Comparing Text Representations: A Theory-Driven Approach

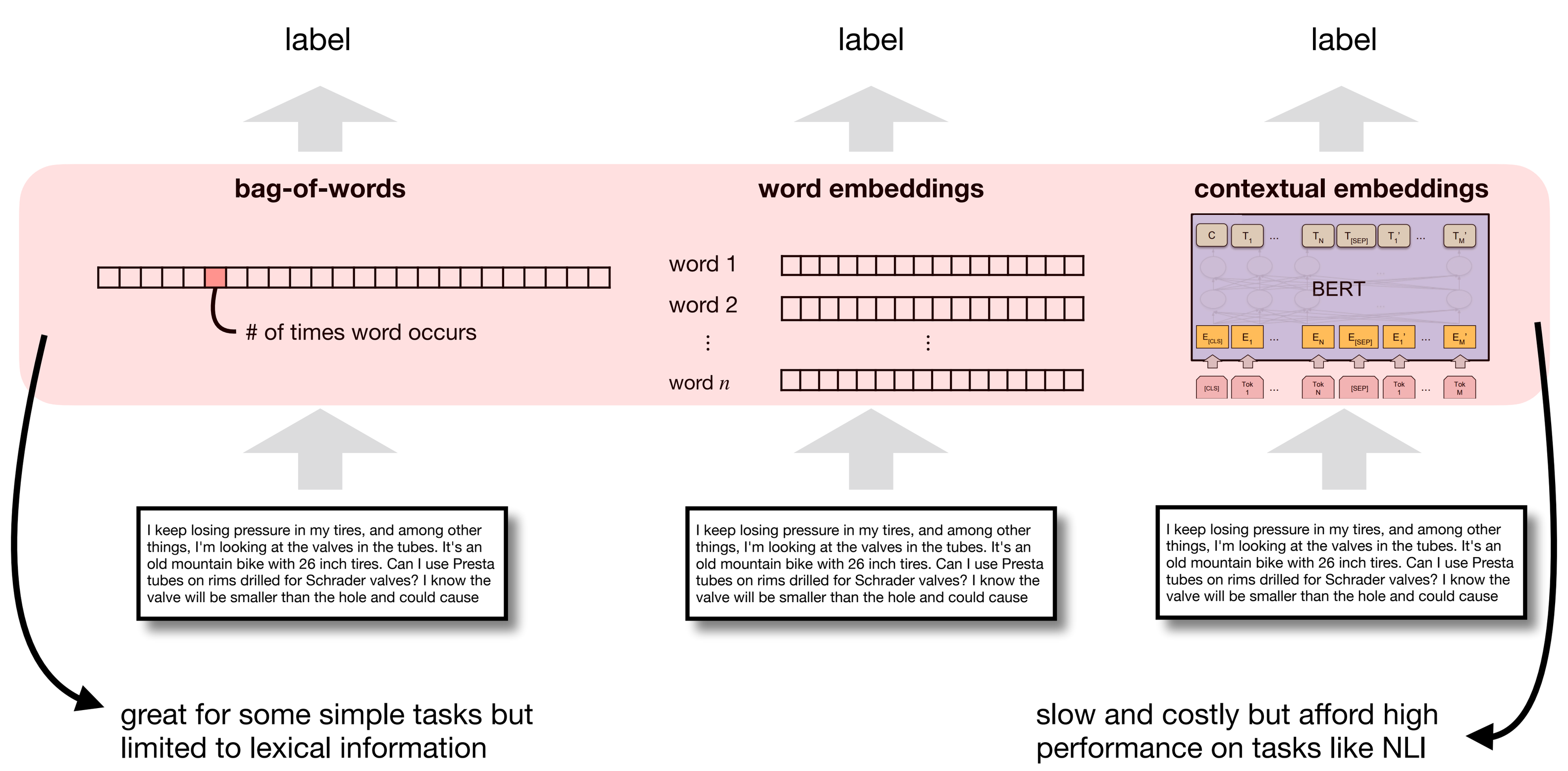
Gregory Yauney
Cornell University
gyauney@cs.cornell.edu

David Mimno
Cornell University
mimno@cornell.edu

What makes some text classification tasks difficult while others are easy?

How can we tell if a task will be difficult without training any classifiers?

We know text representations impact task difficulty:

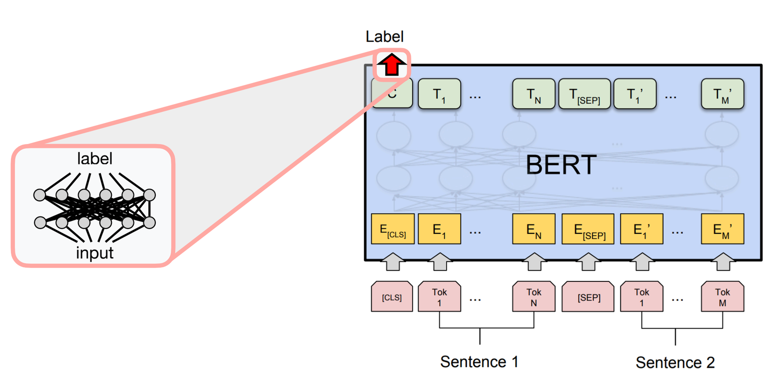


- you want to **solve a task**: do you need the expensive model?
- you want to **build a challenge dataset**: can you be sure it's hard as you want it to be?
- you want to **interpret** model performance: which properties of embeddings affect classification performance?

What factors make a task + representation easy or hard?

Analyze representations, not models

- Small networks are easier to analyze than the large models in NLP
- But they are good analogs of the final classification layers in large language models, so we can use them to study representations



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, 2019

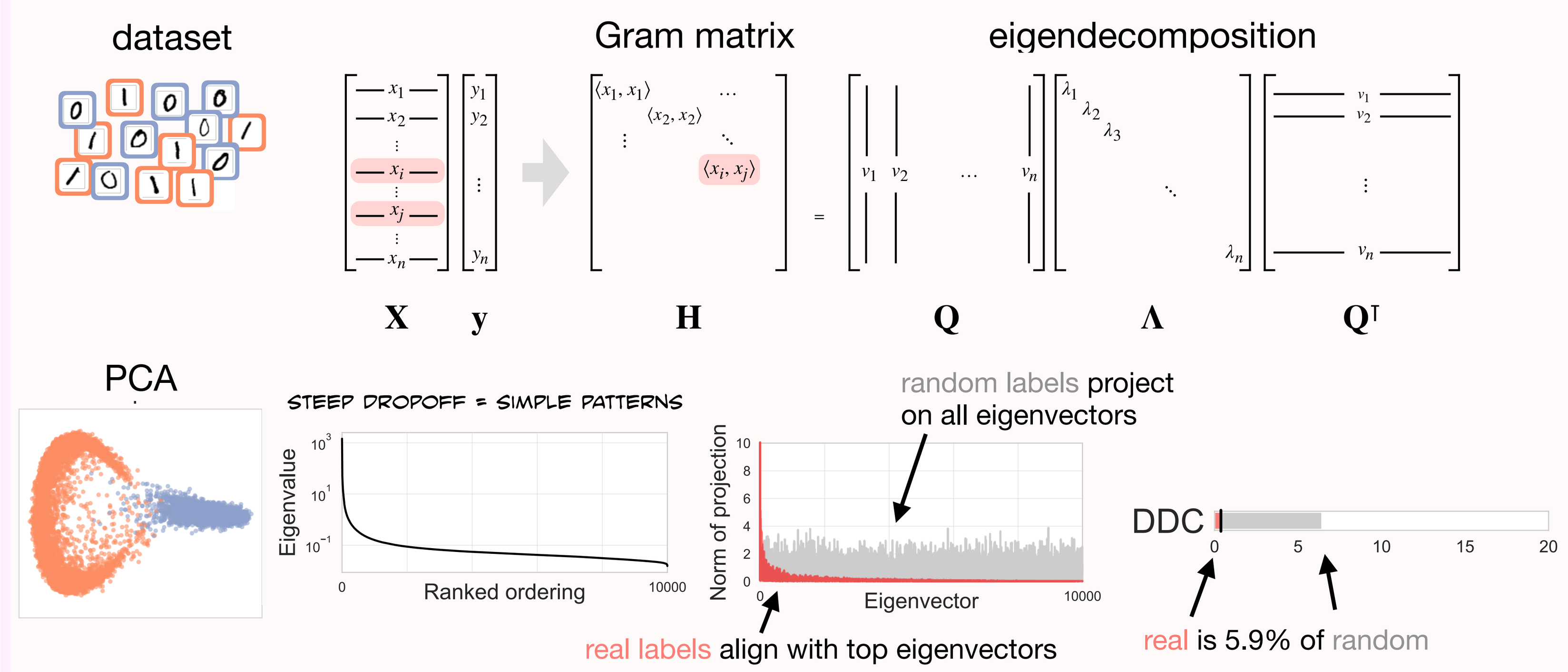
We translate and adapt **data-dependent complexity** (Arora & al. 2019)* to text datasets.

Evaluation patterns:

1. For a given target labeling, is one or another representation more effective?
2. For a given representation, are some labelings more or less compatible with that representation?
3. How can we measure and explain the difficulty of text classification problems between datasets?

Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks, Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang, 2019.

Data-dependent complexity: patterns in data + alignment with labels (Arora & al., 2019)

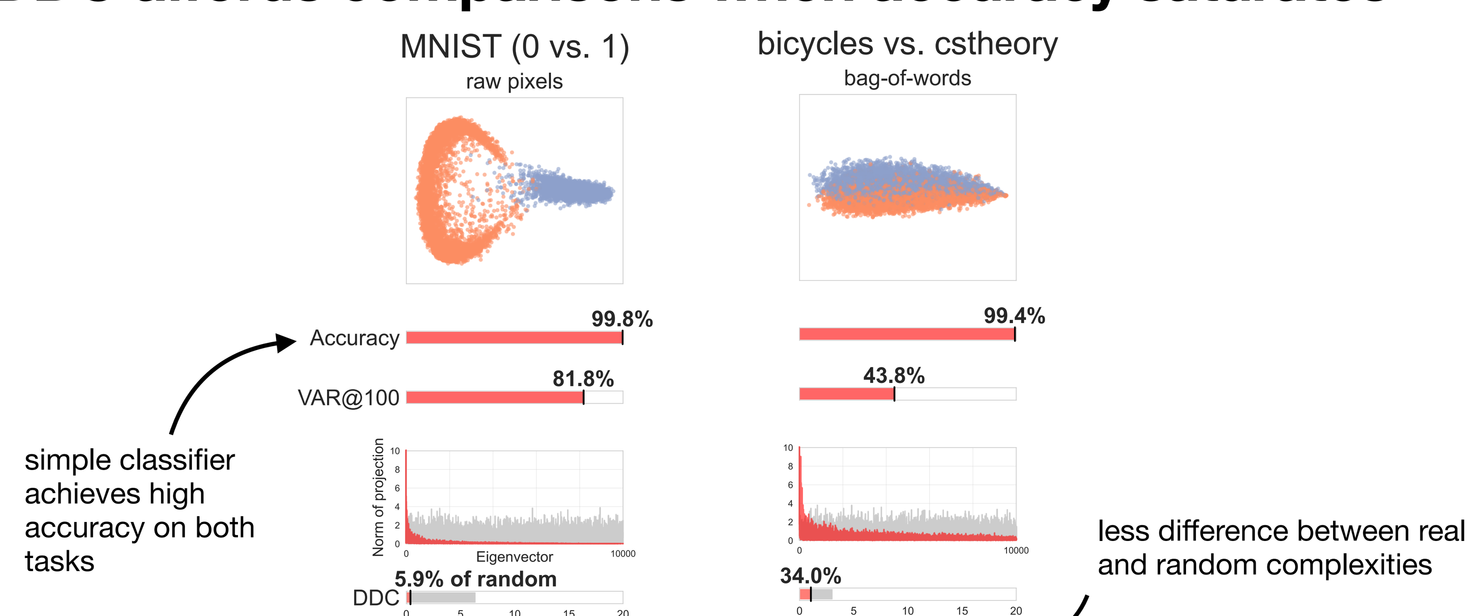


Making DDC a practical tool for text datasets

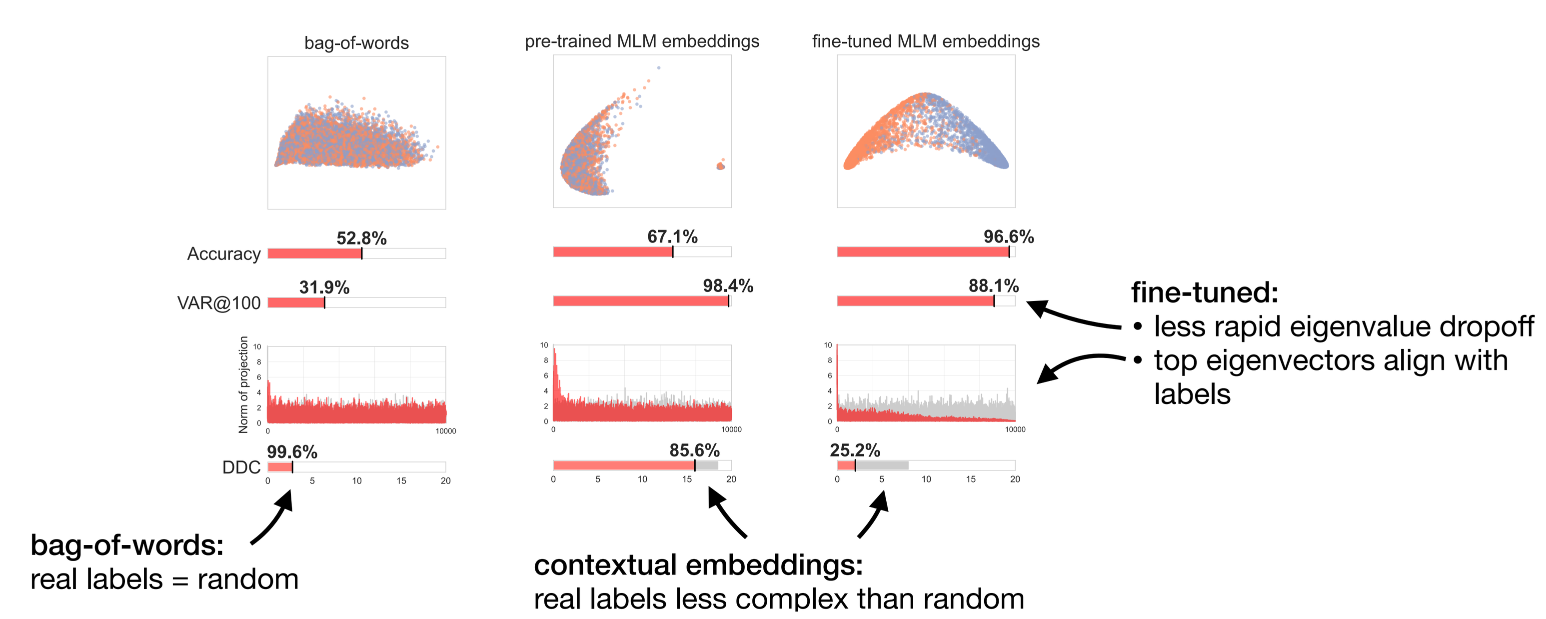
1. Compare to *distribution* of random labels
2. Sampling from large datasets is effective
3. DDC is sensitive to duplicates

DETAILS IN THE PAPER!

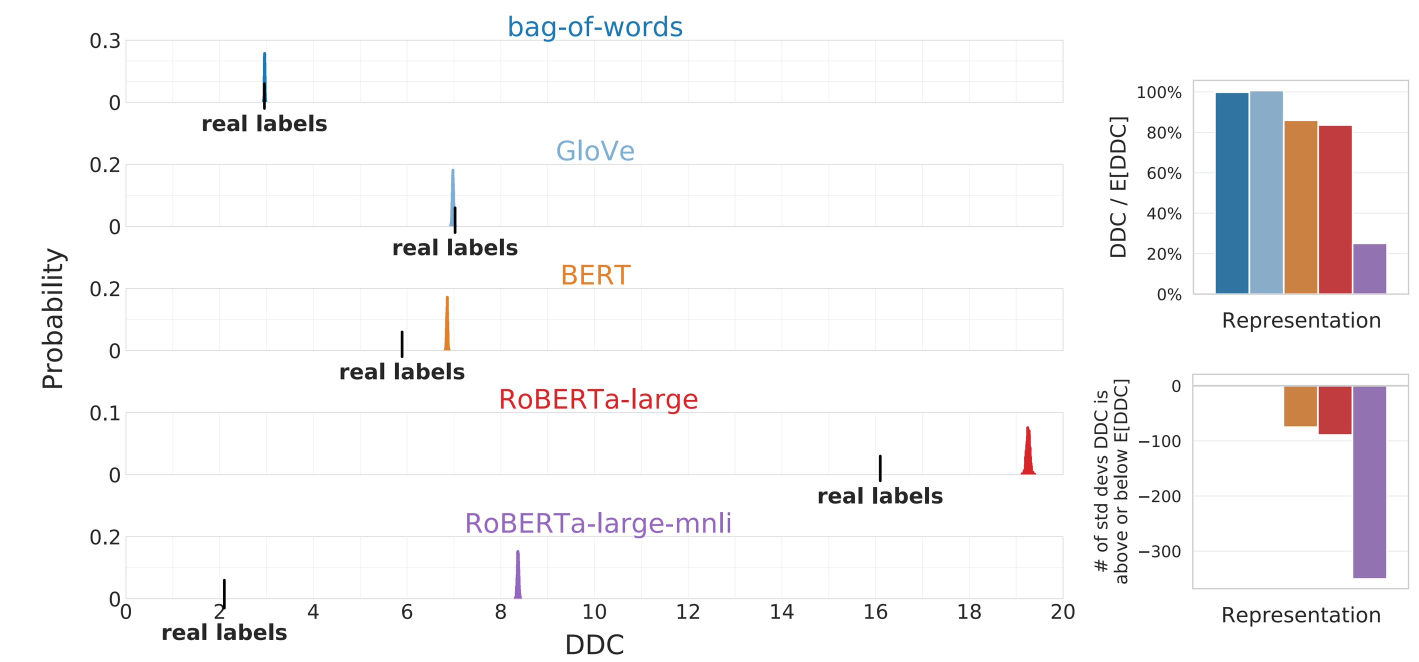
DDC affords comparisons when accuracy saturates



MNLI: contextual embeddings distinguish real and random labels



Case study: MNLI with different representations



1. Real labels are as complex as random labels for bag-of-words.
2. Pre-trained and fine-tuned representations separate real and random labels.

Case study: comparing NLI datasets

MNLI, QNLI, SNLI: pre-trained and fine-tuned representations separate real and random labels.

WNLI: all evaluated representations don't distinguish real and random labels

"How can *I* use this?"

- If you're a **model builder**: get more information about label-representation alignment
- If you're a **dataset designer**: make sure no existing representations separate real and random labels
- If you're interested in **interpretability**: study other changes to embeddings

github.com/gyauney/data-label-alignment