

Perbandingan Performa Klasifikasi Terjemahan Al-Qur'an Menggunakan Metode Random Forest dan Long Short Term Memory

Dhea Putri Aftari, Nazruddin Safaat H^{*}, Surya Agustian, Yusra, Iis Afrianty

Fakultas Sains dan Teknologi, Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia

Email: ¹12050120341@students.uin-suska.ac.id, ^{2,*}nazruddin.safaat@uin-suska.ac.id, ³surya.agustian@uin-suska.ac.id

⁴yusra@uin-suska.ac.id ⁵iis.afrianty@uin-suska.ac.id

Email Penulis Korespondensi: nazruddin.safaat@uin-suska.ac.id

Submitted: 10/05/2024; Accepted: 18/05/2024; Published: 30/05/2024

Abstrak—Penelitian ini berfokus pada penggunaan Al-Qur'an sebagai sumber utama ajaran Islam dengan tujuan mempermudah pemahaman umat Islam terhadap isinya. Untuk mencapai tujuan ini, dilakukan pengelompokan ayat terjemahan Al-Qur'an melalui proses klasifikasi. Dua metode yang jarang digunakan untuk data terjemahan Al-Qur'an adalah *Random Forest* (RF) dan *Long Short Term Memory* (LSTM), karena keduanya mampu mengolah data besar dan kompleks. Data yang digunakan dalam penelitian ini adalah terjemahan Al-Qur'an yang telah diklasifikasikan ke dalam 15 topik oleh penelitian sebelumnya, namun penelitian ini hanya berfokus pada 6 topik. Tujuan penelitian ini adalah membandingkan kinerja RF dan LSTM dalam mengklasifikasikan terjemahan Al-Qur'an ke dalam 6 kategori berbeda. Hasil penelitian menunjukkan bahwa pada kategori dakwah, LSTM secara konsisten mengungguli RF, dengan nilai *F1-Score* sebesar 57,3% dan *accuracy* 96,8%, sedangkan RF memiliki *F1-Score* sebesar 49,4% dan *accuracy* 97,5%. Temuan ini menunjukkan bahwa LSTM memiliki kinerja yang lebih baik, terutama dengan *preprocessing* yang tepat, penyetelan parameter optimal, dan data yang seimbang. Penelitian ini memberikan wawasan penting dalam pengembangan model klasifikasi untuk teks terjemahan Al-Qur'an, serta menekankan pentingnya proses *preprocessing* dan penyetelan parameter yang tepat.

Kata Kunci: *Accuracy*; Klasifikasi; *Long Short Term Memory*; *Preprocessing*; Penyetelan Parameter; Terjemahan Al-Qur'an; *Random Forest*

Abstract—This study focuses on the use of the Qur'an as the primary source of Islamic teachings, aiming to facilitate Muslims' understanding of its content. To achieve this, the classification of translated Qur'anic verses was conducted. Two methods that are rarely used for Qur'anic translation data are Random Forest (RF) and Long Short Term Memory (LSTM) due to their ability to process large and complex data. The data used in this study are translations of the Qur'an that have been classified into 15 topics by previous research, but this study will only focus on 6 topics. The objective of this research is to compare the performance of RF and LSTM in classifying Qur'anic translations into 6 different categories. The results show that in the preaching category, LSTM consistently outperformed RF, with an F1-Score of 57.3% and an accuracy of 96.8%, whereas RF achieved an F1-Score of 49.4% and an accuracy of 97.5%. These findings indicate that LSTM has better performance, especially with proper preprocessing, optimal parameter tuning, and balanced data. This study provides important insights into the development of classification models for Qur'anic translation texts, highlighting the importance of proper preprocessing and parameter tuning.

Keywords: *Accuracy*; Classification; Quran Translation; Parameter Tuning; Preprocessing; Long Short Term Memory; Random Forest

1. PENDAHULUAN

Menurut Kemenag pada tahun 2023, proyeksi populasi dunia dan umat Muslim menunjukkan peningkatan signifikan dalam beberapa dekade mendatang. Diperkirakan populasi dunia akan tumbuh sebesar 32%, sementara umat Muslim diproyeksikan meningkat sekitar 70% hingga tahun 2060[1]. Islam, sebagai agama yang diperkenalkan oleh Nabi Muhammad SAW., memiliki sumber utama dari Al-Qur'an dan hadis. Al-Qur'an dianggap sebagai firman Allah yang diwahyukan kepada Nabi Muhammad SAW. melalui ruh al-amin (Jibril) dalam bahasa Arab. Kandungan Al-Qur'an dijamin kebenarannya dan dianggap sebagai panduan kerasulan Nabi, petunjuk ibadah, sumber hukum, dan landasan utama dalam kehidupan manusia [2]. Sebagai umat Islam, tentunya diwajibkan untuk memahami dan mengamalkan Al-Qur'an sebagai pedoman hidup. Untuk memaksimalkannya, upaya yang dapat dilakukan yaitu dengan dilakukannya proses mengkategorikan ayat-ayat Al-Qur'an berdasarkan tema atau topik, melalui proses pengelompokan yang memanfaatkan terjemahan untuk memberikan kemudahan bagi umat Islam dalam memahami isi Al-Qur'an. Seperti pada penelitian yang mengklasifikasikan ayat-ayat Al-Qur'an menggunakan *Doc2vec* [3].

Klasifikasi teks saat ini menjadi tren populer, seperti yang ditunjukkan dalam penelitian oleh Pane & Mubarak pada tahun 2018 yang mengklasifikasikan Al-Qur'an berdasarkan struktur, tema, dan gaya bahasa. Topik yang dibahas ayat-ayat Al-Qur'an dapat diklasifikasikan ke dalam 15 kelas, yaitu (1) Arkanul Islam, (2) Iman, (3) Al-Qur'an, (4) Ilmu dan Cabang-cabangnya, (5) Amal, (6) Dakwah, (7) Jihad, (8) Manusia dan Hubungan Kemasyarakatan, (9) Akhlak, (10) Peraturan yang Berhubungan dengan Harta, (11) Hal-hal yang Berkaitan dengan Hukum, (12) Negara dan Masyarakat, (13) Pertanian dan Perdagangan, (14) Sejarah dan Kisah-kisah, dan (15) Agama-agama[4]. Dalam klasifikasi Al-Qur'an, setiap ayat memiliki potensi untuk termasuk ke beberapa kelas, yang dikenal sebagai klasifikasi multi-label. Namun, kelemahan penelitian ini

adalah *accuracy* yang belum optimal karena ketidakseimbangan data antar kelas Al-Qur'an. Saran yang diajukan adalah melakukan penyeimbangan penggunaan data untuk meningkatkan hasil klasifikasi terjemahan ayat Al-Qur'an.

Random Forest sangat banyak digunakan dalam proses klasifikasi, seperti untuk memprediksi win ratio pemain PUBG (*Player Unknown Battle Ground*) berdasarkan data statistik yang dimana tahap awal melibatkan pengumpulan dan seleksi atribut relevan [5]. Klasifikasi dengan *Bayesian weighted Random Forest* juga digunakan untuk klasifikasi data yang berdimensi tinggi [6]. Disisi lain, penggunaan algoritma *Random Forest* untuk klasifikasi kebakaran hutan di Riau [7], Implementasi *Random Forest* juga dilakukan untuk menentukan penerima bantuan raskin [8], dan untuk klasifikasi kata sarkasme pada media social facebook [9]. Klasifikasi dengan *Random Forest* juga digunakan untuk dataset credit approval [10], dan pada kasus gagal jantung diklasifikasikan dengan menggunakan *Random Forest* dan Naïve Bayes [11].

Random Forest, sebagai algoritma Machine Learning, memiliki keunggulan dalam memproses data besar secara efisien dan telah menjadi populer dalam penelitian beberapa tahun terakhir. Sebagai contoh, Penelitian [12] melakukan penelitian menggabungkan metode SIFT dan *Random Forest* dalam klasifikasi motif Songket Palembang, mencapai *accuracy* rata-rata sebesar 92,98%. Meskipun demikian, kelemahan muncul karena tidak semua objek citra dapat diklasifikasikan sepenuhnya, disarankan untuk memperbanyak dataset agar klasifikasi menggunakan *Random Forest* berjalan lebih baik. Penelitian lain oleh Husin, tentang membandingkan algoritma *Random Forest*, Naive Bayes, dan BERT untuk klasifikasi multi-class pada artikel CNN. Meski mencapai *accuracy* sempurna pada tahap pelatihan, terdapat kekurangan dalam lamanya proses pencarian kombinasi parameter, dengan saran untuk menggunakan dataset seimbang dan perangkat dengan spesifikasi memadai guna meningkatkan efisiensi selama penelitian [13].

Selain algoritma Machine Learning, proses klasifikasi juga menggunakan pendekatan *Deep Learning*, khususnya dengan menggunakan *Long Short Term Memory* (LSTM). Keunggulan *Deep Learning* terletak pada kemampuannya untuk menganalisis hubungan nonlinier yang kompleks melalui representasi fitur hierarkis dan terdistribusi. LSTM dapat mengatasi ketergantungan jangka panjang dalam data deret waktu, yang seringkali menjadi hambatan bagi metode tradisional [14]. Penelitian [15] menunjukkan bahwa LSTM memiliki kinerja lebih baik daripada Convolutional Neural Network (CNN) dalam klasifikasi teks. Namun, masalah muncul pada waktu pelatihan yang lama ketika memproses teks dengan panjang yang signifikan. Untuk meningkatkan kinerja klasifikasi, disarankan untuk memaksimalkan penggunaan label, membagi data label, dan memperluas fitur node pada analisis sintaksis untuk meminimalkan masalah waktu pelatihan yang lama.

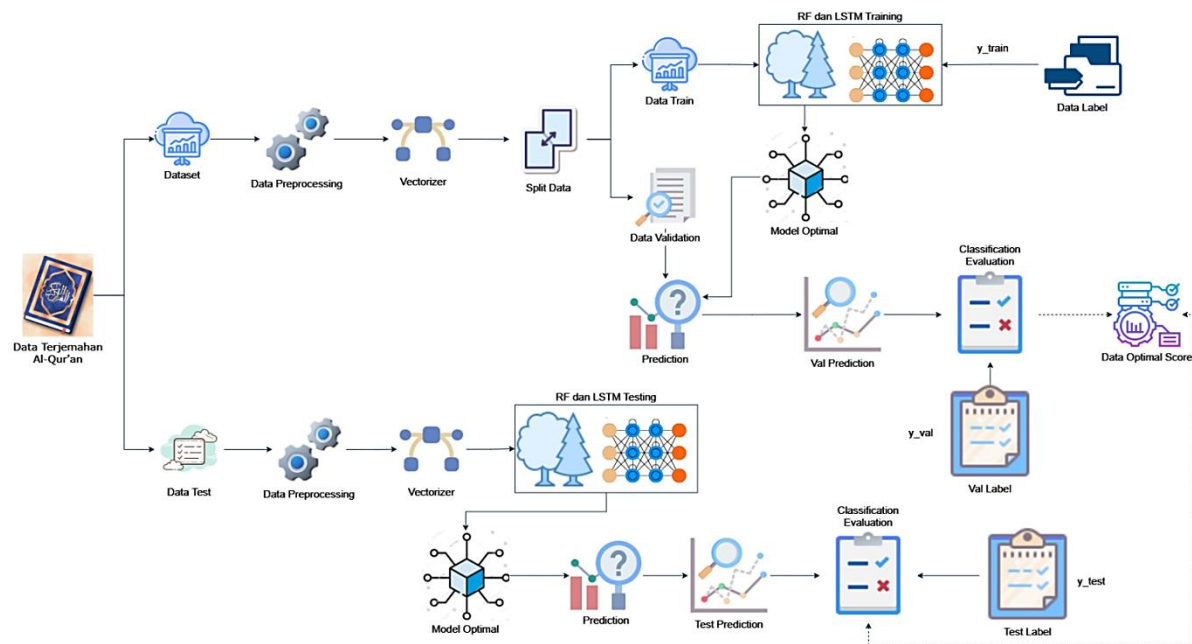
Dalam konteks klasifikasi, LSTM digunakan untuk mengatasi masalah pemrosesan teks yang panjang dan urutan data. LSTM memiliki memori sel yang membantu sistem untuk mempertahankan kondisinya, sehingga dapat mengatasi masalah "*exploding*" dan "*vanishing gradient*" yang sering terjadi pada jaringan saraf rekuren. Dengan demikian, LSTM memungkinkan pengolahan teks yang lebih panjang dan kompleks, sehingga cocok digunakan dalam analisis sentimen terhadap teks panjang seperti ulasan hotel [16]. Di sisi lain, LSTM digunakan pada penelitian klasifikasi Sentimen Vaksin Covid-19 pada twitter. Yang mana hasil dari penelitian tersebut dapat menghasilkan model menjadi kalimat baru [17]. Penelitian lain yang menggunakan metode *Long Short Term Memory* (LSTM) untuk memprediksi harga saham, menguji beberapa parameter seperti jumlah layer, epoch, dan time step untuk mendapatkan model prediksi yang optimal [18]. Penelitian yg serupa melakukan prediksi harga listrik berdasarkan model hibrida jaringan syaraf tiruan LSTM yang dioptimasi adam dan transformasi wavelet [14].

Dari isu yang telah dijelaskan sebelumnya diatas, Penulis berencana melakukan penelitian dengan memanfaatkan dataset dari penelitian [4] yang berjumlah 15 topik. Tetapi dalam penelitian ini hanya akan berfokus pada 6 topik terjemahan Al-Qur'an, yakni Dakwah, Jihad, Manusia dan Hubungan Kemasyarakatan, Akhlak, dan Peraturan yang Berhubungan dengan Harta, serta satu topik tambahan untuk data yang tidak termasuk ke dalam kelas manapun (*undefined*). Penelitian ini menerapkan metode *Random Forest* dan *Long Short Term Memory*, yang masih jarang digunakan dalam penelitian klasifikasi Al-Qur'an, yang bertujuan untuk meningkatkan *accuracy* dari hasil klasifikasi pada data terjemahan Al-Qur'an yang tidak seimbang. Perbandingan hasil *accuracy* antara metode *Random Forest* dalam Machine Learning dan LSTM dalam Deep Learning juga menjadi fokus penelitian ini.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Dalam penelitian ini, berbagai tahapan metodologi akan dijalankan secara sistematis untuk mencapai tujuan yang telah ditetapkan. Berikut gambar 1 adalah langkah-langkah dari proses metodologi penelitian yang akan dijalankan pada penelitian ini.



Gambar 1. Alur metode penelitian Random Forest dan Long Short Term Memory

Dari Gambar 1, dapat dijelaskan bahwa penelitian ini menggunakan data dari terjemahan Al-Qur'an yang kemudian diolah dengan proses *preprocessing*, *vectorizer*, dan *split data* pada data *training* lalu diuji kedalam algoritma *Machine Learning* dengan menggunakan *Random Forest* dan algoritma *Deep Learning* menggunakan *Long Short Term Memory*. Sedangkan untuk data *Testing* tidak dilakukan split karena data sudah di split dari awal. Dari proses tersebut menghasilkan hasil *optimal score* yang berisi nilai optimal dari masing-masing data.

2.2 Dataset

Penelitian ini memanfaatkan dataset Al-Qur'an yang mulia[19], sebagai sumber data terjemahan Al-Qur'an berbahasa Indonesia, kemudian data telah diberi label oleh Pane dan Mubarak pada penelitiannya yang berjumlah 15 label [4]. Adapun 15 topik yang terdapat pada data Al-Qur'an yaitu Arkanul Islam, Iman, Al-Qur'an, Ilmu dan Cabang-cabangnya, Amal, Dakwah, Jihad, Manusia dan Hubungan Kemasyarakatan, Akhlak, Peraturan yang Berhubungan dengan Harta, Hal-hal yang Berkaitan dengan Hukum, Negara dan Masyarakat, Pertanian dan Perdagangan, Sejarah dan Kisah-kisah, dan Agama-agama. Dan 1 topik untuk kelas yang tidak termasuk ke dalam topik manapun (*undefined*). Tetapi pada penelitian ini hanya membahas 6 kelas saja yaitu Dakwah, Jihad, Manusia dan Hubungan Kemasyarakatan, Akhlak, Peraturan yang Berhubungan dengan Harta dan *undefined class*. Dataset bersumber dari data Al-Qur'an yang mulia berbahasa Indonesia.

1. Data Training dan Validation

Data training terdiri dari 22% yang digunakan pada penelitian ini menggunakan surah ke 2 (Al-Baqarah) sampai dengan surat ke 6 (Al-An'am) yang berjumlah 917 ayat, serta pada surah ke 67 (Al-Mulk) berjumlah 431 ayat sampai dengan surah ke 77 (Al-Mursalat) total dari data training yaitu 1244. Adapun untuk data *Validation* diambil dari 10% data training.

2. Data Test

Data *Test* yang digunakan dalam penelitian ini jumlah keseluruhannya terdapat 16 juz dan 3551 ayat. Pembagian juz berdasarkan kategori ayat-ayat panjang (juz 10 sampai juz 20) sebanyak 2003 ayat, ayat-ayat sedang (juz 25 sampai juz 28) sebanyak 977 ayat dan ayat-ayat pendek (juz 30 dan Al-Fatihah) sebanyak 571 ayat.

2.3 Text Preprocessing

Text preprocessing pada klasifikasi teks Al-Qur'an bertujuan untuk mencapai kinerja optimal dengan merepresentasikan dokumen sebagai vektor fitur. Proses ini melibatkan tokenisasi dokumen, representasi dalam ruang vektor data, penghapusan fitur non-informatif seperti *stop words*, angka, dan karakter khusus. Fitur-fitur yang tersisa mengalami standarisasi melalui proses *stemming* untuk mereduksi kata-kata ke bentuk akar[20]. Meskipun telah dilakukan penghapusan fitur tidak informatif, proses *stemming*, dan standarisasi, diperlukan penerapan ambang batas untuk mengurangi dimensi ruang fitur pada setiap dokumen teks[21].

2.4 Vectorizer

Feature Construction adalah proses kunci dalam analisis data yang melibatkan pembuatan fitur baru berdasarkan hubungan antara fitur yang ada, dengan tujuan meningkatkan kinerja algoritma klasifikasi [22]. Dalam penelitian

ini, menggunakan proses vectorizer yang merupakan sebuah konsep dalam pemrosesan teks yang menggabungkan teknik TF.IDF (*Term Frequency-Inverse Document Frequency*) untuk mengekstraksi fitur dari teks. TF.IDF memberikan bobot pada kata-kata berdasarkan frekuensi mereka dalam dokumen dan dataset secara keseluruhan. Dengan menggunakan vectorizer, teks dapat diubah menjadi representasi vektor numerik yang berguna untuk tugas seperti klasifikasi atau analisis sentimen menggunakan algoritma pembelajaran mesin. Untuk menentukan nilai dari hasil klasifikasi pada penelitian ini menggunakan *F1-Score*. *F1-Score* adalah ukuran statistik yang menggabungkan presisi (*precision*) dan *recall* dalam satu metrik tunggal. Presisi mengukur seberapa banyak dari prediksi positif yang sebenarnya benar, sedangkan *recall* mengukur seberapa banyak dari kelas positif yang diprediksi dengan benar. *F1-Score* memberikan keseimbangan antara presisi dan *recall*, dan nilainya berkisar dari 0 hingga 1, di mana 1 menunjukkan kinerja yang sempurna. *F1-Score* sangat berguna dalam evaluasi model klasifikasi, terutama ketika kelas yang dihasilkan oleh model tidak seimbang. Ini memberikan gambaran yang lebih komprehensif tentang seberapa baik model dapat mengidentifikasi kelas positif dan menghindari kesalahan klasifikasi. Adapun rumus untuk *precision*, *recall*, dan *F1-Score* adalah sebagai berikut:

Precision (P) dihitung dengan rumus:

$$P = \frac{TP}{(TP+FP)} \quad (1)$$

Recall (R) dihitung dengan rumus:

$$R = \frac{TP}{(TP+FN)} \quad (2)$$

F1-Score dihitung dengan rumus:

$$F1 = 2 \times \frac{P \times R}{P + R} \quad (3)$$

Di mana TP adalah *true positive*, yaitu jumlah dokumen teks yang secara benar diklasifikasikan ke dalam kategori tertentu. FP adalah *false positive*, yaitu jumlah dokumen teks yang secara keliru diklasifikasikan ke dalam kategori tertentu. FN adalah *false negative*, yaitu jumlah dokumen teks yang secara keliru ditolak diklasifikasikan ke dalam kategori tertentu.

2.5 Random Forest

Random Forest adalah algoritma pembelajaran mesin yang bekerja dengan cara menggabungkan sejumlah besar pohon keputusan (*decision trees*) secara acak. Setiap pohon keputusan dihasilkan dari sampel data yang diambil secara acak dari kumpulan data pelatihan. Ketika melakukan prediksi, setiap pohon memberikan prediksi dan hasil akhir diambil berdasarkan mayoritas suara dari semua pohon keputusan. Hal ini memungkinkan *Random Forest* untuk mengatasi *overfitting* dan meningkatkan *accuracy* prediksi [23]. Dalam penelitian ini, metode *Random Forest* digunakan untuk memproses data terjemahan Al-Qur'an. Terdapat 2 pengaturan model yang diterapkan pada penelitian ini dengan menggunakan *Random Forest* yaitu, pengaturan pertama model yang tidak dilakukan penyetelan parameter. Kemudian model kedua dilakukan penyetelan parameter dengan proses pencarian best parameter RF untuk mendapatkan hasil yang optimal. Untuk *Random Forest* yang terdiri dari N -trees dapat digunakan rumus sebagai berikut :

$$l(y) = \operatorname{argmax}_c (\sum_{n=1}^N I_{h_n(y)=c}) \quad (4)$$

Di sini, h_n adalah tree ke- n dari total N pohon dalam ensemble, dan l adalah fungsi indikator yang bernilai 1 jika tree h_n memprediksi instance y sebagai kelas c , dan 0 jika tidak. Penjumlahan $\sum_{n=1}^N I_{h_n(y)=c}$ menghitung frekuensi prediksi kelas c oleh semua pohon. Operasi argmax_c kemudian mencari kelas c yang paling sering diprediksi oleh pohon-pohon tersebut, sehingga $l(y)$ adalah kelas dengan prediksi mayoritas, memastikan keputusan akhir berdasarkan konsensus terbanyak dalam *Random Forest*.

2.6 Long Short Term Memory

Long Short Term Memory (LSTM) adalah variasi dari RNN, yang dapat berhubungan dengan konteks dan memproses data secara lebih baik. LSTM merupakan konfigurasi khusus dari RNN yang terdiri dari blok memori berulang, yang memiliki tiga gerbang: yaitu gerbang masukan, gerbang keluaran, dan gerbang lupa. Gerbang-gerbang ini memungkinkan LSTM untuk mengingat informasi dalam jangka waktu yang lama, secara selektif melupakannya ketika suatu nilai tidak lagi penting. Pada penelitian ini, LSTM digunakan setelah lapisan *embedding* untuk memproses input teks, diikuti oleh *dropout layer* untuk mengurangi *overfitting*, dan dilanjutkan dengan lapisan *dense* untuk melakukan klasifikasi. Penggunaan *callback ModelCheckpoint* juga memastikan penyimpanan model pada setiap *epoch* hanya jika kinerja pada data *Validation* meningkat, yang memilih pemulihan model terbaik selama pelatihan. Untuk menghasilkan nilai LSTM dapat digunakan persamaan berikut [24]:

$$i_t = \sigma(W_{if} x_t + W_{hi} h_{t-1} + W_{ci} c_t + b_i) \quad (5)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_t + b_f) \quad (6)$$

$$o_t = \sigma(W_{xo}f_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (7)$$

$$c_t = \sigma(f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (8)$$

$$h_t = o_t \tanh(c_t) \quad (9)$$

Persamaan diatas merupakan komponen dari *Long Short-Term Memory* (LSTM), sebuah tipe khusus dari Recurrent Neural Network (RNN) yang digunakan untuk memproses urutan data. Fungsi σ adalah fungsi sigmoid logistik yang mengatur nilai antara 0 dan 1. Rumus (5) menghitung nilai pintu masuk i_t , rumus (6) menghitung nilai pintu lupa f_t , dan rumus (7) menghitung nilai pintu keluar o_t , semuanya berdasarkan input saat ini x_t , status tersembunyi sebelumnya h_{t-1} , dan status sel sebelumnya c_t dengan bobot terkait dan bias masing-masing. Rumus (8) memperbarui nilai sel c_t dengan menggabungkan informasi dari pintu lupa dan pintu masuk. Terakhir, rumus (9) menghitung vektor tersembunyi h_t dengan mengalikan nilai pintu keluar o_t dengan aktivasi tangens hiperbolik dari nilai sel yang diperbarui c_t . Pintu-pintu ini membantu LSTM mengendalikan aliran informasi dan mengatasi masalah vanishing gradient yang sering terjadi pada RNN tradisional.

3. HASIL DAN PEMBAHASAN

Dari percobaan yang dilakukan, penulis menggunakan *python* sebagai bahasa pemrograman untuk memproses data terjemahan Al-Qur'an. Adapun proses yang digunakan terdiri dari dua macam yaitu *Baseline* dan *Optimal*. Berikut merupakan proses *Baseline* yang dilakukan pada penelitian ini.

3.1 Model Baseline

Pada model *baseline*, dilakukan percobaan tanpa proses *balancing* dengan menggunakan metode *Random Forest* dan *Long Short Term Memory* tanpa mengatur parameternya. Dari percobaan tersebut di dapatkan hasil sebagai berikut:

Tabel 1 Hasil *Baseline* RF pada Data *Train* dan *Validation*

Kelas	Train		Validation	
	<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>
Dakwah	100,0%	100,0%	48,4%	93,6%
Jihad	100,0%	100,0%	47,9%	92,0%
Manusia dan hubungan kemasyarakatan	99,8%	99,9%	46,6%	87,2%
Akhlaq	100,0%	100,0%	58,1%	88,8%
Peraturan yang berhubungan dengan harta	99,4%	99,8%	63,3%	96,0%
Undefined	98,2%	99,6%	48,4%	93,6%

Pada tabel 1, menunjukkan hasil penggunaan metode *Random Forest* untuk klasifikasi. Meskipun model berhasil dengan baik pada data pelatihan dengan *F1-Score* dan *accuracy* mencapai tingkat tinggi (bahkan mencapai 100%), kinerjanya menurun drastis pada data *Validation*. Ini mengindikasikan adanya *overfitting*, di mana model tidak dapat menggeneralisasi dengan baik pada data baru yang belum pernah dilihat sebelumnya. Perlu dilakukan penyesuaian atau pengoptimalan lebih lanjut untuk meningkatkan kinerja pada data *Validation*. Selain pada data *Training* dan *Validation*, metode RF juga diujikan kedalam data *Test*. Dimana hasil dari pengujian proses *baseline* RF sebagai berikut :

Tabel 2 Hasil *Baseline* RF pada Data *Test*

Kelas	Data Test Juz 30		Data Test Juz 10-20		Data Test Juz 25-28	
	<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>
Dakwah	48,7%	94,8%	54,5%	97,4%	49,4%	97,4%
Jihad	49,8%	99,1%	50,0%	94,0%	48,1%	92,6%
Manusia dan hubungan kemasyarakatan	49,7%	84,1%	51,3%	91,0%	50,8%	91,0%
Akhlaq	48,0%	92,1%	54,4%	87,2%	52,7%	91,0%
Peraturan yang berhubungan dengan harta	48,0%	92,3%	51,6%	96,3%	53,9%	96,2%
Undefined	48,8%	89,5%	44,7%	80,9%	47,8%	91,5%
Rata-Rata	48,9%	91,4%	50,4%	89,9%	50,7%	92,5%

Pada tabel 2, menampilkan hasil evaluasi metode *Random Forest* pada data uji yang dibagi berdasarkan bagian-bagian Al-Qur'an. Meskipun kinerja bervariasi, secara umum model menunjukkan kinerja yang baik dengan *F1-Score* sekitar 48-55% dan *accuracy* 80-99%. Namun, ada penurunan kinerja pada beberapa kasus,

seperti kelas Undefined pada beberapa bagian terjemahan Al-Qur'an, menunjukkan perlunya analisis dan penyesuaian lebih lanjut pada model.

Selain metode *Random Forest*, penelitian ini juga dilakukan dengan menggunakan metode *Deep Learning* yaitu *Long Short Term Memory* (LSTM). Pada metode LSTM baseline pengaturan parameter yang digunakan yaitu *dense layer* = 286, *dropout* = 0.2, *epochs* = 50, *batch size* = 64. Berikut merupakan hasil dari pengujian *baseline* dengan LSTM :

Tabel 3 Hasil *Baseline* LSTM pada Data *Train* dan *Validation*

Kelas	Parameter <i>Tuning</i>	<i>Train</i>		<i>Validation</i>	
		<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>
Dakwah	<i>dense layer</i> = 286, <i>dropout</i> = 0.2, <i>epochs</i> = 50, <i>batch size</i> = 64	100,0%	100,0%	74,6%	98,6%
Jihad	<i>dense layer</i> = 286, <i>dropout</i> = 0.2, <i>epochs</i> = 50, <i>batch size</i> = 64	100,0%	100,0%	76,2%	94,2%
Manusia dan hubungan kemasyarakatan	<i>dense layer</i> = 286, <i>dropout</i> = 0.2, <i>epochs</i> = 50, <i>batch size</i> = 64	100,0%	100,0%	77,2%	92,8%
Akhlak	<i>dense layer</i> = 286, <i>dropout</i> = 0.2, <i>epochs</i> = 50, <i>batch size</i> = 64	93,2%	93,2%	47,6%	90,7%
Peraturan yang berhubungan dengan harta	<i>dense layer</i> = 286, <i>dropout</i> = 0.2, <i>epochs</i> = 50, <i>batch size</i> = 64	99,8%	99,8%	59,8%	95,0%
Undefined	<i>dense layer</i> = 286, <i>dropout</i> = 0.2, <i>epochs</i> = 50, <i>batch size</i> = 64	99,5%	99,5%	47,9%	92,1%

Pada tabel 3, menampilkan hasil penggunaan metode LSTM dengan penyetelan parameter pada data pelatihan dan *Validation*. Meskipun model mencapai kinerja yang tinggi pada data pelatihan, terutama dengan beberapa kelas mencapai 100% pada *F1-Score* dan *Accuracy*, kinerja menurun pada data *Validation*, menunjukkan adanya penurunan kemampuan generalisasi. Perlu dilakukan analisis lebih lanjut dan penyesuaian tambahan pada model untuk meningkatkan kinerja pada data yang belum terlihat sebelumnya.

Selain pada data *Training* dan *Validation*, metode LSTM juga diujikan kedalam data *Test*. Dimana hasil dari pengujian proses *baseline* LSTM sebagai berikut :

Tabel 4 Hasil *Baseline* LSTM pada Data *Test*

Kelas	Data <i>Test</i> Juz 30		Data <i>Test</i> Juz 10-20		Data <i>Test</i> Juz 25-28	
	<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>
Dakwah	48,7%	94,8%	54,0%	97,2%	54,9%	96,8%
Jihad	49,8%	99,1%	57,1%	94,2%	55,5%	92,8%
Manusia dan hubungan kemasyarakatan	52,0%	82,5%	57,6%	89,1%	63,3%	86,7%
Akhlak	48,1%	92,8%	46,6%	87,1%	47,6%	90,9%
Peraturan yang berhubungan dengan harta	48,0%	92,3%	52,8%	96,3%	55,8%	96,1%
Undefined	47,2%	89,3%	44,7%	80,9%	47,8%	91,5%
Rata-Rata	49,0%	91,2%	51,8%	89,5%	54,0%	91,6%

Pada tabel 4 menampilkan hasil evaluasi metode LSTM pada data uji yang dibagi berdasarkan bagian-bagian terjemahan Al-Qur'an. Secara umum, model menunjukkan kinerja yang baik dengan *F1-Score* berkisar antara 47-63% dan *accuracy* mencapai 80-99%. Namun, ada penurunan kinerja pada beberapa kelas dan bagian-bagian terjemahan Al-Qur'an tertentu, menunjukkan perlunya analisis lebih lanjut dan mungkin penyesuaian pada model.

3.2 Model Optimal

Untuk mendapatkan model dengan hasil yang optimal dilakukan *balance* data pada setiap kelas dengan menggunakan teknik *oversampling*. Terdapat 4 macam eksplorasi untuk mendapatkan nilai paling optimal pada data terjemahan ayat Al-Qur'an yaitu :

- Data yang tidak dilakukan proses *balance* dan *preprocessing* tetapi tidak dilakukan pencarian *best parameter* pada metode RF dan *best model* pada metode LSTM
- Data yang tidak dilakukan proses *balance* yang dilakukan proses *preprocessing* tetapi tanpa *stemming* dan *stopword* dan tidak dilakukan pencarian *best parameter* pada metode RF dan *best model* pada metode LSTM
- Data yang dilakukan proses *balance* dan *preprocessing* serta dilakukan pencarian *best parameter tuning* pada metode RF. Lalu pada metode LSTM dilakukan pencarian *best model* dan penyesuaian *parameter tuning*.

- d. Data yang dilakukan proses *balance* dan *preprocessing* dengan tidak menggunakan proses *stemming* dan *stopword* serta dilakukan *pencarian best parameter tuning* pada metode RF. Lalu pada metode LSTM dilakukan *pencarian best model* dan penyesuaian *parameter tuning*.

Berikut merupakan hasil penelitian berupa *F1-Score* dan *Accuracy* tiap kelas pada metode *Random Forest* terhadap data *Training* dan *Validation* :

Tabel 5 Hasil Pengujian Paling Optimal dari Data *Training* dan *Validation* dengan menggunakan RF

Kelas	Optimasi Proses	Parameter Tuning	Train		Validation	
			<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>
Dakwah	<i>Full Preprocessing</i>	<i>Criterion : entropy, Max_depth : 35, Max_features : sqrt, n_estimators : 90</i>	100,0%	100,0%	49,5%	97,8%
Jihad	<i>Full Preprocessing</i>	<i>Criterion : entropy, Max_depth : 20, Max_features : sqrt, n_estimators : 75</i>	97,3%	97,3%	75,3%	92,8%
Manusia dan hubungan kemasyarakatan	<i>Full Preprocessing</i>	<i>Criterion : entropy, Max_depth : 88, Max_features : log2, n_estimators : 60</i>	99,1%	99,1%	56,9%	88,5%
Akhlaq	<i>Full Preprocessing</i>	<i>Criterion : gini, Max_depth : 68, Max_features : sqrt, n_estimators : 50</i>	99,8%	99,8%	55,6%	87,1%
Peraturan yang berhubungan dengan harta	<i>Full Preprocessing</i>	<i>Criterion : entropy, Max_depth : 50, Max_features : sqrt, n_estimators : 50</i>	100,0%	100,0%	48,5%	94,2%
Undefined	<i>Full Preprocessing</i>	<i>Criterion : entropy, Max_depth : 46, Max_features : sqrt, n_estimators : 36</i>	98,8%	98,8%	56,5%	92,8%

Pada tabel 5 tersebut menampilkan hasil optimal dari *training* dan *Validation* menggunakan metode *Random Forest* dengan pengaturan parameter yang berbeda untuk setiap kelas. Meskipun sebagian besar kelas mencapai tingkat kinerja yang tinggi pada data pelatihan, kinerja pada data *Validation* menunjukkan variasi. Ini menunjukkan pentingnya penyesuaian parameter dan teknik pemrosesan yang tepat untuk meningkatkan generalisasi model pada data yang belum terlihat sebelumnya, dengan tujuan mencapai kinerja yang lebih konsisten pada berbagai kelas.

Selain pada data *Training* dan *Validation*, metode RF Optimal juga diujikan kedalam data *Test*. Dimana hasil dari pengujiannya dapat dilihat pada tabel 6 sebagai berikut :

Tabel 6 Hasil Pengujian Paling Optimal pada Data *Test* menggunakan metode RF

Kelas	Data Test Juz 30		Data Test Juz 10-20		Data Test Juz 25-28	
	<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>	<i>F1-Score</i>	<i>Accuracy</i>
Dakwah	48,7%	94,8%	52,8%	97,4%	49,4%	97,5%
Jihad	49,7%	99,0%	53,1%	94,6%	56,5%	92,8%
Manusia dan hubungan kemasyarakatan	46,6%	83,7%	51,7%	91,0%	51,8%	91,0%
Akhlaq	53,1%	91,2%	59,4%	87,1%	58,3%	90,4%
Peraturan yang berhubungan dengan harta	52,2%	92,3%	60,6%	96,7%	62,2%	96,5%
Undefined	51,5%	89,1%	45,7%	80,9%	47,7%	91,2%
Rata-Rata	50,6%	91,1%	56,1%	90,0%	55,3%	92,4%

Dari tabel 6 tersebut, dapat dilihat performa model pada setiap klasifikasi dan bagian-bagian data tes. Misalnya, kelas Peraturan yang Berhubungan dengan Harta dengan proses *Full Preprocessing* memiliki *F1-Score* yang bervariasi tergantung pada bagian data tes yang digunakan, dengan nilai tertinggi pada Data Test Juz 25-28 (62,2%) dan terendah pada Data Test Juz 30 (46,6%). Hal ini menunjukkan bahwa performa model bisa berbeda tergantung pada bagian-bagian tertentu dari data tes. Selain itu, perbedaan dalam proses pengolahan data juga mempengaruhi performa model, seperti yang terlihat dari perbedaan *F1-Score* dan *Accuracy* antara klasifikasi dengan dan tanpa penggunaan *stemming* dan *stopword preprocessing*.

Selain metode *Random Forest*, penelitian ini juga dilakukan dengan menggunakan metode *Deep Learning* yaitu *Long Short Term Memory* (LSTM). Berikut merupakan hasil dari pengujian dengan LSTM :

Tabel 7 Hasil Paling Optimal dari Proses *Training* dan *Validation* dengan Metode LSTM

Kelas	Optimasi Proses	Parameter Tuning	Train		Validation	
			F1-Score	Accuracy	F1-Score	Accuracy
Dakwah	Full Preprocessing	dense layer = 347, dropout = 0.5, best model epochs = 10, batch size = 256	100,0%	100,0%	78,0%	97,8%
Jihad	Preprocessing without stemming and stopword	dense layer = 300, dropout = 0.5, best model epochs = 14, batch size = 64	100,0%	100,0%	78,5%	94,2%
Manusia dan hubungan kemasyarakatan	Full Preprocessing	dense layer = 340, dropout = 0.5, best model epochs = 14, batch size = 128	99,7%	99,7%	74,5%	91,4%
Akhlaq	Full Preprocessing	dense layer = 290, dropout = 0.5, best model epochs = 6, batch size = 256	93,2%	93,2%	58,9%	84,2%
Peraturan yang berhubungan dengan harta	Full Preprocessing	dense layer = 340, dropout = 0.5, best model epochs = 9, batch size = 256	99,8%	99,8%	71,8%	95,0%
Undefined	Full Preprocessing	dense layer = 340, dropout = 0.5, best model epochs = 4, batch size = 32	99,5%	99,5%	62,4%	92,8%

Dari tabel 7 tersebut merupakan hasil dari proses pelatihan (*training*) dan *Validation* menggunakan metode *Long Short Term Memory* (LSTM) untuk klasifikasi. Dari tabel tersebut, dapat dilihat bahwa performa model bervariasi tergantung pada klasifikasi dan proses yang dilakukan sebelumnya. Misalnya, kelas Jihad dengan proses *Preprocessing without stemming and stopword* memiliki *F1-Score* tertinggi pada data *Validation* (80,3%), sementara kelas Akhlak memiliki *F1-Score* terendah (58,9%). Penyetelan parameter tuning pada LSTM sangat berpengaruh untuk memberikan hasil optimal. Hal ini menunjukkan bahwa penggunaan LSTM dengan penyetelan parameter yang tepat dan proses *preprocessing* yang sesuai dapat meningkatkan kinerja model dalam melakukan klasifikasi.

Selain pada data *Training* dan *Validation*, metode LSTM juga diujikan kedalam data *Test*. Dimana hasil dari pengujian proses optimal LSTM sebagai berikut :

Tabel 8 Hasil Pengujian Data *Test* paling optimal dari proses *Test* pada metode LSTM

Kelas	Data Test Juz 30		Data Test Juz 10-20		Data Test Juz 25-28	
	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy
Dakwah	48,6%	94,8%	54,3%	96,6%	57,3%	96,8%
Jihad	49,7%	99,0%	62,2%	94,4%	57,0%	92,5%
Manusia dan hubungan kemasyarakatan	52,5%	83,2%	59,2%	88,2%	64,2%	89,1%
Akhlaq	59,1%	91,6%	62,4%	85,8%	60,6%	87,4%
Peraturan yang berhubungan dengan harta	56,8%	91,9%	67,9%	96,4%	67,7%	95,9%
Undefined	51,7%	87,6%	52,3%	79,7%	54,1%	89,7%
Rata-Rata	54,0%	90,6%	60,8%	88,9%	60,7%	90,9%

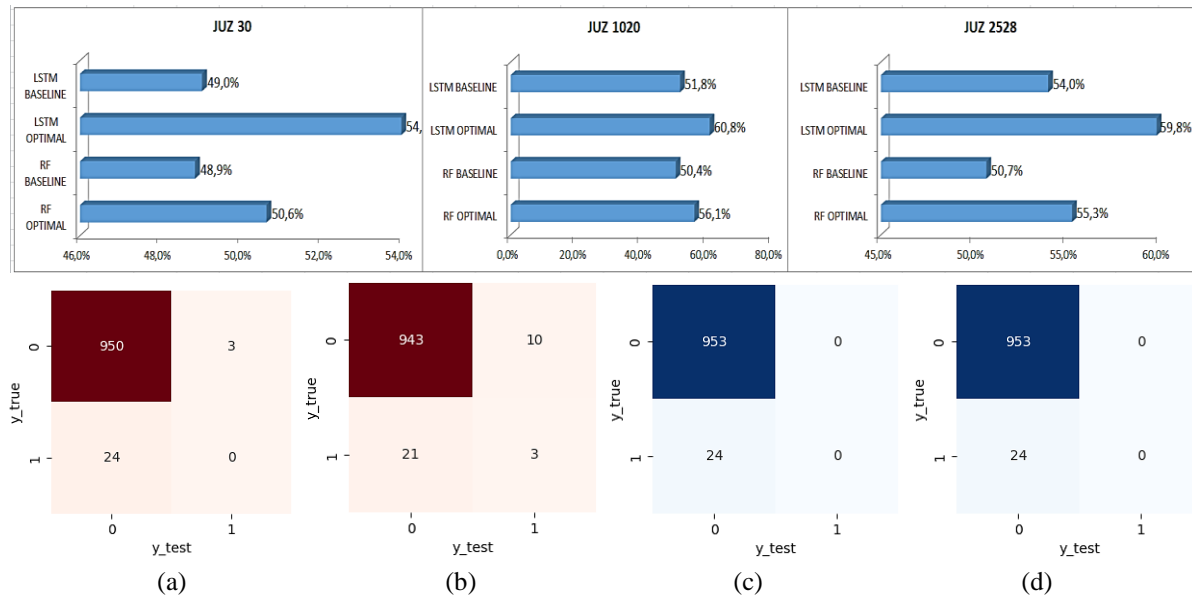
Pada tabel 8 tersebut menunjukkan hasil pengujian menggunakan metode *Long Short Term Memory* (LSTM) pada data uji, yang diproses dengan parameter optimal dari proses pelatihan dan *Validation* sebelumnya. Hasilnya menunjukkan variasi kinerja antara kelas dan bagian-bagian Al-Qur'an yang berbeda. Sebagian besar kelas mencapai tingkat kinerja yang cukup baik, Kinerja model bervariasi antar kategori dan bagian data tes, dengan rata-rata *F1-Score* berkisar antara 54,5% hingga 61,2% dan *accuracy* antara 88,9% hingga 90,8%, menunjukkan kemampuan model dalam mengklasifikasikan teks Al-Quran dengan baik meskipun ada perbedaan dalam tingkat kinerja antar kategori dan bagian data tes yang diuji.

Tabel 9 Hasil Rata-Rata *F1-Score* dari Data *Test* dengan metode RF dan LSTM

Model	Data Test Juz 30		Data Test Juz 1020		Data Test Juz 2528	
	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy
RF Optimal	50,6%	91,1%	56,1%	90%	55,3%	92,4%
RF Baseline	48,9%	91,4%	50,4%	89,9%	50,7%	92,5%
LSTM Optimal	54,0%	90,6%	60,8%	88,9%	60,7%	91,2%

<i>LSTM Baseline</i>	49,0%	91,2%	51,8%	89,5%	54,0%	91,6%
----------------------	-------	-------	-------	-------	-------	-------

Tabel 9 merupakan perbandingan hasil rata-rata *F1-Score* dari data *Test* dengan menggunakan *Random Forest* dan *Long Short Term Memory*. Dimana model RF optimal memiliki nilai rata-rata(50-55%) lebih rendah dari model LSTM optimal yang rata-ratanya(54-61%).



Gambar 1. Grafik Perbandingan *F1-Score* Model *Baseline* dan Optimal

Berdasarkan grafik diatas, yang merupakan perbandingan dari nilai data *Test* juz 30, juz 10-20, dan juz 25-28 antara proses *baseline* dan proses optimal yang dilakukan pada metode RF dan LSTM. Dapat dilihat bahwa pada LSTM optimal memiliki nilai yang lebih dominan tinggi di setiap data tes. Gambar diatas merupakan *confusion matriks* dari salah satu kelas yaitu kelas Dakwah pada *Test* juz 25-28. Pada gambar (a) merupakan *confusion matriks* dengan menggunakan metode RF *baseline* sedangkan pada gambar (b) merupakan *confusion matriks* dengan menggunakan RF optimal. Dapat dilihat dari gambar tersebut pada proses model RF tidak mengalami peningkatan di model optimalnya. Untuk gambar (c) merupakan *confusion matriks* dengan menggunakan metode LSTM *baseline* dan gambar (d) merupakan *confusion matriks* dengan menggunakan metode LSTM optimal. Pada *confusion matriks* di LSTM *baseline* menunjukkan proses *baseline* dapat membaca data 0 dan 1. Tetapi proses optimal masih memiliki nilai lebih tinggi dalam mengelola data 1 daripada proses *baseline*. Berikut merupakan tabel dari kelas jihad yang terdiri dari *baseline* dan optimal yang menunjukkan perbandingan *F1-Score* dan *Accuracy*.

Tabel 10 Kesalahan Klasifikasi pada Data *Test* 2528 di Kelas Dakwah dengan metode RF Optimal

Terjemahan	Data Label	Hasil Prediksi
Dan orang-orang yang mengambil pelindung-pelindung selain Allah, Allah mengawasi (perbuatan) mereka; adapun engkau (Muhammad) bukanlah orang yang diserahi mengawasi mereka.(42:06)	1	0
Maka sungguh, sekiranya Kami mewafatkanmu (sebelum engkau mencapai kemenangan), maka sesungguhnya Kami akan tetap memberikan azab kepada mereka (di akhirat),(43:41)	1	0

Pada tabel 10, menampilkan hasil kesalahan klasifikasi pada data *Test* pada juz 25-28 dalam kelas dakwah menggunakan metode *Random Forest* (RF) yang dioptimalkan. Di dalam tabel, terdapat dua contoh ayat Al-Qur'an beserta terjemahan dan label asli dari data, serta hasil prediksi yang diberikan oleh model klasifikasi. Angka "1" menunjukkan bahwa ayat tersebut termasuk dalam kategori tertentu, sedangkan angka "0" menunjukkan sebaliknya. Pada contoh pertama, model memprediksi ayat sebagai kategori yang salah, sedangkan pada contoh kedua, hasil prediksi juga tidak sesuai dengan label yang seharusnya. Ini menunjukkan adanya kesalahan dalam klasifikasi oleh model terhadap data uji yang diberikan.

Tabel 11 Kesalahan Klasifikasi pada Data *Test* 2528 di Kelas Dakwah dengan Metode LSTM Optimal

Terjemahan	Data Label	Hasil Prediksi
Dan lenyaplah dari mereka apa yang dahulu selalu mereka sembah, dan mereka pun	0	1

Terjemahan	Data Label	Hasil Prediksi
tahu bahwa tidak ada jalan keluar (dari azab Allah) bagi mereka(48:41). Dan ketika putra Maryam (Isa) dijadikan perumpamaan, tiba-tiba kaummu (Suku Quraisy) bersorak karenanya.(43:57)	1	0

Tabel 11 tersebut menampilkan hasil kesalahan klasifikasi pada data *Test* pada juz 25-28 dalam kelas dakwah menggunakan metode *Long Short-Term Memory* (LSTM) yang dioptimalkan. Dalam tabel ini, disajikan dua contoh ayat Al-Qur'an beserta terjemahannya, label asli dari data, dan hasil prediksi yang diberikan oleh model klasifikasi LSTM. Angka "1" menunjukkan bahwa ayat tersebut termasuk dalam kategori tertentu, sedangkan angka "0" menunjukkan sebaliknya. Pada contoh pertama, model memprediksi ayat sebagai kategori yang salah, sedangkan pada contoh kedua, hasil prediksi juga tidak sesuai dengan label yang seharusnya. Ini menunjukkan adanya kesalahan dalam klasifikasi oleh model terhadap data uji yang diberikan. Kesalahan klasifikasi pada metode RF dan LSTM terjadi karena model gagal memahami makna kompleks dan struktur kalimat yang rumit dalam ayat Al-Qur'an. Hal ini menyebabkan prediksi yang tidak sesuai dengan label yang seharusnya, baik untuk ayat yang seharusnya termasuk dalam kategori tertentu (label 1) maupun yang seharusnya tidak (label 0).

Tabel 12 Perbandingan *F1-Score* dan *Accuracy* pada kelas Dakwah

Kelas	Metode	<i>F1-Score</i>	<i>Accuracy</i>
Dakwah	<i>RF Baseline</i>	49,4%	97,5%
Dakwah	<i>RF Optimal</i>	49,4%	97,5%
Dakwah	<i>LSTM Baseline</i>	54,9%	96,8%
Dakwah	<i>LSTM Optimal</i>	57,3%	96,8%

Dapat dilihat pada tabel 12, model optimal memiliki nilai lebih tinggi daripada model *baseline*. Pada metode *Random Forest* optimal tidak mengalami kenaikan dari *baseline*. Sedangkan pada metode *Long Short Term Memory* model optimal, *F1-Score* mengalami peningkatan sebesar 2,4% dari *baseline* dan *Accuracy* yang masih sama dengan proses *baseline*. Dapat tarik kesimpulan bahwa model optimal LSTM memiliki peningkatan lebih besar daripada model optimal RF. Sehingga *Long Short Term Memory* dianggap sebagai model yang lebih baik dalam melakukan proses klasifikasi pada data terjemahan Al-Qur'an daripada model *Random Forest*.

4. KESIMPULAN

Dari percobaan yang telah dilakukan terhadap 6 kelas yaitu Dakwah, Jihad, Manusia dan Hubungan Kemasyarakatan, Akhlak, Peraturan yang Berhubungan dengan Harta dan *undefined class*. Metode *Long Short Term Memory* (LSTM) menghasilkan *F1-Score* dan *Accuracy* yang lebih baik daripada metode *Random Forest* (RF). Dari hasil penelitian yang dilakukan menggunakan metode *Random Forest* (RF) dan *Long Short Term Memory* (LSTM) untuk melakukan klasifikasi terhadap terjemahan Al-Qur'an, dapat disimpulkan bahwa performa model klasifikasi dipengaruhi oleh berbagai faktor, termasuk jenis proses *preprocessing*, penyetelan parameter, dan metode klasifikasi yang digunakan. Percobaan menunjukkan bahwa penggunaan *preprocessing* yang lengkap, penyetelan parameter yang optimal, dan data yang seimbang cenderung menghasilkan performa yang lebih baik dalam hal *F1-Score* dan *Accuracy*. Selain itu, pentingnya generalisasi model terhadap berbagai bagian data tes juga menjadi faktor kunci dalam mengembangkan model yang handal. Dengan demikian, hasil penelitian ini memberikan wawasan penting bagi peneliti atau praktisi dalam pengembangan model klasifikasi untuk kasus-kasus yang melibatkan teks terjemahan Al-Qur'an. Adapun saran untuk pengembangan penelitian selanjutnya yaitu dapat membahas tentang teknik lainnya seperti *undersampling* dalam mencapai nilai yang lebih optimal. Pada model RF dengan menggunakan *best parameter* tentunya telah menghasilkan nilai terbaik. Sedangkan pada metode LSTM tentunya masih bisa di kembangkan lagi dalam proses penyetelan parameternya agar mencapai hasil yang lebih optimal.

REFERENCES

[1] Kemenag, 'Menjadi Muslim, Menjadi Indonesia (Kilas Balik Indonesia Menjadi Bangsa Muslim Terbesar)', <https://kemenag.go.id>. Accessed: Nov. 28, 2023. [Online]. Available: <https://kemenag.go.id/opini/menjadi-muslim-menjadi-indonesia-kilas-balik-indonesia-menjadi-bangsa-muslim-terbesar-03w0yt>

[2] M. A. S. Nuruddaroini and Muh. H. Zubaidillah, 'Penghafal Alquran Perspektif Sikap Kognitif', *AM*, vol. 1, no. 2, p. 112, Feb. 2023, doi: 10.35931/am.v1i2.1472.

[3] M. Alshammeri, E. Atwell, and M. Alsalka, 'Classifying Verses of the Quran using Doc2vec', *Proceedings of the 18th International Conference on Natural Language Processing*, pp. 284–288, 2021.

[4] R. A. Pane and M. S. Mubarak, 'Klasifikasi Multi-Label Pada Topik Ayat Al-Quran Terjemahan Bahasa Inggris Menggunakan Multinomial Naive Bayes', 2018.

[5] R. A. Haristu and P. H. P. Rosa, 'Penerapan Metode Random Forest untuk Prediksi Win Ratio Pemain Player Unknown Battleground', *MEANS*, pp. 120–128, Oct. 2019, doi: 10.54367/means.v4i2.545.

- [6] O. R. Olaniran and M. A. A. Abdullah, 'Bayesian weighted random forest for classification of high-dimensional genomics data', *Kuwait Journal of Science*, vol. 50, no. 4, pp. 477–484, Oct. 2023, doi: 10.1016/j.kjs.2023.06.008.
- [7] A. Efendi, I. Iskandar, R. Kurniawan, and M. Affandes, 'Klasifikasi Kebakaran Hutan Riau Menggunakan Random Forest dan Visualisasi Citra Sentinel-2', vol. Vol 4, pp. 1602–1612, 2023, doi: 10.30865/klik.v4i3.1521.
- [8] I. Kurniawan, D. C. P. Buani, A. Abdussomad, W. Apriliah, and R. A. Saputra, 'Implementasi Algoritma Random Forest Untuk Menentukan Penerima Bantuan Raskin', *JTIK*, vol. 10, no. 2, pp. 421–428, Apr. 2023, doi: 10.25126/jtik.20231026225.
- [9] C. S. Arsyah and M. Elsera, 'Implementasi Random Forest Dalam Melakukan Klasifikasi Kata Sarkasme Pada Media Sosial Facebook', *Com, Engine, Sys, Sci*, vol. 4, no. 1, pp. 216–223, Jul. 2023, doi: 10.46576/djtechno.v4i1.3361.
- [10] N. Widjiyati, 'Implementasi Algoritme Random Forest Pada Klasifikasi Dataset Credit Approval', *J. Janitra Inform. Sis. Inf.*, vol. 1, no. 1, pp. 1–7, Apr. 2021, doi: 10.25008/janitra.v1i1.118.
- [11] B. R. Khalil, 'Heart Failure Patients Are Classified Using The Random Forest And Naïve Bayes Algorithms', *Journal of Northeastern University*, vol. 26, no. 03, 2023.
- [12] S. Devella, Y. Yohannes, and F. N. Rahmawati, 'Implementasi Random Forest Untuk Klasifikasi Motif Songket Palembang Berdasarkan SIFT', *JATISI*, vol. 7, no. 2, pp. 310–320, Aug. 2020, doi: 10.35957/jatisi.v7i2.289.
- [13] N. Husin, 'Komparasi Algoritma Random Forest, Naïve Bayes, dan Bert Untuk Multi-Class Classification Pada Artikel Cable News Network (CNN)', *Infokom*, vol. 7, no. 1, pp. 75–84, May 2023, doi: 10.55886/infokom.v7i1.608.
- [14] Z. Chang, Y. Zhang, and W. Chen, 'Electricity price prediction based on hybrid model of adam optimized LSTM neural network and wavelet transform', *Energy*, vol. 187, p. 115804, Nov. 2019, doi: 10.1016/j.energy.2019.07.134.
- [15] H. Wang and F. Li, 'A text classification method based on LSTM and graph attention network', *Connection Science*, vol. 34, no. 1, pp. 2466–2480, Dec. 2022, doi: 10.1080/09540091.2022.2128047.
- [16] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, 'Sentiment Analysis Using Word2vec And Long Short-Term Memory (LSTM) For Indonesian Hotel Reviews', *Procedia Computer Science*, vol. 179, pp. 728–735, 2021, doi: 10.1016/j.procs.2021.01.061.
- [17] M. Ihsan, Benny Sukma Negara, and Surya Agustian, 'LSTM (Long Short Term Memory) for Sentiment COVID-19 Vaccine Classification on Twitter', *Digitalzone*, vol. 13, no. 1, pp. 79–89, May 2022, doi: 10.31849/digitalzone.v13i1.9950.
- [18] G. Budiprasetyo, M. Hani'ah, and D. Z. Aflah, 'Prediksi Harga Saham Syariah Menggunakan Algoritma Long Short-Term Memory (LSTM)', *TEKNOSI*, vol. 8, no. 3, pp. 164–172, Jan. 2023, doi: 10.25077/TEKNOSI.v8i3.2022.164-172.
- [19] 'Al-Quran yang mulia', Quran.com. Accessed: Feb. 13, 2024. [Online]. Available: <https://quran.com/id>
- [20] R. Ulgasesa, A. B. P. Negara, and T. Tursina, 'Pengaruh Stemming Terhadap Performa Klasifikasi Sentimen Masyarakat Tentang Kebijakan New Normal', *justin*, vol. 10, no. 3, p. 286, Sep. 2022, doi: 10.26418/justin.v10i3.53880.
- [21] A. I. Kadhim, 'An Evaluation of Preprocessing Techniques for Text Classification', *International Journal of Computer Science and Information Security (IJCSIS)*, vol. Vol. 16, No. 6, no. 6, pp. 22–32, Jun. 2018.
- [22] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, 'Selecting Critical Features For Data Classification Based On Machine Learning Methods', *J Big Data*, vol. 7, no. 1, p. 52, Dec. 2020, doi: 10.1186/s40537-020-00327-4.
- [23] T. Kam Ho, 'Random Decision Forest', in *Proceedings of 3rd international conference on document analysis and recognition*, IEEE, Aug. 1995, pp. 278–282.
- [24] A. Qayyum, S. Latif, and J. Qadir, 'Quran Reciter Identification: A Deep Learning Approach', in *2018 7th International Conference on Computer and Communication Engineering (ICCCCE)*, Kuala Lumpur: IEEE, Sep. 2018, pp. 492–497. doi: 10.1109/ICCCCE.2018.8539336.