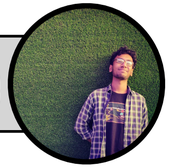
Evaluating AI-enabled Software Development Techniques

Dr. Antonio Mastropaolo

Instructor

Mr. Alvi Haque



Teaching Assistant

WILLIAM & MARY

CHARTERED 1693

Spring 2026

Evaluating AI-enabled Software Development Techniques

Beyond Exact Matches: Metrics for evaluating Technical (N)atural (L)anguage and Code

Evaluating AI-enabled Software Development Techniques

Beyond Exact Matches: Metrics for evaluating Technical (N)atural (L)anguage and Code

AUTOMATED METRICS *vs.* MANUAL METRICS

+ Quick and cheap way to assess the output of PTM recommendations (code or NL)

- Chances are that the **semantic content** is not captured, especially for NL

Evaluating AI-enabled Software Development Techniques

Reads the contents of this source as a string.	PR
Get the textual information from this source and represent it as a string.	GT

```
public String read() throws IOException {
  Closer closer = Closer.create();
  try {
     Reader reader = closer.register(openStream());
     return CharStreams.toString(reader);
  } catch ( Throwable e ) {
     throw closer.rethrow (e);
  } finally { closer.close(); }
}
```

Evaluating AI-enabled Software Development Techniques

Reads the contents of this source as a string.	PR
Get the textual information from this source and represent it as a string.	GT

```java
public String read() throws IOException {
  Closer closer = Closer.create();
  try {
     Reader reader = closer.register(openStream());
     return CharStreams.toString(reader);
  } catch ( Throwable e ) {
     throw closer.rethrow (e);
  } finally { closer.close(); }
}
```

BLEU SCORE: 0.21

Evaluating AI-enabled Software Development Techniques

Reads the contents of this source as a string.	PR
Get the textual information from this source and represent it as a string.	GT

BLEU SCORE

Evaluating AI-enabled Software Development Techniques

Reads the contents of this source as a string.	PR
Get the textual information from this source and represent it as a string.	GT

Semantically Equivalent Code Descriptions

BLEU SCORE

Evaluating AI-enabled Software Development Techniques

Beyond Exact Matches: Metrics for evaluating Technical (N)atural (L)anguage and Code

AUTOMATED METRICS

+ Quick and cheap way to assess the output of PTM recommendations (code or NL)

- Chances are that the **semantic content** is not captured, especially for NL

VS.

MANUAL METRICS

+ An expert from that domain (i.e., developer) manually analyzes and comments on how closely that predictions resembles the GT

- Expensive (i.e., Time, $$$$)

Evaluating AI-enabled Software Development Techniques

Beyond Exact Matches: Metrics for evaluating Technical (N)atural (L)anguage and Code

CODE		Technical Natural Language	
BLEU	CrystalBLEU	BLEU	Rouge
CodeBLEU	Embedding-based	Meteor	chrF

Evaluating AI-enabled Software Development Techniques

Beyond (N)atural

BLEU

CodeBLEU

...chnical

...guage

...Rouge

...chrF

Ren, S., Guo, D., Lu, S., Zhou, L., Liu, S., Tang, D., ... & Ma, S. (2020). Codebleu: a method for automatic evaluation of code synthesis. arXiv preprint arXiv:2009.10297.

Eghbali, A., & Pradel, M. (2022, October). CrystalBLEU: precisely and efficiently measuring the similarity of code. In Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering (pp. 1-12).

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting of the Association for Computational Linguistics (pp. 311-318).

Banerjee, S., & Lavie, A. (2005, June). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. ... summarization (pp. 65-72).

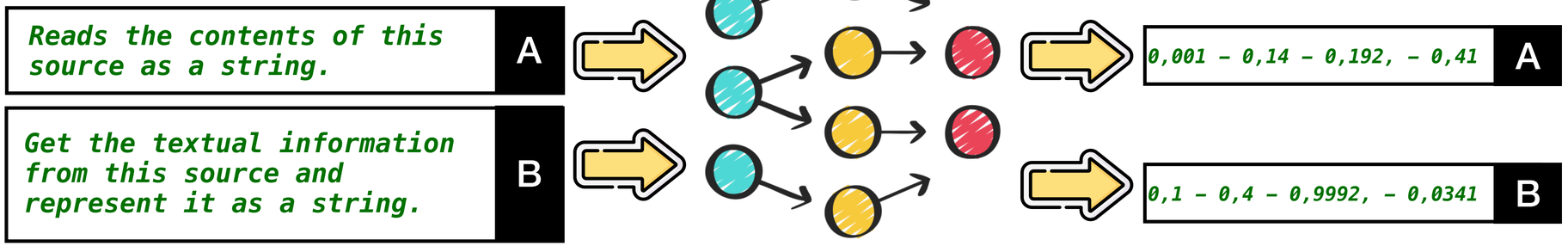
Lin, C. Y. (2004, July). Rouge: A package for automatic evaluation of summaries. In Text summarization branches out (pp. 74-81).

Popović, M. (2015, September). chrF: character n-gram F-score for automatic MT evaluation. In Proceedings of the tenth workshop on statistical machine translation (pp. 392-395).

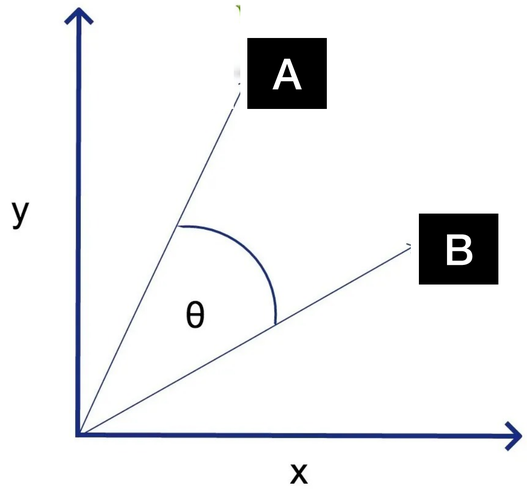
Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). Bertscore: Evaluating text generation with bert. arXiv preprint arXiv:1904.09675.

Evaluating AI-enabled Software Development Techniques

Embedding-based Metric



Cosine Similarity



Evaluating Pre-trained Models for SD

Embedding-based Metric

If I swap the embedding model, say, to one trained on code, I will produce code-specific embeddings. If I use a model trained on natural language, I will get natural-language embeddings. And if the model was trained on both code and NL, then it can generate embeddings for both modalities, letting us apply the same methods we discussed across each.

Evaluating AI-enabled Software Development Techniques

CodeBLEU

CodeBLEU is an evaluation metric designed to assess the quality of generated code.

Evaluating AI-enabled Software Development Techniques

CodeBLEU

CodeBLEU is an evaluation metric designed to assess the quality of generated code.

Unlike traditional metrics like BLEU, which originate from natural language processing, CodeBLEU enhances evaluation by accounting for the distinct structural and semantic characteristics of programming languages.

Evaluating AI-enabled Software Development Techniques

CodeBLEU

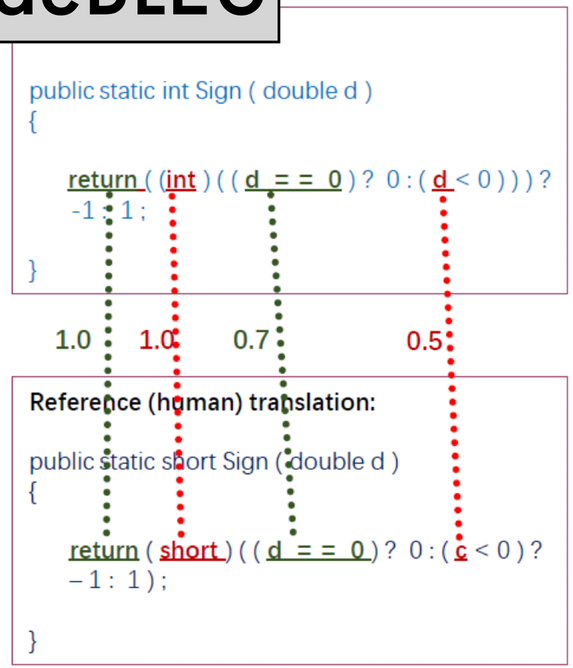
CodeBLEU is an evaluation metric designed to assess the quality of generated code.

Unlike traditional metrics like BLEU, which originate from natural language processing, CodeBLEU enhances evaluation by accounting for the distinct structural and semantic characteristics of programming languages.

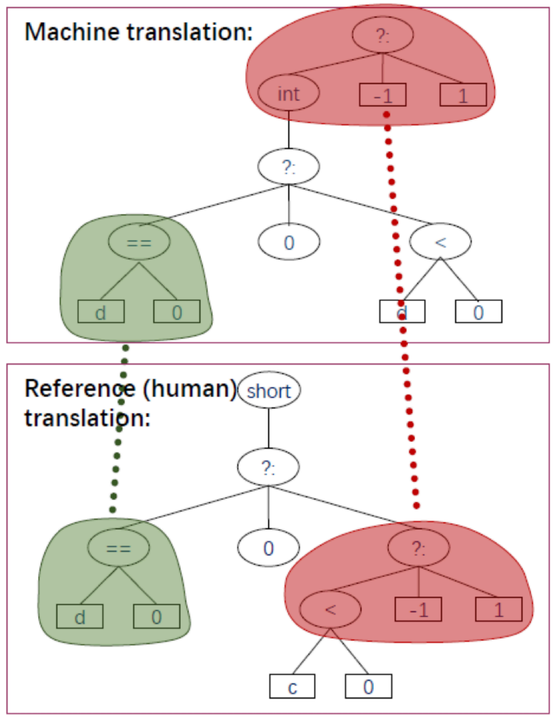
BLEU evaluates similarity based on n-gram overlap between the generated and reference text. In programming: (i) two code snippets can appear different yet perform the same function, and (ii) structural correctness — like balanced brackets, valid syntax, and proper variable usage — is crucial.

Evaluating AI-enabled Software Development Techniques

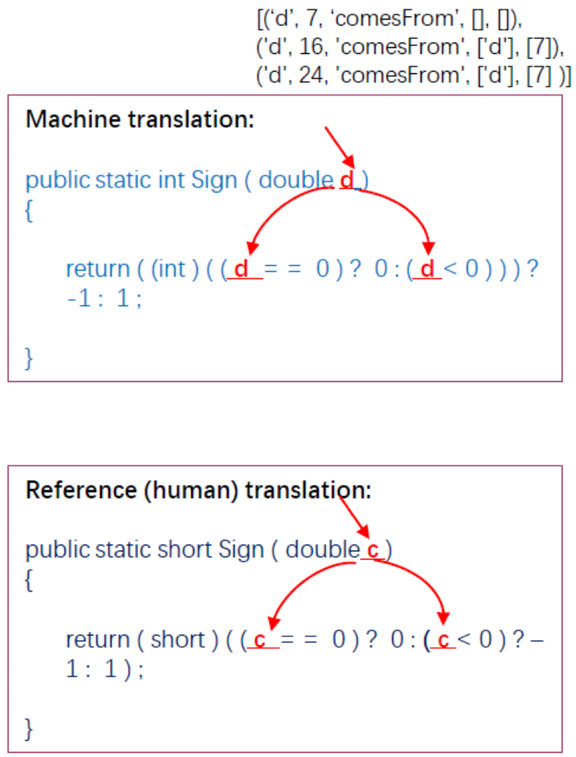
CodeBLEU



Weighted N-Gram Match



Syntactic AST Match



Semantic Data-flow Match

$$\text{CodeBLEU} = \alpha \cdot \text{N-Gram Match (BLEU)} + \beta \cdot \text{Weighted N-Gram Match} + \gamma \cdot \text{Syntactic AST Match} + \delta \cdot \text{Semantic Data-flow Match}$$

0.25 0.25 0.25 0.25

Evaluating AI-enabled Software Development Techniques

CrystalBLEU

CrystalBLEU just as CodeBLEU is an evaluation metric designed to assess the quality of generated code.

Evaluating AI-enabled Software Development Techniques

CrystalBLEU

CrystalBLEU just as CodeBLEU is an evaluation metric designed to assess the quality of generated code.

In contrast to BLEU and CodeBLEU—which focus on token-level or structural similarity to reference code—CrystalBLEU rewards models for generating correct yet novel outputs, discouraging direct copying from training examples.

Evaluating AI-enabled Software Development Techniques

CrystalBLEU

CrystalBLEU just as CodeBLEU is an evaluation metric designed to assess the quality of generated code.

In contrast to BLEU and CodeBLEU—which focus on token-level or structural similarity to reference code—CrystalBLEU rewards models for generating correct yet novel outputs, discouraging direct copying from training examples.

In other words, if we know that there are code tokens that contribute less to the evaluation of the generated code — for example, the names of boilerplate variables or temporary identifiers — we might want to **reduce their impact** in the scoring process.

Evaluating AI-enabled Software Development Techniques

What if our approach generates natural language tokens?
E.g, we train a model that automatically documents the code?

CONTRASTIVE LEARNING

Contrastive Learning

The goal of contrastive learning is to learn an embedding space in which similar sample pairs stay close to each other while dissimilar ones are far apart

Contrastive Learning

Contrastive Learning Objectives

Contrastive Loss

Chopra et al. 2005

We would like to learn a function that encodes into an embedding vector examples in such a way that examples from the same class have similar embeddings and samples from different classes have very different ones

Contrastive Learning

Contrastive Learning Objectives

Contrastive Loss

Chopra et al. 2005

We would like to learn a function that encodes into an embedding vector examples in such a way that examples from the same class have similar embeddings and samples from different classes have very different ones

$$L = (1 - y) * D(x1,x2)^2 + y * max(0,margin - D(x1,x2))^2$$

D(x1, x2) is the **distance** between the two samples (e.g., euclidean).

margin is a threshold that defines how far apart the dissimilar samples should be.

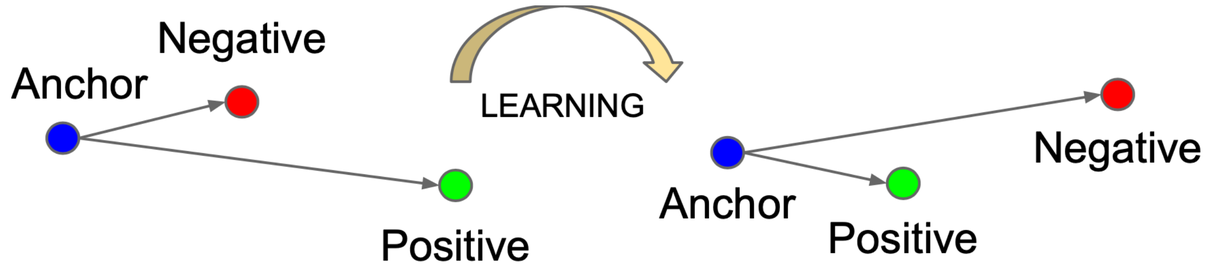
y is the **binary label** (1 if the samples are similar, 0 if they are dissimilar).

Contrastive Learning

Contrastive Learning Objectives

Triplet Loss

Schroff et al. 2015

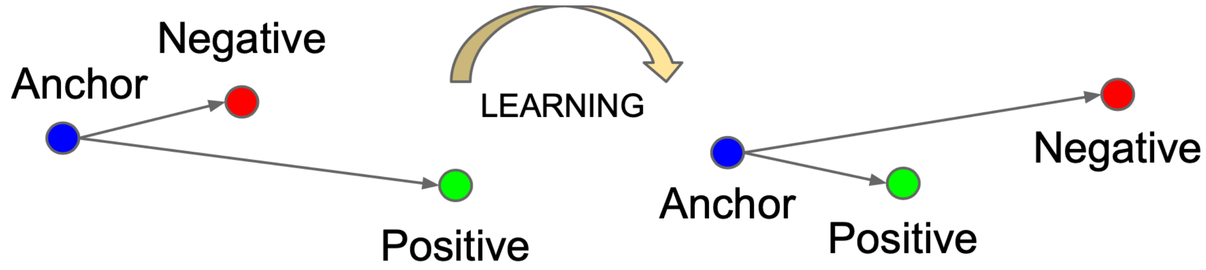


Contrastive Learning

Contrastive Learning Objectives

Triplet Loss

Schroff et al. 2015



$$L(A, P, N) = max(0, D(A, P) - D(A, N) + m)$$

D(A, P) is the distance between the anchor and positive embeddings

D(A, N) is the distance between the anchor and negative embeddings

m is the **margin** that ensures a sufficient gap between the positive and negative pairs.

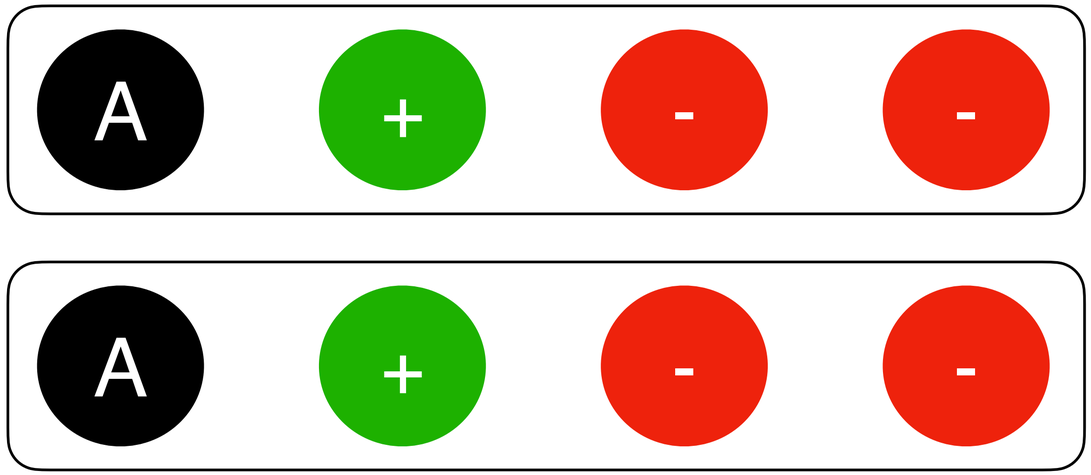
Contrastive Learning

Contrastive Learning Objectives

N-pair Loss

Sohn 2016

generalizes triplet loss to include comparison with multiple negative samples.



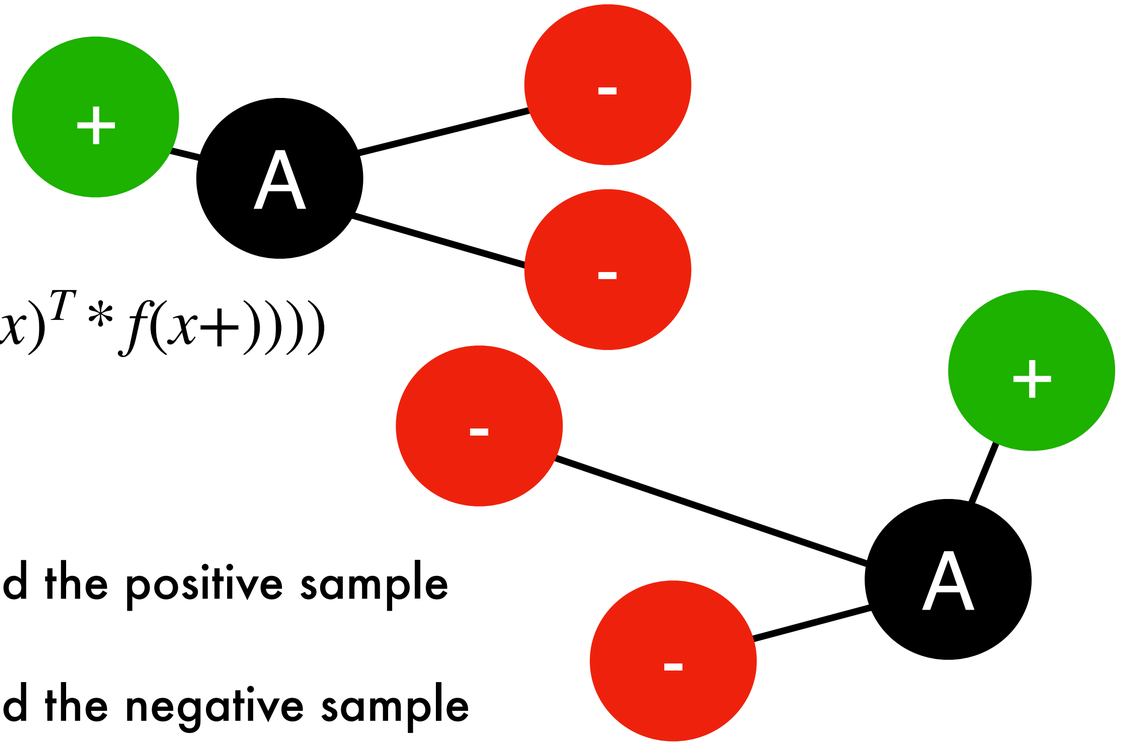
Contrastive Learning

Contrastive Learning Objectives

N-pair Loss

Sohn 2016

generalizes triplet loss to include comparison with multiple negative samples.



$$L(x, x+, x_i-) = log(1 + sum(exp(f(x)^T * f(x_i-) - f(x)^T * f(x+))))$$

$f(x)$ represents the embedding function (the model).

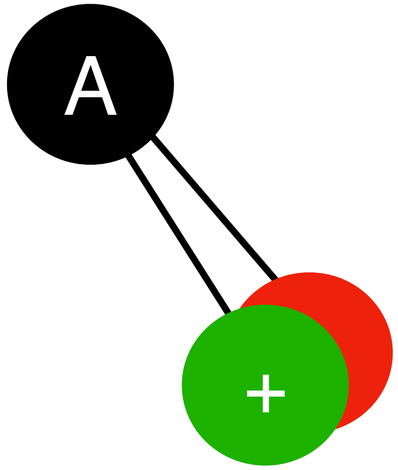
$f(x)^T * f(x+)$ measures the similarity between the anchor and the positive sample

$f(x)^T * f(x_i-)$ measures the similarity between the anchor and the negative sample

Contrastive Learning

How to develop a Contrastive Learning Approach?

- Hard-Negatives Mining

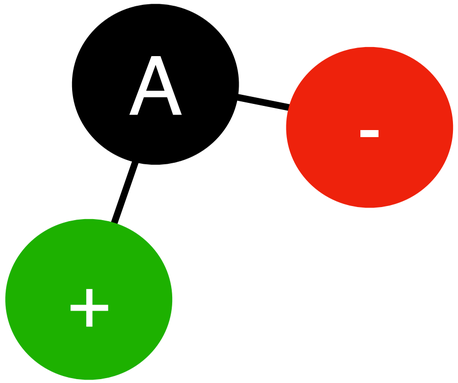


A **hard** negative refers to a negative sample that is challenging for the model to differentiate from a positive pair.

Contrastive Learning

How to develop a Contrastive Learning Approach?

- Hard-Negatives Mining



A **hard** negative refers to a negative sample that is challenging for the model to differentiate from a positive pair.

This happens when the negative sample is positioned very close to the anchor (reference point) in the embedding space

Contrastive Learning

How to develop a Contrastive Learning Approach?

- **Hard-Negatives Mining**

Anchor Sentence (A): *"The dog is playing with the bone"*

Positive Sentence (P+): "The dog is enjoying his bone"

Negative Sentence (N-): "The dog is not playing with the bone"

Contrastive Learning

How to develop a Contrastive Learning Approach?

- Large Batch Size

Usually Batch Sizes are generated by randomly sampling a fixed number of training elements

A large batch size enables the model to encounter a diverse set of negative samples, which provides a sufficient challenge for the model to learn meaningful representations that can effectively differentiate between various examples.

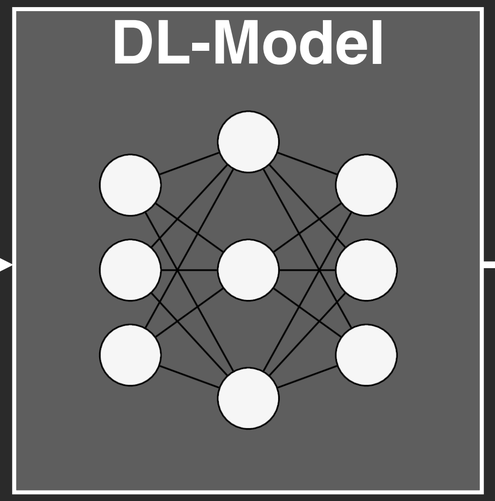
Contrastive Learning Method for developing new metrics in Software Engineering

Code Summarization

INPUT

```
public ConnectionConsumer createConnectionConsumer
    (final Destination destination)
       throws JMSException{

   if(LOGGER.isTraceEnabled())
   {
       ActiveMQRALogger.LOGGER.
         trace("Create connectionConsumer");
   }
   else {
       throw new IllegalStateException(ISE);
   }
}
```



PREDICTION

Create a connection to the Consumer.

Contrastive Learning Method for developing new metrics in Software Engineering

Contrastive Learning Method for developing new metrics in Software Engineering

GROUND TRUTH		*MODEL'S PREDICTION*
`Create a connection to the consumer.`	EXACT MATCHES	`Create a connection to the consumer.`

Contrastive Learning Method for developing new metrics in Software Engineering

GROUND TRUTH

```
Create a connection
to the consumer.
```

EXACT MATCHES

MODEL'S PREDICTION

```
Create a conn to
the consumer.
```

Contrastive Learning Method for developing new metrics in Software Engineering

BLEU SCORE

ROUGE SCORE

METEOR SCORE

chrF SCORE

Contrastive Learning Method for developing new metrics in Software Engineering

Reads the contents of this source as a string.	PR
Get the textual information from this source and represent it as a string.	GT

```
public String read() throws IOException {
  Closer closer = Closer.create();
  try {
     Reader reader = closer.register(openStream());
     return CharStreams.toString(reader);
  } catch ( Throwable e ) {
     throw closer.rethrow (e);
  } finally { closer.close(); }
}
```

Contrastive Learning Method for developing new metrics in Software Engineering

Reads the contents of this source as a string.	PR
Get the textual information from this source and represent it as a string.	GT

```java
public String read() throws IOException {
  Closer closer = Closer.create();
  try {
     Reader reader = closer.register(openStream());
     return CharStreams.toString(reader);
  } catch ( Throwable e ) {
     throw closer.rethrow (e);
  } finally { closer.close(); }
}
```

SEMANTICALLY EQUIVALENT CODE SUMMARIES

Contrastive Learning Method for developing new metrics in Software Engineering

Reads the contents of this source as a string.	PR
Get the textual information from this source and represent it as a string.	GT

```java
public String read() throws IOException {
  Closer closer = Closer.create();
  try {
     Reader reader = closer.register(openStream());
     return CharStreams.toString(reader);
  } catch ( Throwable e ) {
     throw closer.rethrow (e);
  } finally { closer.close(); }
}
```

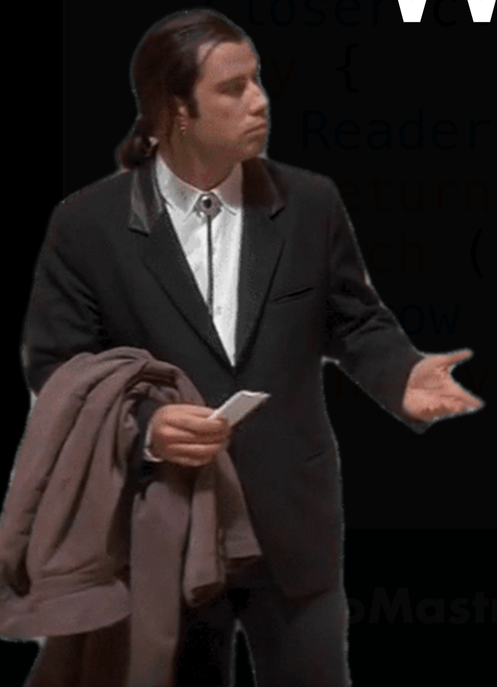
BLEU SCORE: 0.21

Contrastive Learning Method for developing new metrics in Software Engineering

Reads the contents of this source as a string. PR

Get the textual information from this source and represent it as a string. GT

What about source code?



BLEU SCORE: 0.21

Contrastive Learning Method for developing new metrics in Software Engineering

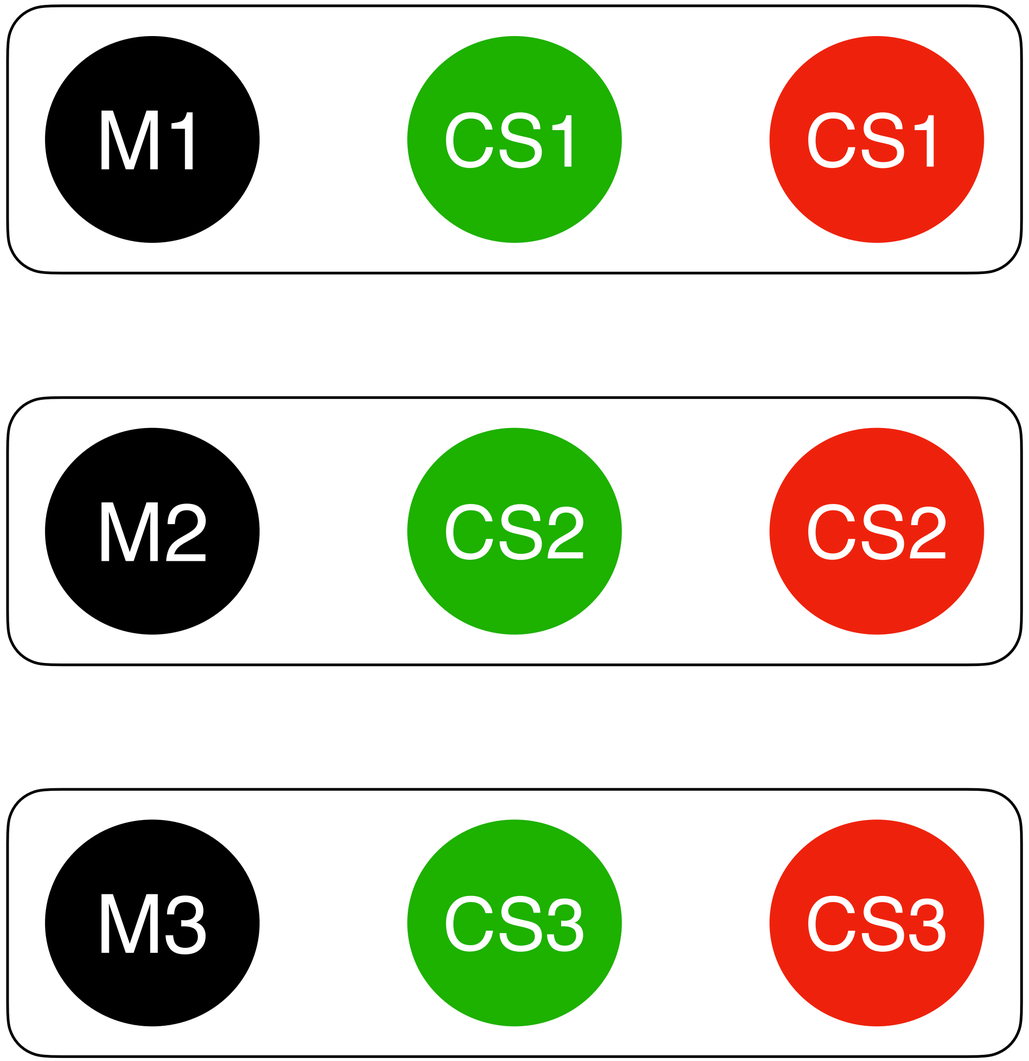
Get the textual information from this source and represent it as a string. ✓

Connect to the server and return the status ✗

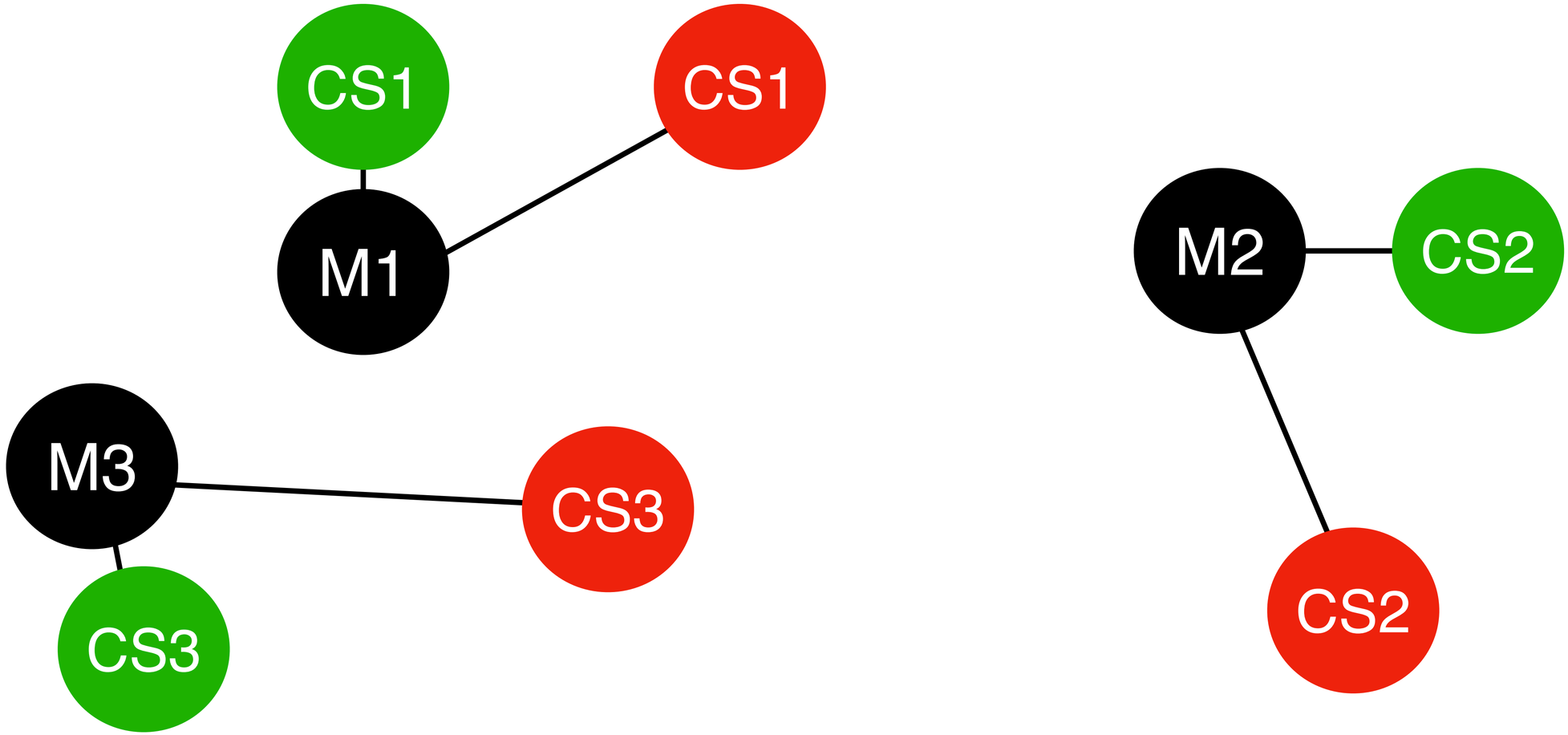
```
public String read() throws IOException {
  Closer closer = Closer.create();
  try {
     Reader reader = closer.register(openStream());
     return CharStreams.toString(reader);
  } catch ( Throwable e ) {
     throw closer.rethrow (e);
  } finally { closer.close(); }
}
```

Summary Alignment to CoDe Semantic

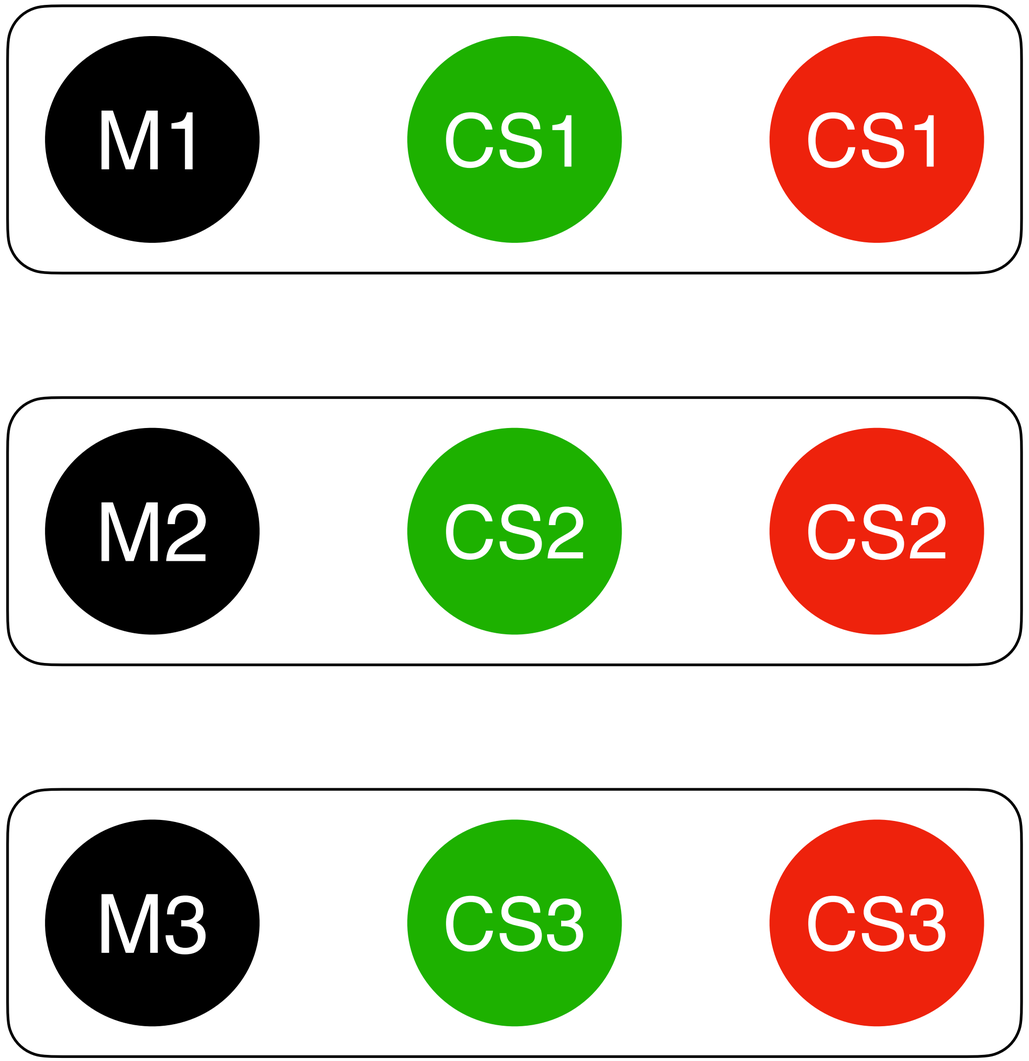
Contrastive Learning Method for developing new metrics in Software Engineering

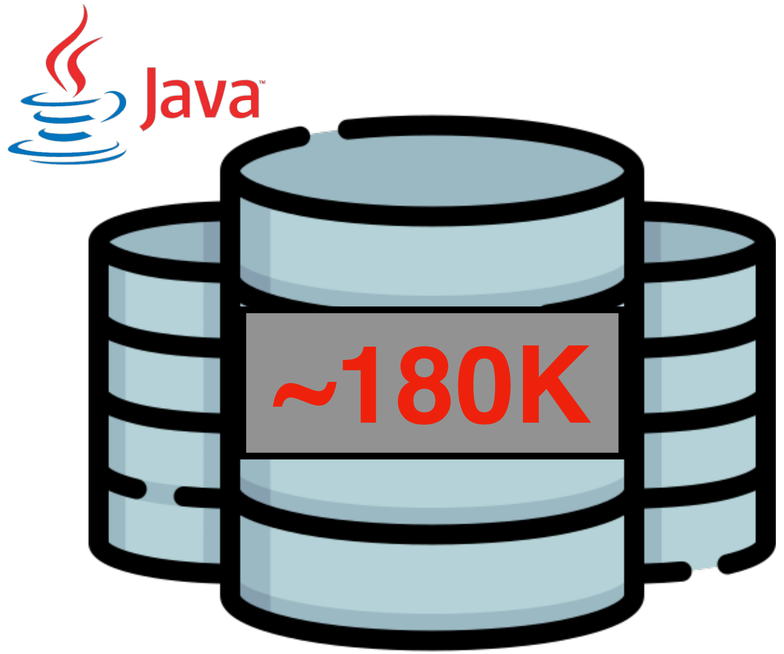


Contrastive Learning Method for developing new metrics in Software Engineering



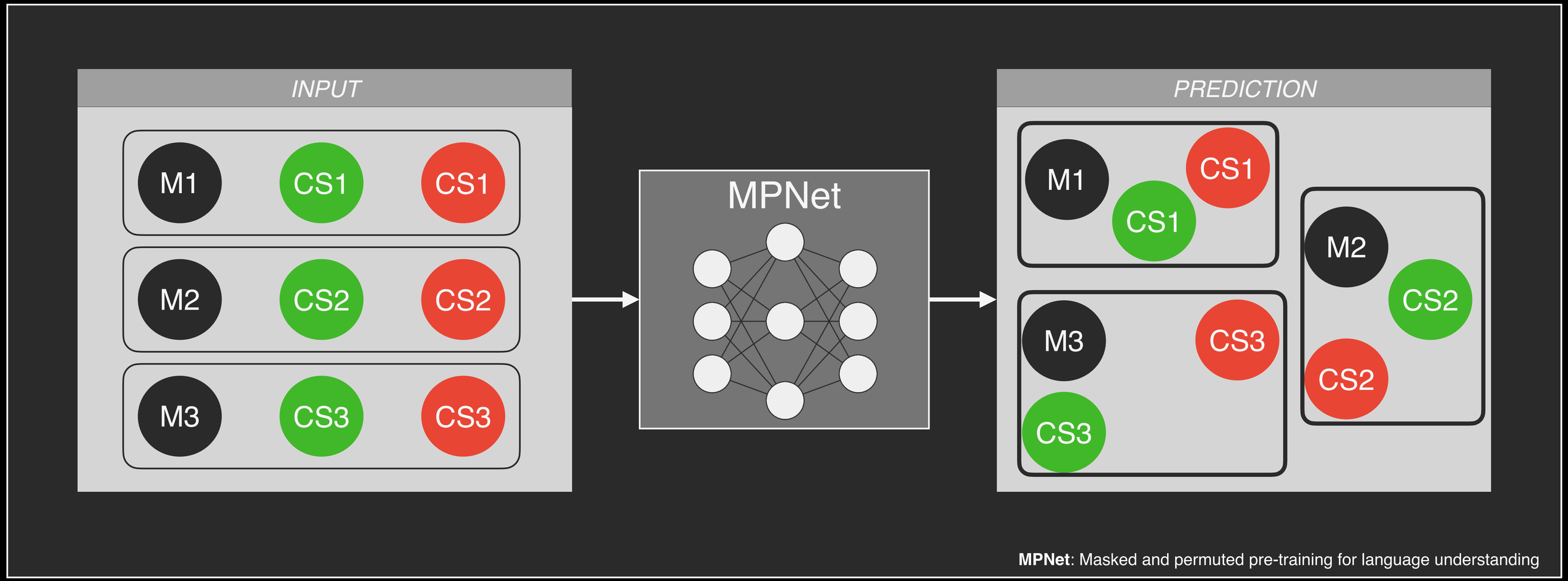
Contrastive Learning Method for developing new metrics in Software Engineering





CodeXGLUE: A Machine Learning Benchmark Dataset for Code Understanding and Generation —*Lu et al.*—

Contrastive Learning Method for developing new metrics in Software Engineering



MPNet: Masked and permuted pre-training for language understanding

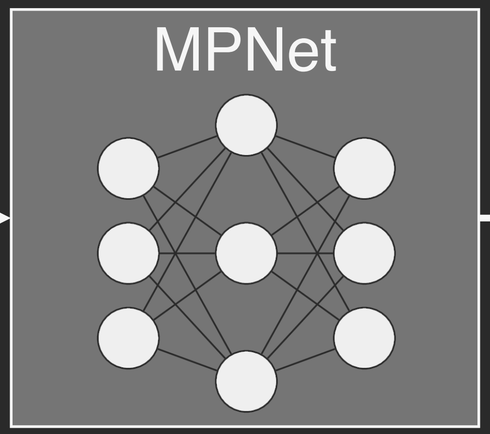
Contrastive Learning Method for developing new metrics in Software Engineering

SIDE (Summary alIgnment to coDe sEmantic)

INPUT

Create a connection to the consumer.

```
public ConnectionConsumer
        createConnectionConsumer
        …{
      if(LOGGER.isTraceEnabled())
      {
         ActiveMQRALogger.LOGGER.
            trace("Create
               connectionConsumer");
      }
      else {
         …
     }
}
```



PREDICTION

0.81

MPNet: Masked and permuted pre-training for language understanding