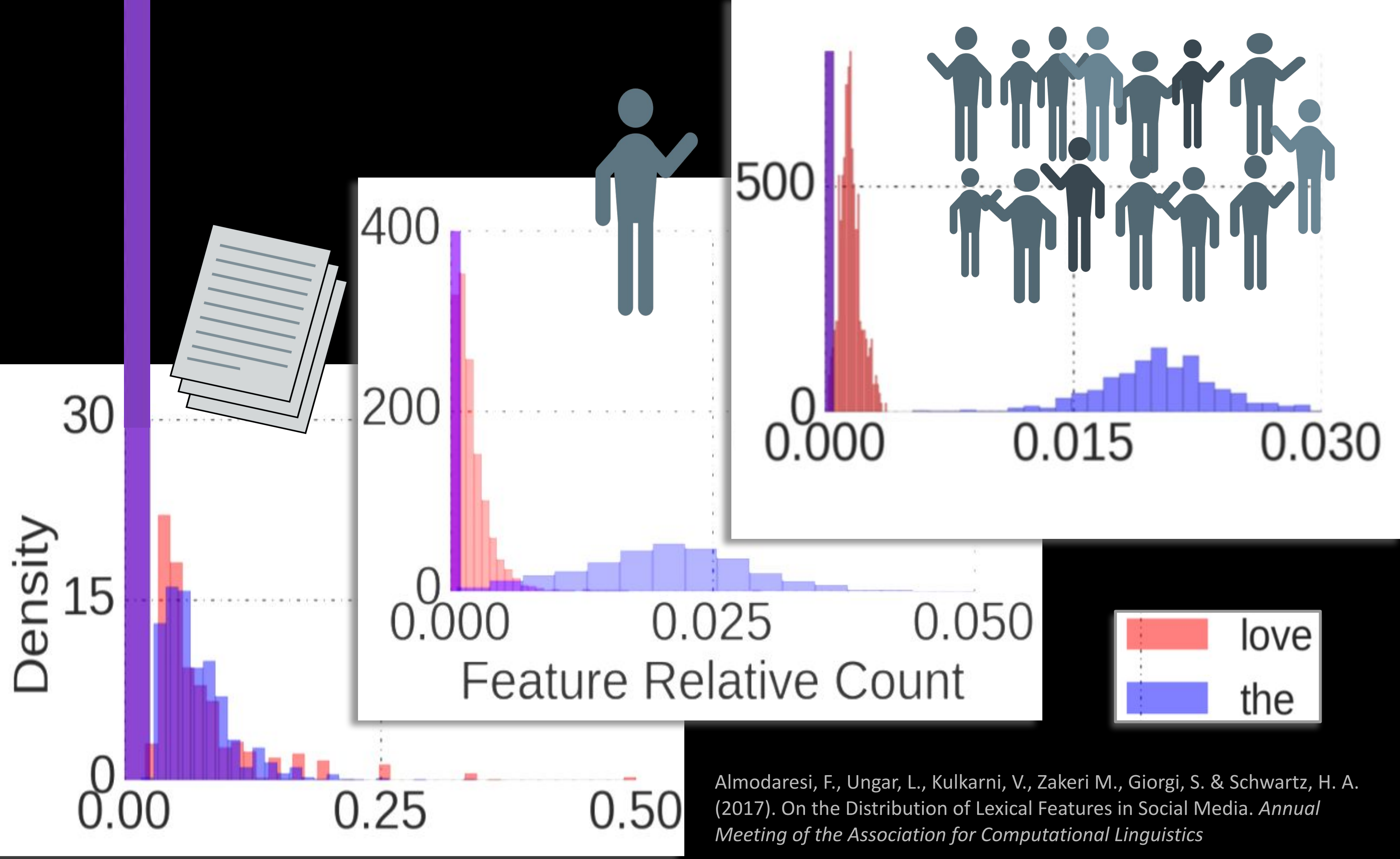




Almodaresi, F., Ungar, L., Kulkarni, V., Zakeri M., Giorgi, S. & Schwartz, H. A. (2017). On the Distribution of Lexical Features in Social Media. *Annual Meeting of the Association for Computational Linguistics*



Almodaresi, F., Ungar, L., Kulkarni, V., Zakeri M., Giorgi, S. & Schwartz, H. A. (2017). On the Distribution of Lexical Features in Social Media. *Annual Meeting of the Association for Computational Linguistics*



Almodaresi, F., Ungar, L., Kulkarni, V., Zakeri M., Giorgi, S. & Schwartz, H. A. (2017). On the Distribution of Lexical Features in Social Media. *Annual Meeting of the Association for Computational Linguistics*

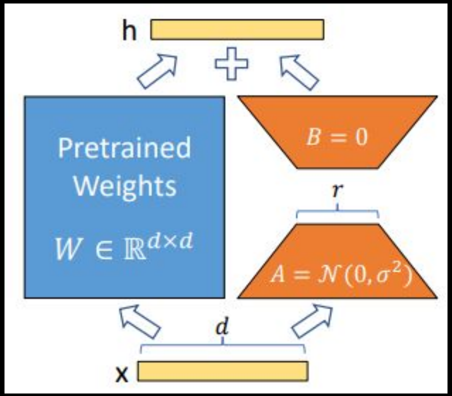
Data are inherently multi-level.

Distribution	Message			User			County		
	1-gram	topic	Lex.	1-gram	topic	Lex.	1-gram	topic	Lex.
Power Law	**.71**	.10	.00	.04	.00	.00	.07	.00	.00
Log-Normal	.25	**.89**	**1.00**	**.96**	**.97**	**.64**	**.92**	**.86**	.44
Normal	.04	.01	.00	.00	.03	.36	.01	.14	**.56**

Proportion best fit by the given distribution.

Almodaresi, F., Ungar, L., Kulkarni, V., Zakeri M., Giorgi, S. & Schwartz, H. A. (2017). On the Distribution of Lexical Features in Social Media. *Annual Meeting of the Association for Computational Linguistics*

LoRA: Fine-tuning LMs with Low Rank Approximation



- LoRA is a memory efficient form of training LLMs without significant loss in performance
- LoRA performs gradient updates for only 4M out of 7B parameters to improve Llama2's social understanding

(slides from Yang, 2023; based on slides based on Jesse Mu, Ivan Vulic, Jonas Pfeiffer, and Sebastian Ruder)

LoRA: Fine-tuning LMs with Low Rank Approximation

- For each downstream task, we learn a different set of parameters $\Delta\phi$
  - $|\Delta\phi| = |\phi_o|$
  - GPT-3 has a $|\ \phi_o|$ of 175 billion
  - Expensive and challenging for storing and deploying many independent instances

- Key idea: encode the task-specific parameter increment $\Delta\phi = \Delta\phi(\Theta)$ by a smaller-sized set of parameters Θ, $|\Theta| \ll |\phi_o|$

- The task of finding $\Delta\phi$ becomes optimizing over Θ

$$\max_{\Theta} \sum_{(x,y)} \sum_{t=1}^{|y|} \log(P_{\phi_o+\Delta\phi(\Theta)}\,(y_t|x, y_{<t}))$$

(slides from Yang, 2023; based on slides based on Jesse Mu, Ivan Vulic, Jonas Pfeiffer, and Sebastian Ruder)

LoRA: Fine-tuning LMs with Low Rank Approximation

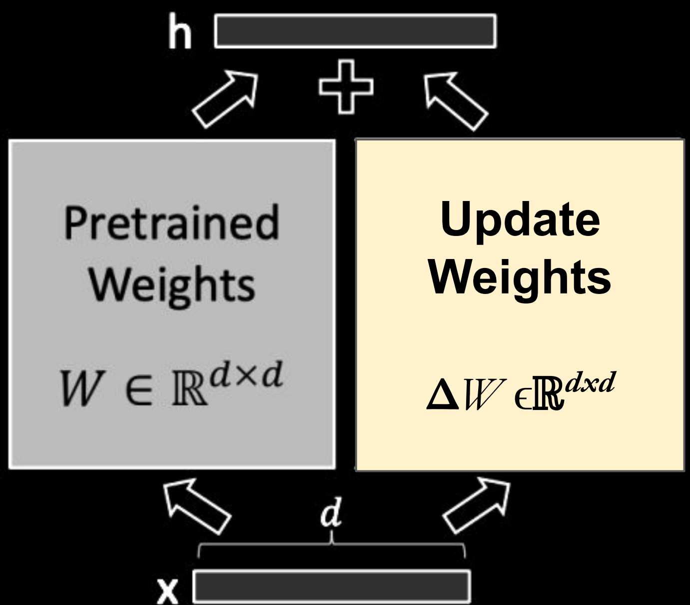
Low-rank-parameterized update matrices

- Updates to the weights have a low "intrinsic rank" during adaptation (Aghajanyan et al. 2020)
- $W_0 \in \mathbb{R}^{d \times k}$: a pretrained weight matrix
- Constrain its update with a low-rank decomposition:

$$W_0 + \Delta W = W_0 + BA$$

where $B \in \mathbb{R}^{d \times r}, A \in \mathbb{R}^{r \times k}, r \ll \min(d, k)$

- Only A and B contain **trainable** parameters



$h = Wx$

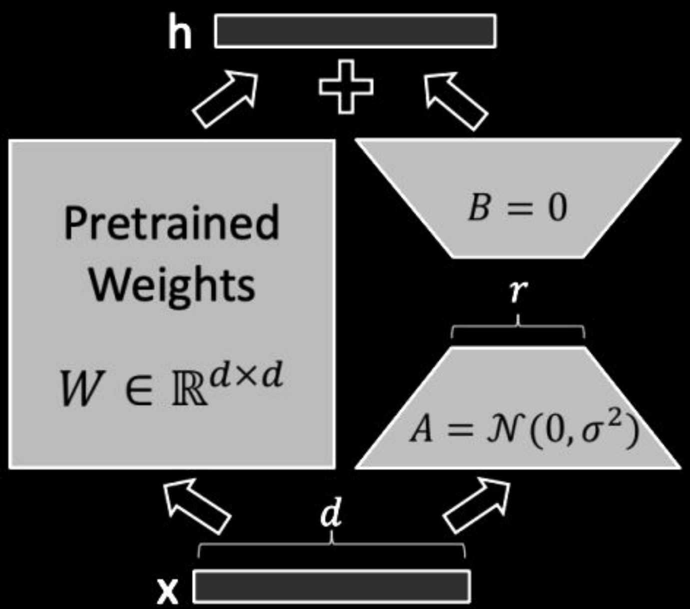
(slides from Yang, 2023; based on slides based on Jesse Mu, Ivan Vulic, Jonas Pfeiffer, and Sebastian Ruder)

LoRA: Fine-tuning LMs with Low Rank Approximation

Low-rank-parameterized update matrices

- As one increase the number of trainable parameters, training LoRA converges to training the original model
- **No additional inference latency:** when switching to a different task, recover W_0 by subtracting BA and adding a different $B'A'$
- Often LoRA is applied to the weight matrices in the self-attention module

just query and value is enough



$h = Wx$

(slides from Yang, 2023; based on slides based on Jesse Mu, Ivan Vulic, Jonas Pfeiffer, and Sebastian Ruder)