VarCLR: Variable Semantic Representation Pre-training via Contrastive Learning


Qibin Chen


Jeremy Lacomis


Edward J. Schwartz


Graham Neubig


Bogdan Vasilescu


Claire Le Goues

Carnegie Mellon University
School of Computer Science

Variable Names are an Important Part of a Project

Variable Names are Important to Human Programmers

```
def minimumTotal(self, t):
    return reduce(
        lambda a, b: [f + min(d,
e) for d, e, f in zip(a, a[1:],
b)], t[::-1]
    )[0]
```

120. Triangle

Medium 4795 380 Add to List Share

Given a `triangle` array, return *the minimum path sum from top to bottom*.

For each step, you may move to an adjacent number of the row below. More formally, if you are on index `i` on the current row, you may move to either index `i` or index `i + 1` on the next row.

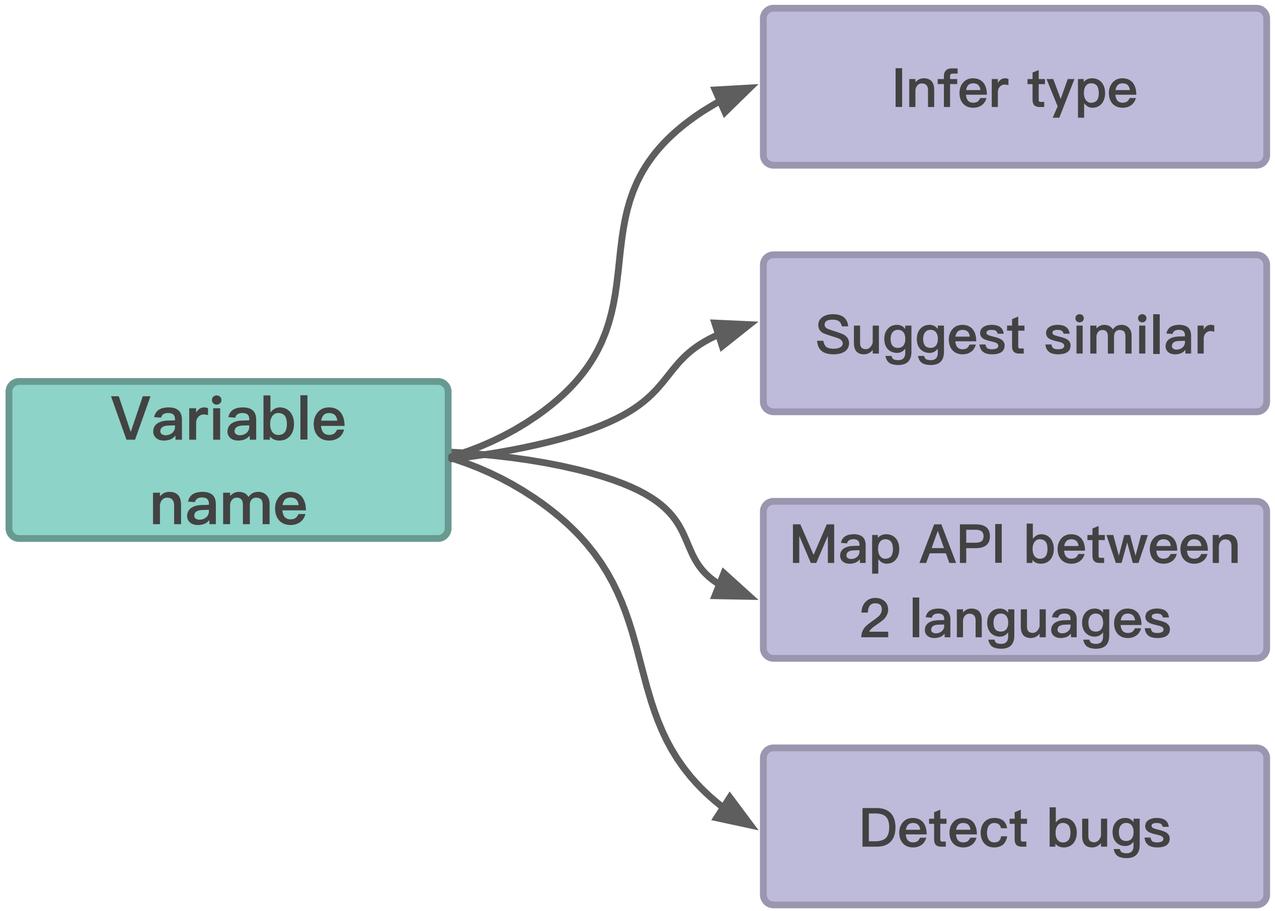
Example 1:

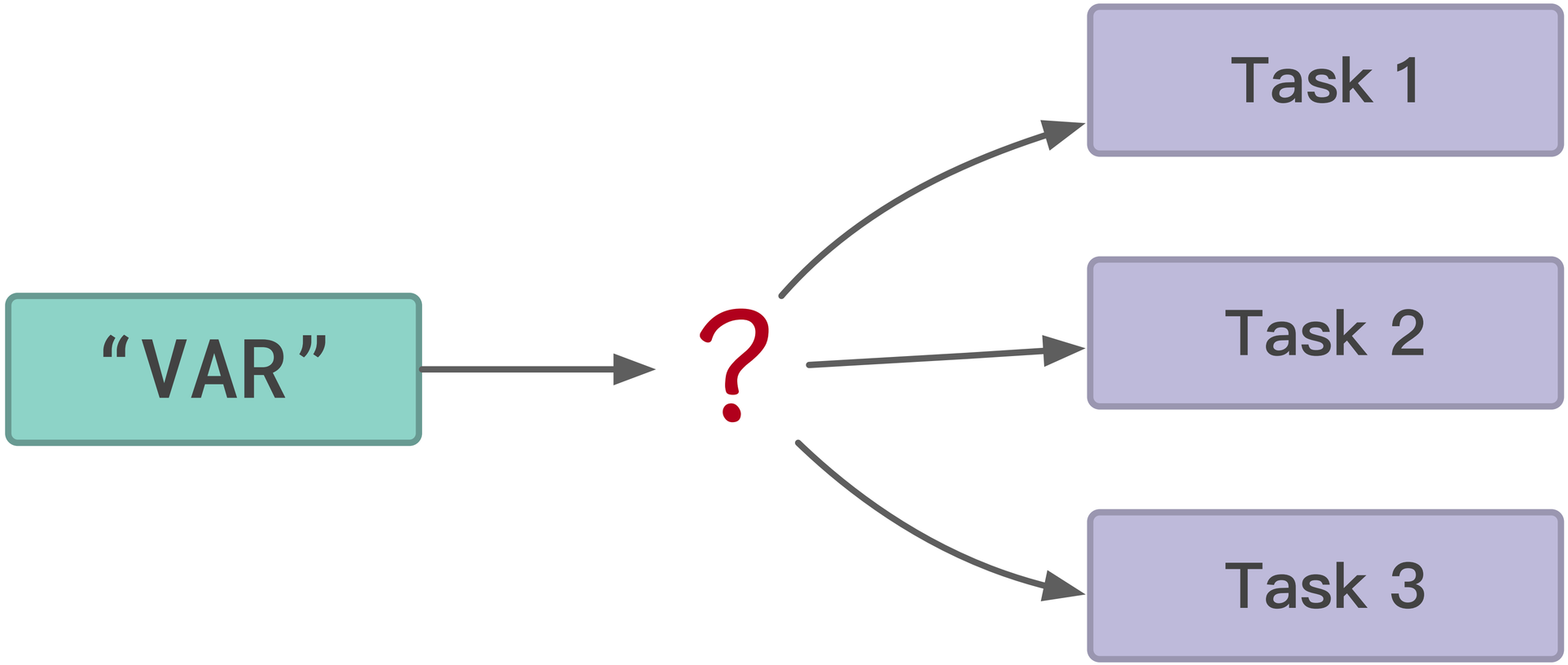
```
Input: triangle = [[2],[3,4],[6,5,7],[4,1,8,3]]
Output: 11
Explanation: The triangle looks like:
   2
  3 4
 6 5 7
4 1 8 3
```

“Full word > abbreviation >> single character”
(128 participants)

–Lawrie et al., 2006, “What’s in a Name?”

Variable Names are Important to Name-based Analysis Tasks



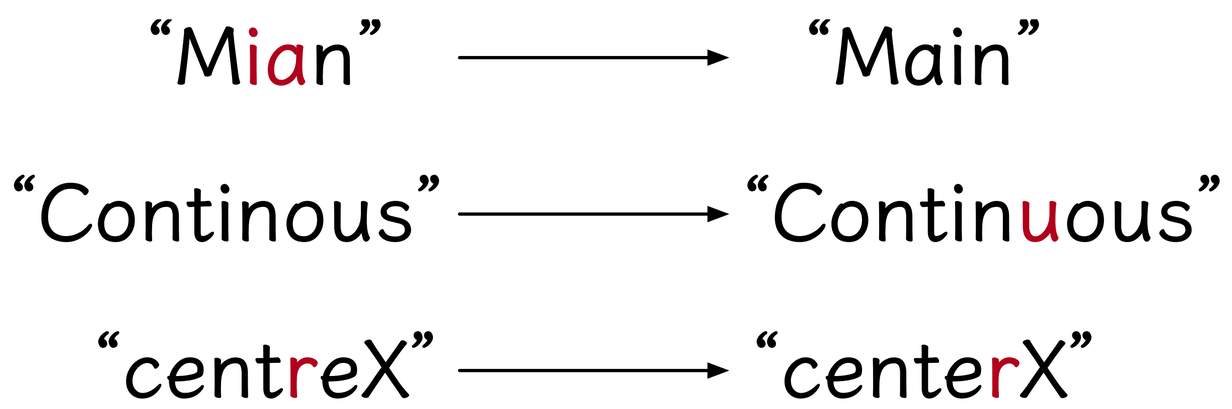


This variety of applications have a shared underlying problem:
Variable Representation

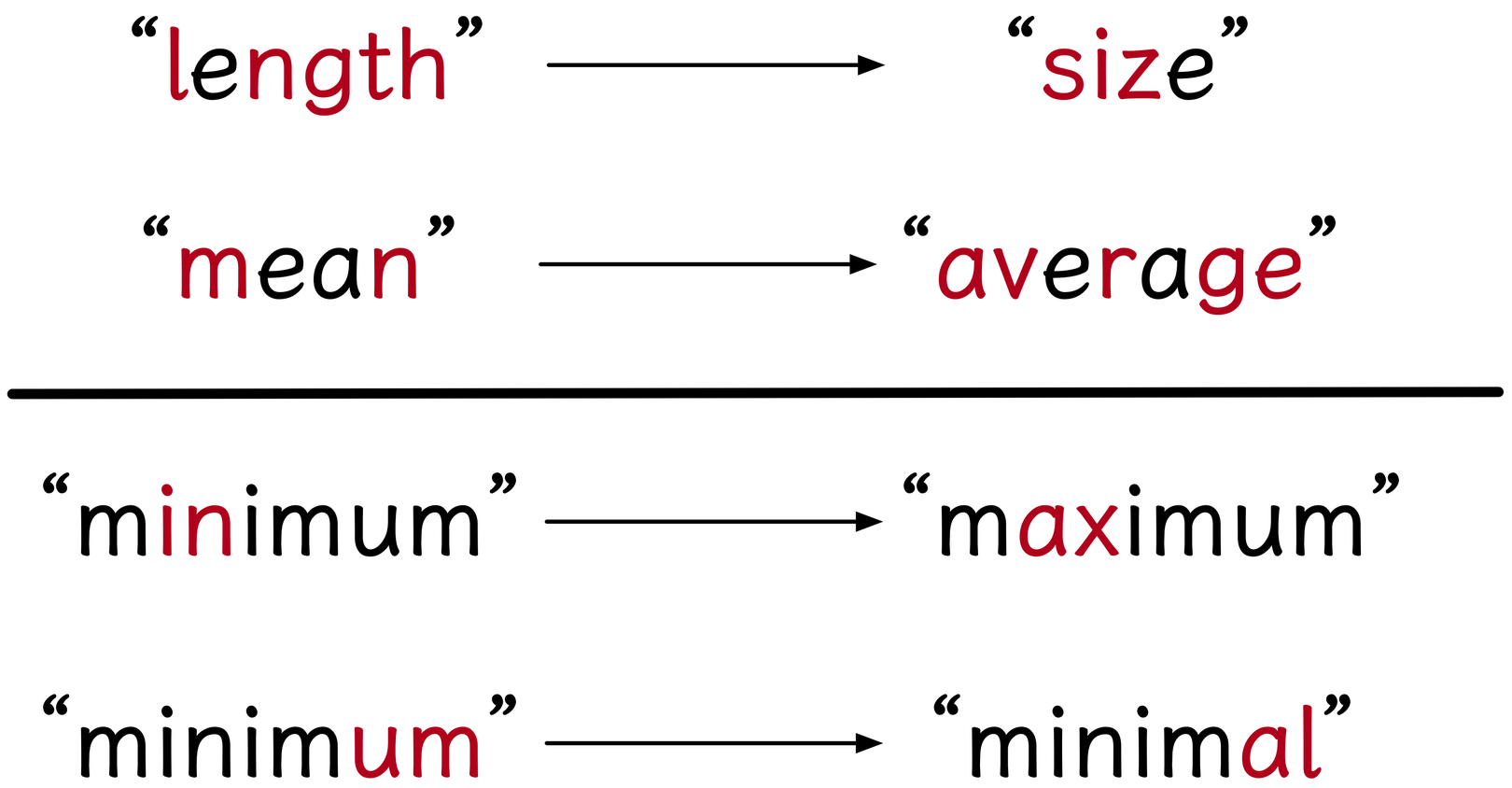
What Properties Make Good Variable Representations?

- Character similarity
- Open vocabulary
- Subword importance and order
- ⭐ **Relatedness**
- ⭐ **Similarity** **(interchangeability)**

Desired Property of Variable Representation: Character Similarity



Character Similarity Is Not Enough and Can Be Misleading

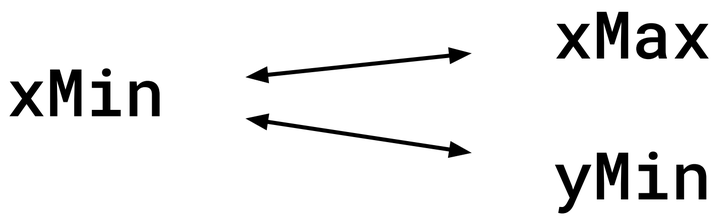


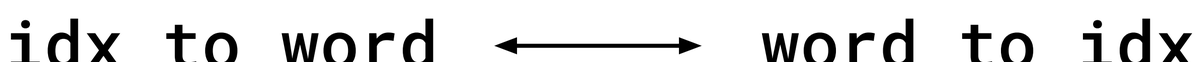
Desired Property of Variable Representation: Open Vocabulary

- Variables are like languages with no whitespaces
  - `arrayListCompletedFromForm_withoutDuplicate`
- Split by `camelCase` or `snake_case`?
- Arbitrary abbreviations and contractions
  - `filenames, fnames, displayMessage, displayMsg`

Desired Property: Subword Importance and Order

- Subword importance
- Subword importance can change dynamically
- Subword order affects the meaning





What Properties Make Good Variable Representations?

- Character similarity
- Open vocabulary
- Subword importance and order
- ⭐ **Relatedness**
- ⭐ **Similarity** **(interchangeability)**

Desired Property of Variable Representation: Relatedness

Relatedness: The Distributional Hypothesis

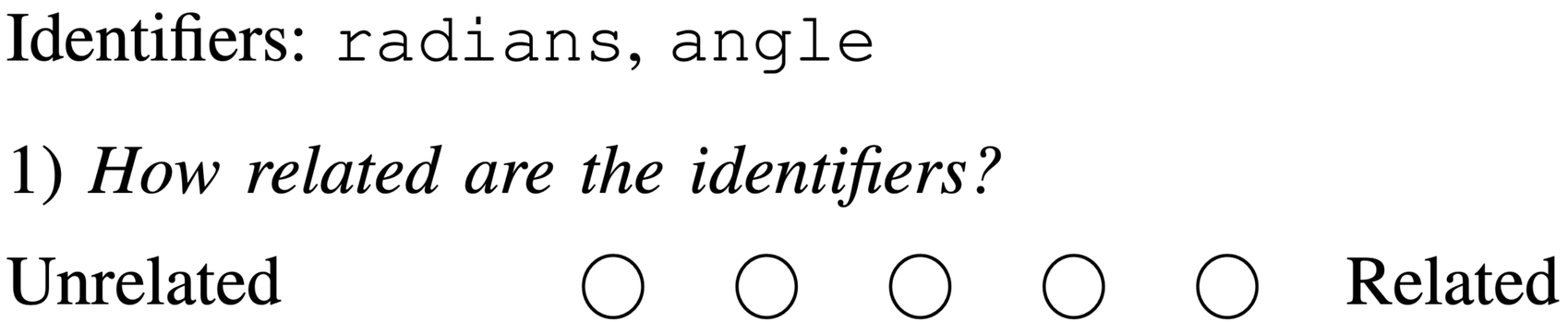
- Words that occur in the same contexts tend to have similar meanings (Harris, 1954)
- “A word is characterized by the company it keeps" was popularized by (Firth, 1957)

J. R. Firth, English linguist

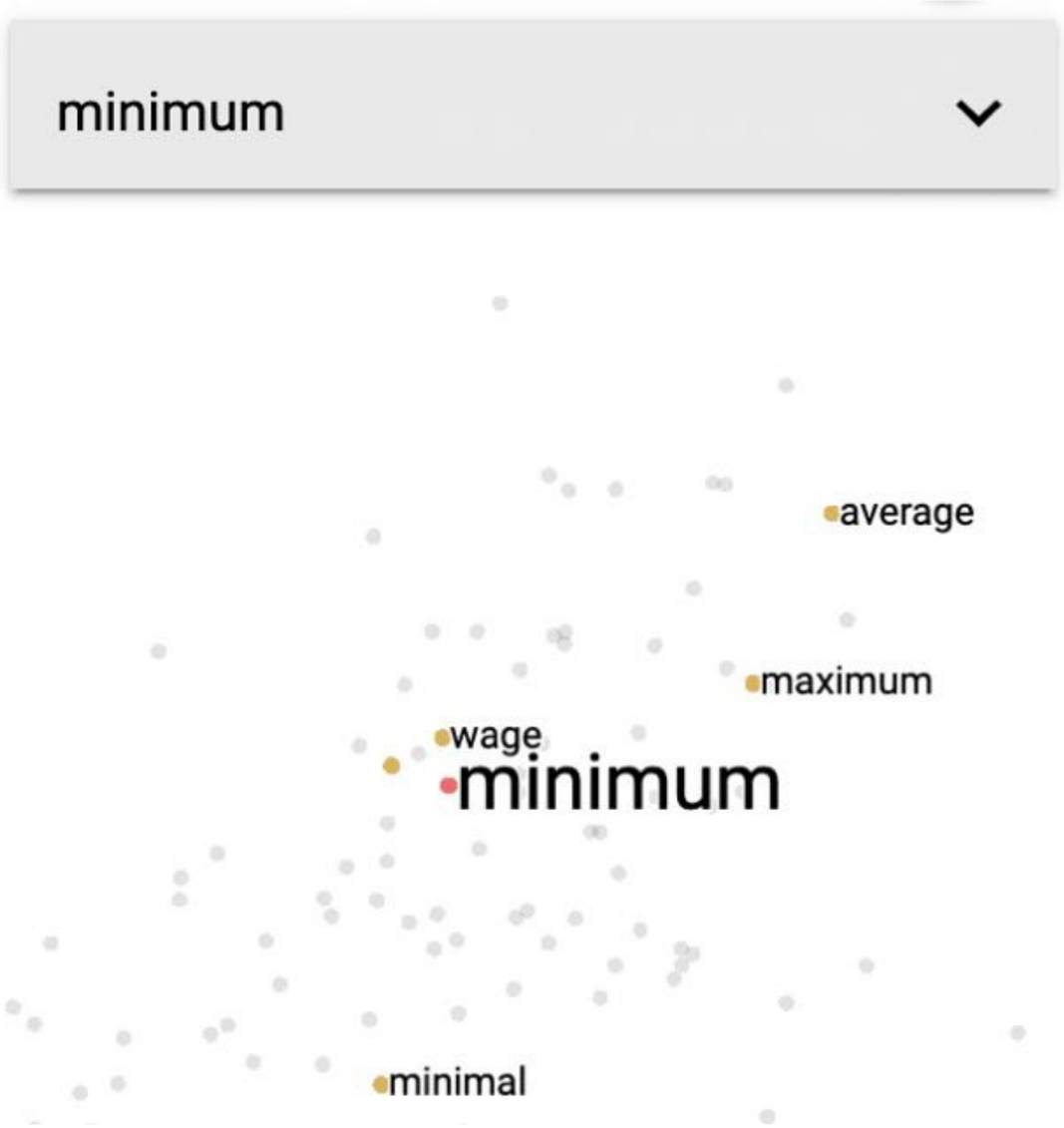
Relatedness: Evaluation Benchmarks and State-Of-The-Art Methods

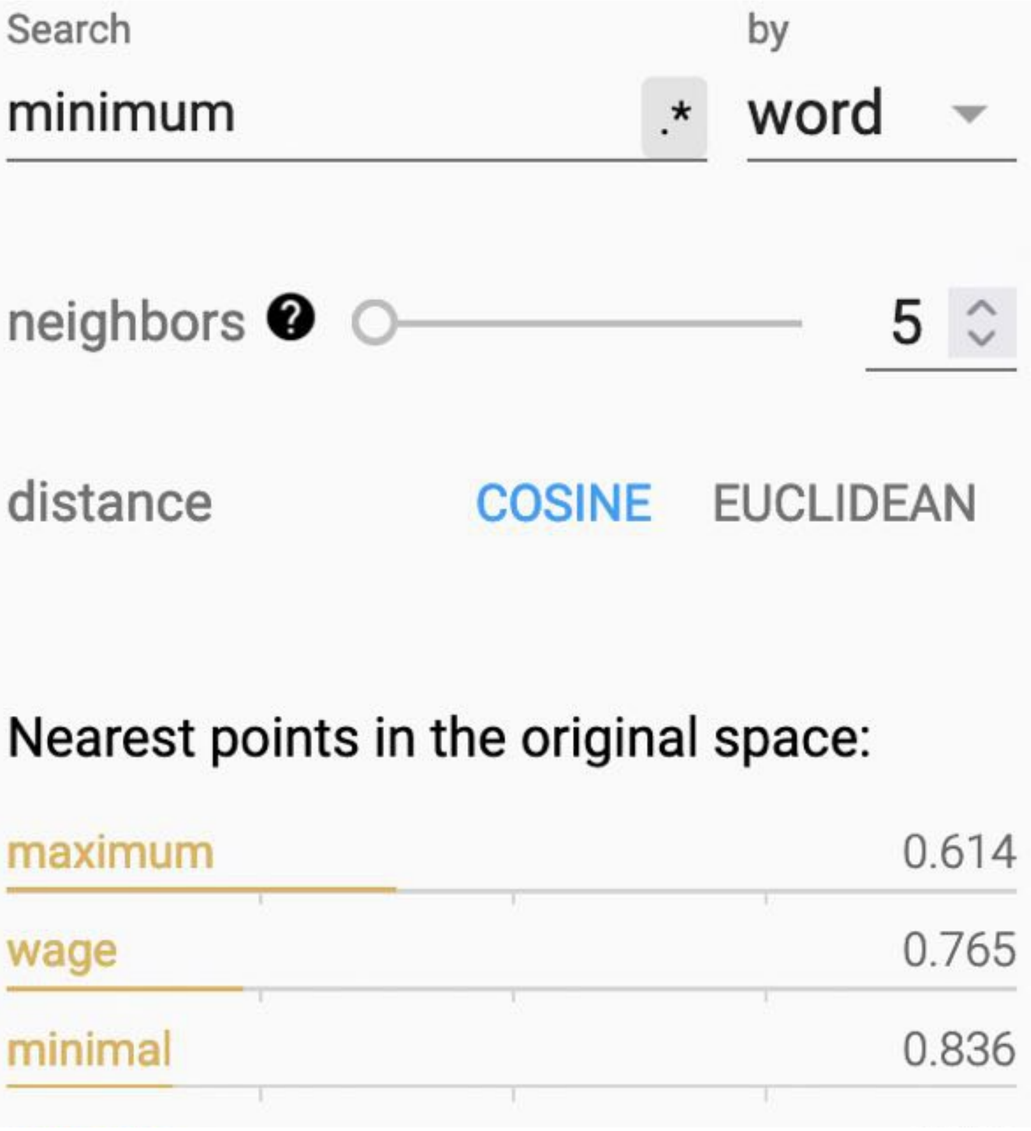
- **IdBench** (Wainakh, et al., ICSE 2021)
  - Word2vec
  - FastText
  - Path-based

Current Methods Are Effective at Representing Relatedness!



Limitation of Relatedness: It Is Not All You Need





https://projector.tensorflow.org/

Desired Property of Variable Representation: Similarity (substitutable)

Desired Property of Variable Representation: Similarity (substitutable)

	FastText (IdBench)	Human
`IdBench(`paddingTop`,`paddingRight`)` ⟶	0.89	0.07
`IdBench(`minText`,`maxText`)` ⟶	0.95	0.05

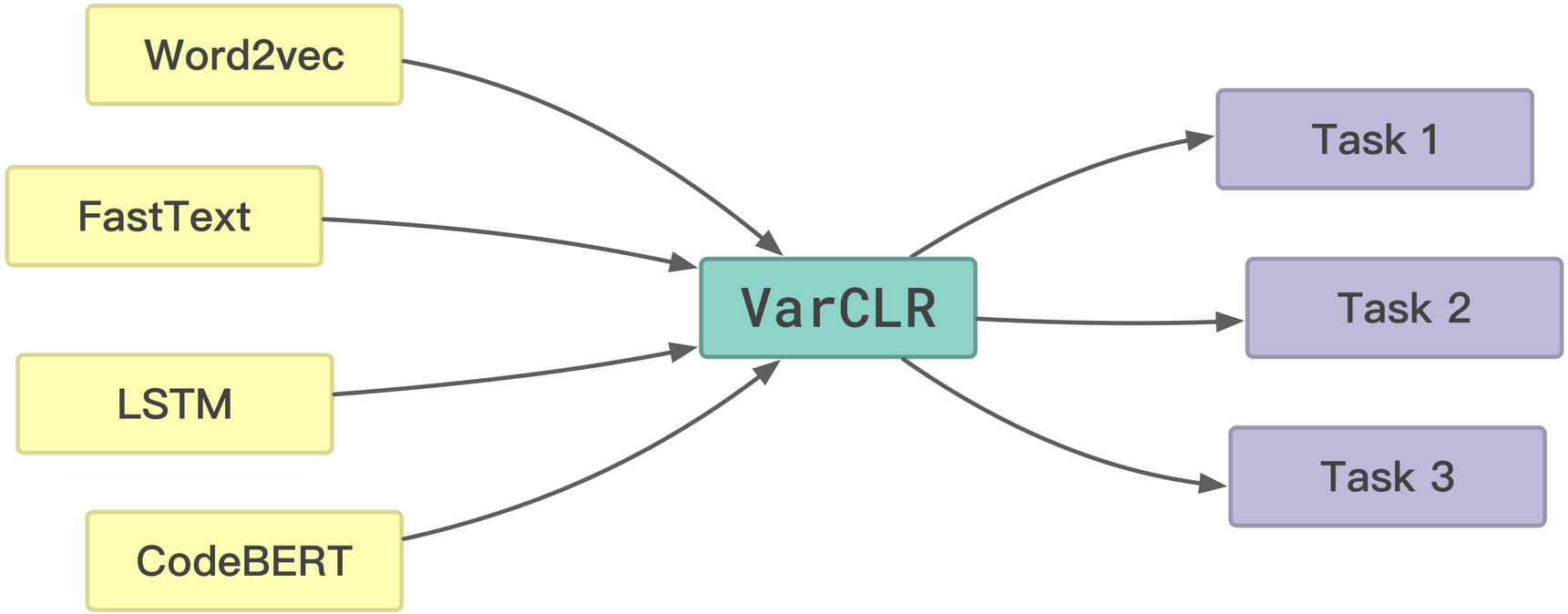
Beware of the underlying assumption when borrowing ML methods as a black box!

What Properties Make Good Variable Representations?

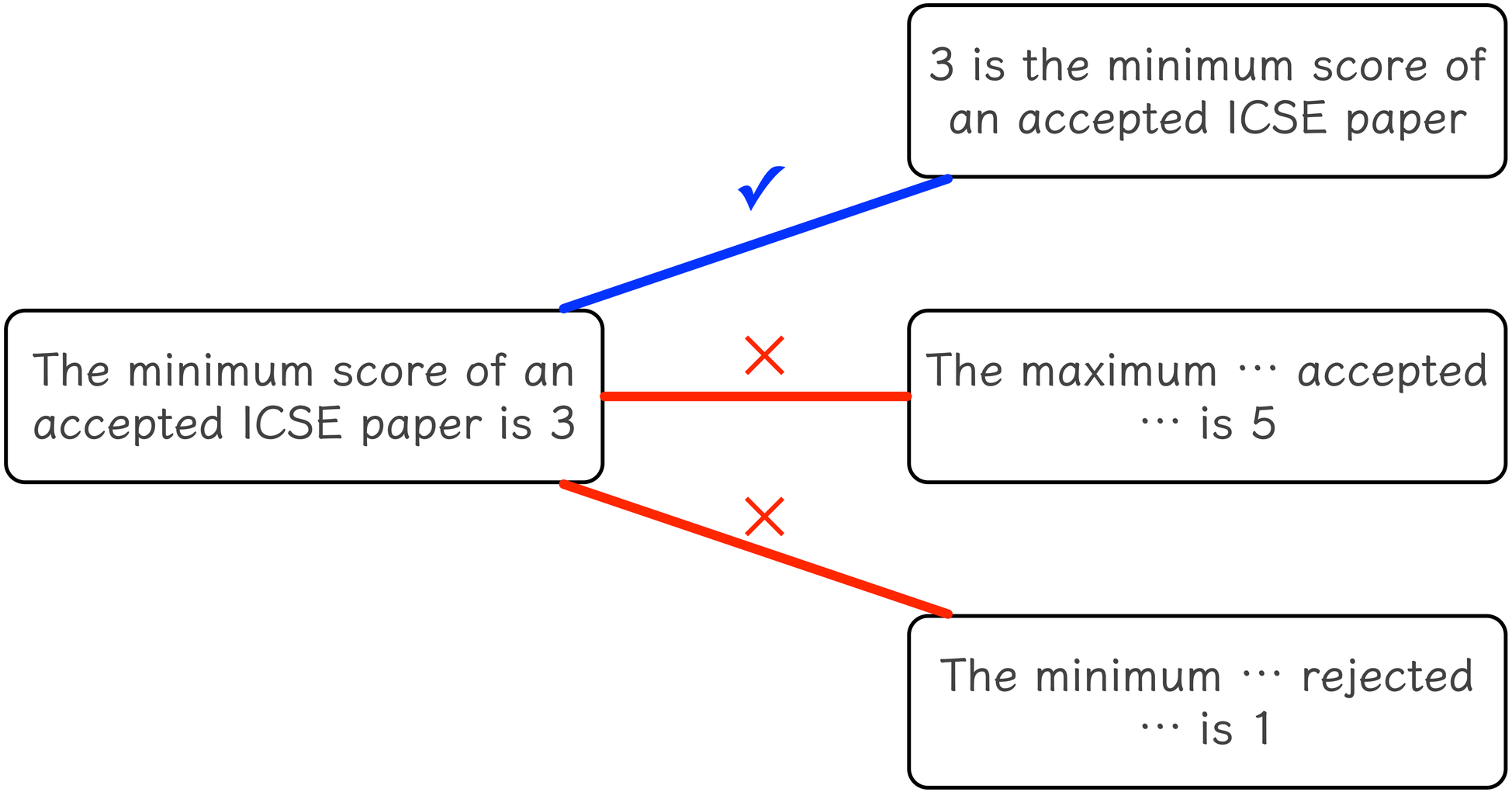
- Character similarity
- Open vocabulary
- Subword importance and order
- ⭐ **Relatedness**
- ⭐ **Similarity** **(interchangeability)**

VarCLR: Towards Better Variable Representations

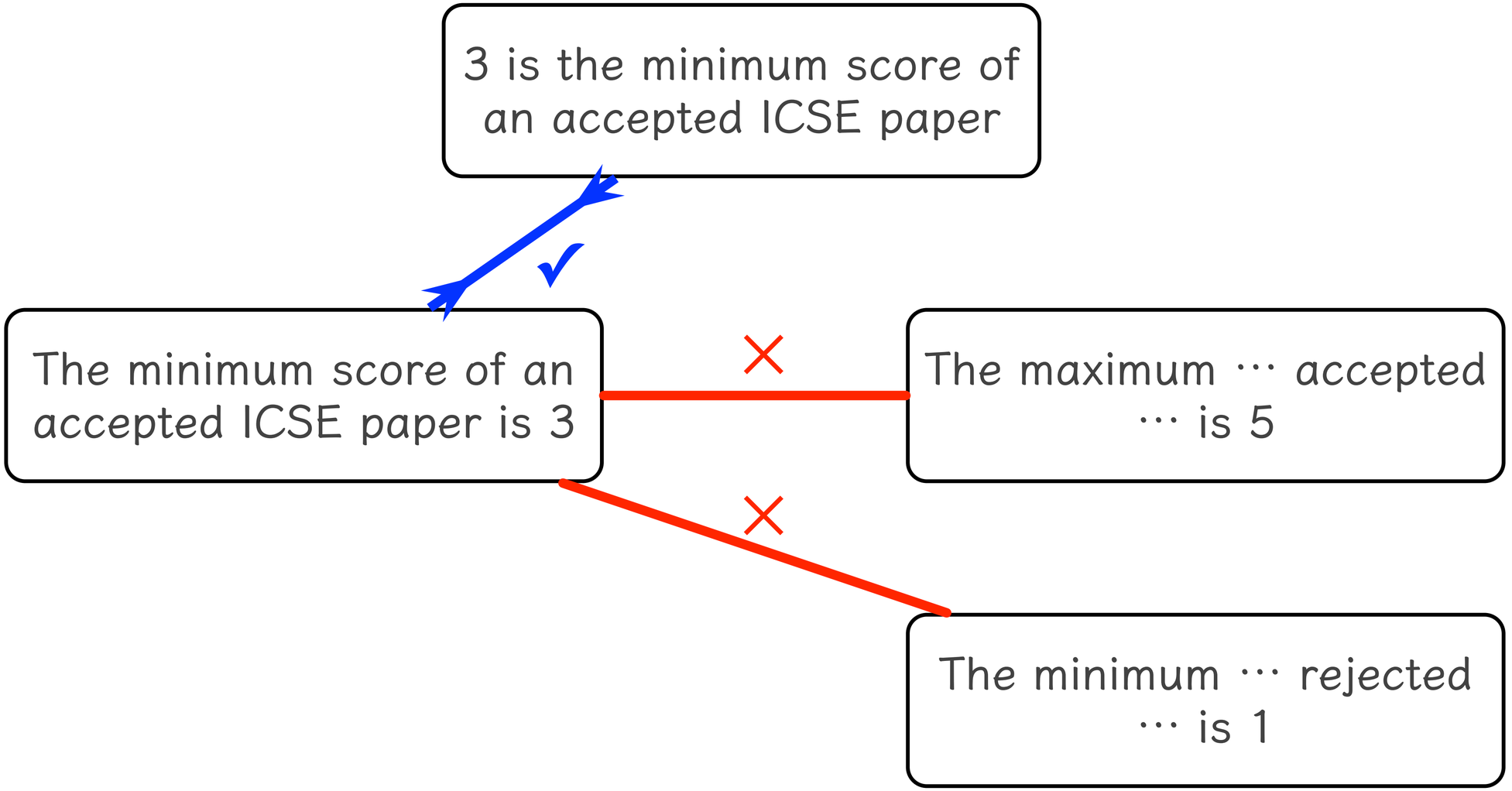
- Built on these SOTA approaches and improve the representations



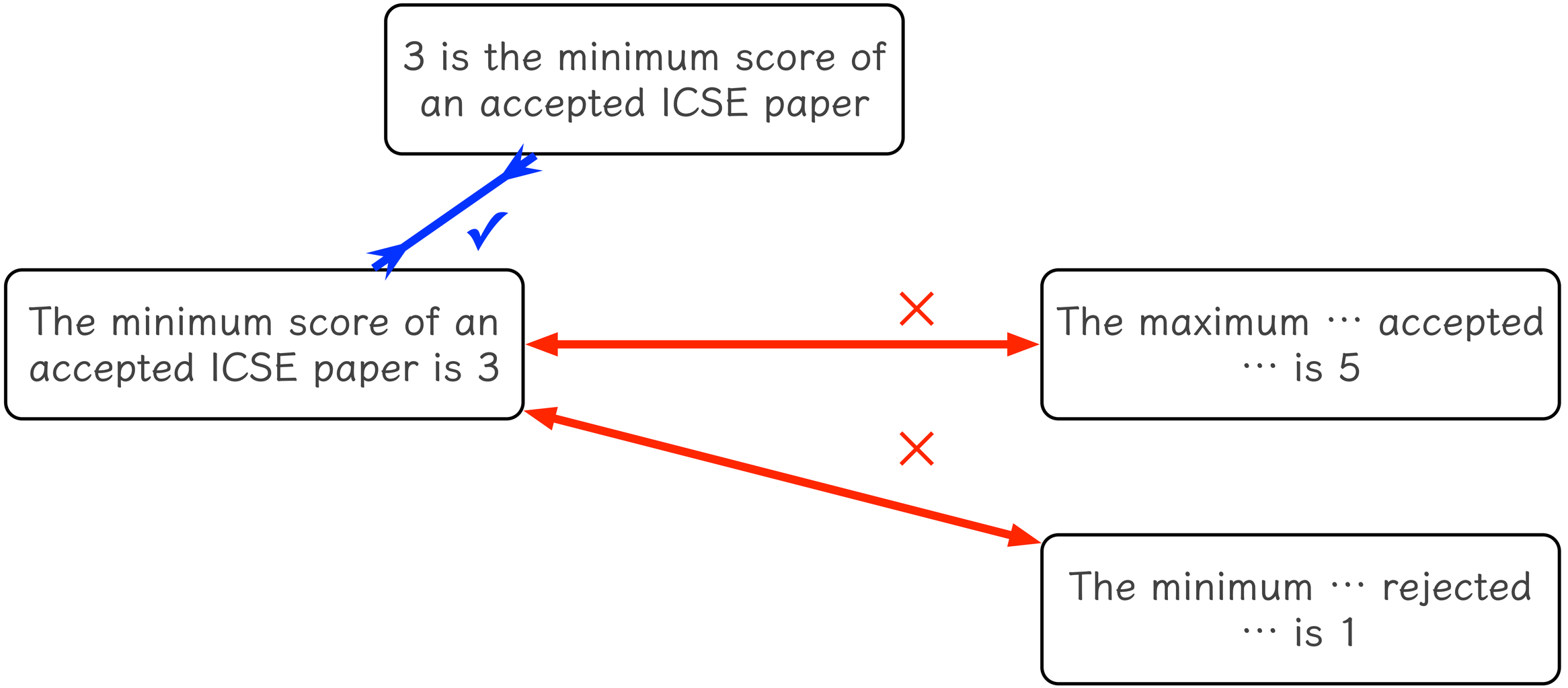
With significantly improved similarity!



Contrastive Learning: A Motivating Example



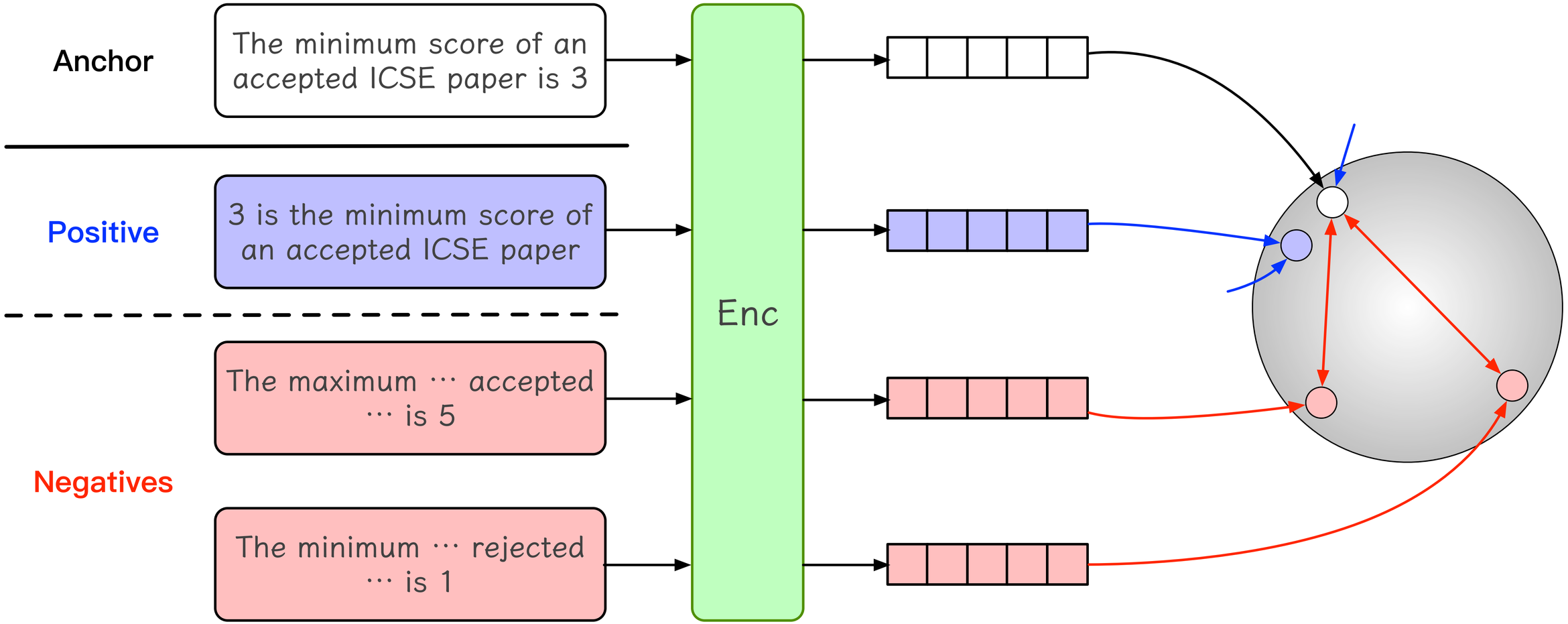
Contrastive Learning: A Motivating Example



Contrastive Learning: A Motivating Example

What Is Contrastive Learning

- Given similar (and dissimilar) pairs
- Learn an embedding space where similar sample pairs are close to each other while dissimilar ones are far apart



Contrastive Learning: A Motivating Example

Fantastic Positive Pairs and Where to Find Them

- Image: augmentations (random crop, resize, clip, color jittering)
- Text: word dropouts, NLI labels (entailment / contradiction)
- Image-text: captions (OpenAI CLIP)

VarCLR GitHubRenames: Collecting Positive Variable Name Pairs

- Collected 66,855 pairs from 568 high star C# projects
- Where the only change in a diff block is the variable name

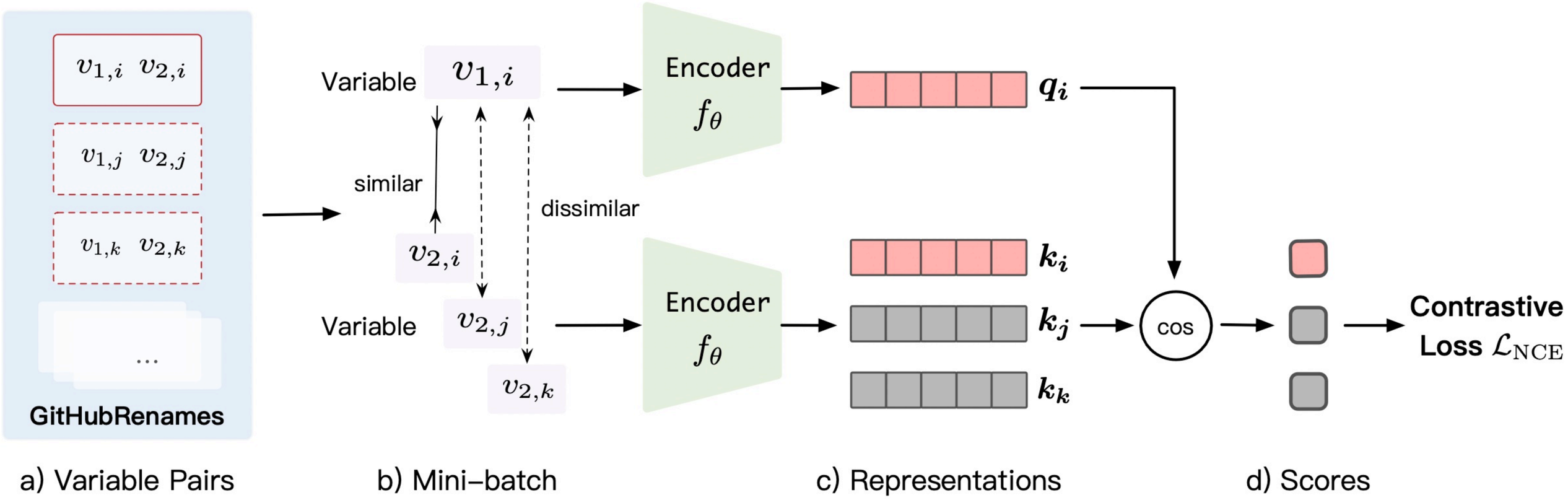
VarCLR GithubRenames: Example

6 docs/samples/Microsoft.ML.Samples/Static/FastTreeRegression.cs

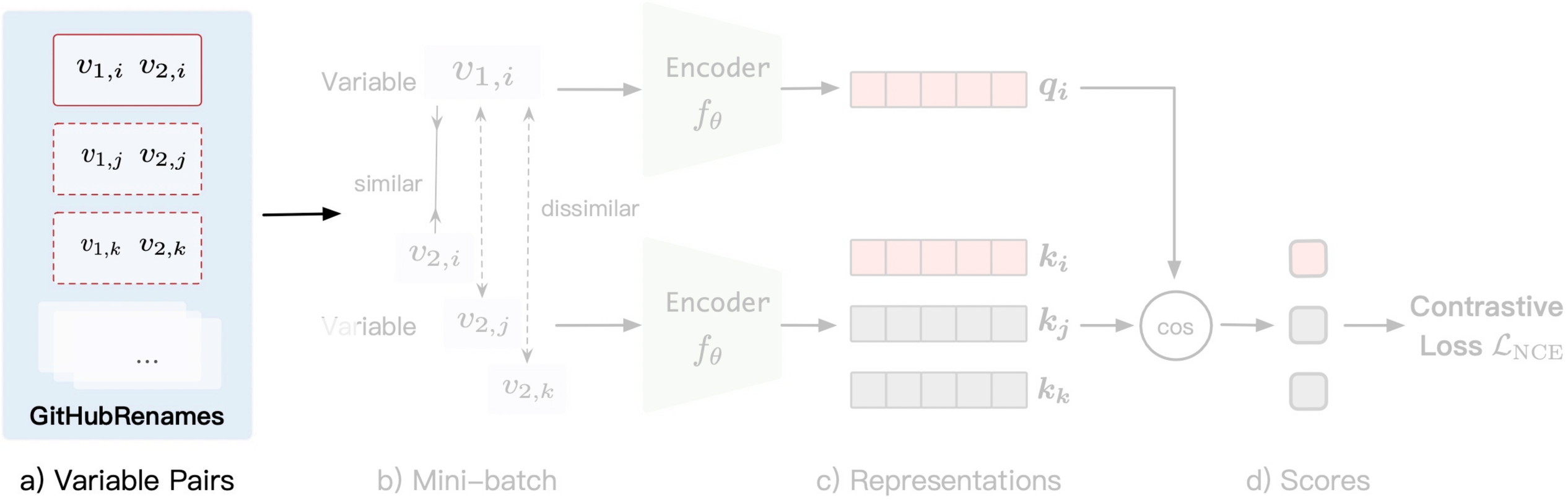
Viewed

```
@@ -38,9 +38,9 @@ public static void FastTreeRegression()
38  38                      .Append(r => (r.label, score: mlContext.Regression.Trainers.FastTree(
39  39                                                    r.label,
40  40                                                    r.features,
41      -                                                 numTrees: 100, // try: (int) 20-2000
42      -                                                 numLeaves: 20, // try: (int) 2-128
43      -                                                 minDatapointsInLeaves: 10, // try: (int) 1-100
        41 +                                              numberOfTrees: 100, // try: (int) 20-2000
        42 +                                              numberOfLeaves: 20, // try: (int) 2-128
        43 +                                              minimumExampleCountPerLeaf: 10, // try: (int) 1-100
44  44                                                    learningRate: 0.2, // try: (float) 0.025-0.4
45  45                                                onFit: p => pred = p)
46  46                      )
```

VarCLR: Overview

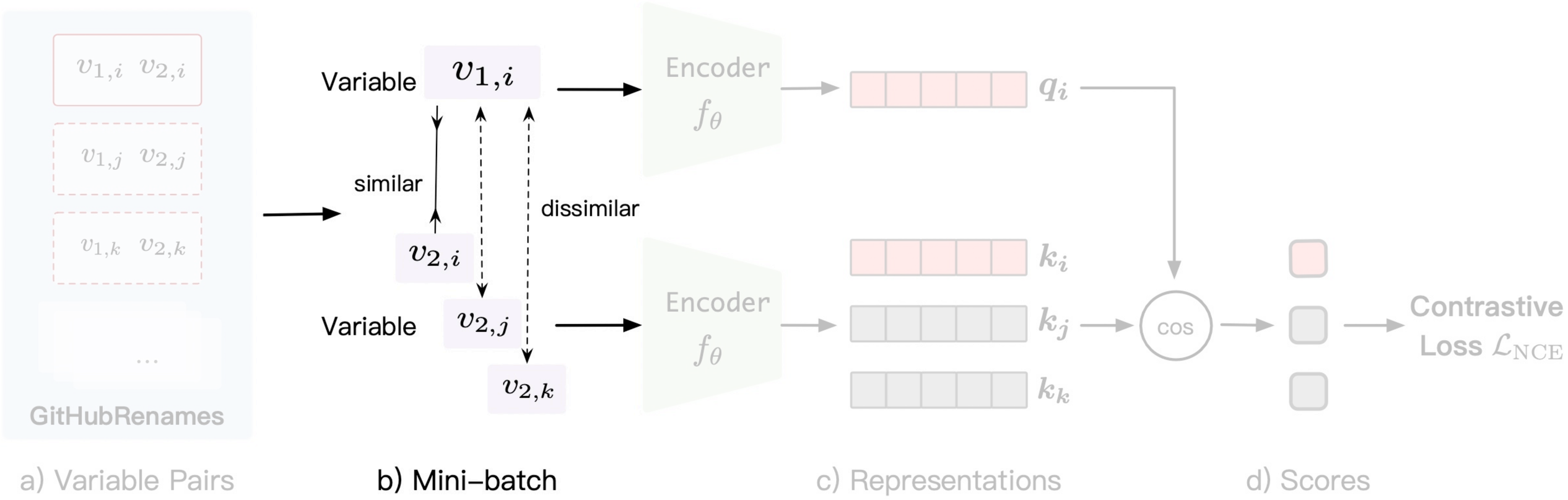


VarCLR: Overview



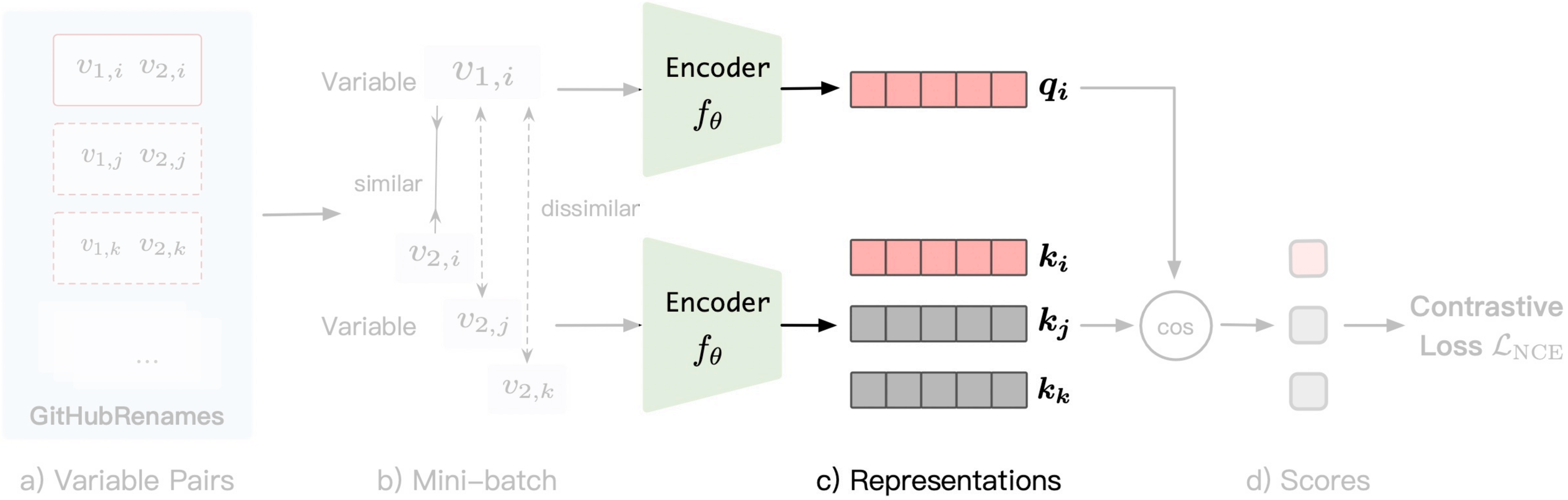
positives

VarCLR: Overview



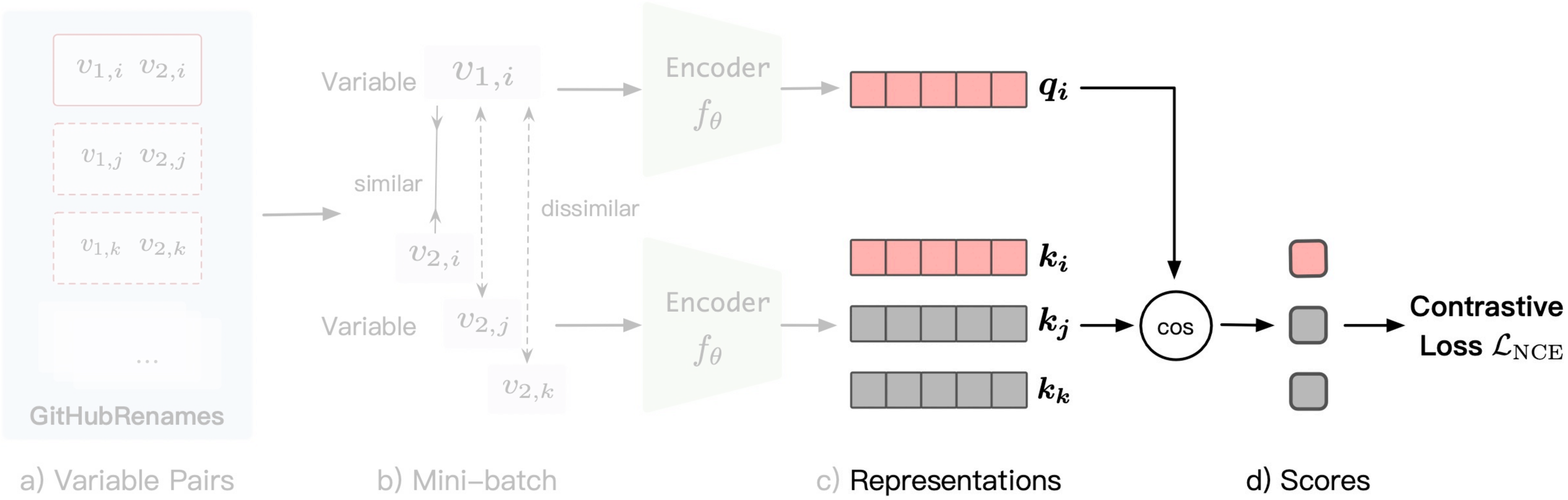
positives in-batch negatives

VarCLR: Overview



positives in-batch negatives encoders

VarCLR: Overview



positives in-batch negatives encoders InfoNCE objective

VarCLR: Experiment Setup

- Benchmark: IdBench
  - Relatedness vs. Similarity
  - Small/Medium/Large
- Metric: Spearman’s rank correlation
- IdBench baselines
  - Levenshtein distance, Word2vec, FastText, Path-based
- VarCLR Model Choices
  - Word2vec, LSTM, CodeBERT

VarCLR Results: Improving Relatedness

0.73 ⟶ 0.80

SCAN ME

IdBench-Large

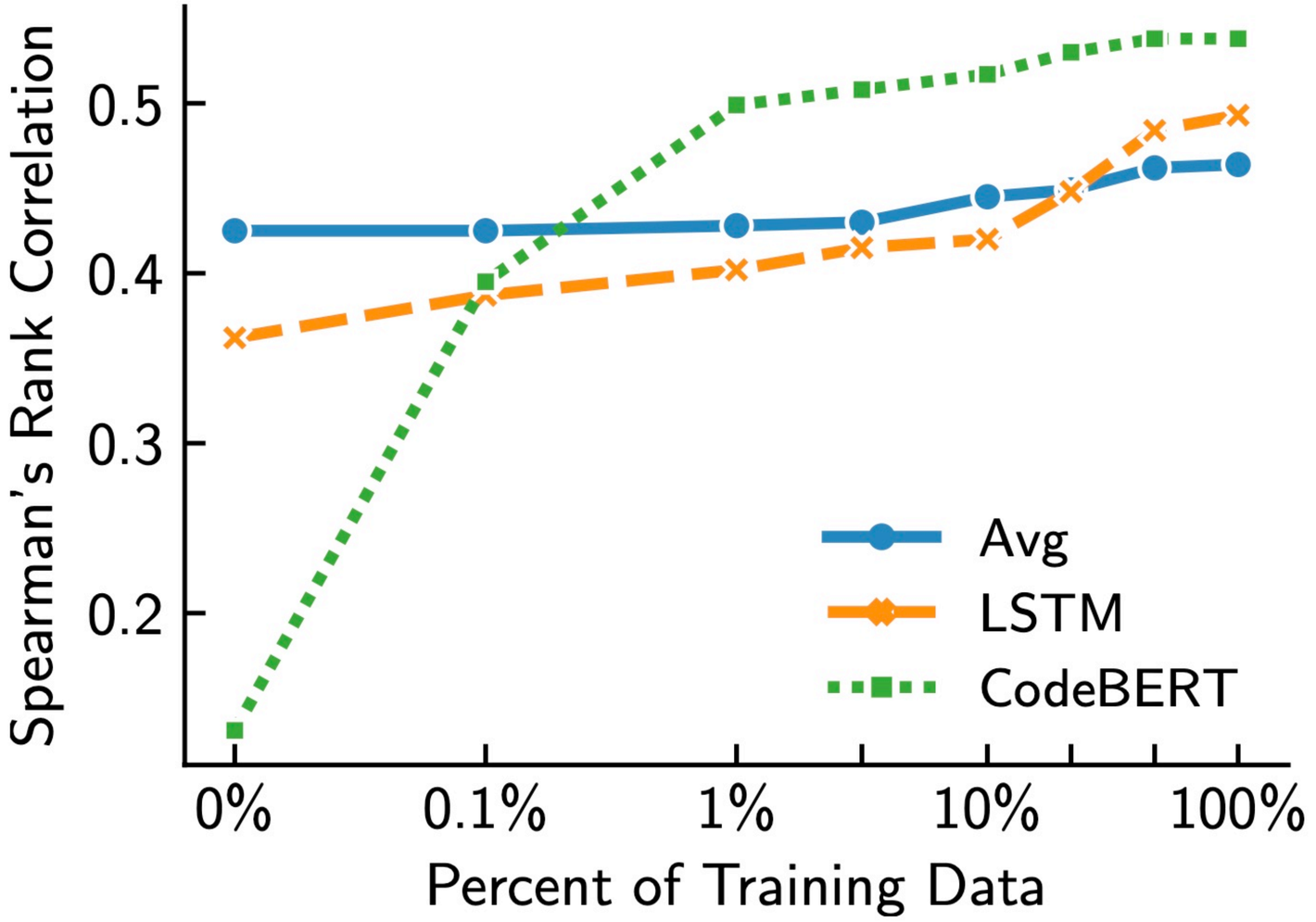
VarCLR Results: Improving Similarity

0.35 ⟶ 0.53

SCAN ME

IdBench-Small

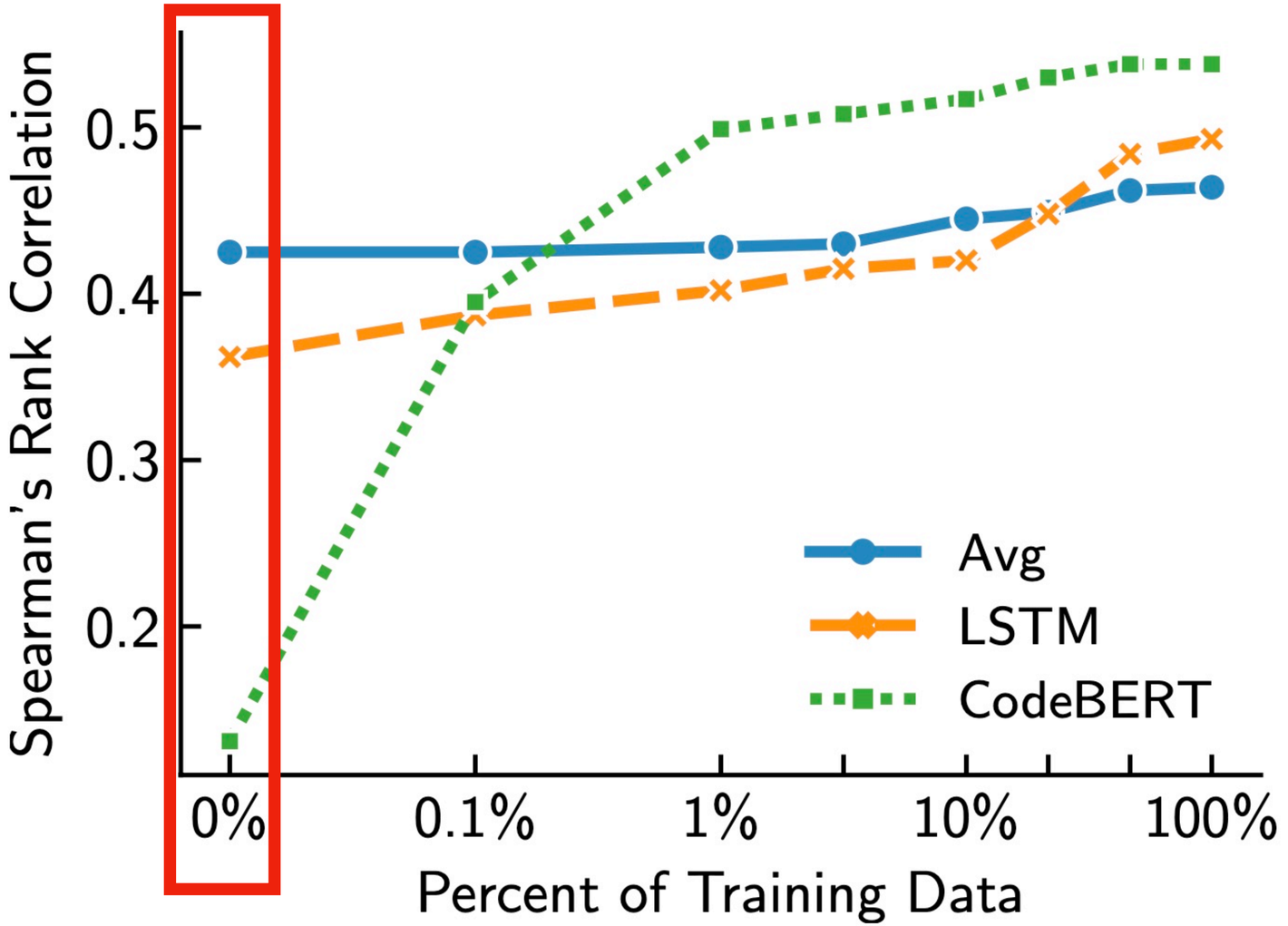
VarCLR: Effect of Contrastive Training



Paper

SCAN ME

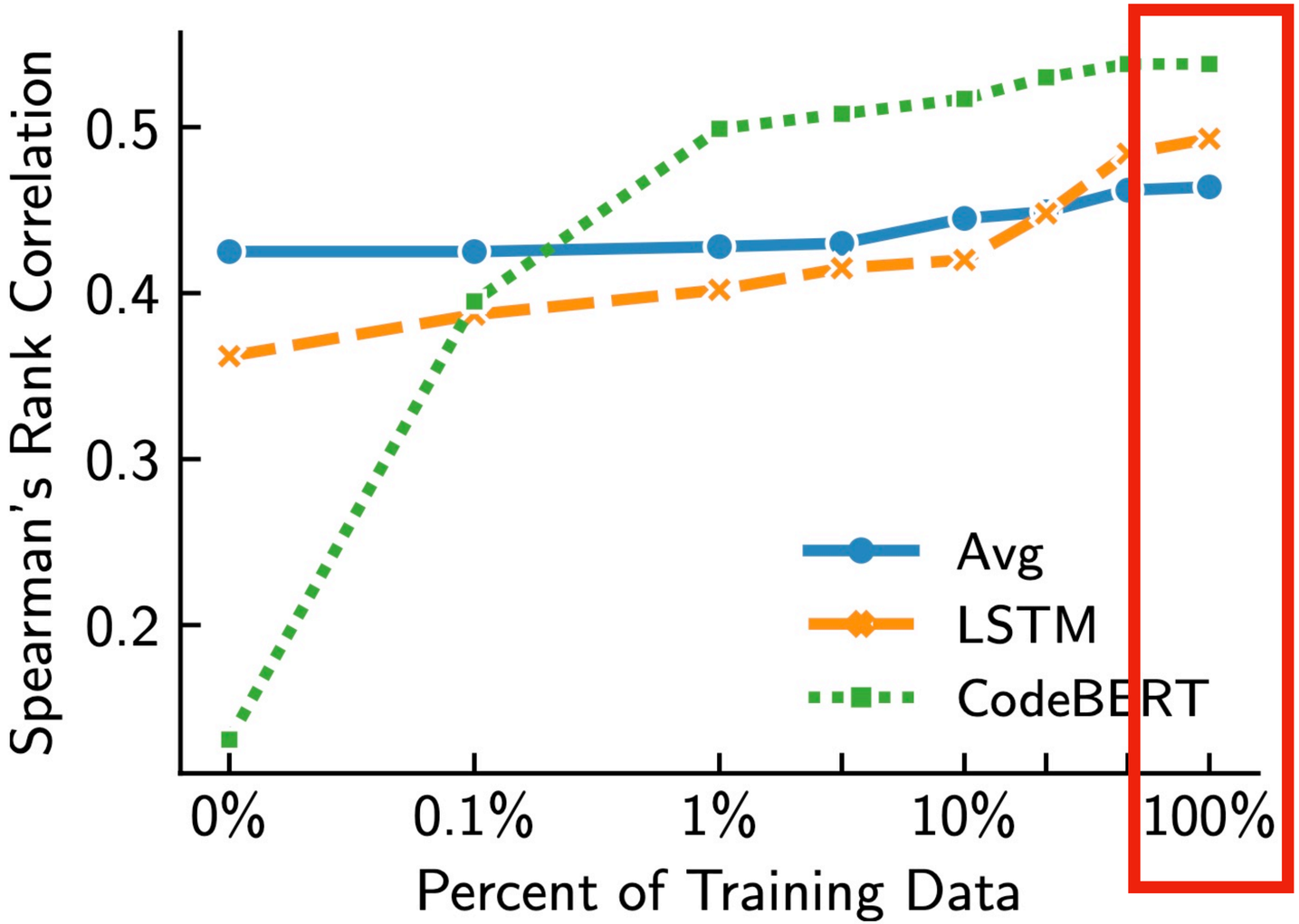
VarCLR: Effect of Training Data



Paper

SCAN ME

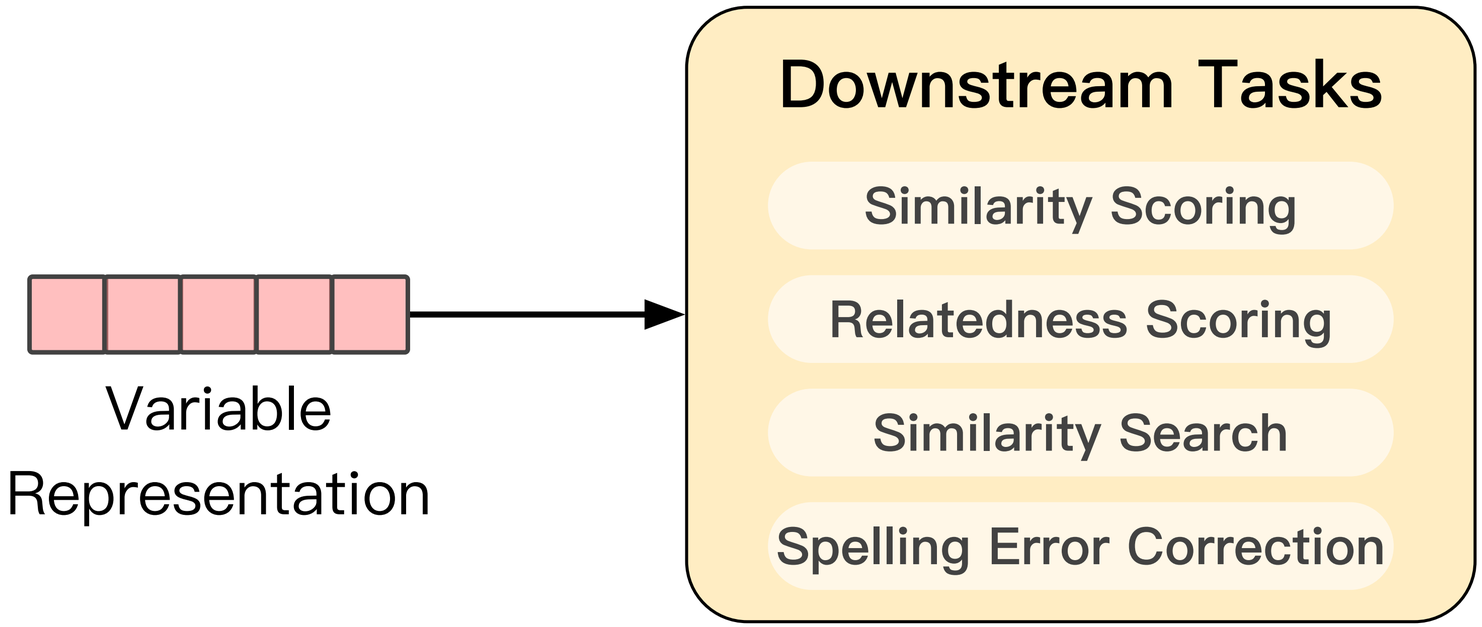
VarCLR: Effect of Training Data



Paper

SCAN ME

More Tasks: Similarity Search and Spelling Error Correction



Paper

SCAN ME

VarCLR Is Easy to Use!

Code

SCAN ME

build passing | stars 22 | license MIT | code style black

VarCLR: Variable Representation Pre-training via Contrastive Learning

New: Paper accepted by ICSE 2022. Preprint at arXiv!

This repository contains code and pre-trained models for VarCLR, a contrastive learning based approach for learning semantic representations of variable names that effectively captures variable similarity, with state-of-the-art results on IdBench@ICSE2021.

- VarCLR: Variable Representation Pre-training via Contrastive Learning
  - Step 0: Install
  - Step 1: Load a Pre-trained VarCLR Model
  - Step 2: VarCLR Variable Embeddings

Off-the-shelf pre-trained embeddings and models!

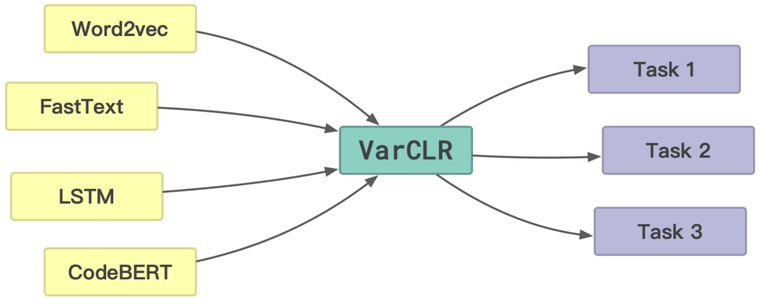
No GPU required!

squaresLab

STROUDEL
SOCIO-TECHNICAL RESEARCH USING DATA EXCAVATION LAB

VarCLR: Towards Better Variable Representations

- Built on these SOTA approaches and improve the representations



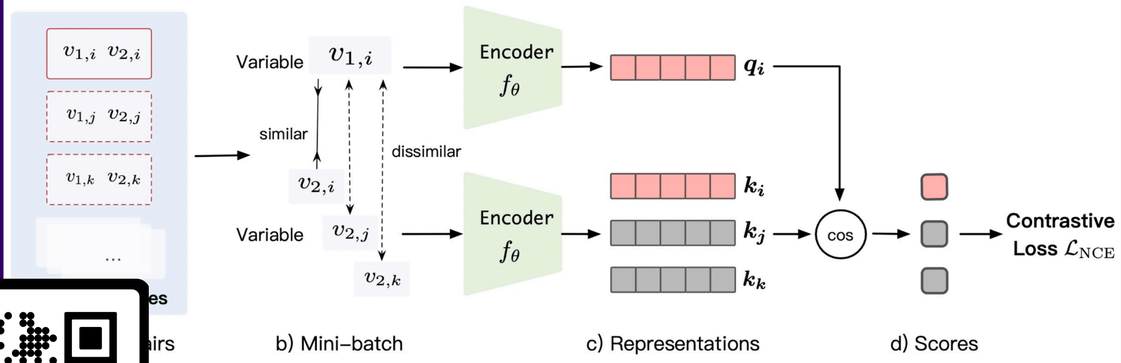
With significantly improved similarity!

VarCLR GithubRenames: Example

docs/samples/Microsoft.ML.Samples/Static/FastTreeRegression.cs — Viewed

```
@@ -38,9 +38,9 @@ public static void FastTreeRegression()
                .Append(r => (r.label, score: mlContext.Regression.Trainers.FastTree(
                                                r.label,
                                                r.features,
-                                               numTrees: 100, // try: (int) 20-2000
-                                               numLeaves: 20, // try: (int) 2-128
-                                               minDatapointsInLeaves: 10, // try: (int) 1-100
+                                               numberOfTrees: 100, // try: (int) 20-2000
+                                               numberOfLeaves: 20, // try: (int) 2-128
+                                               minimumExampleCountPerLeaf: 10, // try: (int) 1-100
                                                learningRate: 0.2, // try: (float) 0.025-0.4
                                            onFit: p => pred = p)
                            )
```

VarCLR: Overview



VarCLR Is Easy to Use!

build passing | stars 22 | license MIT | code style black

VarCLR: Variable Representation Pre-training via Contrastive Learning

New: Paper accepted by ICSE 2022. Preprint at arXiv!

This repository contains code and pre-trained models for VarCLR, a contrastive learning based approach for learning semantic representations of variable names that effectively captures variable similarity, with state-of-the-art results on IdBench@ICSE2021.

- VarCLR: Variable Representation Pre-training via Contrastive Learning
  - Step 0: Install
  - Step 1: Load a Pre-trained VarCLR Model
  - Step 2: VarCLR Variable Embeddings

Off-the-shelf pr… embeddings and…

No GPU require…

squaresLab

STR… SOCIO-TECH… USING DATA…

Paper

Code