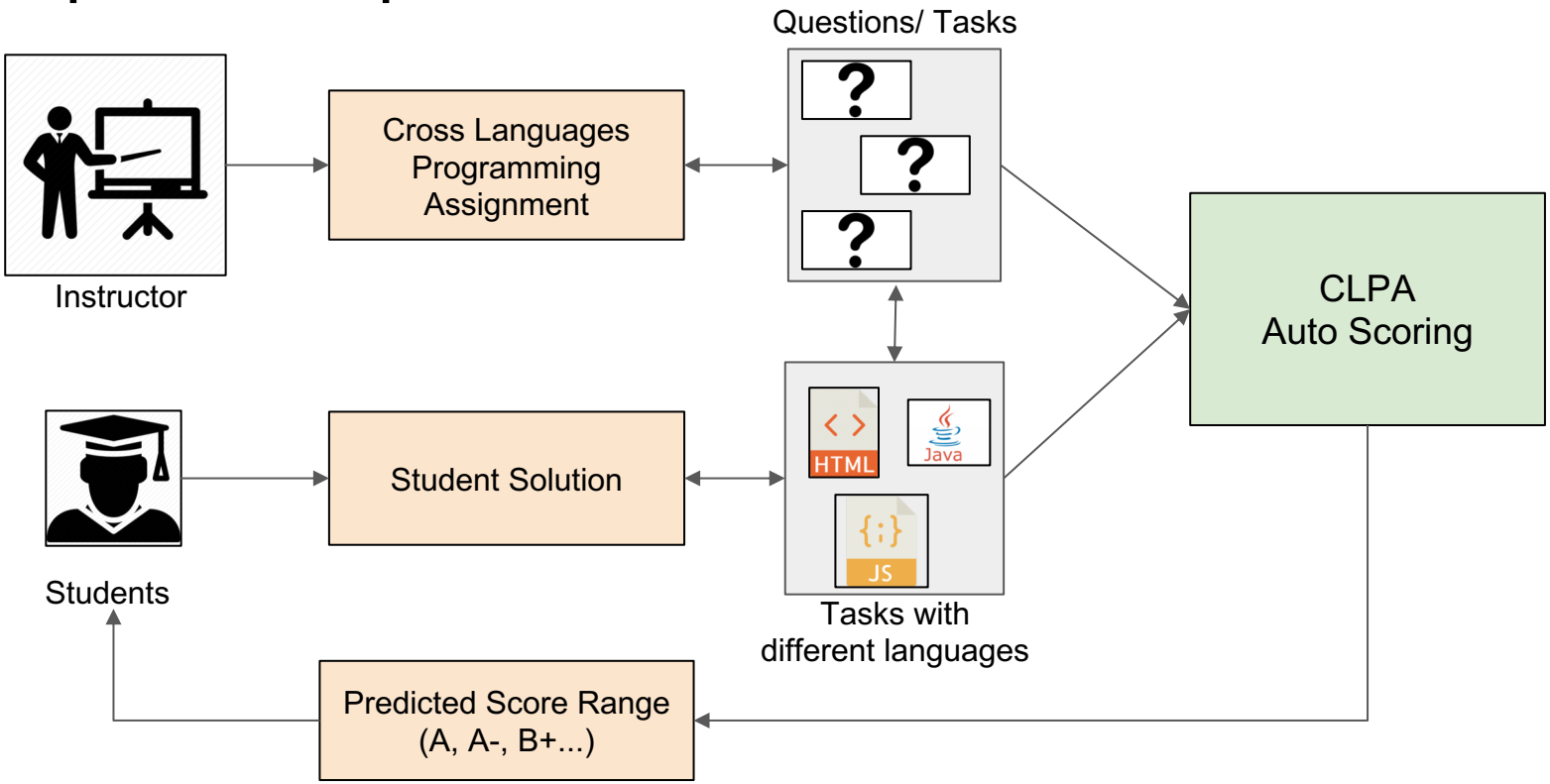
Automatic Scoring for Cross Languages Programming Assignment by Strategies of Code Vectorization and Machine Learning

Hung PHAN

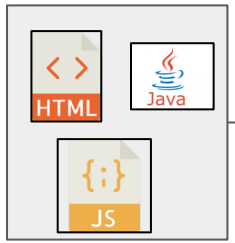
Overview

Grading student programming language assignments is a challenging task for instructors. They need to deal with the variety of software configurations and programming languages that can be used in the each students submissions in the form of programming projects. To alleviate this task, there are works on automatically scoring student works in the form of cross languages programming exercises by providing approaches for generating test cases for testing each programs. However, this trend of research requires to build a compiler system which can be costly to handle multiple types of assignment and types of testing and it supports for single programming language only. In this work, we want to overcome that challenge by proposing CLPAAutoScoring, a grading system that supports instructor to evaluate programming assignments in different programming languages and doesn't require test cases for running the programs. We propose the idea that a complex programming assignment can be represented as a vector, to make it able for Machine Learning for predicting the grade of that assignment. We study and analyze several strategies of modeling vectorization based on the source code, the parsed tree and the distance vector between each type of grade. We use a dataset of 130 valid PAs for a homework which contains Javascript and Html code in a Top-60 US University in Computer Science for evaluation. The result shows that, our proposed strategy using topic distance and parsed tree of source code in JavaScript and Html can classify with the weight accuracy of 94% on our best Machine Learning algorithm for 4 grade levels A, A-, B+, B, which shows the potential of our approach.

Input & Output



Techniques

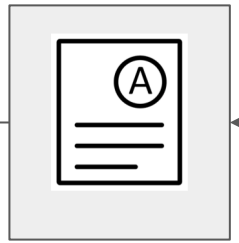


Student Solutions

Vectorization

...
0.2
0.4
0.5
...

Vector



Graded Assignment

Teaching Assistants

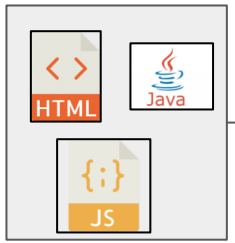
1. **Provide 3 strategies of vectorization:**
- Original in **NLP**: **TF-IDF**.
- Topic Distance on Source Code Tokens
- Topic Distance on AST nodes.

1. **Use 10 popular Machine Learning approaches.**

Machine Learning Approaches

1. `GaussianNB`
2. `LogisticRegression`
3. `DecisionTreeClassifier`
4. `RandomForestClassifier`
5. `AdaBoostClassifier`
6. `LinearDiscriminantAnalysis`
7. `QuadraticDiscriminantAnalysis`
8. `LinearSVC`
9. `MLPClassifier`
10. `GradientBoostingClassifier`

Original Strategy in NLP for Vectorization



United Document with Code Tokens

Term Frequency-Inverse Document Frequency (TF-IDF)

...
0.2
0.4
0.5
...

Vector

Result:
50% by **Decision Tree** Classifier

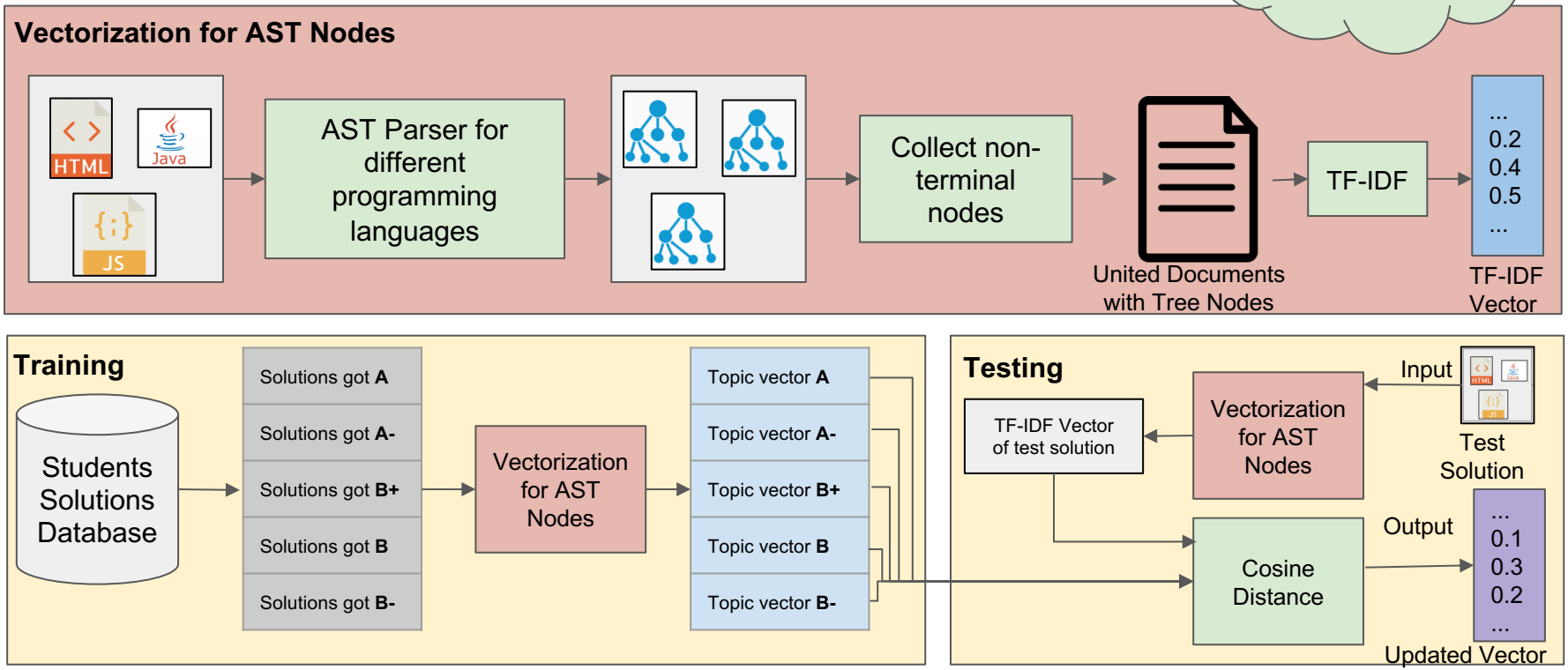
Evaluation:
130 student solutions for Homework 2 (JavaScript and HTML) of **COMS 319** (Construction of User Interfaces) of Iowa State University, Fall 2019.

Disadvantage:
Raw TF-IDF is used for **document summarization** instead of **score prediction**.

Grade	Score Range
A	50
A-	[45-50)
B+	[40-45)
B	[30-40)
B-	<30

Topic Distance on AST Nodes Vector

Result:
91% by
Decision Tree
Classifier



Contributions

1. Provide a vectorizing technique for cross languages programming assignment.
2. Analyzing popular machine learning approaches for the ability of learning the mapping from vector to label as students' actual grade.
3. Implement a base-line model for vectorization using TF-IDF in NLP and shows the drawback.
4. Implement improved approaches for vectorization based on topic distance vector and information of code tokens and AST nodes in cross languages.

In overall, **CLPAAutoScoring** predicts students' score based on their code structure and AST structures.