وزارة التعليم العالي والبحث العلمي
جامعة الموصل
كلية علوم الحاسوب والرياضيات
قسم البرمجيات

تصميم وتنفيذ نموذج لكشف نسخ البرامج غير المعتمد على اللغة

رسالة مقدمة
الى مجلس كلية علوم الحاسوب والرياضيات في جامعة الموصل
كجزء من متطلبات نيل شهادة الماجستير في
البرمجيات

من قبل
شهد سعد عمر خطاب

بإشراف
أ.م.د أسماء ياسين حمو سليمان

1445 هـ 2024 م

المستخلص

في هندسة البرمجيات ، يمكن تعريف استنساخ الشفرة البرمجية البرمجية(Code Clone(CC على أنه تطابق أو تشابه جزء من الشفرة البرمجية مع جزء آخر في البرنامج نفسه أو بين البرامج. كما انه يحدث في مستويات مختلفة: بدءًا من الجملة إلى الدوال أو الوحدات . هناك أربعة أنواع من أستنساخ الشفرة البرمجية وهي النوع الأول(Exact Clone),النوع الثاني(Renamed Clone),النوع الثالث(Gapped Clone),والنوع الرابع (Semantic Clone). على الرغم من لجوء المبرمجين الى استنساخ البرمجيات للسهولة و لأسباب اخرى, الا انه يحوي مخاطر عدة كصعوبة صيانة البرمجيات و انتشار الأخطاء فضلا عن انتهاك حقوق الملكية ... الخ. لذلك فمن الضروري العثور على هذا الاستنساخ عن طريق الكشف عن استنساخ الشفرة البرمجية "Code Clone Detection(CCD) ". هناك عدة انواع من CCD أولها هو الاسلوب المعتمد على النصText Base ثانيا الاسلوب المعتمد على الرموز"Token Based" أما النوع الثالث فيعتمد على تكوين الشجرة"Abstract Syntax Tree(AST)" أما النوع الرابع فيعتمد على تكوين الرسم البياني" Program Dependence Graph(PDG) " خامسا الاسلوب المعتمد على المقاييس "Metric Based" أخيرا الأسلوب المعتمد على التهجين "Hypird based". وبما أن اكتشاف النسخ مرتبط بتركيب الجملة في اللغة, فان اكتشاف النسخ بين برامج مكتوبة بلغات مختلفة يصبح صعبا.

تم في هذه الرسالة تصميم و بناء نموذج مقترح يجمع بين الاسلوبين اسلوب الطرائق التقليدية مع اسلوب خوارزميات التشابه لايجاد اكبر نسبة تشابه بين ملفين مكتوبين بلغات مختلفة هي (C#,C++,Java, Python). والكشف عن النوع الاول و الثاني و الثالث من نسخ البرامج , بتطبيق التقنية الهجينة Hybird Approach التي تجمع بين تقنية الاعتماد على النص وتقنية الاعتماد على الرموز. يليها تطبيق النموذج على خمس خوارزميات تشابه وهي (Jaccard similarity, Cosine similarity, Longest Common Subsequence, Smith–Waterman, Levenshtien similarity).لايجاد افضل خوارزمية بينهم.

استخدم النموذج التزايدي في التصميم,اذ احتوى على خمس مراحل وهي المعالجة الاولية,مرحلة التقطيع, توحيد المتغيرات, مقارنة التشابه, والتنفيذ و عرض النتائج. لكل مرحلة تم رسم المستوى التفصيلي لمخطط حالة الاستخدام ومخطط الصنف للتصميم الهيكلي, ثم مخطط النشاط في تصميم المكونات.

تم تنفيذ النموذج على 11 برنامجا مختلفا كل منها مكتوب بلغات الأربع المذكورة اعلاه(اي مامجموعه 44 برنامجا)تم اختيارها بحيث تشمل معظم انواع الجمل البرمجية و باستخدام خوارزميات التشابه الخمس أعلاه.بعد الاطلاع على النتائج تبين ان خوارزميةCosine Similarity اعطت افضل النتائج لمعظم حالات الاختبار.وكمعدل حصلت على 0.96, بينما خوارزمية Smith Waterman كانت الأسوأ اعطت 0.37.لذلك يوصى اعتماد خوارزمية Cosine Similarity في كشف النسخ بين

البرامج المكتوبه باللغات اعلاه. أقل نسبة تشابه ظهرت باستخدام خوارزمية Smith Waterman لذلك يوصى بتجنبها عند ايجاد التشابه بين البرامج المكتوبة بلغات مختلفة.

استخدم البرنامج Enterprise architect في تصميم النموذج. وتمت كتابة البرامج باستخدام المحرر PyCharm Community Edition.

Ministry of Higher Education and
Scientific Research
University of Mosul
College of Computer Science and
Mathematics
Department of Software

Designing and Implementation a model for Language-independent Code Clone Detection

**A Thesis Submitted to the Council of the College of
Computer Science and Mathematics
University of Mosul
as a Partial Fulfillment of Requirements
for the Degree of Master
in Software**

By
Shahad Saad Omer Khatab

Supervised by

A.P.D. Asma Yaseen Hamo Suleman

2024 A.D. **1444 A.H.**

Abstract

In software engineering, a Code Clone (CC) can be defined as the match or similarity of one part of a program to another part in the same program or between programs. It can be in different levels: start from statement to functions or units. There are four types of code clone: Type1_Exact Clone ,Type2_Renamed Clone, Type3_Gapped Clone, Type4_Semantic Clone. Although programmers uses software clone to simplify the programming process and for other reasons, it has several risks, such as the difficulty of maintaining the software, the spread of errors, the violation of property rights and etc. It is therefore necessary to find this cloning by detecting the reproduction of the code "Code Clone Detection CCD." There are several techniques of CCD. First of all, it's a "Text-Based" method. Second, method based on the symbols "Token Based", the third Abstract Syntax Tree (AST)". The fourth technique "Program Dependence Graph (PDG)" , then "Metric based". Finally, the "Hybrid based." Since the discovery of copies is linked to the composition of the sentence in the language, it is difficult to detect copies between programmes that are written in different languages.

In this thesis, a proposed model was designed and built that combines the traditional style of methods with machine learning to find the most similar ratio between two files written in different languages (C#, C++, Java, Python). The model detects type I, type II and type III of code clone through the applying of Hybird Approach. The Hybrid approach combines text-based with Token-based CCD techniques. Five of similar algorithms are used to find which one is the best. They are (Jaccard similarity, Cosine similarity, Longest Common Subsequence, Smith-Waterman, Levenshtien similarity).

The incremental model is used in the design, which consists of five stages: the Preprocessing, the Tokenization, the standardization of variables, the comparison of similarities, and the implementation and presentation of results. For each phase in five stages, the Use Case diagram is given to represent the analysis stage and for the structural design the class diagram is used. The activity diagram is used in the design of the components.

The model is implemented on 11 different programmes, each are written in four languages (a total of 44 programmes) that were selected to cover some types of software sentences and using the five similarity algorithms above.

The results show that the Cosine Similarity algorithm gave the best results in most cases and as an average of 0.96, while the Smith Waterman algorithm was 0.37 the worst algorithms.It is therefore recommended that the Cosine Similarity algorithm be used to detect copies between programmes written in languages (C#, C++, Java, Python).

The lowest degree of similarity was shown using the Smith Waterman algorithm, so it is recommended that it be avoided when matching programmes written in different languages.

Also, Python's language and because of its nature which rely on the calling of libraries, has given the least result of a CCD between its programmes and the rest of the languages.

Enterprise architecture was used in model design. The programs were written in PyCharm Community Edition.