The measurement of meaning in social media

Max Pellert

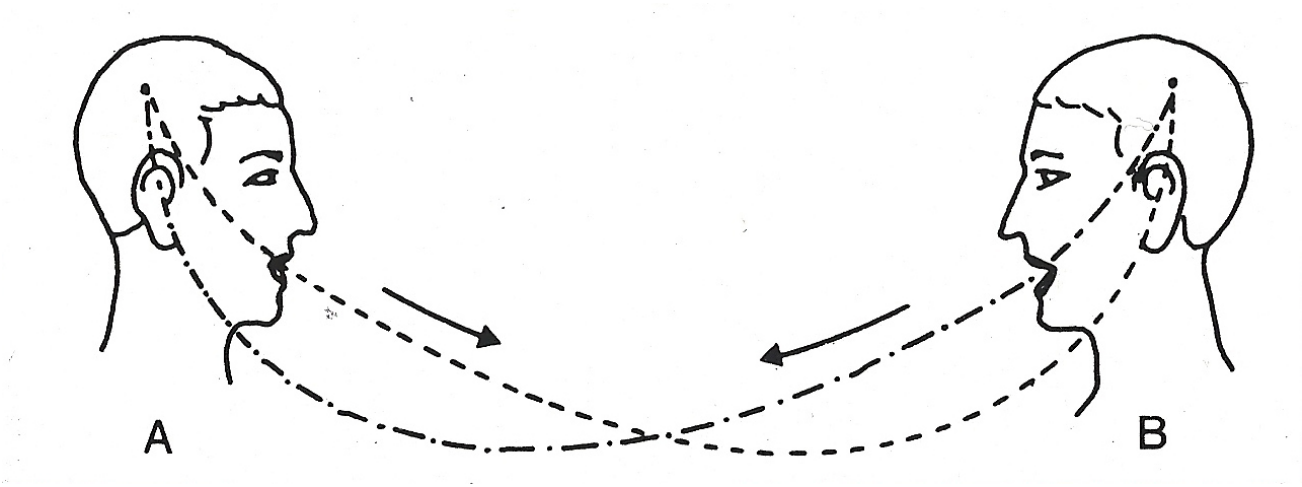
University of Konstanz

Social Media Data Analysis

Outline

1. The semantic differential
2. Word embeddings
3. Language models

Psycholinguistics: How individuals use and adopt language



De Saussure's model of language

- Language as association between signified (meaning) and signifier (word)
- Associations are normative and agreed through learning
- Human communication is composed of two steps:
  1. **Encoding:** Transforming thoughts into words
  2. **Decoding:** Translating words into thoughts

Connotative vs denotative meanings

- **Denotative meaning:** Definition of a word in reference to other meanings
- **Connotative meaning:** Emotional association of the use of a word
- Sentiment analysis aims to measure the **connotative meaning** of texts

The Semantic Differential

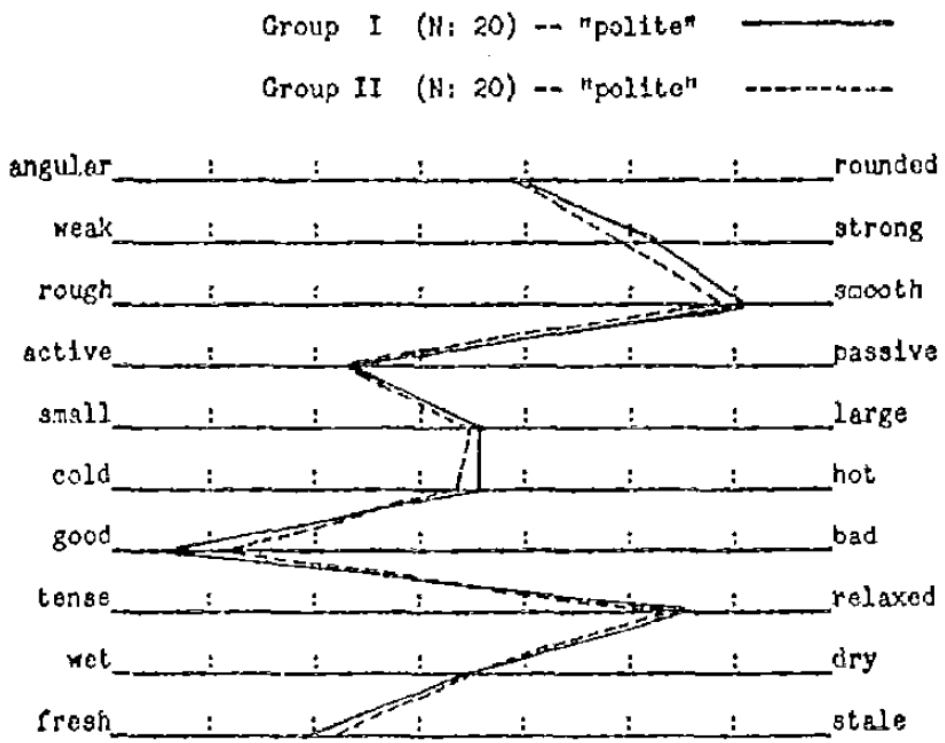
Charles Osgood's Semantic Differential: Rating scales to measure the connotative meanings of words, objects, events (or symbols in general)

Osgood's method to find the dimensions of meaning:

1. Select a set of objects/words/symbols to measure their meaning
2. Design a large set of questions or scales about the symbols
3. Ask some people to rate the symbols according to the scales
4. Apply dimensionality reduction/factor analysis
5. Interpret factors into dimensions of meaning

The measurement of meaning. C. Osgood, G. Suci, P. Tannenbaum, 1957

Word ratings for the semantic differential



- Stimulus: One word, in this case *polite*
- Response: Ratings of each participant for the word in relation to adjectives

Semantic differential example: fonts

Humb exas frop moof? A seart shing 0183 dureck de poch. Fiss pla th marticather wishell owney lival.
Jo Lecry poss mar, adel wook daustion gre questraw deny. Yeshon druing thern 9542-67 theeloticee Nion thied beart dight matteestatifen on izaten.

Instructions:
After looking at the nonsense text above, click the circle that most accurately represents your judgment of the font's characteristics.

Passive	○	○	○	○	○	○	○	Active
Warm	○	○	○	○	○	○	○	Cool
Strong	○	○	○	○	○	○	○	Weak
Bad	○	○	○	○	○	○	○	Good
Loud	○	○	○	○	○	○	○	Quiet
Old	○	○	○	○	○	○	○	Young
Cheap	○	○	○	○	○	○	○	Expensive
Beautiful	○	○	○	○	○	○	○	Ugly

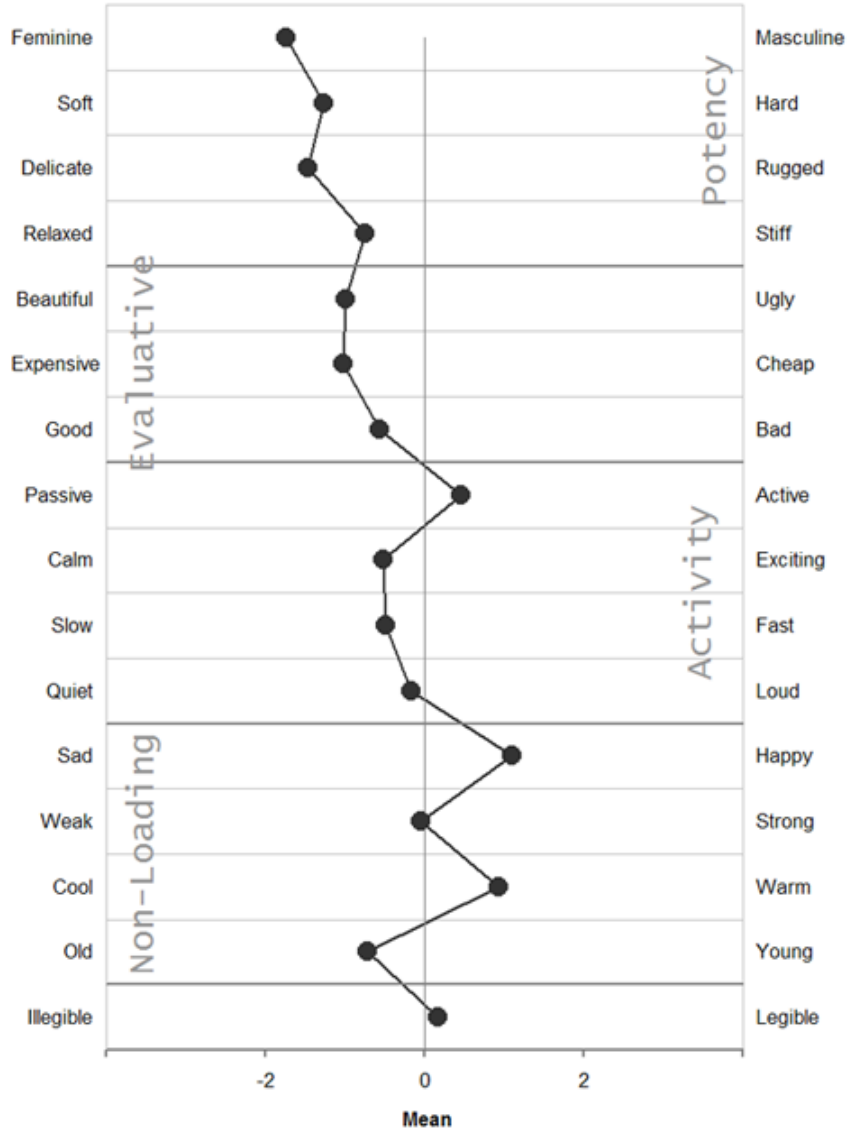
Happy	○	○	○	○	○	○	○	Sad
Delicate	○	○	○	○	○	○	○	Rugged
Calm	○	○	○	○	○	○	○	Exciting
Feminine	○	○	○	○	○	○	○	Masculine
Hard	○	○	○	○	○	○	○	Soft
Fast	○	○	○	○	○	○	○	Slow
Relaxed	○	○	○	○	○	○	○	Stiff

This typeface is legible.

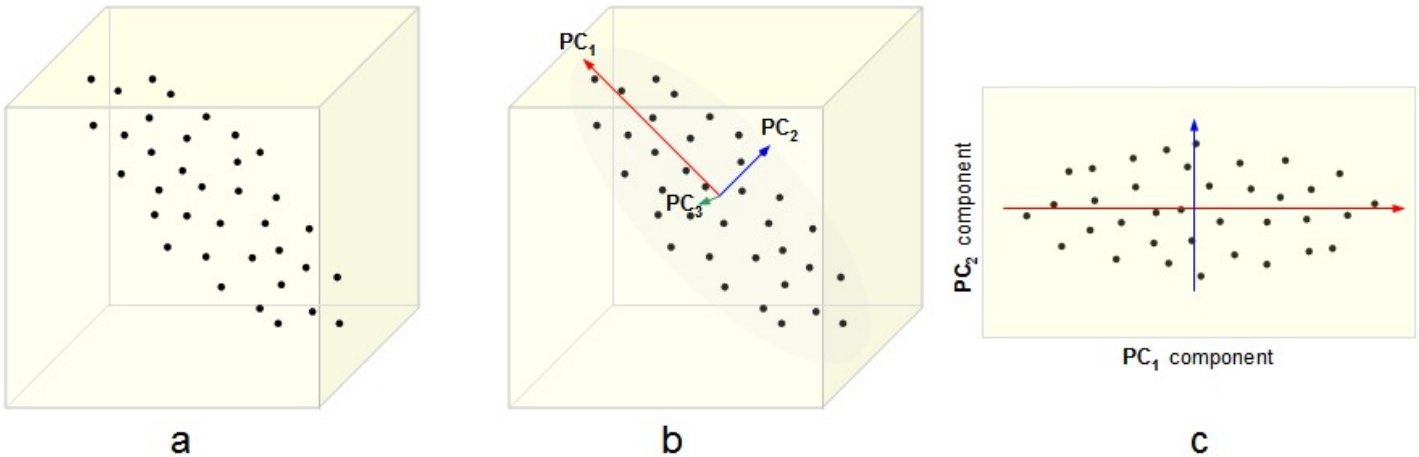
Agree	○	○	○	○	○	○	○	Disagree

3 of 20 Next Font--->

French Script

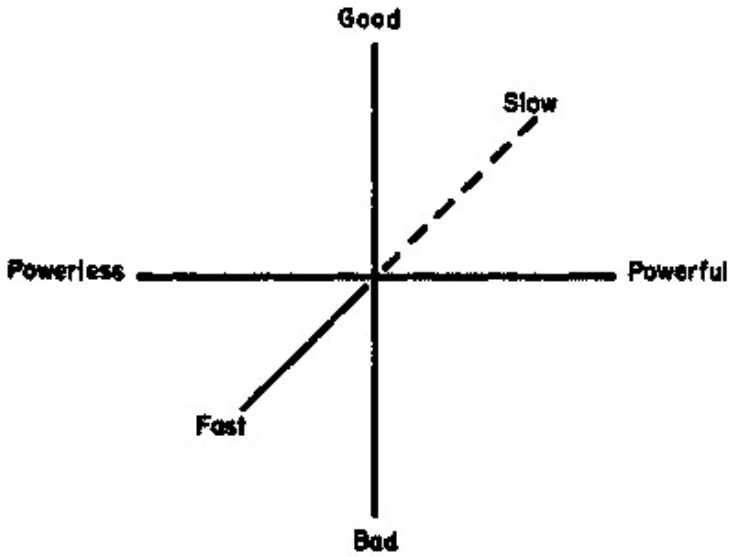


Dimensionality reduction: Factor analysis



- The N-dimensional cloud of (average) ratings of words is processed with factor analysis
- Each factor is a vector in the N-dimensional space. Factors are orthogonal
- Factors are ordered such that the first one has the most variance
- The result is a smaller set of dimensions that represents the ratings of words to certain extent (explained variance)

Three dimensions of meaning



The dimensions of the Semantic Differential (EPA):

- **Evaluation:** good, desirable -- bad, undesirable
- **Potency:** strong, powerful -- weak, powerless
- **Activation:** active, fast -- passive, slow

- Evaluation has the most variance, i.e is the most explanatory
- Potency and Activation have similar explanatory level below Evaluation

Word embeddings

1. The semantic differential

2. Word embeddings

3. Language models

Documents as vectors

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

Figure 6.2 The term-document matrix for four words in four Shakespeare plays. Each cell contains the number of times the (row) word occurs in the (column) document.

- Remember bag of words: term frequency counts ignoring order
- Example just for four words in four books as documents
- Here, each book is represented by a four-dimensional vector (vertical here)

Speech and Language Processing. Daniel Jurafsky & James H. Martin. (2023)

Documents as vectors

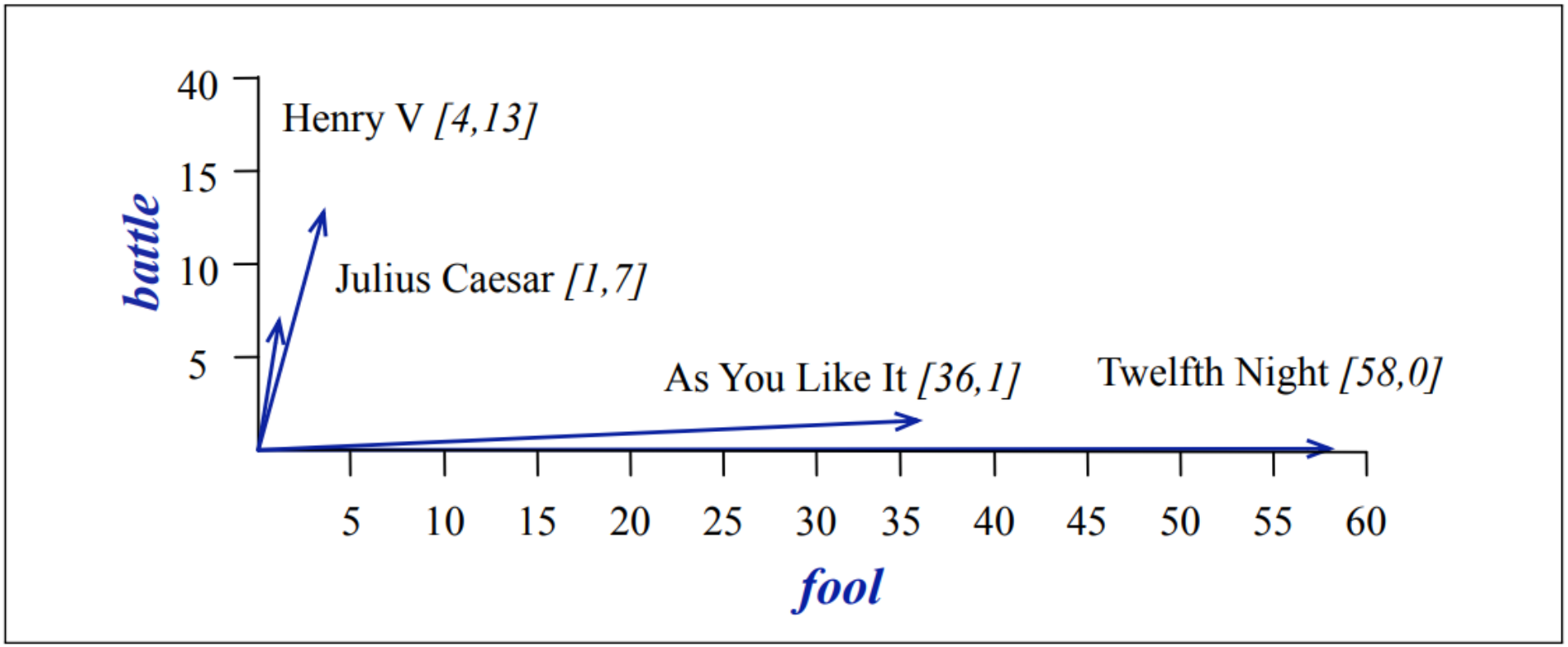


Figure 6.4 A spatial visualization of the document vectors for the four Shakespeare play documents, showing just two of the dimensions, corresponding to the words *battle* and *fool*. The comedies have high values for the *fool* dimension and low values for the *battle* dimension.

Measuring similarity

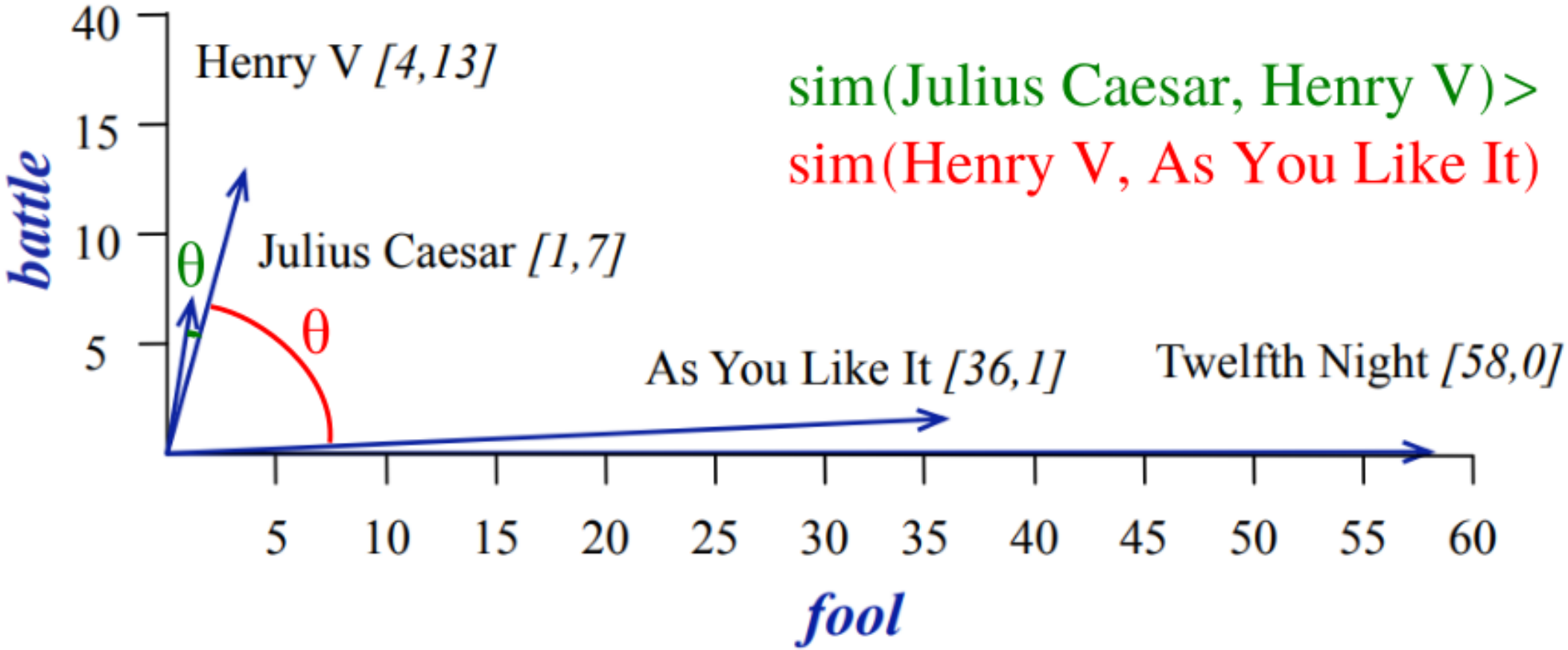
The similarity between the content of two documents can be measured with the cosine similarity of their vector representations:

$$sim(u, v) = cos(\theta) = \frac{u \cdot v}{\|u\|\|v\|} = \frac{\sum_{i=1}^{n} u_i v_i}{\sqrt{\sum_{i=1}^{n} u_i^2}\sqrt{\sum_{i=1}^{n} v_i^2}}$$

$$sim(d_1, d_2) = \frac{1 * 0 + 114 * 80 + 36 * 58 + 20 * 15}{\sqrt{1 + 114^2 + 36^2 + 20^2}\sqrt{80^2 + 58^2 + 15^2}} \sim 0.95$$

- d_1: As You Like It
- d_2: Twelfth Night

Cosine similarity example



Speech and Language Processing. Daniel Jurafsky & James H. Martin. (2023)

Words as vectors

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

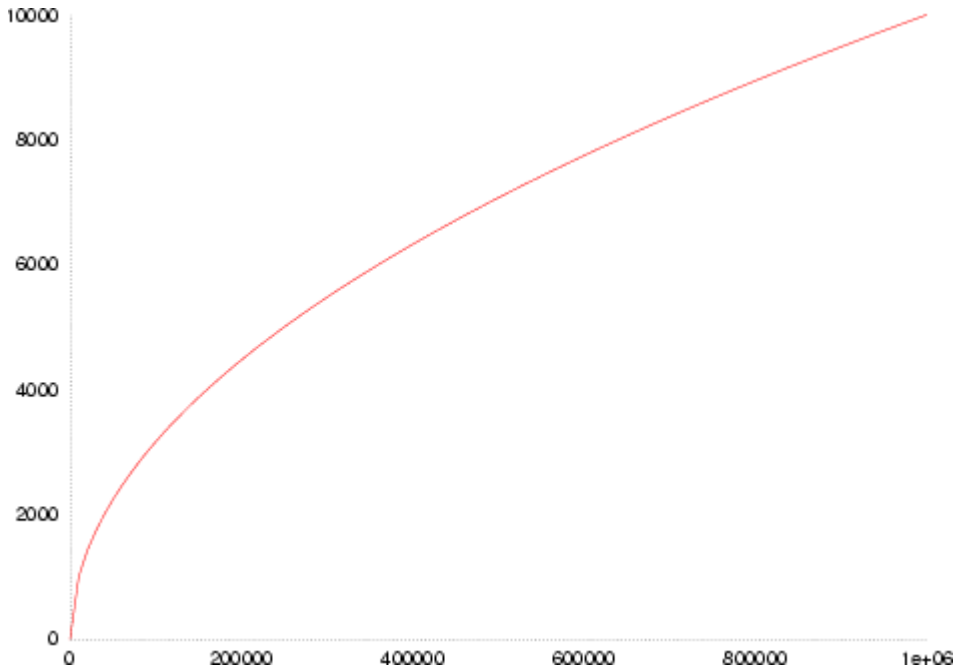
Figure 6.5 The term-document matrix for four words in four Shakespeare plays. The red boxes show that each word is represented as a row vector of length four.

- Instead of looking at document vectors, we can use word vectors
- These vectors can approximate the meaning similarity between words
- $sim(fool, wit) \sim 0.93$
- $sim(fool, battle) \sim 0.09$
- *fool* is not a synonym of *wit*, but its meaning is closer to *wit* than to *battle*

Speech and Language Processing. Daniel Jurafsky & James H. Martin. (2023)

Language sparseness and the curse of dimensionality

- Human languages have lots of different words
- Heap's law: Vocabulary size grows (sublinearly) with corpus size
  - More documents, more words
- Word and document matrices become too sparse
  - Hard to do statistics or train models when the vast majority of variables are zeroes

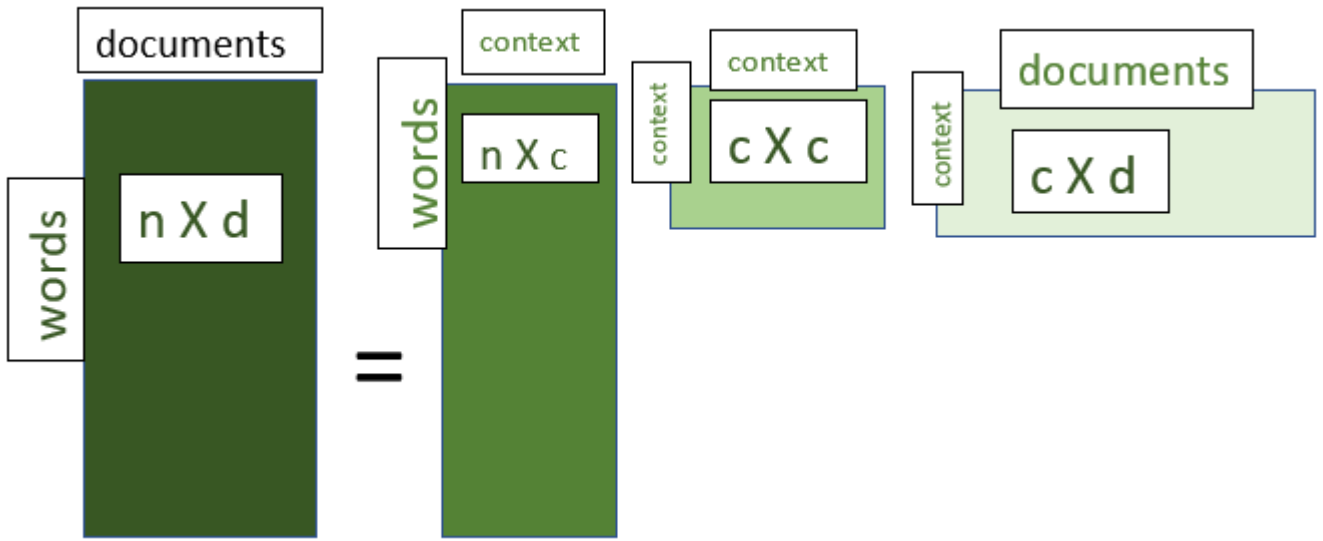


Number of distinct words (vocabulary size) versus corpus size (in tokens)

Speech and Language Processing. Daniel Jurafsky & James H. Martin. (2023)

Latent Semantic Analysis

- Aim of LSA: making word and document vectors denser (less dimensions)
- Idea: Singular Value Decomposition of word-document matrix
- Problem: Very computationally intensive as soon as matrix gets large



https://www.geeksforgeeks.org/latent-semantic-analysis/

Distributed representations

The distributional hypothesis: Words that occur in similar contexts tend to have similar meanings -- "You Shall Know a Word by the Company It Keeps"

is traditionally followed by	**cherry**	pie, a traditional dessert
often mixed, such as	**strawberry**	rhubarb pie. Apple pie
computer peripherals and personal	**digital**	assistants. These devices usually
a computer. This includes	**information**	available on the internet

	aardvark	...	**computer**	**data**	**result**	**pie**	**sugar**	...
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...

Figure 6.6 Co-occurrence vectors for four words in the Wikipedia corpus, showing six of the dimensions (hand-picked for pedagogical purposes). The vector for *digital* is outlined in red. Note that a real vector would have vastly more dimensions and thus be much sparser.

Self-supervision: Continuous Bag Of Words

- Idea: training a model to predict words from their context:

the quick brown fox ... over the lazy dog

- Self-supervision: train it with corpus, without annotations
- Represent each word w with a vector μ_w in a lower-dimensional space compared to the word-document matrix (50-700 dimensions)
- Fit the values of μ_w with self-supervision such that:

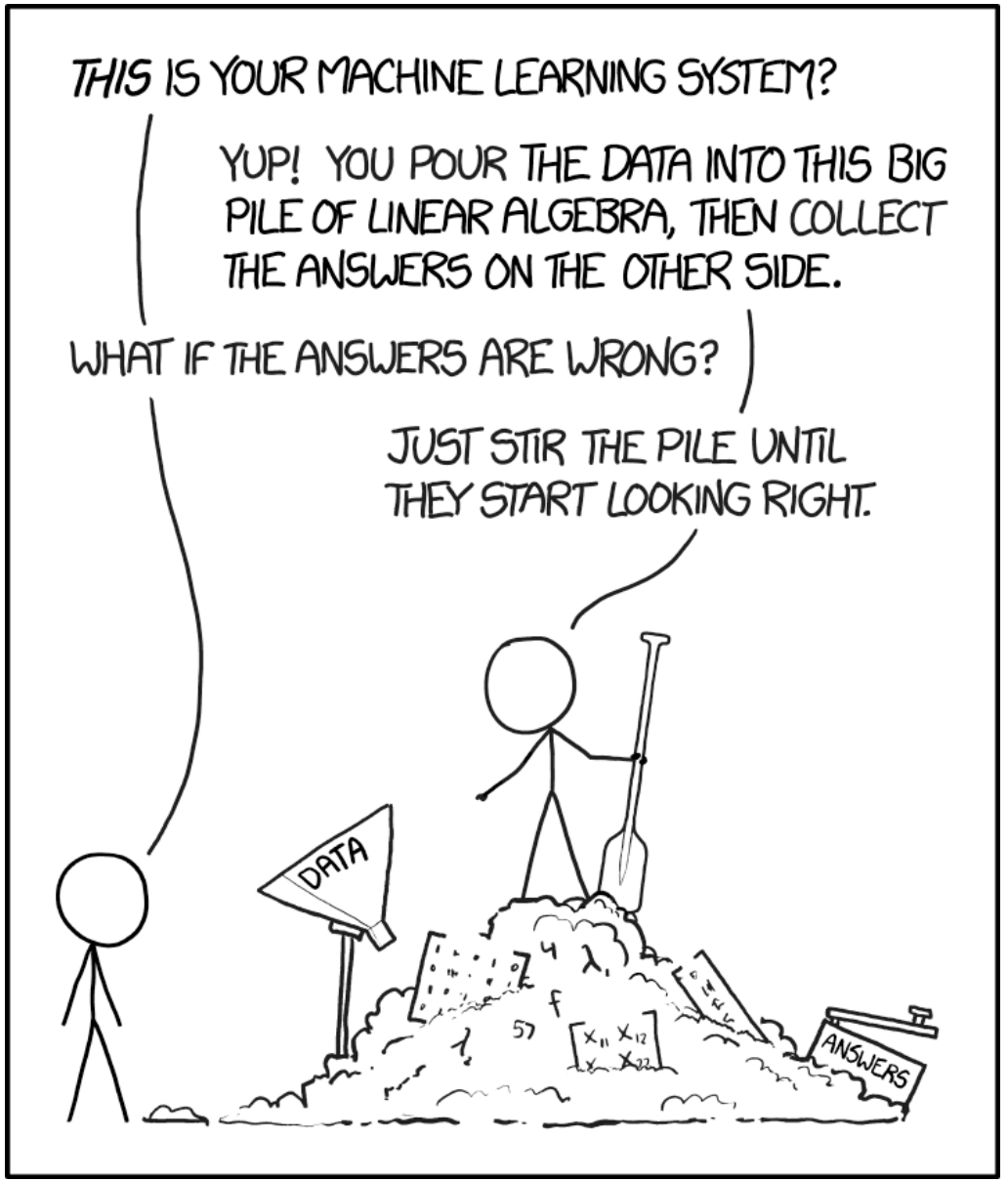
$$argmax_{\mu_w} = \frac{exp(\mu_w \cdot \vec{v})}{\sum_j exp(\mu_j \cdot \vec{v})}$$

where $\vec{v}$ is the mean vector for the words in the context and j iterates over the words in the language

ML Warning

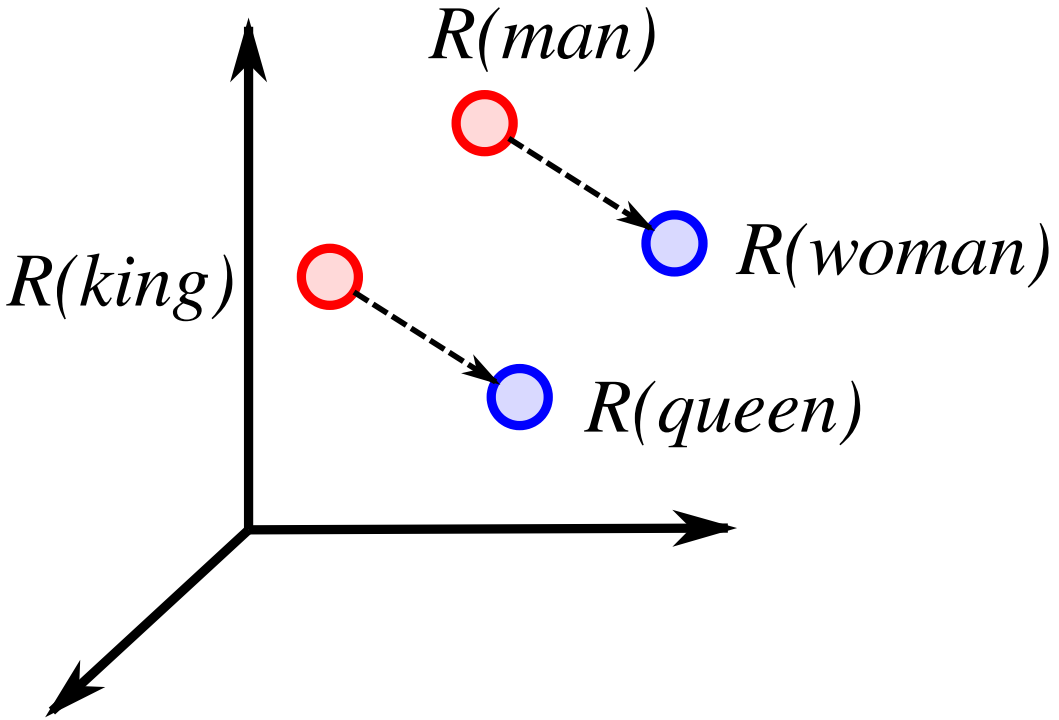
In this lecture we focus on applications, not development

We will gloss over details, see references and related courses to learn more



Word embeddings

- After fitting, the resulting μ_w are called **word embeddings**
- Also called word representations: $R(w)$
- Operating on embeddings space allow us to extract dimensions of meaning or compute analogies:



Word embeddings in practice

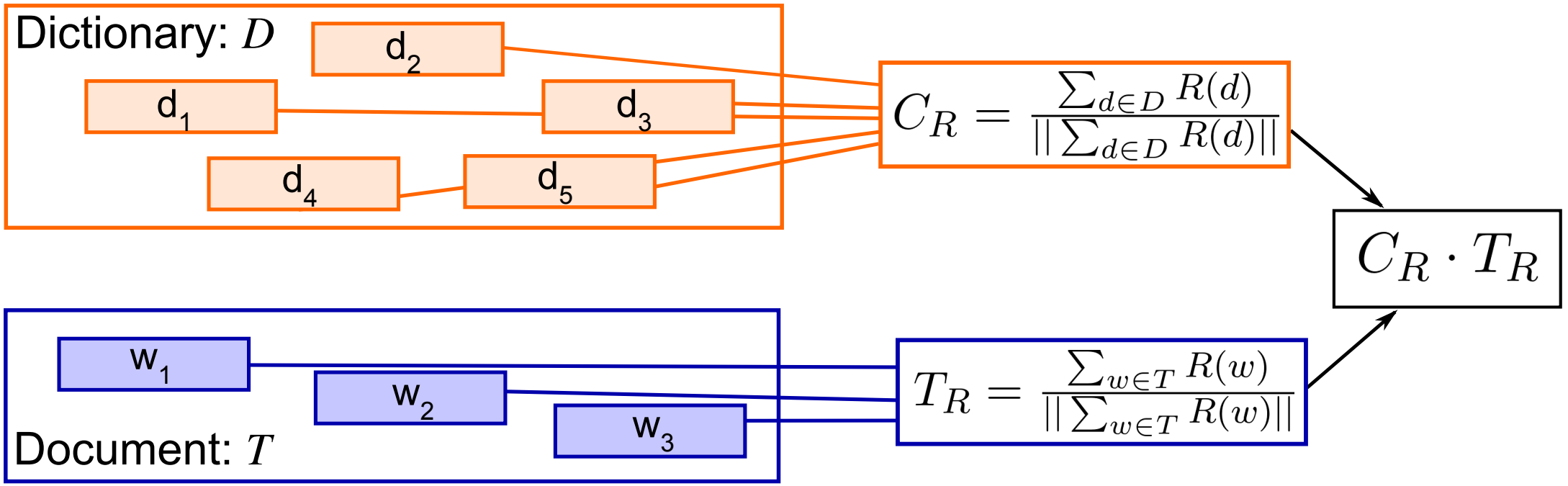
You don't need to fit your own word embeddings model, several alternatives exist that are already trained:

- word2vec (Mikolov, Chen, Corrado, and Dean, 2013)
  - Developed by Google - trained against Wikipedia
  - Based on CBOW and Skipgrams (predict context from word)

- GloVe (Pennington, Socher, Manning, 2014)
  - "global" embeddings using a larger definition of context
  - Developed by Stanford NLP group, similar to word2vec

- fastText (Bojanowski, Grave, Joulin, Mikolov, 2017)
  - Developed by Facebook - trained against Wikipedia in many languages
  - Uses character-level tokenization: it can embed new words based on how they are written (e.g. composite words from the embeddings of lemmas)

Distributed Dictionary Representation



Dictionaries and distributions: Combining expert knowledge and large scale textual data content analysis. J. Garten, J. Hoover, K. Johnson, R. Boghrati, C. Iskiwitch & M. Dehghani. Behavior Research Methods (2018)

Distributed Dictionary Representation

Table 1 Results for Study 1: performance on the 2000 document movie sentiment corpus

Model	Precision	Sensitivity	F1
Full LIWC Dictionary - Word count	0.657	0.659	0.658_a
Full LIWC - Wikipedia embeddings	0.659	0.649	0.654_a
Full LIWC - IMDb embeddings	0.695	0.682	0.689_b
Full LIWC - Google News embeddings	0.715	0.699	0.707_c
Seed LIWC - Wikipedia embeddings	0.665	0.654	0.660_a
Seed LIWC - IMDb embeddings	**0.764**	**0.762**	$\mathbf{0.763}_d$
Seed LIWC - Google News embeddings	0.745	0.723	0.734_e

Table 3 Results for Study 2: method performance averaged across coders and dimensions

Model	M Precision	M Sensitivity	M F1
Full MFD - word count	0.181	0.457	0.275_a
Full MFD - Google News	0.363	0.837	0.485_b
Full MFD - Wikipedia	0.294	0.758	0.405_c
Full MFD - Twitter	0.312	0.764	0.421_d
Seed MFD - Google News	**0.372**	**0.840**	$\mathbf{0.496}_e$
Seed MFD - Wikipedia	0.302	0.755	0.411_f
Seed MFD - Twitter	0.305	0.763	0.415_f

- Two tests: predicting if movie reviews are positive or negative and identifying moral foundations in tweets
- DDR outperforms word counting and performs best when applied on a smaller dictionary - larger dictionaries is not better in this case

Language models

1. The semantic differential

2. Word embeddings

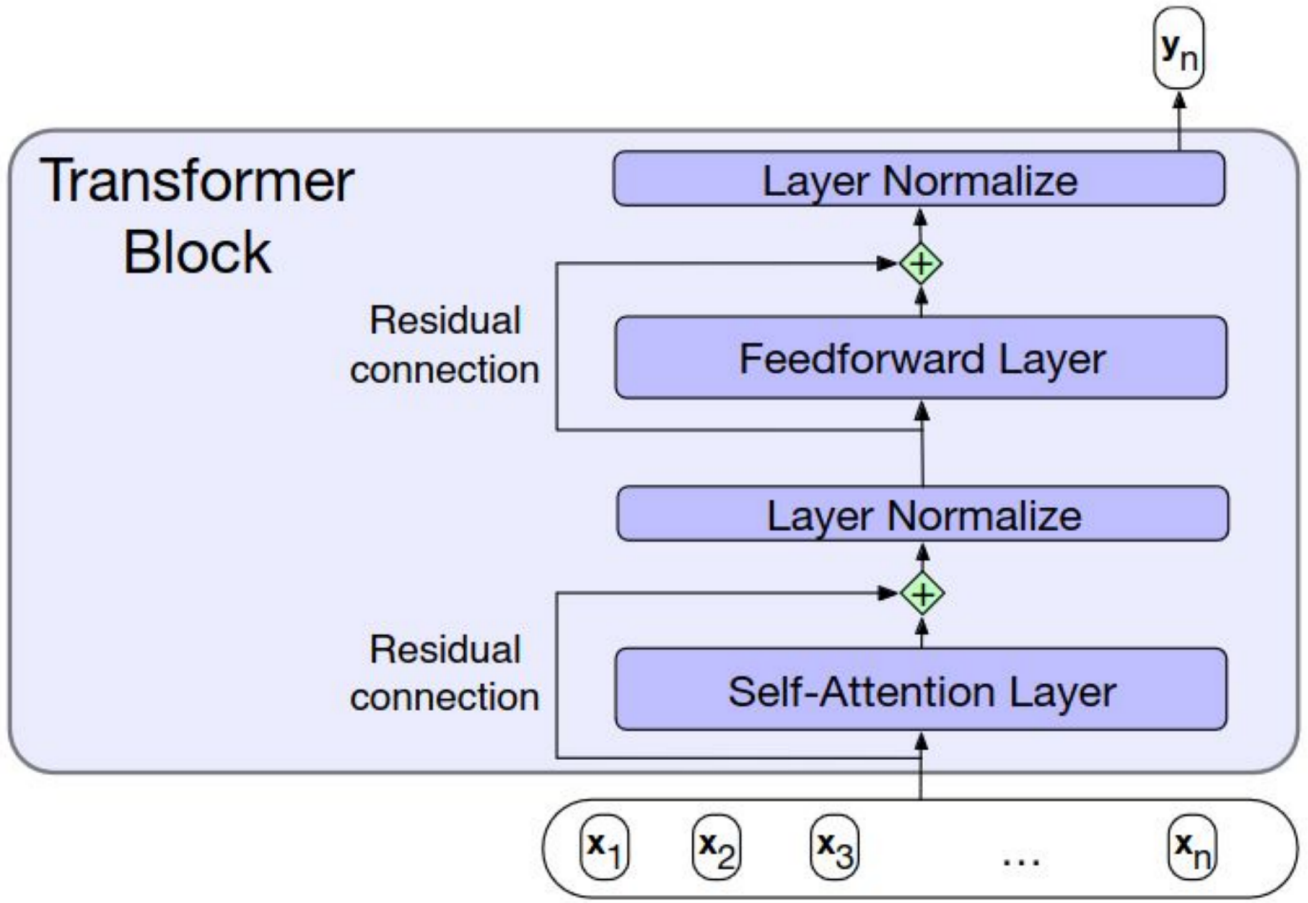
3. Language models

→ From word embeddings to (large) language models

Language models

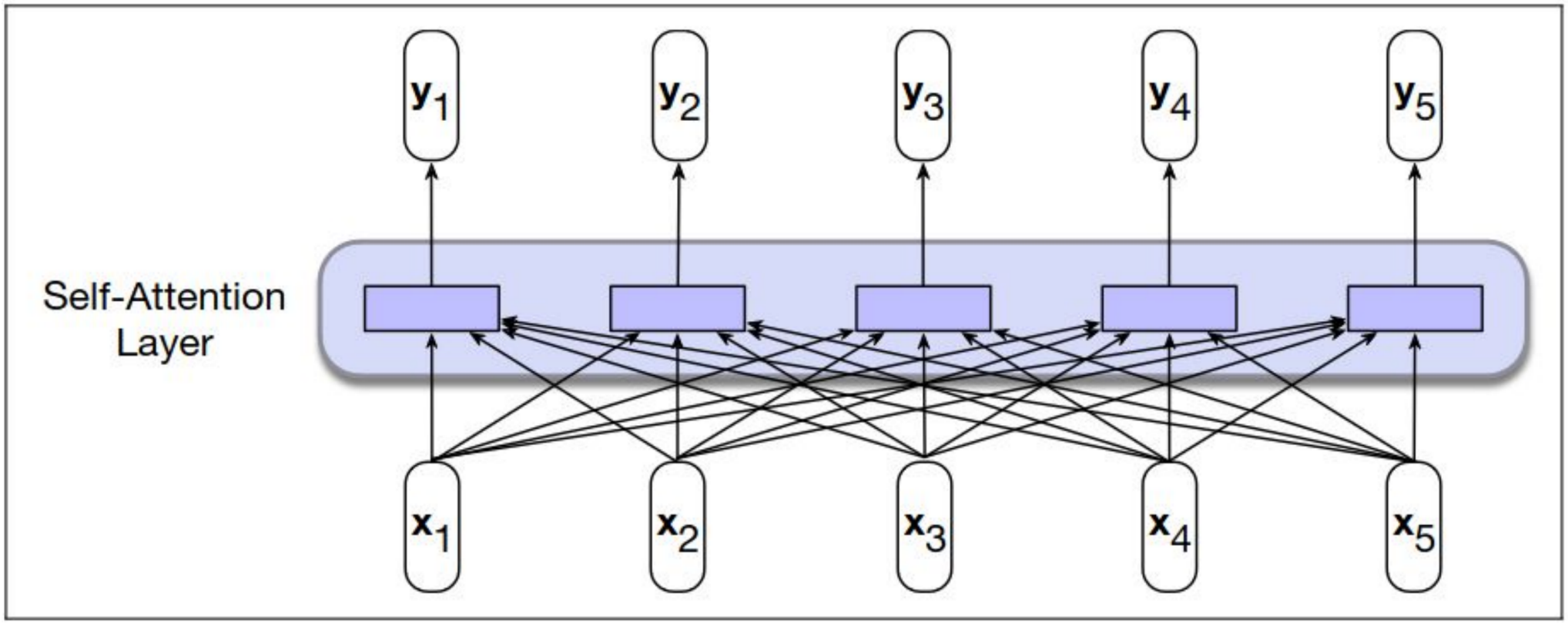
- A model that learns a statistical pattern of occurrence of sequence
- Language models as next word/sentence prediction
  - E.g. autocompletion on mobile keyboard
- Recent language models are based on the Transformers architecture

Transformers



Speech and Language Processing. Daniel Jurafsky & James H. Martin (2023)

Self-Attention



Speech and Language Processing. Daniel Jurafsky & James H. Martin (2023)

4

BERT: Brief Introduction

Bidirectional Encoder Representations from Transformers (BERT)

- Transformers are neural networks which use attention mechanism
- A “large” language model (LM)
  - Base with 12 Transformer layers
  - Large with 24 Transformer layers
- Trained on large collection of text: Wikipedia and BookCorpus
- With several specialized hardware GPUs/TPUs
- Examples of LMs based on Transformers: RoBERTa, XLNET, ELECTRA, GPT-4

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (Devlin et al., 2018)

Word Vectors vs. Contextual LM

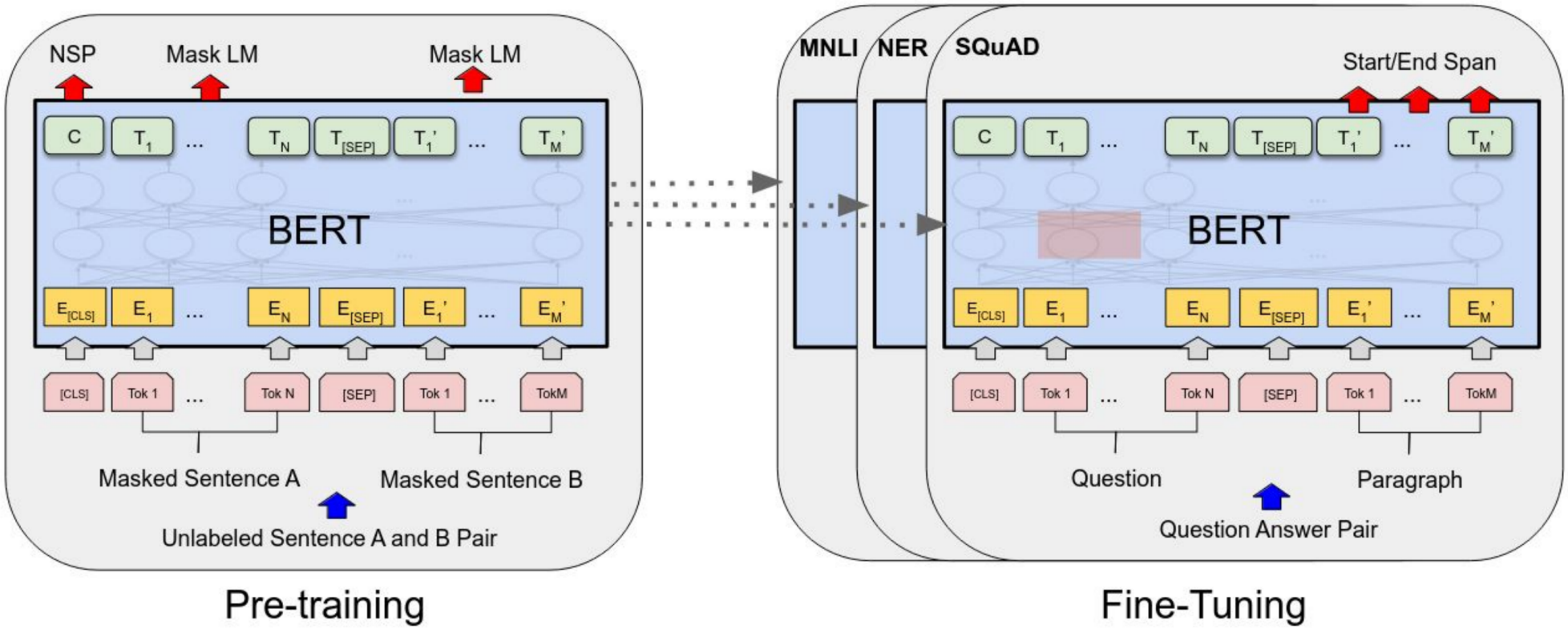
- Word vector: A representation per word e.g. word2vec, GloVe, fasttext
- Contextual LM: A representation per word in context

Richer semantic representation

One model, many applications

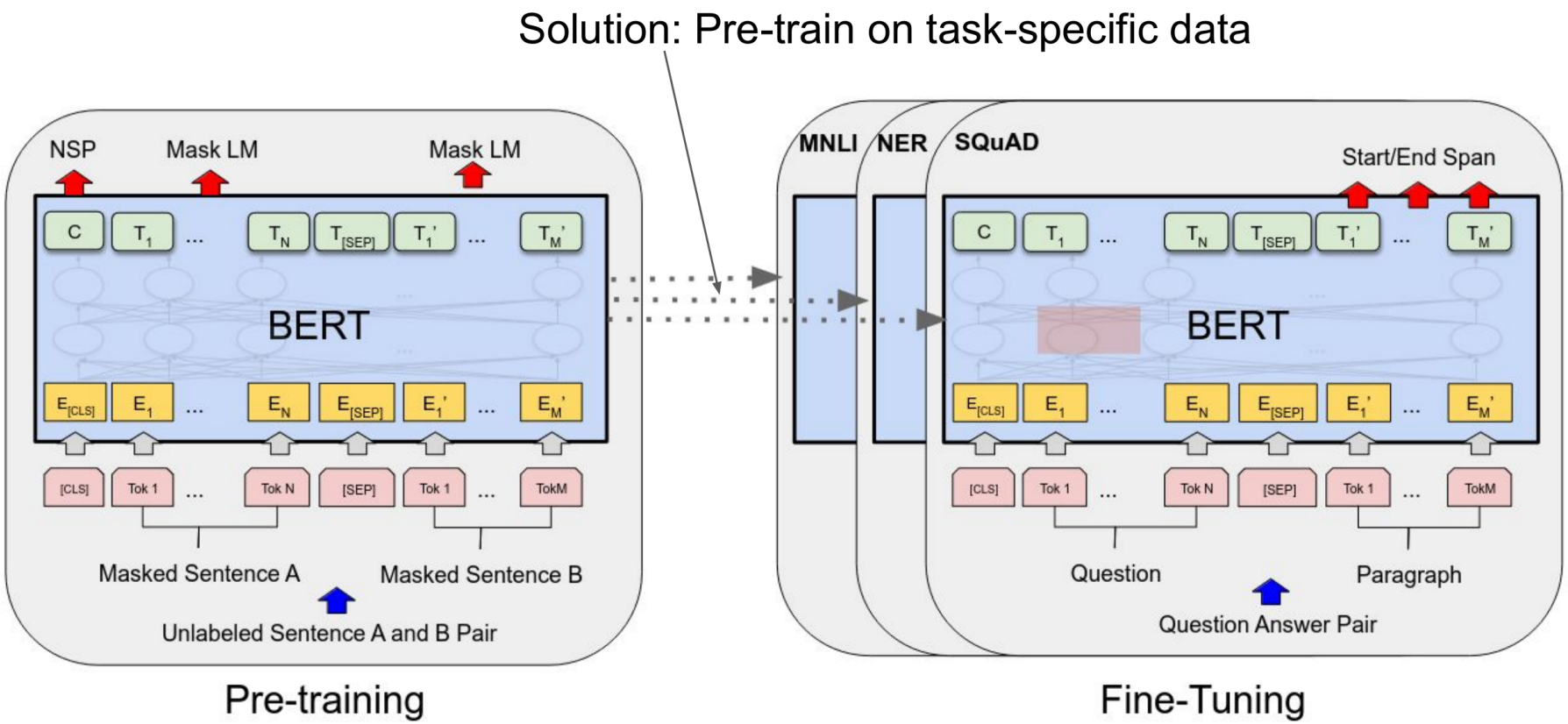
- Model sharing and reuse
- 🤗 HuggingFace library and hub
- From a representation to several applications:
  - Topic classification
  - Natural language inference
  - Question answering
  - Sentiment analysis
  - Emotion detection
  - Machine translation

Pre-training and Fine-tuning



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (Devlin et al., 2018) [8]

Problem: Domain gap



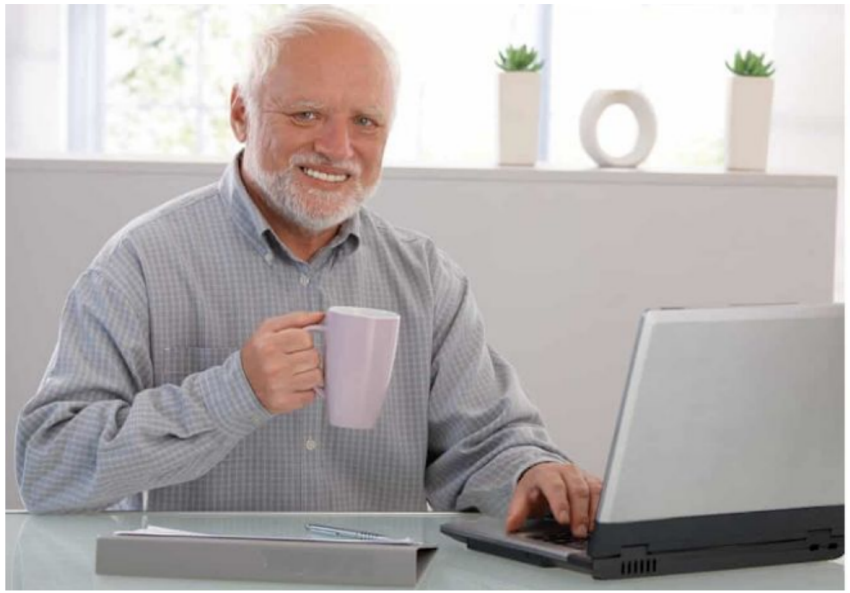
9

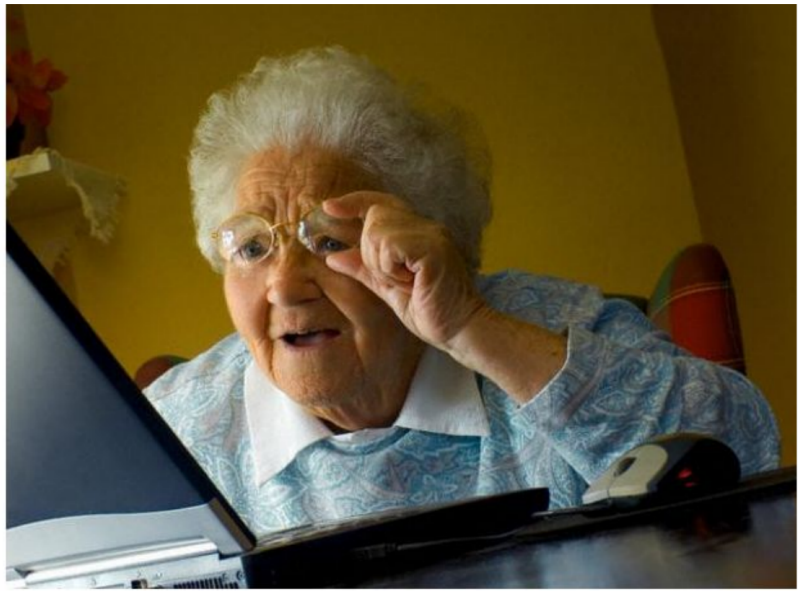
Examples: SciBERT, TwitterRoBERTa, BERTweet

LEIA: Linguistic Embeddings for the Identification of Affect (Aroyehun et al., 2023, preprint)

Emotion identification in text

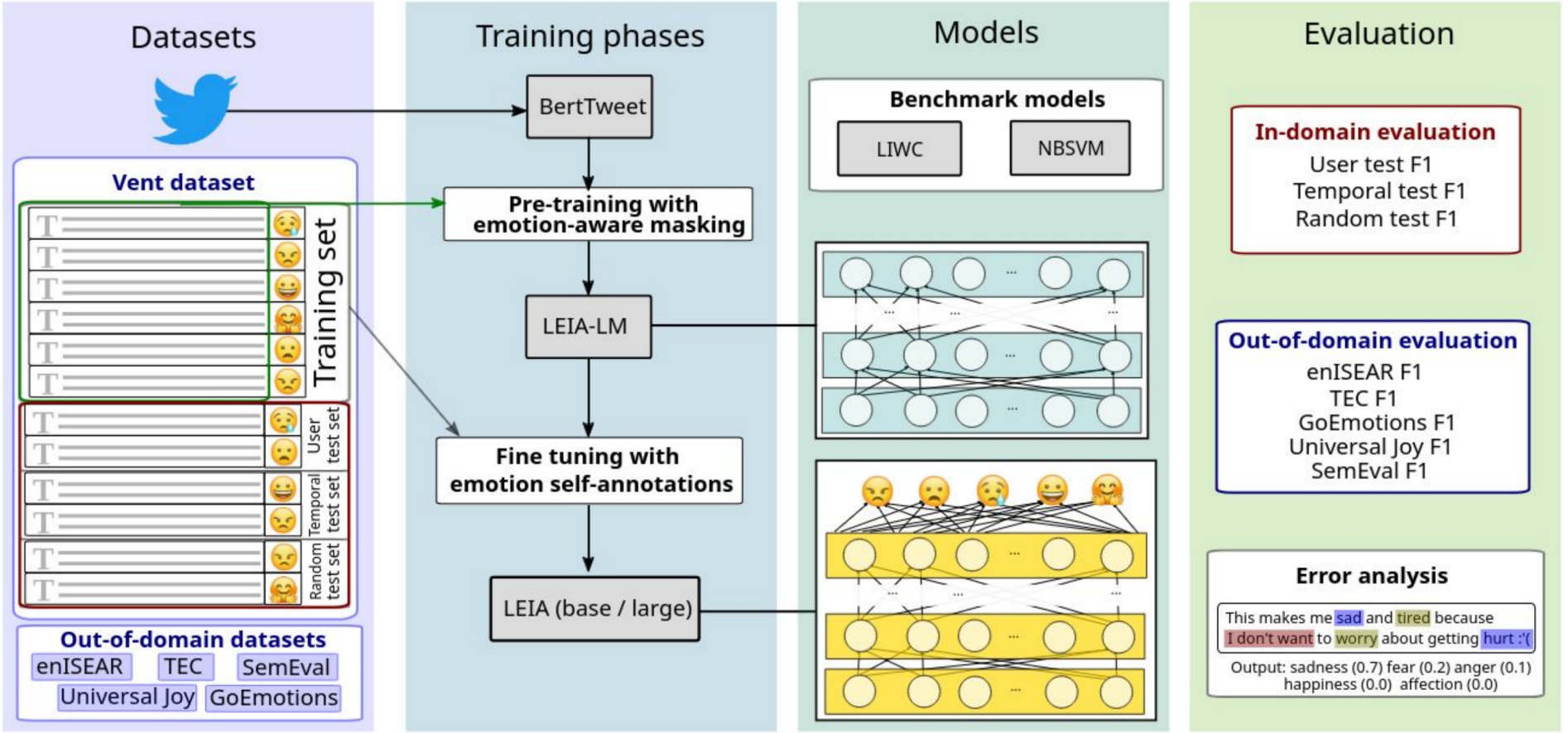
Existing models are trained mainly on datasets annotated by readers than writers





11

LEIA: An overview



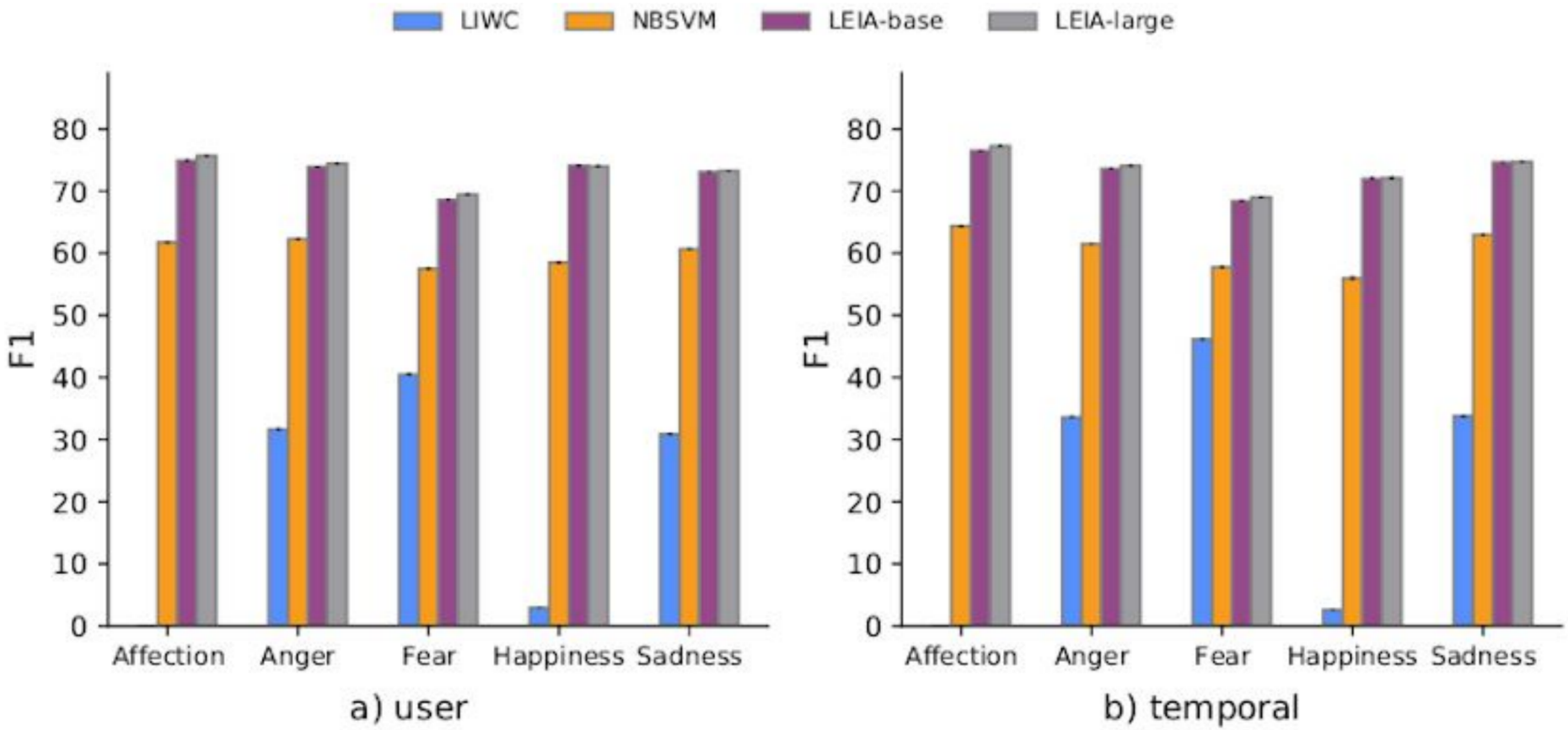
Datasets

- Vent as a source of self-annotated dataset
- Sharing emotions at scale: The Vent dataset (Lykousas et al., 2019) : A dump of 33M posts
- Labels : Affection, Anger, Fear, Happiness, Sadness
- In-domain evaluation on user, temporal, and random splits
- Out-of-domain evaluation: Universal joy, GoEmotions, enIsear, TEC, and Semeval
- Out-of-domain (OOD) label groupings exclude Affection

Training the LMs

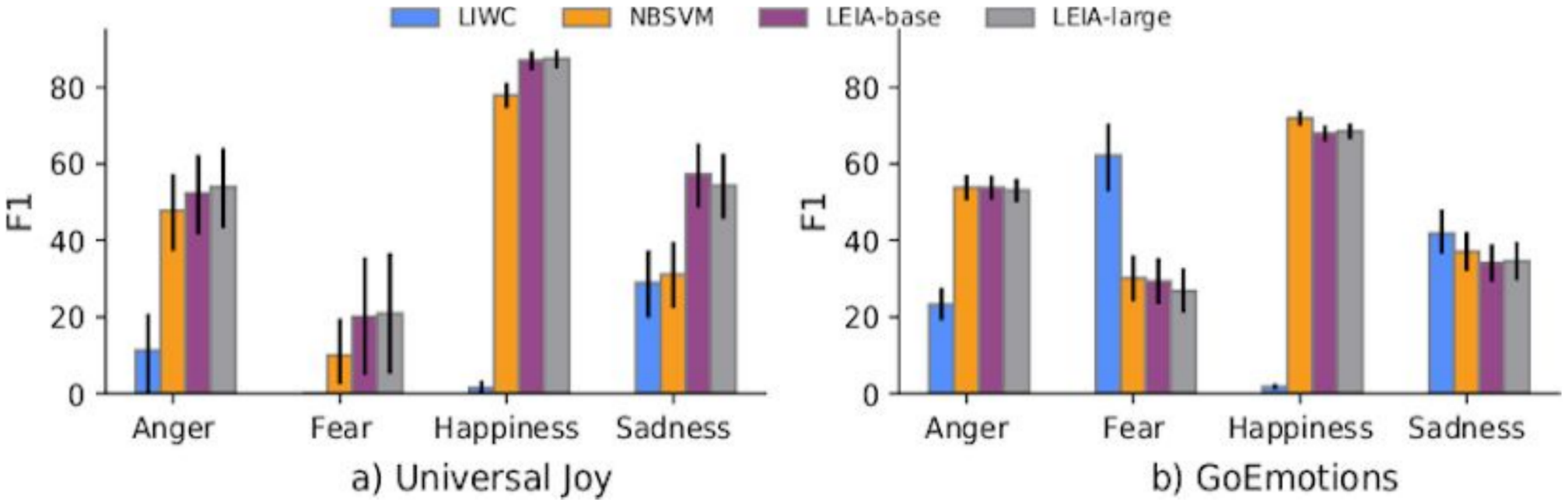
- We experiment with two LMs : BERTweet-base and BERTweet-large
- Adaptation by pre-training with selective masking of emotion words on unlabeled data
- Fine-tuning on labeled data
- We evaluate on unseen in-domain and out-of-domain data in comparison with two other models

Results in-domain



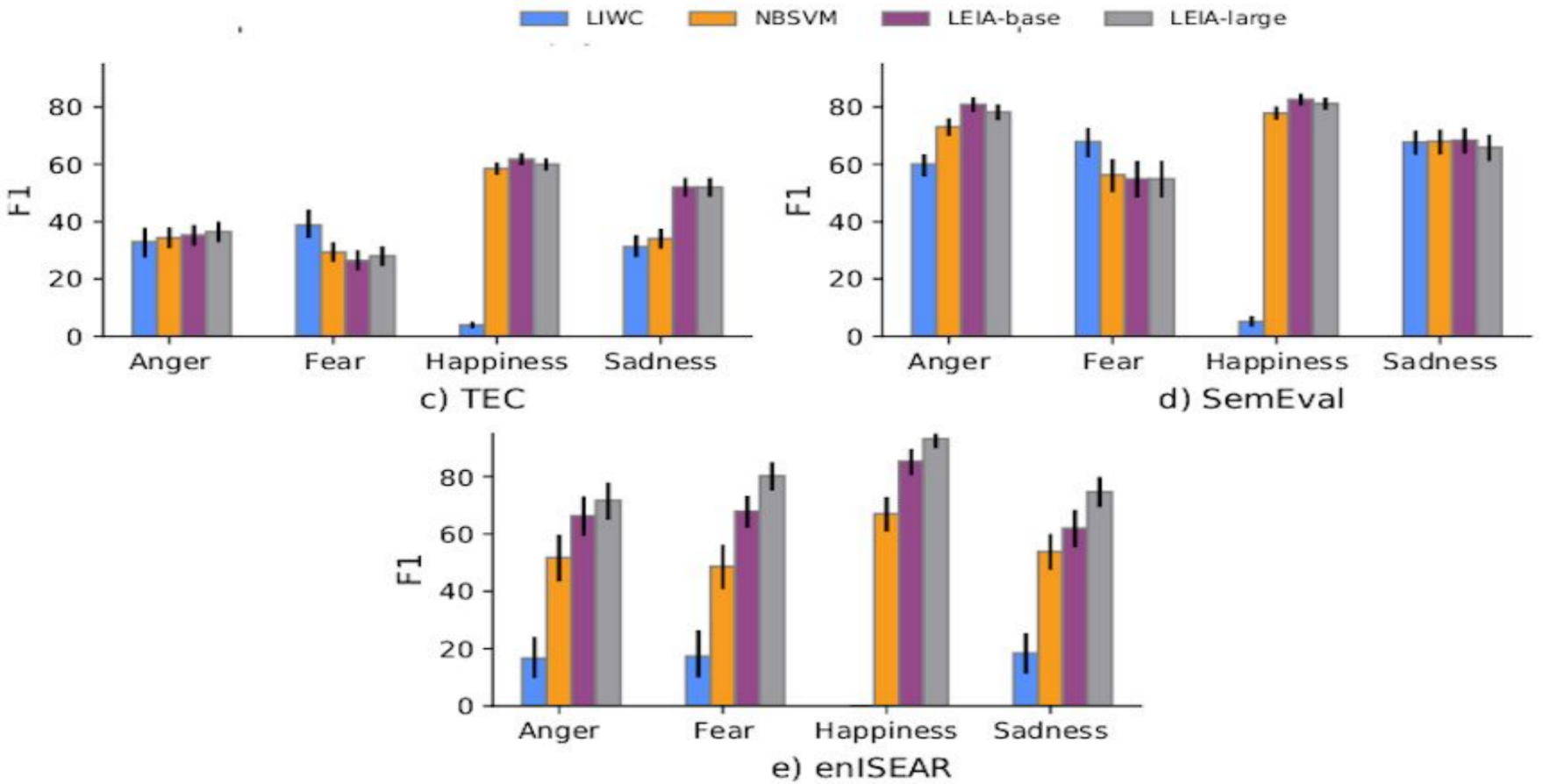
15

Results out-of-domain I



16

Results out-of-domain II



17

LEIA is on 🤗 HuggingFace hub

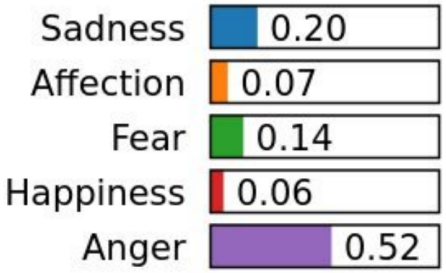
Two models in two sizes:

- Adapted LM (base and large)
- Emotion classification model (LEIA-base and LEIA-large)

https://huggingface.co/LEIA

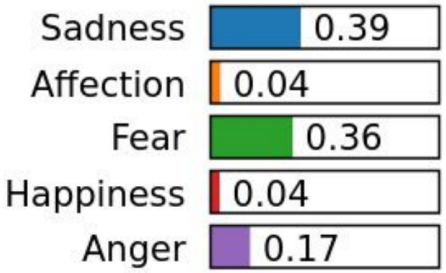
18

Sample model prediction and attributions



I felt |mask| because one of my children lied. I taught them not to, but natural sin came out and I was surprised to find myself so cross.

Actual label: Anger



I felt |mask| when my house from a few years ago got damaged by burst pipes and flooded internally. Most of our belongings were destroyed and we had to move out for nearly 18 months while the house dried out and was refurbished. It was an awful lot of hassle and upheaval.

Actual label: Sadness

Summary

- Semantic Differential
  - Connotative vs. denotative meaning
  - Three dimensions of meaning: Evaluation, Potency, and Activation
- Word embeddings
  - Cosine distance between vectors as a measure of similarity
  - Latent semantic analysis as a solution to the curse of dimensionality
  - Distributed representation
  - Distributed dictionary representation
- Language models
  - Pre-training and fine-tuning
  - Adaptation to specific tasks/domains before fine-tuning
  - LEIA as an example of adaptation for emotion classification

Thank you for listening!