Plagiarism detection techniques

Abstract

Academic dishonesty is one of the critical measures to evaluate research papers, theses and students' assignments. Therefore, plagiarism detection is an area of concern for many researchers especially in the academic field. Other fields such as plagiarized news, magazine articles and web resources are also area of concern. In that regard, many detection techniques and tools have been developed to address the problem of plagiarism. Different types of texts require different techniques to detect plagiarism. Documents to be retrieved, searched and thence judged according to the existence of plagiarism can be classified into two types: programming source code documents and natural language documents. The first type of documents is programming source code. Several researches have been developed for source code plagiarism detection or so-called code clones detection (John et al. 1981; Sam 1981; Marguerite et al. 1988; Parker et al. 1989; Wise 1992; Edward 2001a, 2001b; Shauna 2001; Belkhouche et al. 2004; Kim and Choi 2005; Mike et al. 2005; Mozgovoy et al. 2005; Peter and Julian 2005; Seunghak and Iryoung 2005; Chao et al. 2006; Christian and Tahaghoghi 2006; Samuel and Zelda 2006; Son et al. 2006; Jeong-Hoon et al. 2007; Lingxiao et al. 2007). This type of documents has specific structure which is language dependent. The word "language" here refers to one of the programming languages such as FORTRAN, PASCAL, C, JAVA and many more. Thus, the detection algorithm is based on what programming language is used. Most of the early techniques were used for one programming language. For instance, John et al. (1981) developed plagiarism detection system for FORTRAN source code, Sam (1981) developed a tool that detect plagiarism in PASCAL programs and some other systems that can be found in the literature. In addition, there exist other techniques used to detect code clones in two or more programming languages. For example, Whale (1990) developed a system called Plague that works with Pascal and Prolog source code. Xin et al (2004) developed SID system (Shared Information Distance) which supports Java and C++ source code.

Early code clone detection techniques focus on keeping track of metrics such as number of lines, variables, statements, subprograms, call to subprograms and other parameters. However, current research makes a quantum leap and uses the structure or style of the source code. Thus, such technique is called stylometric (i.e. based on the style or structure) since some research has also been involved to use this technique in natural language plagiarism detection. The latest trends for code clone detection use artificial neural networks (Steve et al., 2007) in which neural networks were trained based on some common features of the submitted documents. The network input uses number of metrics as input unites. The network output with low error rate can measure how relevance two documents are. In brief, code clones detection techniques aim to locate plagiarized code in one or more programming language(s) and rely on either metrics or style/structure of the code. The second type of documents is natural language documents written in English, Arabic or any other languages. Detecting plagiarism in this type of documents is much more difficult than the first type because natural languages are not easy to be modeled. In contrast to code clone detection techniques, neither metrics nor structures can be maintained easily in natural language documents. Although the research of detecting plagiarism started more than a decade after the first type (1981 for code clones vs. 1997 for natural language documents), many applicable techniques and useful tools have been developed for plagiarism detection in natural language documents (Antonio et al. 1997; Culwin et al. 2001; Zaslavsky et al. 2001; Monostori et al. 2002; Bao et al. 2003; Bao et al. 2004; Daniel and Mike 2004; Weir et al. 2004; Xin et al. 2004; Ye et al. 2004; Heon et al. 2005; Hui and Jamie 2005; Stefan and Stuart 2005; Yerra and Ng 2005; Bao et al. 2006a; Bao et al. 2006b; Byung-Ryul et al. 2006; Eissen and Stein 2006; Hui and Jamie 2006; Kang et al. 2006; Koberstein and Ng 2006; Manuel et al. 2006; Sebastian and Thomas 2006; Sorokina et al. 2006; Benno et al. 2007; Liu et al. 2007; Meyer zu Eissen et al. 2007; Rehurek 2007; Romans et al. 2007; Steve et al. 2007). The following sections discuss different representations of natural language documents for use in plagiarism detection.