Scientific Journal of Impact Factor(SJIF): 4.14

e-ISSN(O): 2348-4470
p-ISSN(P): 2348-6406

International Journal of Advance Engineering and Research Development

Special Issue for ICPCDECT 2016,Volume 3 Issue 1

SOFTWARE PLAGIARISM DETECTION

Pooja Patil[1], Kajal Thapa[2], Rohit Patil[3], Aniruddha Wathare[4]

[1-4]*Computer Science, AISSMS's Institute of Information Technology*

Abstract - *With the development of internet and electronic devices, software plagiarism hasbecomeprevalentinsoftwareindustriesaswellaseducationalinstitutes, violating one's intellectual integrity. Certain techniques such as watermarking and semantics-preserving code obfuscations were introduced to tackle this issue. However, besides the need to insert additional data in the original program, code obfuscations can often destroy watermarks. Also, it was found that a sufficiently determined attacker may be able to destroy any watermark. In order to overcome these issues, birth-marking technique is proposed, which extracts a set of characteristics that uniquely identify the original program. Our work focuses on extracting birthmarks from source codes, implementing algorithms to measure the similarity between them and displaying the results on a dedicated user interface with respect to a pre-set threshold.*

Keywords- *Plagiarism, Birth-marking, Jaccard, Cosine, Dice, Reflect API, feature set*

I. Introduction

Software Plagiarism is the practice of claiming, or implying, original authorship, or incorporating the code from someone else's source code in whole or in part, into one's own, without adequate acknowledgement.

Software Plagiarism is of major concern for the software industry as well as the Academia. Finding ways to identify and combat Plagiarism is central to maintaining one's intellectual integrity, and therefore a tool to detect Software Plagiarism is required. There are different tools that detect software plagiarism using different techniques like tokenization, parameterized-matching etc. but the idea was to conquer the inefficiencies of those techniques.Our work focuses on extracting birthmarks from source codes using Reflect API, implementing algorithms to measure the similarity between them and displaying the results on a dedicated user interface with respect to a pre-set threshold.

II. Proposed System

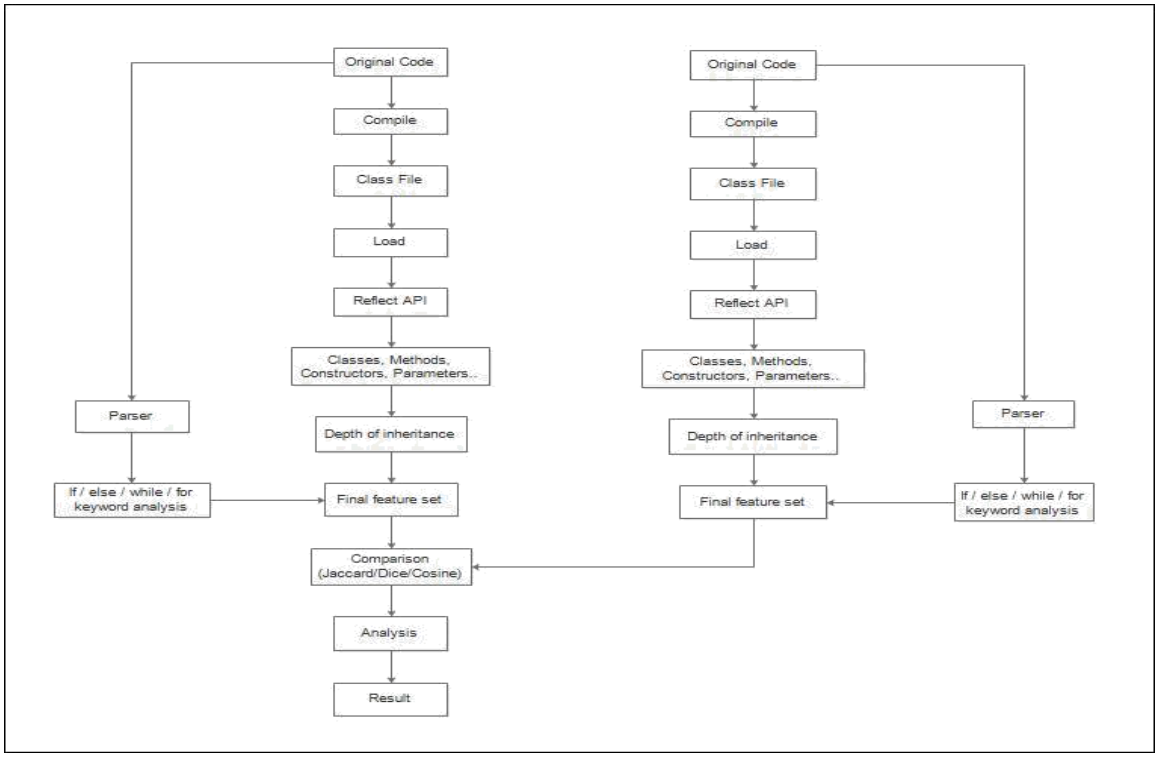
Plagiarism detection in software is a complex process, but there are tools that have lexical basis which are not that powerful, so we have designed a framework with the following salient features:

(1) Designs, loops, structures are compared instead of code

(2) Comparison follows a stepwise process according to the abstraction levels.

(3)Ratherthanjustcomparingvariableswecomparethekeywords,numberoffunctions, number of constructors, number of parameters etc.

The tool uses Reflect API of the java framework to extract certain characteristics such as number of constructors, number of functions, depth of inheritance etc. Consequently, it parses the code to find the keywords used in the software source code. A feature set is made out of the extracted characteristics and stored in the database.



The source code of the software to be checked for plagiarism is processed in the above mentioned fashion and the extracted features are compared with those in the database using algorithms Cosine, Jaccard and Dice to find the similarity.

I. Cosine Similarity

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

II. Jaccard Index

$$f(A,B) = \frac{A \cdot B}{|A|^2 + |B|^2 - A \cdot B}$$

III. Dice Coefficient

$$s_v = \frac{2|A \cdot B|}{|A|^2 + |B|^2}$$

III. Conclusion

With the growing trend of illegal code reuse, software thefts are becoming more common, compromising one's intellectual integrity. The proposed system can be used to settle the disputes of software thefts in software industries as well as in academic institutes to check the uniqueness of the software projects.

Mainly, the system computes and displays the similarity value between the source code of the software to be checked for plagiarism and those in the database. Compared to conventional systems that are used to check for plagiarism, this system proposes a different view of plagiarism detection by using utilities of java framework to extract and compare the feature sets.

IV. References

[1] Zhenzhou Tian, Qinghua Zheng, Ting Liu, Ming Fan, Eryue Zhuang and Zijiang Yang, "Software Plagiarism Detection with Birthmarks based on Dynamic Key Instruction Sequences", 2015 IEEE Transactions.

[2] Yoon-Chan Jhi, Xinran Wang, XiaoqiJia, Sencun Zhu, Member, Peng Liu and Dinghao Wu, "Program Characterization Using Runtime Values and Its Application to SoftwarePlagiarismDetection" ,IEEETRANSACTIONSONSOFTWARE ENGINEERING, 2015

[3] Fangfang Zhang,Dinghao Wu,Peng Liu andSencun Zhu, "Program Logic Based Software Plagiarism Detection", 2014 IEEE 25th International Symposium on Software Reliability Engineering.

[4] ShanmugasundaramHariharan, "Automatic Plagiarism Detection Using Similarity Analysis", The International Arab Journal of Information Technology, Vol. 9, No. 4, July 2012.

[5] SamerAbd El -Wahed, Ahmed Elfatatry and Mohamed S. Abougabal,"A New Look at Software Plagiarism Investigation and Copyright Infringement", 2007 IEEE.

[6] Tapan P. Gondaliya,Hiren D. Joshi and Hardik Joshi,"Source Code Plagiarism Detection SCPDet: A Review", International Journal of Computer Applications.

[7] VikasThada and Dr VivekJaglan,"Comparison of Jaccard, Dice, Cosine Similarity Coefficient To Find Best Fitness Value for Web Retrieved Documents Using Genetic Algorithm", International Journal of Innovations in Engineering and Technology (IJIET).

[8] Georgina Cosma and Mike Joy, "An Approach to Source-Code Plagiarism Detection and Investigation Using Latent Semantic Analysis", IEEE TRANSACTIONS ON COMPUT-ERS, VOL. 61, NO. 3, MARCH 2012

[9] AsakoOhnoand Hajime Murano, "A TWO-STEP IN-CLASS SOURCE CODE PLAGIARISM DETECTION METHOD UTILIZING IMPROVED CM ALGORITHM AND SIM", International Journal of Innovative Computing, Information and Control ICIC International Volume 7, Number 8, August 2011.

[10] SuphakitNiwattanakul, JatsadaSingthongchai, EkkachaiNaenudorn and SupachanunWanapu,"Using of Jaccard Coefficient for Keywords Similarity", Proceedings of the International MultiConference of Engineers and Computer Scientists 2013 Vol I, IMECS 2013, March 13 - 15, 2013, Hong Kong.

