

# State of the art

## Predicția unei note cu input sursă de cod



Luca Marian

Odobesteanu Alexandru

Popescu Valentin

Ratundu Andrei

## Introducere

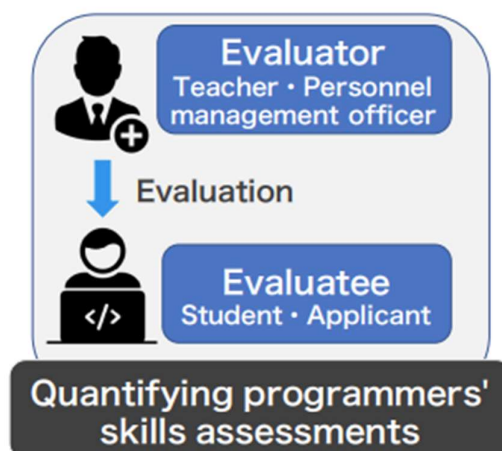
Domeniul inteligenței artificiale s-a dezvoltat rapid în decursul ultimilor decenii, odată cu apariția și expansiunea rețelelor sociale, care a dus la un mare aport în colectarea datelor. Astfel, inteligența artificială s-a infiltrat în mai multe domenii precum medicină, industrie, studiul comportamentelor oamenilor și animalelor și multe altele.

Educația este o problemă foarte importantă în dezvoltarea unei țări și a cetățenilor acesteia. O modalitate prin care se poate crește nivelul educației în școli este prin prezicerea performanței academice pe care a elevilor și studenților. Inteligența artificială este folosită și în analiza datelor și predicția comportamentelor. În acest document se va prezenta utilitatea AI în învățământ, în predicția notelor studenților.

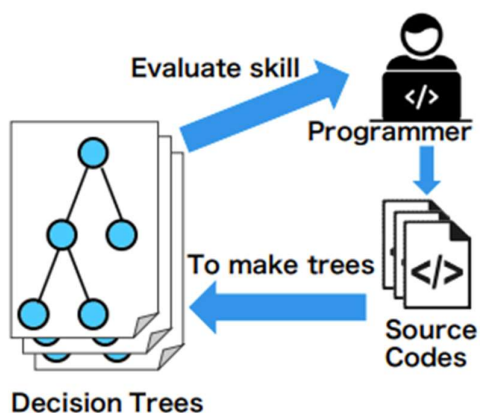
## Visualization of Programming Skill Structure by Log-Data Analysis with Decision Tree

### Introducere

Pentru a evalua capacităților unui programator în universități sau în companii, evaluatorii efectuează o analiză asupra sursei de cod în care studenții rezolvă probleme pe care le încarca pe GitHub. Codul sursă colectat este mai apoi analizat de un factor uman. Acest lucru poate avea diferite dezavantaje precum: codurile sursă nu pot fi evaluate simultan, astfel este dificil de menținut coerența criteriilor de evaluare în rândul evaluatorilor. Astfel, se propun metode de estimare a aptitudinilor unui programator prin analiza surselor de cod colectate automat. Studiul (Oeda & Chieda, 2019) vizează caracteristicile codului sursă al studenților folosind un arbore de decizie, pentru a estima abilitățile de programare.



Figură 1 Metoda tradițională



Figură 2 Metoda Propusă

La orele de programare, comportamentul elevilor poate fi confirmat folosind date-log. Am dezvoltat o înregistrare sistem care poate dobândi istoricul intrărilor comenzilor UNIX și istoricul editării codului sursă. Setul de date a fost luat de la elevii din școala și este format din datele de la lecțiile de programare, formate din date de la 39 de studenți.

## Metoda propusă

În acest studiu, arborele de decizie este utilizat în domeniul educației pentru a vizualiza caracteristicile astfel încât să fie ușor de înțeles de cei care nu sunt familiarizați cu învățarea automată. Arborii de decizie sunt modele de predicție bazate pe structuri de arbore și sunt potrivite pentru astfel de probleme, deoarece rezultatele predicției sunt scoase într-o formă care este ușor de înțeles pentru oameni. Prin urmare, în acest studiu, am folosit arbori de decizie pentru a vizualiza rezultatele pentru regresie. Procedura pentru generarea unui arbore de decizie este următoarea:

1. Introducerea tuturor datelor în nodul rădăcină
2. Împărțirea datelor în noduri astfel încât coeficientul Gini să fie minimizat
3. Repetarea pașilor 1 și 2 la fiecare nod divizat până când eroarea medie pătrătică din nod scade sub o anumită valoare sau arborele ajunge la o rădăcină anume

Coeficientul Gini este definit după cum urmează. Pentru un nod  $t$  cu un arbore de decizie, luați în considerare cazul în care există  $n$  mostre în nod și clase  $c$  în nod. În nodul  $t$ , presupunând că numărul de eșantioane aparținând clasei  $i$  este  $n_i$ , raportul  $p(i|t)$  eșantioanelor aparținând clasei  $i$  este dat de

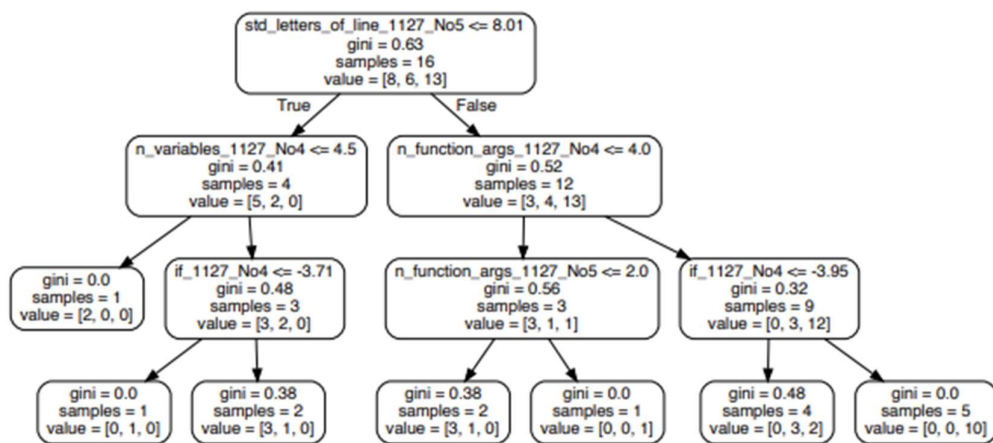
$$p(i|t) = \frac{n_i}{n}.$$

În acest studiu, jurnalul de comenzi și codul sursă colectat în fiecare minut sunt convertite în mai multe caracteristici pentru examinare periodică a disciplinei „Limbaajul de programare” în anul II al Colegiului de Tehnologie Kisarazu. Caracteristicile extrase sunt organizate într-un fișier CSV și utilizate pentru a genera un model de arbore de decizie. Unele dintre caracteristici utilizate sunt prezentate în tabelul următor:

Features	Abbreviation
Frequency of appearance of self-made function declaration	ln_functions
String length of self-made function name	mean_function_name
Number of function arguments	n_function_args
String length of variable name	mean_variable_name
Indentation method (BSD type / kernel type)	indentation_style
Frequency of appearance of blank line	ln_blank_lines
Number of characters per line	mean_letters_of_line
Distribution of the number of characters per line	std_letters_of_line
Indent character (tab / space)	tab_space_style
Frequency of appearance of “do”	do
Frequency of appearance of “while”	while
Frequency of appearance of “for”	for
Frequency of appearance of “if”	if
Frequency of appearance of “else if”	elseif
Frequency of appearance of “else”	else
Frequency of appearance of “switch”	switch
Number of lines in self-made function	mean_function_lines
Frequency of appearance of single-line comment	online_comment
Frequency of appearance of multi-line comments	multiline_comment

Figură 3 Tabel caracteristici

Arborele de decizie generat:



## Concluzii

În predicția notelor unui student trebuie să se țină cont de anumiți indicii și numeroase variabile. Trebuie setați parametrii care să măsoare atât calitatea, cât și cantitatea muncii unui elev/student. Astfel, după părerea mea, o predicție a notei bazată doar pe fișiere sursă în care să măsoară numărul de linii de cod, numărul de funcții, numărul de clase, etc., poate fi uneori eronată.

Monitorizarea și sprijinirea elevilor este considerată foarte importantă în multe instituții de învățământ. Dacă un profesor poate identifica un elev slab într-un stadiu incipient, poate lua măsuri pentru a se asigura că elevul nu va abandona cursul. Prin urmare, este necesară identificarea acestor elevi în timpul fiecărei lecții, astfel încât profesorul să poată acorda o atenție deosebită lor.

## Bibliografie

Oeda, S., & Chieda, M. (2019). *Visualization of Programming Skill Structure by Log-Data Analysis*. Kisarazu City: Procedia Computer Science 159.