Abstrak

Sebagian besar metode ekstraksi data pada halaman web menggunakan *wrapper induction* dan *automatic data extraction*. Metode *automatic data extraction* muncul karena metode sebelumnya dianggap terlalu rumit. Dalam proses ekstraksi data, *automatic data extraction* membentuk *pattern* yang akan dicocokan dengan tag HTML pada halaman web.

Pada Tugas Akhir ini mengimplementasikan metode *automatic data extraction* dengan menggunakan algoritma yang disebut IDE (*Instance-based Data Extraction*). Teknik ini melibatkan user dalam pembentukan *pattern* dengan memberikan label pada halaman web. Pada proses *instance-based data extraction* ini ada tiga langkah yang utama yaitu, *page labeling*, *similarity measure* dan *data extraction*.

Ketepatan dalam membentuk *pattern* ekstraksi dapat dilakukan dengan cara mengisi nilai *range node* sebanyak jumlah *node* yang terdapat dalam satu template dari *target item.*

Performansi algoritma IDE dipengaruhi oleh nilai *range node* yang diberikan. . Jika *node* yang diambil semakin mendekati *pattern target item* maka performansi akan semakin baik. Selain itu jenis website yang diekstrak juga ikut mempengaruhi performansi. Website yang memiliki *pattern target item* (struktur HTML dari data yang akan diekstrak) sederhana akan lebih mudah untuk diekstrak.

Ketika data yang akan diekstrak tidak memiliki *pattern* yang unik maka algoritma IDE akan kesulitan untuk mengekstrak data yang sesuai dengan keinginan user. Akibatnya di dalam hasil ekstraksi masih terdapat data yang tidak relevan.

Tahap analisis dan pengujian dengan parameter pengujian berupa *recall ratio* dan *precision ratio* memberikan hasil bahwa algoritma IDE yang dibangun terbukti bisa mendapatkan informasi sesuai dengan keinginan user meskipun ada beberapa *noise*.

Kata kunci: *Automatic Data Extraction, pattern, pattern target item, page labeling, similarity measure, data extraction.*