ASTCODA: Abstract Syntax Tree Convolutions Operating on Domain Attention

Liudmila Paskonova [1], Lomonosov Moscow State University
Supervised by Dr. Alexander Chernov [2]

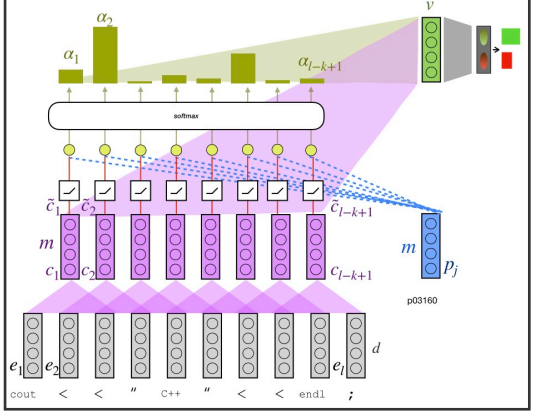
Background: Approaches to code segmentation have limitations

- require fixed segment annotations
- miss domain nuances
- limited to specific programming languages

Problem:
We want to extract features from *unlabeled* source code *automatically*.

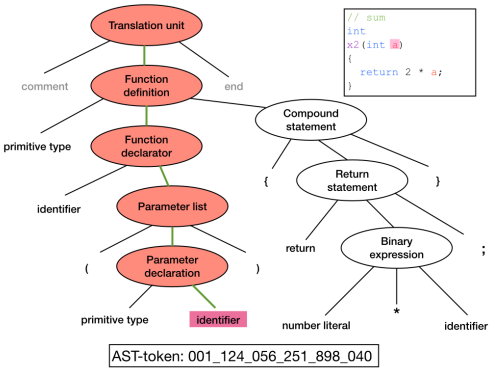
Solution:
Features are sequences of consecutive tokens. We employ AST-based tokenizer and a CNN with attention to assess their importance. Our framework is **AST-based**, **self-contained**, **domain-aware** and **multi-language**.

Model Architecture



AST-based tokenization preserves the structure of the program code

- To construct AST we use Tree-Sitter
- AST-tokenization worked better than word-based approaches



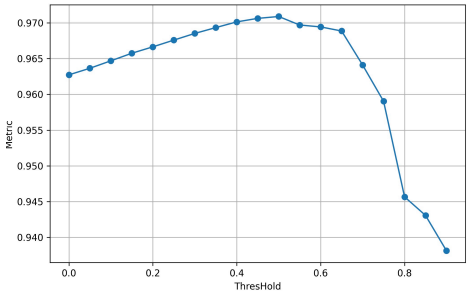
AST-token: 001_124_056_251_898_040

Higher attention weights highlight more important features

$$\text{RSDM}(\text{pred}, \text{gt}) = 1 - \frac{\text{distance}(\text{pred}, \text{gt})}{\text{worst\_distance}(L, \text{gt})}$$

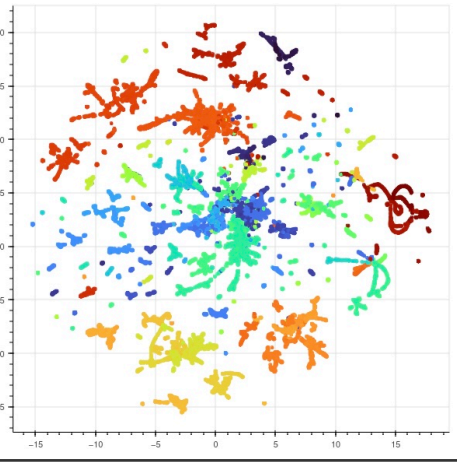
$$\text{distance}(\text{pred}, \text{gt}) = \sum_{i=0}^{M-1} \min_{j \in \{0 \dots K-1\}} |\text{pred}_i - \text{gt}_j|$$

$\tilde{\alpha}_i \geq \text{threshold} \implies$ feature i is more likely to correspond the predicted class



Embeddings of neighboring AST-tokens form syntax clusters

This helps solve OOV problem.



Our model finds logical errors

```cpp
#include<bits/stdc++.h>
using namespace std;
int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);cout.tie(0);
    long i=0;
    long n;
    vector <long> r;
    cin>>n;
    long J;
    for (int j=0;j<n;j++)
    {
        cin>>J;r.push_back(J);
    }
    J=0;
    while(i<n)
    {
        if (abs(r.at(i)-r.at(i+1))>abs(r.at(i+2)-r.at(i)))
        {
            i+=2;
            J+=abs(r.at(i+2)-r.at(i));
        }
        else
    {i++;
        J+=abs(r.at(i+1)-r.at(i));
    }

    }
    cout<<J<<endl;
    return 0;
}
```

Task	Source	Samples	Domain	#	Class	#	lang.
Vulnerability detection	FormAI dataset	336523	Entire dataset	1	Correct/ Vulnerable	2	C
Error localization in student programming submissions	Project CodeNet	449950	Problem	30	Correct/ Partial Solution	2	C++

Code, data and pretrained models

https://github.com/Liudmila-Paskonova/ASTCODA

[1] paskonovalv@gmail.com

PLDI'25 Student Research Competition

[2] blackav@gmail.com