

Pengembangan sistem pendeteksi plagiarisme berbasis algoritma Latent Semantic Analysis (LSA) dengan implementasi paralel computing pada infrastruktur komputasi awan = Development of Latent Semantic Analysis (LSA) algorithm based plagiarism detection system using paralel computing implementation on cloud computing infrastrukture

Nanda Girindratama, author

Deskripsi Lengkap: <https://lib.ui.ac.id/detail?id=20489249&lokasi=lokal>

Abstrak

Pada penelitian ini, dikembangkan HPC yang menerapkan multicore processing pada program Sistem Pendeteksi Plagiarisme dengan memanfaatkan infrastruktur komputasi awan berbasis OpenStack. Sistem Pendeteksi Plagiarisme merupakan program yang dikembangkan untuk mendeteksi tingkat plagiarisme dari suatu karya ilmiah. Algoritma program yang digunakan untuk penelitian kali ini adalah latent semantic analysis (LSA). Implementasi HPC dilakukan dengan bantuan library OpenMP yang didesain untuk bahasa pemrograman C. Diterapkan dua jenis paralelisme pada program, yaitu paralelisme fungsi dan paralelisme data. Setelah dilakukan pengujian, didapati hasil bahwa kedua metode paralelisme ini mempercepat eksekusi program. Paralelisme fungsi mempercepat waktu eksekusi hingga sebesar 1,03 kali waktu eksekusi serial dan paralelisme data mempercepat waktu eksekusi hingga 1,34 kali waktu eksekusi serial.

In this research, HPC with multicore processing is developed on Plagiarism Detection System using OpenStack based cloud computing infrastructure. Plagiarism Detection System is a software developed to detect plagiarism level of a scientific papers. The algorithm used in this program is latent semantic analysis (LSA). HPC implementation is done using OpenMP library which is designed to be used in C programming language. There are two types of parallelism in this program, which are function parallelism and data parallelism, both accelerate the execution time. Function parallelism accelerates program by up to 1,03 times of serial execution while data parallelism decreases the execution time by up to 1,34 times serial execution time.