Aplikasi Sistem Seleksi Pelamar Kerja dengan menggunakan Metode *Random Forest*

Daniel Gentha Ivan Desantha[1], Kemas Muslim Lhaksmana[2], Donni Richasdy[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]rexsantha@students.telkomuniversity.ac.id, [2]kemasmuslim@telkomuniversity.ac.id,
[3]donnir@telkomuniversity.ac.id

Abstrak

Indonesia merupakan negara berkembang yang memiliki jumlah penduduk yang demikian banyak dan masuk dalam 5 besar populasi penduduk di Dunia. Jumlah penduduk di Indonesia juga meningkat setiap tahunnya, dan seiring dengan bertambahnya jumlah penduduk, tenaga kerja juga mengalami peningkatan setiap tahunnya. Perusahaan yang sedang membuka lowongan pekerjaan pasti akan mendapatkan lebih banyak lamaran kerja setiap tahunnya, hal ini mengakibatkan semakin lambatnya proses seleksi pelamar kerja ini. Salah satu cara yang efektif untuk menyeleksi pelamar, yaitu dengan menyeleksi data hasil wawancara pelamar kerja yang telah diberikan skor oleh ahli. Skor ini terbagi penjadi 9 macam penilaian, yaitu *action*, *enthusiams*, *focus*, *imagine*, *integrity*, *smart*, *solid*, *speed* dan *totality*. Penelitian ini dilakukan untuk mendapatkan model penilaian dengan menggunakan klasifikasi teks yang dapat membantu sebuah perusahaan dalam menyeleksi data hasil wawancara pelamar kerja dengan otomatis, waktu yang lebih singkat, mengurangi biaya, dan penilaian yang objektif. Dalam penelitian ini digunakan *word embeddings* untuk merubah kata menjadi vektor. Metode klasifikasi yang dipakai dalam penelitian ini adalah *Random Forest*. Hasil dari penelitian pada jurnal ini menunjukuan bahwa algoritma klasifikasi *Random Forest* dapat dipakai untuk klasifikasi teks wawancara dengan *hyperparameter tuning*. Performa algoritma *Random Forest* pada klasifikasi teks wawancara terlihat cukup bagus dengan akurasi rata-rata 71% dari seluruh 9 poin penilaian yang ada. Dengan akurasi rata-rata 71%, maka bisa disimpulkan bahwa algoritma *Random Forest* memiliki akurasi yang cukup baik, dan dapat digunakan sebagai klasifikasi teks wawancara.

Kata kunci : klasifikasi teks, *random forest*, wawancara, seleksi pelamar kerja

Abstract

Indonesia is a developing country which has a large population and included in the top 5 population in the world. The population in Indonesia also increased every year, and along with the increase in population, the workforce has also increased every year. Companies that are currently opening job vacancies will definitely get more job applications each year, this has resulted in the slower process of selecting job applicants. One of the effective ways to select applicants is by selecting the data from job applicant interviews that have been scored by experts. This score is divided into 9 types of assessment, namely action, enthusiams, focus, imagine, integrity, smart, solid, speed and totality. This research was conducted to obtain an assessment model using text classification that can assist a company in selecting job applicant interview data automatically, with shorter time, reducing costs, and an objective assessment. In this study, word embeddings were used to convert words into vectors. The classification method used in this study is the Random Forest. The results of research in this journal show that the Random Forest classification algorithm can be used to classify interview texts with some hyperparameter tuning. The performance of the Random Forest algorithm in the interview text classification looks quite good with an average accuracy of 71% of all 9 assessment points. With an average accuracy of 71%, it can be concluded that the Random Forest algorithm has a fairly good accuracy, and can be used as a classification of interview texts.

Keywords: text classification, *random forest*, interviews, selection of job applicants

1. Pendahuluan

1.1. Latar Belakang

Indonesia merupakan negara yang memiliki jumlah penduduk yang demikian banyak, dan jumlah tenaga kerja terus mengalami peningkatan setiap tahunnya, dari berbagai jenis tingkat pendidikan. Banyaknya calon pelamar juga menjadikan salah satu penyebab lambatnya hasil pengumuman hasil wawancara. Salah satu cara mengatasi masalah tersebut adalah adanya suatu sistem yang dapat memberikan rekomendasi untuk pengambilan keputusan secara cepat dan tepat.

Proses pencarian pelamar kerja ini diawali dari perusahaan yang memberitahukan jika perusahaannya membuka lowongan pekerjaan bagi pegawai baru. Lalu pelamar yang tertarik akan mengirim berkas yang dibutuhkan perusahaan seperti curriculum vitae(CV), Surat Lamaran, dan portofolio ke perusahaan yang dituju. Ketika pelamar berhasil lolos dan terpilih, maka pelamar akan melanjutkan ke tahap seleksi wawancara. Pelamar yang diterima sebagai pegawai baru akan diberitahukan surat yang menandakan bahwa sang pelamar sudah di terima di perusahaan tersebut.

Banyak perusahaan yang menerima pelamar kerja atau kandidat yang tidak sesuai, hal itu dapat disebabkan oleh adanya ketidaksesuaian, seperti penilaian yang subjektif ketika melakukan penyeleksian, perbedaan standar penilaian pewawancara, waktu dan biaya melakukan wawancara yang lama dan mahal. Dan saat melakukan seleksi juga dibutuhkannya ketelitian dan memakan waktu.

Penelitian ini berfokus pada tahap wawancara. Data hasil wawancara akan dilabeli secara manual oleh tim ahli. Pelabelan dilakukan dengan memberikan tanda kelas atau label terhadap suatu teks [1]. Setelah data terlabeli, maka dilakukan proses klasifikasi untuk mengetahui suatu data (content) masuk kedalam kelas mana. Penggunaan *machine learning* banyak digunakan untuk mengklasifikasikan suatu teks[2][3][4].

Pada penelitian ini, *dataset* diperoleh dari hasil wawancara pada suatu perusahaan. Data tersebut berjumlah sebanyak 55 data jawaban wawancara pelamar kerja yang terdiri dari *content* dan *core values*. Pada kasus penelitian ini dilakukan analisis hasil kinerja metode klasifikasi menggunakan *Random Forest* untuk mengetahui keakuratan akurasi dalam menangani klasifikasi untuk seleksi pelamar kerja.

1.2. Topik dan Bahasannya

Permasalahan yang dibahas dalam tugas akhir ini adalah bagaimana merancang model klasifikasi seleksi pelamar kerja dengan data hasil wawancara menggunakan metode *Random Forest*. Bagaimana performa, kinerja dan akurasi dari klasifikasi *Random Forest*.

Batasan masalah dalam penelitian tugas akhir ini adalah *dataset* yang digunakan dalam penelitian merupakan data teks wawancara pelamar kerja yang berjumlah 55 content dengan 9 (sembilan) core values . *Dataset* disimpan dalam bentuk file *Microsoft Excel Comma Separated Values* (.csv). *Dataset* yang digunakan pada penelitian ini sangat sedikit dengan jumlah 55 data. Label kelas yang digunakan dalam klasifikasi terbagi menjadi dua kelas, yaitu tidak memuaskan dan memuaskan. Pada kolom *core values*, label kelas yang memuaskan ditandai dengan angka 2 (dua), sedangkan label kelas yang tidak memuaskan ditandai dengan angka 1 (satu).

1.3. Tujuan

Tujuan dari penulisan tugas akhir ini untuk melakukan klasifikasi data teks hasil wawancara pelamar kerja dengan menggunakan metode *Random Forest*, membuat model *Random Forest* yang optimal serta menganalisis hasil kinerja dan akurasi sistem klasifikasi yang dibangun dengan menggunakan metode *Random Forest* dan membantu otomatisasi proses rekrutmen pelamar kerja.

2. Studi Terkait

2.1. Klasifikasi Teks

Klasifikasi teks merupakan suatu cara untuk mengkategorikan teks ke dalam kelas yang telah ditentukan sebelumnya berdasarkan konten dari teks [6]. Terdapat dua jenis klasifikasi teks yaitu *supervised* dan *unsupervised*. Klasifikasi *supervised* adalah proses klasifikasi teks dengan menggunakan metode *learning* pada data teks yang sudah memiliki kelas. Sementara klasifikasi *unsupervised* adalah metode klasifikasi teks yang tidak memakai label kelas.

2.2. *Word Embeddings*

Word embeddings (representasi kata terdistribusi) merupakan cara yang merepresentasikan kata-kata bahasa alami dengan cara mempertahankan kemiripan semantik dan sintaksis di antara kata-kata tersebut. Hal ini didapat melalui representasi kata-kata sebagai vektor berdimensi tinggi, yaitu hubungan spasial di antara *embeddings* merepresentasikan hubungan di antara kata-kata.

2.3. *Preprocessing*

Teks yang digunakan pada umumnya memiliki beberapa karakteristik diantaranya adalah memiliki dimensi yang tinggi, terdapat noise pada data, dan terdapat struktur teks yang tidak baik [7]. *Preprocessing* adalah cara yang digunakan dalam mempelajari suatu data teks dengan terlebih dahulu menentukan fitur-fitur yang mewakili setiap kata untuk setiap fitur yang ada pada dokumen. *Preprocessing* terdiri dari beberapa tahap seperti berikut:

- *Case folding*: Proses mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (biasanya huruf kecil atau *lowercase*) [8]
- *Tokenizing*: Proses mengubah kalimat menjadi kumpulan kata tunggal dan menjadikan sebuah kalimat menjadi lebih bermakna atau berada dengan cara memecah kalimat tersebut menjadi kata-kata atau frase-frase [8]
- *Stopword* : Proses menghilangkan kata-kata yang dianggap tidak penting melalui pengecekan kata-kata hasil parsing (*stoplist*).
- *Stemming*: Proses mengembalikan kata ke bentuk dasar (*root word*) dengan menghilangkan aditif yang ada [10]
- *Lemmazation*: Proses untuk mengubah kata berimbuhan menjadi bentuk aslinya [11]

2.4. *Random Forest*

Random Forest terdiri dari banyak *Decision Tree* yang banyaknya dimulai dari *Tree-1* sampai *Tree-n*, dimana *n* adalah jumlah seluruh *tree* yang ada pada *Random Forest* tersebut. *Instance* adalah data yang akan diklasifikasi, data ini selanjutnya akan diproses oleh semua *tree* yang ada pada dalam *forest*. Masing-masing *tree* akan menghasilkan hasil klasifikasi masing-masing. Hasil klasifikasi yang dihasilkan terbagi menjadi berbagai kelas sesuai yang telah ditentukan. Hasil voting terbanyak yang dihasilkan dari semua *tree* akan dijadikan *final-class*, *final-class* ini akan dijadikan sebagai hasil klasifikasi dari algoritma *Random Forest*

2.5. Parameter Pengukuran

Parameter pengukuran yang dipakai pada penelitian ini digunakan untuk mengevaluasi performansi dari model yang telah dibuat. Parameter pengukuran dilihat dari kemampuan model tersebut dalam melakukan prediksi. Pengukuran dilakukan dengan menggunakan confusion matrix. Confusion Matrix ini dibuat untuk memetakan kinerja algoritma dalam bentuk tabulasi. Matriks ini menunjukkan hubungan antara benar tidaknya sebuah data dikategorikan.

3. Sistem yang Dibangun

3.1. Persiapan Data

Sumber data yang digunakan dalam penelitian ini diambil dari hasil wawancara pelamar kerja yang dilakukan dengan berbagai tahap seperti Gambar 1 pada. Pertama, pelamar kerja melakukan rangkaian proses dengan mengirim lamaran yang akan diseleksi oleh divisi SDM, jika dinyatakan memenuhi syarat akan melakukan proses wawancara. Dari proses wawancara ini didapatkan hasil transkrip wawancara. Data teks yang telah dikumpulkan akan diklasifikasi dengan cara dikelompokkan, dilabeli, berdasarkan 9 (Sembilan) *core values* dan dikonversi dari skor skala 1 (satu) sampai 100 (seratus) mejadi label memuaskan dan tidak memuaskan secara manual oleh tim ahli sesuai dengan kategorinya *core values*-nya masing-masing [12] [13] [14].

Kumpulan data teks wawancara dari pengguna juga dibuat menjadi satu dokumen / satu *string* panjang. Dimana data ini akan diprediksi atau dinilai Gambar 2 pada. Proses Pengumpulan *Dataset* berdasarkan 9 (sembilan) *core values* yaitu: *action*, *enthusiams*, *focus*, *imagine*, *integrity*, *smart*, *solid*, *speed* dan *totality*. Dan pelabelannya dilakukan dengan memberikan poin pada setiap core values yang telah dikumpulkan yakni tidak memuaskan (1) dan memuaskan (2).

Total data yang didapat sebanyak 55 data dari proses wawancara. Kumpulan data secara acak kemudian dibagi menjadi *data training* dan *data testing* dengan perbandingan rasio jumlah set data 80:20. Nilai atau prediksi dari 9 (sembilan) *core values* tersebut akan digunakan sebagai acuan keputusan untuk diterima (direkomendasikan) atau tidak diterima (tidak direkomendasikan) seorang pelamar kerja.

3.2. Struktur Model

Sebelum dataset dilatih dengan metode *Random Forest*, dataset harus diolah. Dataset yang sudah dibagi menjadi data latih dan data testing akan memasuki *preprocessing*. *Preprocessing* bertujuan untuk mengubah data agar mempermudah pada saat diolah, terutama dalam menghilangkan *noise*, memperjelas fitur data, merubah data asli agar diperoleh data yang sesuai dengan kebutuhan. Langkah selanjutnya adalah tahap ekstraksi fitur (*feature extraction*) yang digunakan untuk mengambil ciri unik dari pada data masukan dan mengubahnya kedalam bentuk vektor. Di penelitian ini menggunakan teknik lanjutan dari *word2vec*, yaitu *FastText* yang digunakan sebagai proses konversi teks menjadi bentuk vektor.

3.3. Proses *Feature Extraction (Word Embeddings)*

Teknik *Word Embeddings* yang digunakan pada penelitian kali ini adalah *Fast Text*. *Fast Text* akan mengubah suatu kata menjadi vektor dengan algoritma *Continuous Bag of Word* (CBOW). Dengan algoritma ini, hasil vektor klasifikasi *Fast Text* akan mempunyai makna yang dapat dimengerti oleh komputer, seperti contohnya jika komputer diminta mencari sinonim dari kata “aku”, maka akan dihasilkan “saya”, “akupun”, kata tersebut bisa dihasilkan karena kata-kata tersebut berada di kelas yang berdekatan. Kata-kata tersebut digolongkan menjadi kata yang kelasnya berdekatan karena memiliki arti yang hampir sama dan bentuk kata yang hampir sama.

Proses mengubah kata menjadi vektor akan dilakukan pada semua kata yang ada dalam kolom wawancara, ketika kumpulan vektor kata telah ditemukan, maka vektor kata tersebut akan dicari rata-ratanya. Rata-rata dari hasil penjumlahan vektor kata tersebut akan menjadi identitas sebuah vektor teks wawancara. Vektor teks wawancara ini yang nantinya akan menjadi *input* untuk klasifikasi *Random Forest*.

3.4. Pelatihan Model Klasifikasi

Pada penelitian ini digunakannya pelatihan *supervised learning* dengan algoritma pengklasifikasian *Random Forest*. Tahapan pelatihan model algoritma menggunakan data latih yang sudah dibagi dari *dataset*. Proses pengambilan data latih ke model dilakukan secara acak, proses *training* data latih ke model dilakukan secara berulang hingga menghasilkan model yang mempunyai akurasi maksimal dan optimasi parameter sebaik mungkin.

3.5. Metode Validasi

Metode validasi disini melakukan evaluasi terhadap model klasifikasi dengan melihat keakuratan metode prediksi biner melalui *confussion matrix* dan tabel akurasi dan presisi untuk model dengan beberapa parameter statistik seperti sensitivitas(*recall*), *precision*, dan akurasi yang nilainya akan semakin bagus bila mendekati angka 1. Sensitivitas(*recall*) adalah rasio prediksi TP dibandingkan dengan keseluruhan data yang benar positif. *Precision* adalah rasio prediksi TP dibandingkan dengan keseluruhan hasil prediksi positif. Akurasi adalah rasio prediksi benar (TP dan TN) dengan keseluruhan data. Parameter yang didapatkan dari hasil *confussion matrix* akan digunakan untuk mengevaluasi model yang sudah dibuat. Parameter yang digunakan adalah : *true positive* (TP), *false positive* (FP), *true negative* (TN), *false negative* (FN)

$$\text{Sensitivitas} = \text{SE} = \frac{TP}{TP+FN}$$

$$\text{Spesifisitas} = \text{SP} = \frac{TN}{TN+FP}$$

$$\text{Presisi} = \frac{TP}{TP+FP}$$

$$\text{MCC} = \frac{(\text{TP}\times \text{TN}) - (\text{FP}\times \text{FN})}{\sqrt{(\text{TP}+\text{FN})(\text{TP}+\text{FP})(\text{TN}+\text{FN})(\text{TN}+\text{FP})}}$$

3.6. Akurasi Prediksi

Untuk dapat menganalisis dan menilai akurasi dari model klasifikasi dilakukukan dengan menggunakan sejumlah *data test*. Pada tahapan ini *data test* secara individual masuk ke dalam model klasifikasi, lalu mengeluarkan hasil prediksinya. Label hasil prediksi menggunakan pemodelan *Random Forest* dibandingkan dengan label yang ada pada *data test* sebenarnya, dengan membandingkan hasil dan data aslinya, maka akan didapatkan performa akurasi dari model klasifikasi yang dibuat.

$$\text{Akurasi} = \text{Q} = \frac{TP+TN}{TP+FP+FN+T}$$

4. Hasil dan Analisis

Pada tahap ini ditampilkan hasil eksperimen model klasifikasi *dataset* wawancara menggunakan algoritma *Random Forest*. Diketahui bahwa jumlah *dataset* teks wawancara sebanyak 55 konten, 9 (sembilan) *core values* dan 2 (dua) jenis kelas. Tahap selanjutnya dilakukan pemisahan *data training* dan *testing* menggunakan pembagian data menjadi *data training* 80% dan *data testing* 20%. *Data training* untuk dijadikan model dan *data testing* untuk mendapatkan akurasi dengan sebaran data.

4.1. Pengolahan *Dataset*

Dataset asli terdiri dari konten yang berisi hasil wawancara pelamar kerja, yang dinilai berdasarkan 9 *core values* yaitu: *action*, *enthusiams*, *focus*, *imagine*, *integrity*, *smart*, *solid*, *speed* dan *totality*. Label yang diberikan kepada 9 *core values* tersebut adalah 1 (tidak memuaskan) dan 2 (memuaskan). *Dataset* asli digambarkan pada Tabel 2. Pada proses *training*, *dataset* diubah melalui proses *one hot encoding* dimana akan menghasilkan sebuah *array* 2 dimensi yang bernilai 1 dan 0 yang ditunjukan pada tabel 3. Perubahan ini dilakukan untuk meningkatkan prediksi dan perbedaan antara data.

4.2. *Feature Extaction*

Pada penelitian ini, agar data teks dapat diklasifikasi, maka data yang berbentuk teks perlu diubah bentuknya menjadi vektor. Proses merubah teks menjadi vektor adalah *word embbedings*. Metode *word embeddings* yang digunakan pada penelitian ini adalah *Fast Text*. *Fast Text* merupakan kelanjutan dari *word2vec* yang merupakan suatu metode klasifikasi text *Unsupervised Learning,* yang menggunakan *Neural Network*. *Fast Text* dapat merubah suatu kata menjadi vektor yang mempunyai arti, serta melakukan klasifikasi kata tersebut. Setelah potongan-potongan kata dalam suatu text dirubah menjadi vektor, maka kumpulan vektor kata telah didapatkan, kumpulan vektor kata tersebut diperlukan untuk mencari nilai vektor dari teks pewawancara. Nilai vektor dari teks pewawancara dihitung dari rata-rata nilai vektor kata yang ada dalam teks tersebut. Selanjutnya nilai vektor teks akan dipakai dalam metode klasifikasi yang digunakan dalam penelitian ini.

Pada penelitian ini, saya menggunakan *pre-trained* model yang disediakan oleh *Fast Text*. Model ini adalah model *Neural Network* yang telah dibuat oleh *Fast Text* sendiri menggunakan data dump artikel *Wikipedia* dan *Common Crawl*. Parameter yang digunakan pada pelatihan model klasifikasi kata ini adalah,

- Algoritma : CBOW (*Continuous Bag of Word*)
- N-Gram : 5
- *Window Size* : 5

4.3. Model *Random Forest*

Pada penelitian ini, untuk meningkatkat tinkat akurasi prediksi dari *Random Forest*, maka ditetapkan *Random Forest Hyperparameter*. *Hyperparameter* ditetapkan untuk mendapatkan nilai-nilai *Hyperparameter* yang optimal dan meningkatkan kinerja model. Proses yang dilakukan seperti penyesuaian *n_estimator* (Jumlah *Decision Tree*) yang digunakan adalah 2 sampai 64, max_depth diantara 1 sampai 10,. Hasil dari penyesuaian *Hyperparameter* disajikan pada tabel 4. Hasil penelitian menunjukan bahwa *n_estimator, max_depth* ditemukan nilai yang paling optimal, yaitu 200, 10, 2, 1.

Nilai parameter tersebut didapatkan dari nilai rata-rata akurasi terbaik dari model *Random Forest* yang dibuat. Proses pencarian nilai rata-rata akurasi ini dilakukan dengan lima kali percobaan untuk setiap angka parameter yang ada.

Dalam gambar 5 pada lampiran, bisa dilihat bahwa angka *max_depth* yang menghasilkan akurasi terbaik ada di angka 10. Performa *max_depth* juga terlihat stabil ketika *max_depth* > 10, maka untuk angka *max_depth* yang optimal adalah angka 10.

Dalam gambar 6 pada lampiran, bisa dilihat bahwa angka *n_estimator* yang menghasilkan akurasi terbaik ada di angka 64. Performa *n_estimator* juga terlihat stabil ketika *n_estimator* > 64, maka untuk angka *n_estimator* yang optimal adalah angka 64.

4.4. Analisis dan Hasil Uji

Dengan menggunakan parameter yang sudah ditentukan sebelumnya, didapatkan akurasi untuk hasil pelatihan dan pengujian pada tabel. Ditunjukan pada Tabel 5 bahwa hasil teks klasifikasi oleh metode *Random Forest* memberikan akurasi yang cukup baik dalam bentuk akurasi untuk setiap data latih (*training*) dan data uji (*testing*)

Dari hasil percobaan yang dilakukan oleh model dan dibandingkan dengan data test, didapatkan akurasi rata-rata untuk setiap *core value* adalah 71%.

4.5. Hasil Validasi

Hasil klasifikasi menggunakan *Random Forest* yang telah diperoleh akan disajikan dalam bentuk *confussion matrix*. Berdasarkan hasil validasi kinerja model yang disajikan pada Tabel 6, merupakan hasil dari *confussion matrix* 9 (sembilan) *core values* menggunakan metode *Random Forest*. Dalam kasus data pelatihan, ditemukan bahwa semua model dapat secara akurat memprediksi nilai target, yang ditunjukkan oleh nilai SE, SP, Q, dan MMC yang tinggi. Validasi untuk data tes ditemukan menghasilkan nilai-nilai parameter validasi yang lebih rendah dari validasi set pelatihan. Menurut hasil, menunjukkan bahwa hasil *data testing* yang diprediksi dari sembilan *core values*, menunjukkan *core values smart* lebih akurat daripada nilai *core values* lainnya. Ini ditunjukkan oleh nilai parameter validasi *core values smart* yang lebih tinggi dari *core values* lainnya. Nilai SE, SP, Q, dan MCC yang diperoleh dari prediksi pada *core values smart* masing-masing adalah 0,90, 0,50, 0,73, dan 0,40. Dan juga, pada nilai MCC yang mewakili kualitas keseluruhan klasifikasi biner, pada *core values smart* ditemukan sebagai yang tertinggi.

5. Kesimpulan

Berdasarkan hasil analisis dan uji yang dilakukan, menunjukan bahwa metode *Random Forest* berhasil menemukan label prediksi untuk Sistem Seleksi Pelamar kerja. Metode *Random Forest* dapat menentukan layak atau tidak pelamar yang diwawancara. Model ini dikembangkan dengan menggunakan metode *Random Forest* untuk memprediksi kelas untuk 9 (sembilan) *core values* dalam data teks hasil wawancara. Proses penilaian dapat dilakukan secara otomatis dan sesuai kriteria dengan Model tersebut. Meskipun penelitian ini menggunakan kumpulan data yang relatif sedikit dan dimensi yang cukup pendek, Model *Random Forest* telah berhasil melakukan serangkaian tahapan yang diperlukan dan dapat memprediksi data teks wawancara pelamar kerja, meskipun ada prosedur penyesuaian parameter *n_estimator dan max_depth* yang perlu dilakukan untuk meningkatkan kinerja model *Random Forest*. Model *Random Forest* memberikan hasil yang cukup baik dengan nilai akurasi rata-rata 71%. Perbaikan lebih lanjut dapat dilakukan dengan menggunakan lebih banyak dataset untuk meningkatkan akurasi.

Daftar Pusaka

[1] T. Saito and O. Uchida, "Automatic Labeling for News Article Classification Based on Paragraph Vector," 2017 9th Int. Conf. Inf. Technol. Electr. Eng., 2017.

[2] C.D. Manning, P. Raghavan, H. Schutze. Introduction to Information Retrieval. Cambridge UP, 2008

[3] C. Sumner, A Byers, R. Boochever, and G. J Park, "Predicting Dark Triad Personality Traits from Twitter usage and a linguistic analysis of Tweets," in Proc. of 11th IEEE International Conference on Machine Learning and Applications, 2012, pp. 386-393.

[4] A. C. E. S. Lima and L. N. deCastro, "Multi-Label Semi-Supervised Classification Applied to Personality Prediction in Tweets," in Proc. of BRICS Congress on Computational Intelligence 11th Brazilian Congress on Computational Intelligence, Recife, Brazil, 2013, pp. 195-203.

[5] Ana Carolina E.S. Lima, Leandro Nunes de Castro, "A multi-label, semi-supervised classification approach applied to personality Prediction in social media"in Affective Neural Networks and Cognitive Learning Systems for Big Data Analysis, vol.58, October 2014, pp. 122–130

[6] C. C. Aggarwal. A SURVEY OF TEXT CLASSIFICATION ALGORITHMS. Springer, 2012

[7] Adiwijaya, Igg. (2006). Texi Mining dan Knowledge Discovery. Komunitas Data mining Indonesia Softomputing Indonesia.

[8] G. Grefenstette. Syntactic Wordclass Tagging. Springer Netherlands, 1999.

[9] J. L. A. Rajaraman. Mining of Massive Datasets,. 2014.

[10] Fadillah Z Tala. A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia.

[11] H. S. Christopher D. Manning, Prabhakar Raghavan. an Introduction to Information Retrieval. Cambridge, 2009

[12] Mairesse, F. and Walker, M. A. and Mehl, M. R., and Moore, R, K. Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text. In Journal of Artificial intelligence Research, 30(1), pp: 457–500, 2007.

[13] Celli, F., Pianesi, F., Stillwell, D. S., and Kosinski, M. 2013. Workshop on Computational Personality Recognition (Shared Task). The Seventh International AAAI Conference on Weblogsand Social Media. Boston, MA, USA

[14] Rout, D.; Preot iuc-Pietro, D.; Kalina, B.; and Cohn, T. 2013.