Improving API Knowledge Discovery with ML
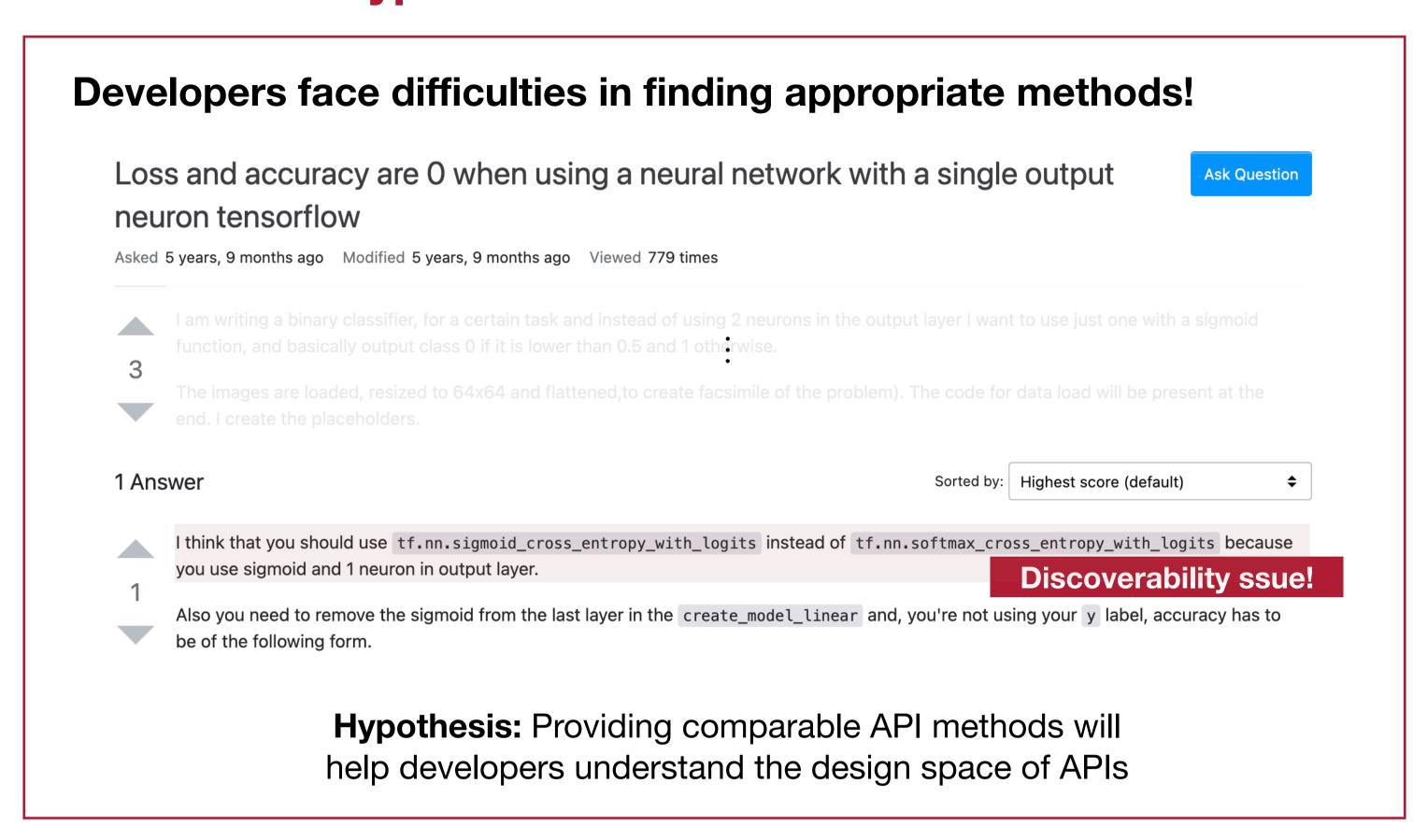
A Case Study of Comparable API Methods

Daye Nam, Brad Myers, Bogdan Vasilescu, Vincent Hellendoorn

> **TL;DR:** We investigated the usefulness of extracting comparable API methods from Stack Overflow posts. We provide evidence that showing comparable API methods can improve developers' understanding of API design space, and showed that an ML-based model can extract such knowledge from Stack Overflow.

Motivation & Hypothesis

Developers face difficulties in finding appropriate methods!

Loss and accuracy are 0 when using a neural network with a single output neuron tensorflow

[Ask Question]

Asked 5 years, 9 months ago   Modified 5 years, 9 months ago   Viewed 779 times

3

I am writing a binary classifier, for a certain task and instead of using 2 neurons in the output layer I want to use just one with a sigmoid function, and basically output class 0 if it is lower than 0.5 and 1 otherwise.

The images are loaded, resized to 64x64 and flattened to create facsimile of the problem. The code for data load will be present at the end. I create the placeholders.

1 Answer   Sorted by: Highest score (default)

I think that you should use `tf.nn.sigmoid_cross_entropy_with_logits` instead of `tf.nn.softmax_cross_entropy_with_logits` because you use sigmoid and 1 neuron in output layer.

Discoverability Issue!

Also you need to remove the sigmoid from the last layer in the `create_model_linear` and, you're not using your `y` label, accuracy has to be of the following form.

Hypothesis: Providing comparable API methods will help developers understand the design space of APIs

Dataset

Manually annotated Stack Overflow Answers containing Tensorflow methods, based on annotation protocol.

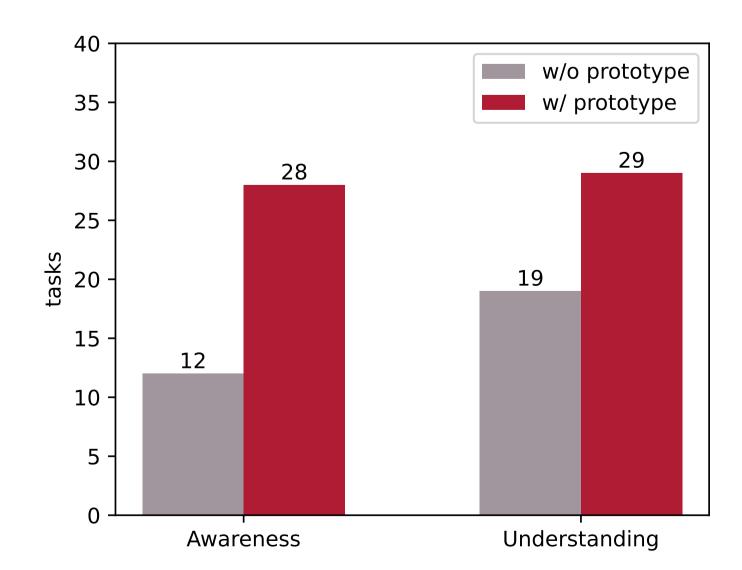| # SO answers	W/ relation	# Comp. Pairs	# Summary sents.
587	198	266	737

User Study for Hypothesis Testing

Study Design

▷ 16 participants who know ML, but not TF.
▷ Participants used general Google search, with and without the prototype, to find appropriate TF methods for given tasks, e.g.:

```
[ ] '''
    Select rows from tensor input1
    if the corresponding value in tensor input2 is True.

    input1: tf.Tensor([[ 7  9]
                       [ 1  6]
                       [12  4]], shape=(3, 2), dtype=int32)
    input2: tf.Tensor([False  True  True], shape=(3,), dtype=bool)
    output: tf.Tensor([[ 1  6]
                       [12  4]], shape=(2, 2), dtype=int32)
    '''
    def task(input1, input2):
        # What Tensorflow function(s) should we use here?
        output = ?
        return output
```

Results

▷ Participants using the prototype were **more aware** of comparable API methods (coeff=3.03, p=0.0015) and had a **better understanding** of the differences (coeff=2.64, 0.0056).
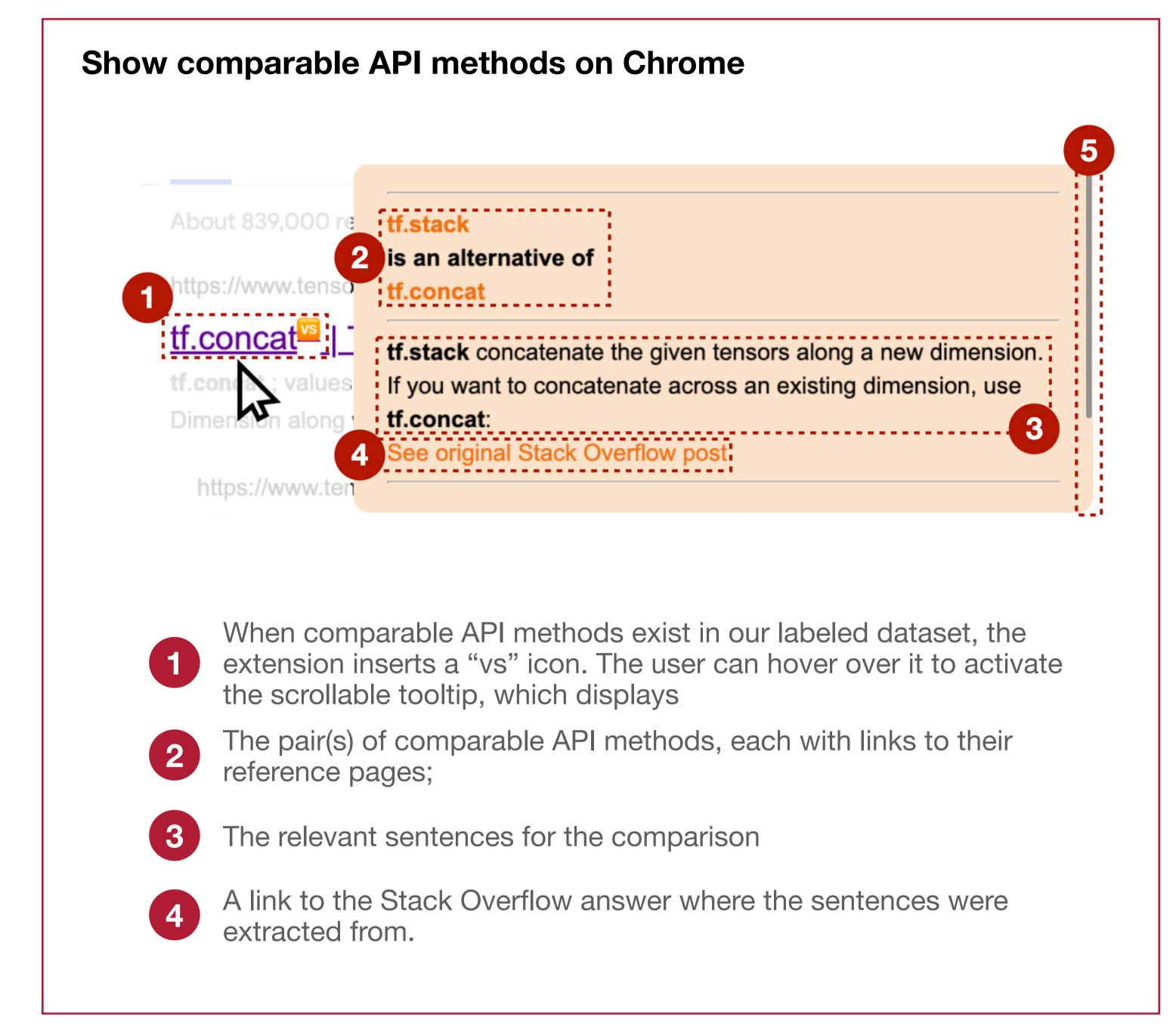

P1: *"I think the tool allowed me to explore more methods, more easily in the same page without retyping the search keyword."*

P12: *"... That [tool] gives the "analogy based search". Search for things that are similar. So if I have a general sense about what I'm looking for and I use the extension, then it can help me to find the right function to use."*
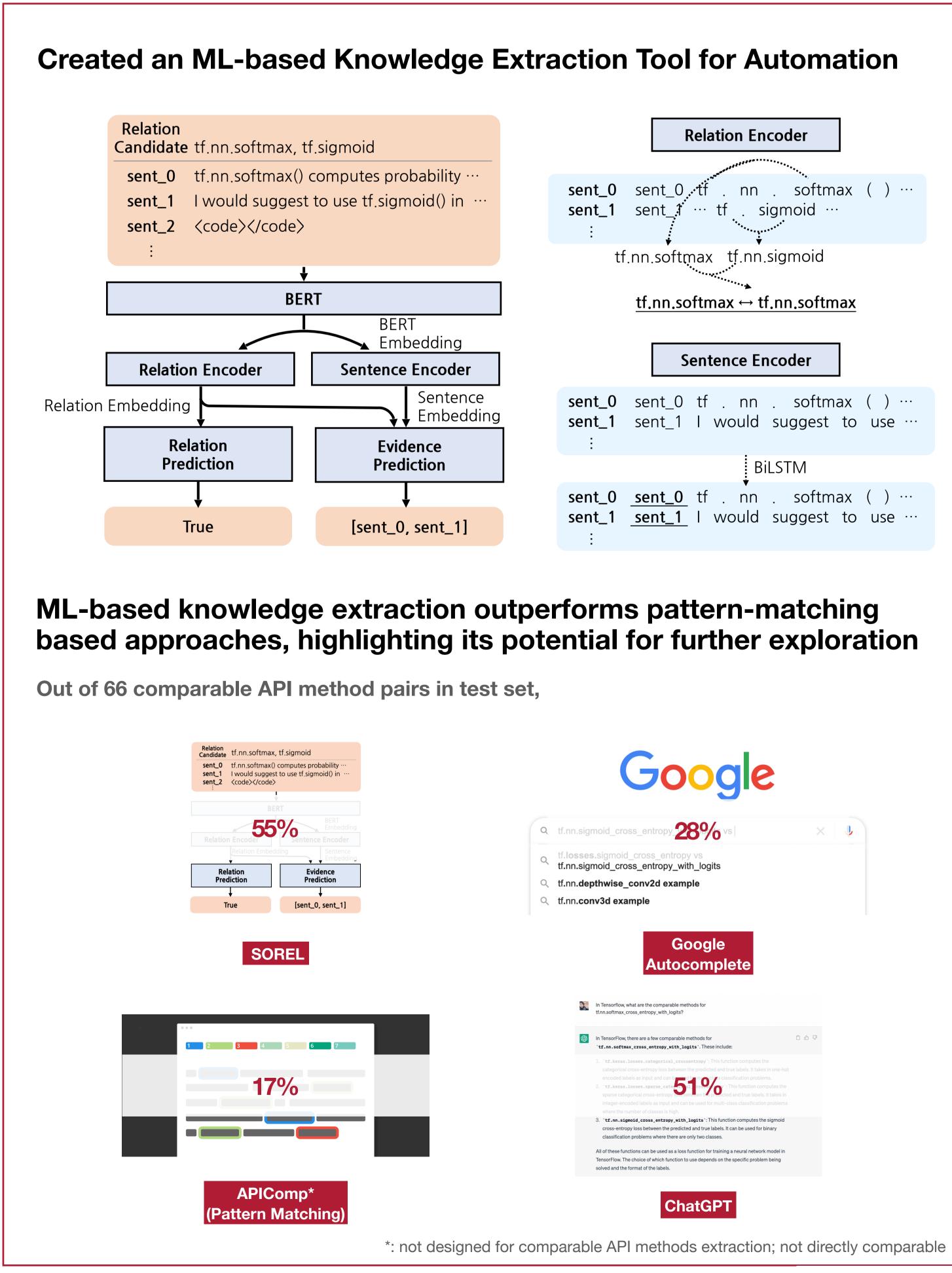
Implications for Automation

▷ Prioritize extracting comparable API methods over summaries.
▷ Prioritize improving recall over precision when extracting pairs
▷ Limit themselves to extractive over abstractive summarization.

Prototype for Hypothesis Testing

Show comparable API methods on Chrome

tf.stack
is an alternative of
tf.concat

tf.stack concatenate the given tensors along a new dimension. If you want to concatenate across an existing dimension, use **tf.concat**:

See original Stack Overflow post.

1. When comparable API methods exist in our labeled dataset, the extension inserts a "vs" icon. The user can hover over it to activate the scrollable tooltip, which displays

2. The pair(s) of comparable API methods, each with links to their reference pages;

3. The relevant sentences for the comparison

4. A link to the Stack Overflow answer where the sentences were extracted from.

SOREL: Stack Overflow RELation extractor

Created an ML-based Knowledge Extraction Tool for Automation

ML-based knowledge extraction outperforms pattern-matching based approaches, highlighting its potential for further exploration

Out of 66 comparable API method pairs in test set,

55%

SOREL

28%

Google Autocomplete

17%

APIComp*
(Pattern Matching)

51%

ChatGPT

*: not designed for comparable API methods extraction; not directly comparable