CloneDetector

Program pentru depistarea proiectelor Java copiate

Adrian-Vlad LEP, Coordonator: conf.dr.ing. Ioana SORA

Universitatea Politehnica Timisoara

Facultatea de Automatica si Calculatoare

Calculatoare si Tehnologia Informatiei

Iunie 2011

Există diverse platforme ce detectează lucrări literare, proiecte şcolare sau progrme software copiate. Programul CloneDetector analizează static codul sursă şi depistează proiectele Java copiate pe baza unui set de reguli Prolog. CloneDetector foloseşte o abordare originală, combinând metodele de depistare bazate pe metrici şi cele bazate pe grafuri de apeluri (call graph) create din arbori sintactici abstracţi(AST). Prin acest instrument se doreşte depistarea clonelor de tipul 3, adică a proiectelor copiate în care pe lângă redenumire s-au operat schimbări de natură sintactică, s-au adăugat sau şters porţiuni de cod.

1. Arhitectura sistemului
Sistemul a fost dezvoltat pentru depistarea proiectelor şcolare Java copiate. Punctul de plecare va fi codul sursă predat. Figura alăturată descrie paşii parcurşi. Pentru compararea a două proiecte, din codul Java se va genera cu ajutorul plugin-ului Eclipse JTransformer arborele sintactic abstract (AST) al fiecărui proiect. AST-ul va fi reprezentat sub forma unor clauze Prolog şi memorat in fişiere qlf (quick load file), fişiere compilate Prolog.

Analiza a două proiecte începe la nivel de clasă. Încercăm să găsim pentru fiecare clasă din primul proiect o clasă echivalentă în cel de al doilea. Două clase vor fi considerate identice dacă vor avea similarităţi la nivel de clasă, dacă metodele lor vor fi similare şi dacă clasele care vor fi apelate din metodele celor două clase vor fi asemănătoare.

Din testele realizate am observat că analiza grafului de apeluri aduce o îmbunătăţire majoră, crescând numărul de împerecheri corecte la nivel de metode, implicit la nivel de clase şi proiecte.

Pentru a satisface nevoile profesorilor cât mai bine şi a uşura munca utilizatorilor, au fost dezvoltate diverse facilităţi, fără a afecta scalabilitatea algoritmului. CloneDetector permite excluderea unor clase, pachete sau foldere din analiză. Această facilitate se dovedeşte foarte utilă în cazul în care toate proiectele au porţiuni de cod date de la început.

Pentru a putea analiza diferit în funcţie de tema dată, am creat două profile de analiză unul larg şi al doilea mai exact. Aceste profile pot fi selectate la rulare sau noi profile pot fi uşor definite în fişierul profiles.pl.

De asemenea, datorită metodei de implementare se pot adăugă sau şterge relativ uşor filtre noi în funcţie de novoile ulterior apărute.

În concluzie CloneDetector se doreşte a fi un tool uşor de folosit şi eficient în depistarea proiectelor software copiate. În urmă testelor realizate asupra unor proiecte concrete, suntem încrezători în utilitatea acestei platforme. S-a obsevat că programele ce folosesc algoritmi de comparare a şirurilor de caractere, deşi foarte rapide, pot depistă doar clonele de tipul 1, eventual 2 şi 3 dacă redenumirile nu sunt complete. Instrumentele ce se bazează pe grafuri de apeluri pot semnala toate tipurile de clone dar datorită algoritmilor complexi, au o durată de procesare mai mare. CloneDetector poate depista toate clonele de tipul 1 şi 2 indiferent de numărul de redenumiri efectuate, semnalând şi clone de tipul 3 cu probabilitatea lor de a fi copiate.