

Sistem penilaian esai otomatis (SIMPLE-O) untuk ujian bahasa Jepang dengan bidirectional LSTM dan Manhattan distance = Automatic essay grading system (SIMPLE-O) for Japanese language exams using bidirectional LSTM and Manhattan distance

Nurjannah Cintya Adiningsih, author

Deskripsi Lengkap: <https://lib.ui.ac.id/detail?id=20514383&lokasi=lokal>

Abstrak

Skripsi ini membahas tentang Sistem penilaian esai Otomatis (SIMPLE-O) untuk ujian Bahasa Jepang dengan Bidirectional LSTM dan Manhattan Distance. Dalam penggunaan Algoritma RNN menggunakan arsitektur Bidirectional LSTM. SIMPLE-O merupakan sistem yang sedang dikembangkan oleh Departemen Teknik Elektro UI yang digunakan untuk menilai esai secara otomatis. Sistem berjalan menggunakan model Bidirectional LSTM, diukur dengan Manhattan Distance serta terdapat metric evaluasi yang terdiri dari Accuracy, Recall, Precision, F1-Measure. Dalam pengolahan sistem dilakukan secara otomatis menggunakan tensorflow. Pengujian yang dilakukan pada sistem yang dibangun terdapat 3 pengujian yaitu : pengujian terhadap epoch, optimizer dan word2vec. Untuk epoch dilakukan terhadap 3 epoch yaitu 20, 5 dan 10. Dari masing – masing epoch dijalankan sebanyak 5 kali. Hasil tertinggi yang didapatkan pada epoch ada pada epoch 20 yaitu 99.02%, untuk hasil pengujian optimizer menggunakan SGD atau stochastic gradient descent dan word2vec sebesar 500.

.....This thesis discusses the Automatic essay scoring system (SIMPLE-O) for Japanese language exams with Bidirectional LSTM and Manhattan Distance. In the use of the RNN Algorithm, the Bidirectional LSTM architecture is used. SIMPLE-O is a system being developed by the Department of Electrical Engineering UI which is used to automatically assess essays. The system runs using the Bi-LSTM model, measured by Manhattan Distance and there is an evaluation metric consisting of Accuracy, Recall, Precision, F1-Measure. In the system processing is done automatically using tensorflow. Tests carried out on the system built have 3 tests, namely: testing the epoch, optimizer and word2vec. For epoch, it is done for 3 epochs, namely 20, 5 and 10. From each epoch, it is run 5 times. The highest result obtained on epoch is at epoch 20, which is 99.02%, for the optimizer test results using SGD or stochastic gradient descent and word2vec of 500.