ISSN No:-2456 –2165

Duplicate Source Code Detection using Adaptive and Progressive Approach

Waghmare Archana B. , Chopade Kajal P. , Kokare Rucha S. , Kumbhar Shivanjali S. , Mokashi Nayan N.
Department of Computer Engineering, S.V.P.M. COE Malegaon(BK),413115 Savitribai Phule Pune University, Maharashtra.

Abstract:-Duplicate detection is a data compression techniquefor identifying duplicate copies of repeating data. Today, duplicate detection technique need to process ever larger datasets in ever shorter time but maintaining the quality of datasets. We present adaptive and progressive approaches that significantly increase efficiency for finding duplicates. In this paper,the adaptive and progressive approaches and different algorithms are used to detect and calculate the percentage of duplications from source code. Duplication is a big concern in academics and it can be a problem in every course. Duplications occurs when someone copy or present others work as their own work. Students make duplications in different areas: homework assignments, essays,projects,coding, etc. In this paper we focus on programming languages and detect the percentage of duplications in programming assignments.

Keywords—Data Cleaning, Stop Word Elimination, Stemming, Code Clone, Duplicate Detection

I. INTRODUCTION

Code clones means similar or identical code. These are obtained by reusing code by copy-paste technique . Duplication of code occurs generally during the development of software system. Code cloning is a form of software reuse and exists in every software project. There are number of reasons for duplications(cloning) of source code. Main reason is that the programmer uses copy and paste feature than writing the code.Sometimes programmer try to implement new logic but find some working code that performs exactly same logic to the desired code then the programmer will copy the entire code and it makes cloning. In software industries, programmers mostly reuse the code by copy-paste technique to reduce development time.The copy-paste technique reduces programming effort and time.Therefore programmers mostly prefer copy-paste technique over writing new code.This code cloning is not only available in software industries but also available in academics.

Generally students copy the source code from his friend or from internet.It was hectic task for teachers to check all students assignments and finding the duplications from that assignments.This system will be used for teachers to check all students assignments easily.

In these system,we give two files first is source file andsecond is target file.We can give any extension to the file.

II. LITERATURE SURVEY

Recently, code clones have received much attention. Code clones means identical or similar code fragments to one another in source code. Software cloning and its detection techniques is an important research area in IT industry.For customers and IT company code clones causes serious threat to the security, quality and maintenance. Therefore, it is necessary to detect and check the various types of code clones. Many researchers have worked in this field.

Pay-As-You-Go Entity Resolution-This paper investigates how we can maximize the progress of ER with a limited amount of work using hints which give information on records that are likely to refer to the same real-world entity.Alsointroduce a family of techniques for constructing hintsefficiently and techniques for using the hints to maximize thenumber of matching records identified using a limited amountof work.Here user can maximize the progress of ER withthe limited amount of work using hints.Gathering individualsprofiles on social sites can yield a huge number of recordsthat need to be resolved. Contrasting each match of recordswith gauge their “similitude” can be costly.[1]

Top-k Set Similarity Joins-In this paper, Xiao et al.studied a variant of the similarity join, termed top-k setsimilarity join. It returns the top-k pairs of records ranked bytheir similarities, thus eliminating the guess work users haveto perform when the similarity values is unknown beforehand. An algorithm, top k-join, is proposed to answer top-ksimilarity join efficiently. It computes most similar recordpairs without the need to specify a similarity threshold. Itsupports interactive near duplicate detection applications,where users are presented with top-k most similar recordpairs progressively.The execution of the top-k similarity joincan be stopped at any time.[2]

Adaptive Windows for Duplicate Detection-In this paperDuplicate Count Strategy (DCS) a variation of SNM thatuses a varying window size. It is based on the intuitionthat there might be regions of high similarity suggesting alarger window size and regions of lower similarity suggestinga smaller window size. Next to the basic variant of DCS,author also propose and thoroughly evaluate a variant calledDCS++ which is probably better than the original SNM interms of efficiency (same results

with fewer comparisons).Itis Perfect Classifier to classify each type of data. DCS++is more efficient than SNM without loss of effectiveness.Our experiments with real-world and synthetic data setshave validated.It increases size of window and also increasecosting.[6]

III. PROPOSED SYSTEM

Proposed system have five sections i.e five modules. Themodules are as follows-

a) ***Preprocessing***- Data pre-processing is an important stepin the data mining process. The phrase ”garbage in,garbage out” is particularly applicable to data miningand machine learning projects. Data-gathering methodsare often loosely con- trolled, resulting in out-of-range,impossible data combinations (e.g., dot, double cot,etc.), missing values, etc. Analyzing data that has notbeen carefully screened for such problems can producemisleading results. Thus, the representation and qualityof data is first and foremost before running an analysis.In Our Project we are removing the special characterfor the code to improve the performance of our system.Here we are also refining the grammar for the code.

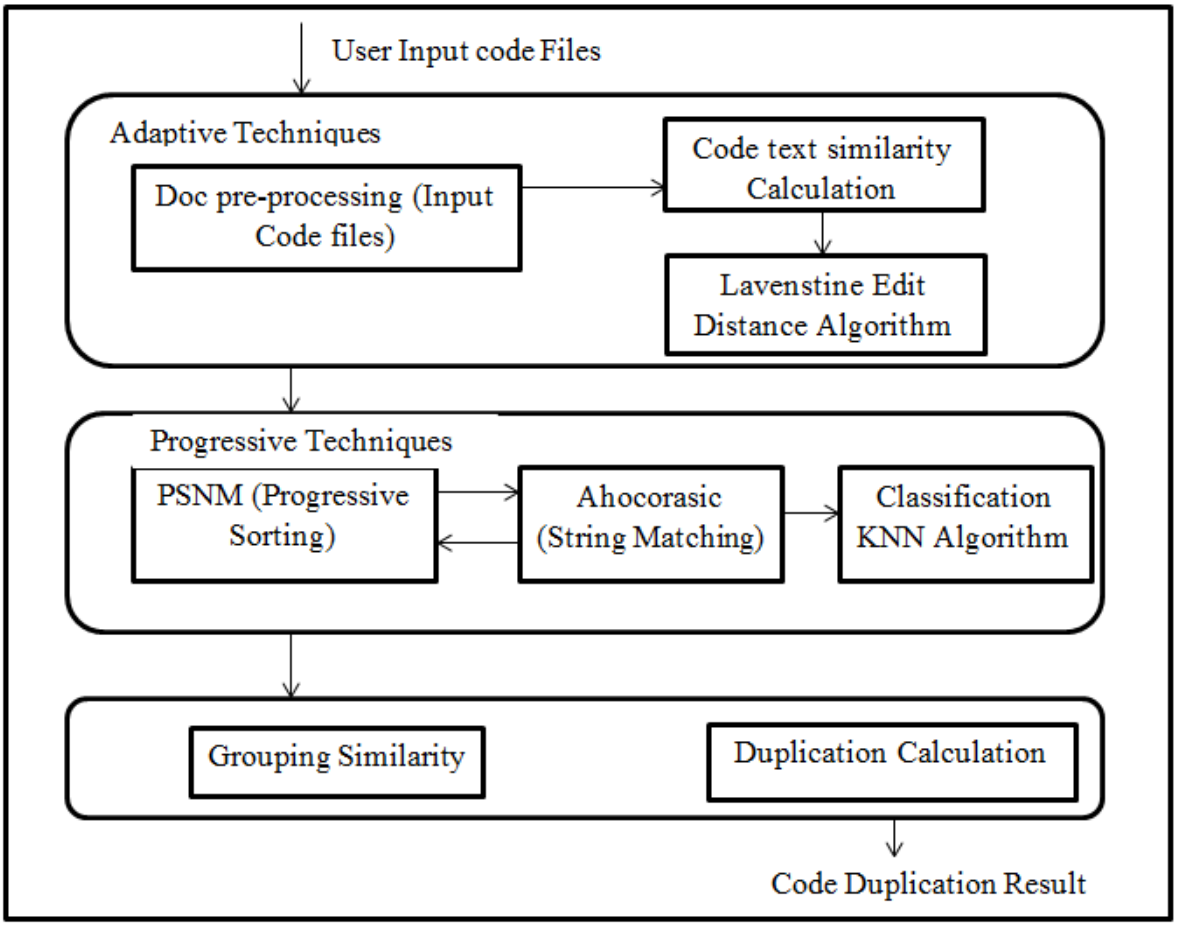

Fig. 1. Proposed System

b) ***Content Similarity***-To check the similarity we are using a LDA technique[8] . Levenshteindistance(LD) is ameasure of the similarity between two strings, the sourcestring (s) and the target string (t). The distance is thenumber of deletions, insertions,or substitutions requiredto transform s into t. The greater the Levenshtein distance, the more different the strings are. In our case, thesource code is the input, and the target string is one ofthe entries in the dictionary. Intuitively ”c=a+b” can betransformed into ”c=a-b” by substituting + for -.

c) ***Progressive Sorting***-In these module we are usingPSNM algorithm[3].By using algorithm we make thepartitions of data after removing the stop words.

d) ***Semantics and Syntactic Matching*** -In this module weare using Aho-Corasickalgorithm[7]for multiple patternmatching tasks. The algorithm consists of two parts:The first part is the building of the tree fromkeywords/patterns you want to search , and thesecond part is searching the text for the keywordsusing the previously built tree (finite state machine,FSM).FSM is a deterministic model of

behaviour com-posed of a finite number of states and transitions between those states.

e) ***Grouping and Duplication Detection***-After gettingthe output from LDA,PSNM and Aho-Corasick thenwe grouping the data into duplicated data and non-duplicated data and calculate the percentage of duplications.

IV. CONCLUSION

Adaptive and Progressive technique is used to increasethe efficiency of duplicate detection for situations with re-stricter execution time. For these ,we present the algorithms LDA, PSNM and Aho-Corasick and from these we analyse thatthe our system gives the efficient solution for finding sourcecode duplication in less time.

REFERENCES

[1]. E. Whang, D. Marmaros, and H. Garcia-Molina, ”Pay-as-you-goentity resolution,” IEEE Trans. Knowl. Data Eng. , vol. 25, no. 5, pp.11111124, May 2012.
[2]. C. Xiao, W. Wang, X. Lin, and H. Shang, ”Top-k set similarityjoins,” inProc. IEEE Int. Conf. Data Eng., 2009, pp. 916927.
[3]. Thorsten Papenbrock, ArvidHeise, and Felix Naumann,”ProgressiveDuplicate Detection, IEEE TRANSACTIONS ON KNOWLEDGE ANDDATA ENGINEERING,MAY 2015.
[4]. U. Draisbach and F. Naumann, ”A generalization of blocking andwindowing algorithms for duplicate detection,” in Proc. Int. Conf. DataKnowl. Eng., 2011, pp. 1824.
[5]. L. Kolb, A. Thor, and E. Rahm, ”Parallel sorted neighborhood blockingwith MapReduce,” in Proc. Conf. Datenbanksysteme in Buro, Technikund Wissenschaft, 2011.
[6]. U. Draisbach, F. Naumann, S. Szott, and O. Wonneberg,”Adaptivewindows for duplicate detection,” in Proc. IEEE 28thInt. Conf. DataEng., 2012, pp. 10731083.
[7]. Tsern-Huei Lee Department of Communication Engineering NationalChiao Tung University,” Generalized Aho-Corasick Algorithm forSignature Based Anti-Virus Applications”, 2007 IEEE.
[8]. RishinHaldar and DebajyotiMukhopadhyay,”Levenshtein DistanceTech-nique in Dictionary Lookup Methods: An Improved approach”,2007 IEEE.