Institut Riset dan Publikasi Indonesia (IRPI)
MALCOM: Indonesian Journal of Machine Learning and Computer Science
Journal Homepage: https://journal.irpi.or.id/index.php/malcom
Vol. 5 Iss. 3 July 2025, pp: 862-874
ISSN(P): 2797-2313 | ISSN(E): 2775-8575

Comparison of Random Forest and Long Short-Term Memory Performance in Multilabel Text Classification of Bukhari Hadith Translation

Perbandingan Performa *Random Forest* dan *Long Short-Term Memory* dalam Klasifikasi Teks Multilabel Terjemahan Hadits Bukhari

Rizmah Zakiah Nur Ahmad[1], Nazruddin Safaat Harahap[2*], Surya Agustian[3], Iwan Iskandar[4], Suwanto Sanjaya[5]

[1,2,3,4,5]Program Studi Teknik Informatika, Fakultas Sains dan Teknologi,
Universitas Islam Negeri Sultan Syarif Kasim Riau, Indonesia

E-Mail: [1]12150122153@students.uin-suska.ac.id, [2]nazruddin.safaat@uin-suska.ac.id,
[3]surya.agustian@uin-suska.ac.id, [4]iwan.iskandar@uin-suska.ac.id, [5]suwantosanjaya@uin-suska.ac.id

Received Apr 09th 2025; Revised Jun 03rd 2025; Accepted Jun 15th 2025; Available Online Jun 23th 2025, Published Jun 23th 2025
Corresponding Author: Nazrudin Safaat Harahap

Abstract

Hadith serves as the second main foundation in Islam, guiding Muslims in interpreting Islamic values and implementing them in real terms in various aspects of life. One of the most respected narrators of hadith is Imam Bukhari, who is known for his thoroughness and strictness in selecting authentic hadith. This study utilizes data from the translation of hadith from Sahih Bukhari into Indonesian, which has been classified into three main categories, namely recommendations, prohibitions, and information. To identify the characteristics of each category, text classification was carried out using two popular methods, namely Random Forest (RF) and Long Short-Term Memory (LSTM), which are known to be effective in processing large-scale and complex text data. The purpose of this study was to examine the difference in performance between the two methods for grouping hadith whose data had been completed. The evaluation results showed that the RF method achieved the highest accuracy of 89.48%, slightly superior to LSTM, which obtained 88.52%. Both methods recorded the same Hamming Loss value, namely 0.1048 (89.52%). These findings suggest that the completeness and quality of Bukhari hadith data contribute to improving classification accuracy by providing better context and variation for the model.

Keyword: Hadits Bukhari, Hamming Loss, Long Short-Term Memory, Random Forest, Text Classification

Abstrak

Hadits merupakan fondasi utama kedua dalam Islam, yang memandu umat Islam dalam menafsirkan nilai-nilai Islam dan mengimplementasikannya secara nyata dalam berbagai aspek kehidupan. Salah satu perawi hadits yang paling dihormati adalah Imam Bukhari, yang dikenal dengan ketelitian dan ketegasannya dalam memilih hadits-hadits yang otentik. Penelitian ini menggunakan data dari terjemahan hadis dari Sahih Bukhari ke dalam bahasa Indonesia yang telah diklasifikasikan ke dalam tiga kategori utama, yaitu anjuran, larangan, dan informasi. Untuk mengidentifikasi karakteristik masing-masing kategori, klasifikasi teks dilakukan dengan menggunakan dua metode populer, yaitu Random Forest (RF) dan Long Short-Term Memory (LSTM), yang dikenal efektif dalam memproses data teks berskala besar dan kompleks. Tujuan dari penelitian ini adalah untuk menguji perbedaan kinerja antara kedua metode tersebut dalam mengelompokkan hadis yang datanya telah lengkap. Hasil evaluasi menunjukkan bahwa metode RF mencapai akurasi tertinggi sebesar 89,48%, sedikit lebih unggul dari LSTM yang memperoleh 88,52%. Kedua metode mencatat nilai Hamming Loss yang sama, yaitu 0,1048 (89,52%). Temuan ini menunjukkan bahwa kelengkapan dan kualitas data hadis Bukhari berkontribusi dalam meningkatkan akurasi klasifikasi dengan memberikan konteks dan variasi yang lebih baik untuk model.

Kata Kunci: Hadits Bukhari*, Hamming Loss*, Klasifikasi Teks*, Long Short-Term Memory, Random Forest*

1. PENDAHULUAN

Hadits merupakan sumber hukum kedua dalam Islam yang berfungsi sebagai penjelas terhadap makna Al-Qur'an [1]. Sebagai pedoman hidup umat Islam, hadits berisi petunjuk mengenai perkataan, perbuatan, serta sikap Nabi Muhammad SAW dalam berbagai aspek kehidupan [2]. Oleh karena itu, mempelajari, memahami,

dan mengamalkan hadits adalah kewajiban bagi setiap Muslim. Imam Bukhari, dengan nama lengkap Abu Abdullah Muhammad bin Ismail bin Ibrahim bin Mughirah al-Ja'fi bin Bardzibah, adalah salah satu perawi hadits yang sangat terkenal [3]. Beliau lahir pada 13 Syawal 194 H di kota Bukhara dan Beliau dikenal sebagai seorang ulama hadits yang sangat teliti dalam memverifikasi keabsahan setiap hadits yang diriwayatkannya [4]. Kitab Shahih Al-Bukhari yang disusunnya memiliki kualitas autentik yang tinggi dan menjadi acuan utama dalam kalangan umat Islam. Kitab ini berisi beragam jenis ajaran yang mencakup anjuran, larangan, serta informasi yang sangat berguna untuk dilaksanakan dalam kehidupan sehari-hari.

Dengan kemajuan teknologi, kajian terhadap hadits semakin berkembang, salah satunya melalui pendekatan klasifikasi hadits melalui karakteristiknya dalam terjemahan bahasa Indonesia. Pendekatan ini penting dalam mendukung pemahaman umat Islam di Indonesia dengan menyajikan makna hadits yang sesuai konteks dan lebih mudah dipahami. Klasifikasi, dalam konteks ini, merujuk pada proses pengelompokan data atau informasi ke dalam kategori tertentu melalui ciri atau fitur yang ada dalam data tersebut [2], [5]. Salah satu cara yang diterapkan dalam pengelompokan hadits adalah dengan mengelompokkan hadits ke dalam tiga kategori, yaitu anjuran, larangan, dan informasi, seperti yang telah dilakukan oleh peneliti Muhammad Yuslan Abu Bakar dalam risetnya [6].

Dalam implementasinya, klasifikasi hadits dapat dilakukan dengan pendekatan multi-label, di mana satu hadits dapat mengandung lebih dari satu kategori dalam teksnya. Sebagai contoh, sebuah hadits dapat memberikan informasi sekaligus mengandung unsur anjuran atau bahkan larangan dalam konteks tertentu. Oleh karena itu, penerapan metode klasifikasi yang dapat menangani multi-label classification menjadi sangat penting dalam penelitian ini.

Beragam pendekatan telah dicoba dalam proses klasifikasi terjemahan hadis Bukhari. Beberapa studi sebelumnya menerapkan metode, seperti *Recurrent Convolution Neural Network* (RCNN) [6] dan metode, *K-Nearest Neighbor* (KNN) [7], serta berbagai pendekatan lainnya [8][9] untuk menyelesaikan permasalahan klasifikasi hadits. Namun, masih jarang ditemukan penelitian yang secara langsung membandingkan performa antara metode *Machine Learning* Konvensional dan pendekatan *Deep Learning* dalam konteks klasifikasi hadis. Maka dari itu, penelitian ini dibuat guna menguji perbedaan hasil antara metode *Random Forest* (RF), sebagai representasi algoritma tradisional, dengan *Long Short-Term Memory* (LSTM) yang merupakan metode dari *Deep Learning*, dalam tugas klasifikasi multi-label terhadap teks terjemahan hadis Bukhari.

RF dikenal dengan kemampuannya dalam menangani data berdimensi tinggi [8], menjadikannya cocok untuk klasifikasi teks. Namun, penerapannya dalam klasifikasi multi-label, khususnya pada data hadits, masih terbatas. Dalam penelitian ini, setiap label pada data hadits akan diproses secara independen, di mana setiap label akan dianalisis secara terpisah tanpa menggunakan transformasi label. Pendekatan ini memungkinkan analisis yang lebih spesifik terhadap setiap kategori, yaitu anjuran, larangan, dan informasi. Metode serupa telah digunakan dalam beberapa penelitian sebelumnya [5], [9], [10] , di mana label diproses secara independen untuk menyempurnakan pemahaman dalam klasifikasi terjemahan ayat suci Al-Qur'an.

Keunggulan RF dalam klasifikasi telah dibahas dalam beberapa literatur. Sebagai contoh, dalam sebuah *Systematic Literature Review* [11], dinyatakan bahwa RF memiliki kelebihan dalam menyelesaikan tugas klasifikasi secara efisien dengan tingkat kesalahan yang relatif rendah. Penelitian lain [9] menunjukkan bahwa RF mencapai akurasi sebesar 96.8% dan *F1-Score* 57.3% dalam klasifikasi terjemahan Al-Qur'an. Selain itu, penelitian [12] terkait identifikasi pasien diabetes melalui penerapan metode normalisasi juga menunjukkan bahwa RF dapat mencapai peforma optimal dengan tingkat akurasi tinggi. Namun, penambahan jumlah pohon (*tree*) dalam algoritma RF tidak selalu berkontribusi pada peningkatan akurasi yang signifikan. Di sisi lain, penelitian [13], yang membandingkan mutual information dan *Chi-Square*, menunjukkan akurasi tertinggi sebesar 91.7%, meskipun tantangan berupa ketidakseimbangan data hadits tetap menjadi masalah dalam sistem klasifikasi.

Selain memanfaatkan model pembelajaran mesin, proses klasifikasi data hadits juga diterapkan melalui pendekatan pembelajaran mendalam, dengan mengandalkan arsitektur LSTM yang dirancang untuk menangani karakteristik data teks yang bersifat urut dan saling terkait. LSTM memiliki keunggulan dalam mengingat informasi dalam jangka panjang, sehingga lebih unggul dalam mengenali pola dalam data teks [9]. Penelitian [14], menunjukkan bahwa LSTM menghasilkan akurasi kompetitif sebesar 97% dalam klasifikasi berita BBC. Hal ini menunjukkan bahwa LSTM cocok digunakan dalam klasifikasi teks berbasis urutan, termasuk data hadits. Akan tetapi, kelemahannya terletak pada durasi pelatihan yang lebih lama serta membutuhkan daya komputasi yang lebih besar.

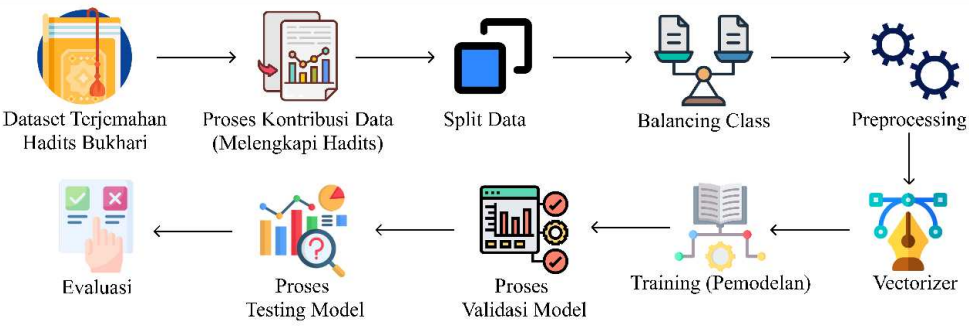
LSTM memiliki kemampuan untuk mengatasi masalah *Vanishing Gradient* yang sering timbul pada *Recurrent Neural Network* (RNN) standar, yang memungkinkan pembelajaran yang lebih stabil pada data teks panjang [15]. Hal ini dicapai melalui arsitektur internal LSTM yang terdiri dari *forget gate*, *input gate*, dan *output gate*, yang secara selektif mengatur aliran informasi. Penelitian [15] juga menunjukkan bahwa LSTM lebih unggul dibandingkan dengan *Convolution Neural Network* (CNN), dengan selisih akurasi sebesar 2.3% dalam klasifikasi teks COVID-19 di Twitter. Sementara itu, penelitian [16], menunjukkan bahwa dalam prediksi harga saham, LSTM mampu mengoptimalkan konfigurasi dengan menguji berbagai parameter, seperti jumlah lapisan, *epoch*, dan panjang *time step*.

Penelitian ini tidak hanya bertujuan untuk membandingkan performa RF dan LSTM, tetapi juga berkontribusi pada perbaikan *dataset* melalui proses pelengkapan teks hadits. Langkah ini dirancang sebagai bagian penting dari keseluruhan alur eksperimen, guna memperoleh hasil yang lebih representatif dan reliabel. Pelengkapan data menjadi bagian esensdaial dalam penelitian ini, karena secara langsung berpengaruh terhadap kinerja model dalam menangani teks hadits yang kompleks dan kontekstual. Penelitian ini menerapkan klasifikasi independen, di mana setiap kategori akan diklasifikasikan secara terpisah. Pendekatan ini berguna untuk menghindari ketergantungan antar kategori dan memungkinkan model menangani setiap kategori secara mandiri, yang dapat meningkatkan akurasi dan pemahaman model pada struktur data yang ada.

Oleh karena itu, penelitian ini secara khusus diarahkan untuk memberikan kontribusi dalam penyempurnaan *dataset* Hadits Bukhari melalui proses pelengkapan data hadits sehingga data memuat *sanad* (rangkaian perawi) serta *matan* (isi hadits) secara lengkap. Hal ini yang membedakan penelitian ini dari penelitian sebelumnya [6], yang hanya menggunakan bagian *matan* hadits pendekatan klasifikasi teks. Selain itu, penelitian ini juga membandingkan performa algoritma RF dan LSTM, serta mengevaluasi hasil klasifikasi menggunakan metrik akurasi, *F1-Score*, dan *Hamming Loss* guna memperoleh gambaran yang lebih komprehensif mengenai kualitas prediksi.

2. METODOLOGI PENELITIAN

Penelitian ini melakukan perbandingan klasifikasi terhadap *dataset* hadits Bukhari menggunakan dua pendekatan, yaitu RF dan LSTM. Tahapan-tahapan dalam proses klasifikasi dapat dilihat pada Gambar 1.



Gambar 1. Alur Penelitian Klasifikasi Hadits Bukhari

2.1. *Dataset*

Dataset hadits Bukhari yang diterapkan pada penelitian merupakan dataset sekunder berupa kumpulan hadits Shahih Bukhari dalam terjemahan bahasa Indonesia. Setiap hadits dalam *dataset* ini telah dilabeli oleh penelitian sebelumnya, yaitu Muhammad Yuslan Abu Bakar [6], yang melakukan pelabelan dengan pendekatan berbasis analisis isi matan. Pada pendekatan ini, setiap hadits dianalisis maknanya untuk menentukan apakah hadits tersebut berisi ajakan (anjuran), larangan terhadap suatu perbuatan, atau sekadar menyampaikan informasi. Pelabelan tersebut kemudian divalidasi oleh ulama untuk memastikan kesesuaian kategori dengan kaidah keislaman. Pada Tabel 1 merupakan representasi data hadits Bukhari multi-label yang diperoleh dari peneliti sebelumnya.

Tabel 1. Representasi *Dataset* Terjemahan Hadits Bukhari [6]

Hadits	Kategori		
	Anjuran	Larangan	Informasi
Barangsiapa tidak mengasihi maka ia tidak akan dikasihi.	1	0	1
Sesungguhnya Allah melaknat orang yang menyambung rambutnya dan yang minta disambung.	0	1	1

Tabel 1 menunjukkan representasi struktur *dataset* terjemahan Hadits Bukhari sebelum dilakukan pelengkapan hadits. Setiap entri terdiri dari teks hadits dan tiga kolom kategori, yaitu anjuran, larangan, dan informasi, yang merepresentasikan pendekatan *multi-label*. Angka "1" mengisyaratkan bahwa hadits bersangkutan masuk dalam golongan itu. Kebalikannya, angka "0" memberi tahu bahwa hadits itu tidak termasuk dalam kategorinya. Penelitian ini menggunakan tiga kelas kategori dalam pelabelan hadits, yaitu anjuran, larangan, dan informasi. *Dataset* ini terdiri dari 7000 data, dengan distribusi data tergolong *imbalanced*. Rincian dari data tersebut ditunjukkan pada Tabel 2.

Tabel 2. Distribusi Data Hadits Bukhari

Kelas	Kategori		
	Anjuran	Larangan	Informasi
1	1.335	844	6.634
0	5.665	6.156	366

Pada Tabel 2, dapat dilihat bahwa kategori informasi memiliki jumlah data hadits yang lebih banyak dibandingkan dengan kategori anjuran dan larangan. Sementara itu, kategori larangan memiliki jumlah data hadits yang lebih sedikit. Ketidakseimbangan jumlah data antar kategori dapat memengaruhi hasil klasifikasi, di mana model cenderung lebih memilih kategori dengan jumlah data yang lebih besar. Sehingga, penting untuk mempertimbangkan teknik penyeimbangan data, seperti *oversampling* atau *undersampling*, dalam proses pelatihan model untuk mengatasi masalah distribusi data yang tidak seimbang ini.

2.2. Proses Kontribusi Terhadap *Dataset* Hadits Bukhari

Kontribusi terhadap *dataset* hadits Bukhari ini dilakukan dengan melengkapi bagian *matan* (isi) hadits menjadi utuh dan lengkap. Proses melengkapi hadits dilakukan dengan menggabungkan data dari sumber tambahan yang diperoleh dari peneliti [17]. Setelah itu, dilakukan pencocokan antara potongan hadits dan teks lengkapnya menggunakan metode otomatis dan manual. Pada tahapan otomatis, digunakan pola pencarian teks dengan *Regex* (*Regular Expression*) untuk menemukan bagian-bagian yang cocok dari teks hadits scenario awal dengan teks hadits lengkap. Jika ada bagian yang belum terdeteksi secara otomatis, penyesuaian dilakukan secara manual. Pencocokan ini menjadi tahap awal dalam proses pelengkapan hadits dari versi sebelumnya. Pada Tabel 3 merupakan perbandingan antara data hadits yang dipereroleh dari peneliti sebelumnya [6] dan data hadits setelah melalui proses perlengkapan.

Tabel 3. Perbandingan hadits sebelum dan sesudah dilengkapi

Teks Hadits Awal [6]	Teks Hadits Versi Lengkap [17]
Sesungguhnya Allah melaknat orang yang menyambung rambutnya dan yang minta disambung.	Telah menceritakan kepada kami [Al Humaidi] telah menceritakan kepada kami [Sufyan] telah menceritakan kepada kami [Hisyam] bahwa dia mendengar [Fathimah binti Mundzir] berkata; saya mendengar [Asma'] berkata; seorang wanita bertanya kepada Nabi shallallahu 'alaihi wasallam katanya; "Wahai Rasulullah, sesungguhnya puteriku menderita penyakit gatal (cacar) hingga rambutnya rontok, sementara saya hendak menikahkannya, apakah saya boleh menyambung rambutnya? Beliau bersabda: "Sesungguhnya Allah melaknat orang yang menyambung rambutnya dan yang minta disambung."

Tabel 3 memperlihatkan perbandingan antara teks hadits sebelum dan sesudah proses pelengkapan data. Pada kolom "Hadits Awal" ditampilkan hadits yang diperoleh dari peneliti sebelumnya [6], yang mana berisi matannya saja, sedangkan pada kolom "Hadits Sesudah Dilengkapi" ditampilkan teks hadits yang sudah disempurnakan dari sumber tambahan. Dengan adanya pelengkapan ini, hadits menjadi lebih utuh, memuat sanad (rangkaian perawi) serta matan (isi hadits) secara lengkap, sehingga memperjelas konteks dan makna hadits tersebut.

2.3. *Split Dataset*

Setelah proses pelengkapan dan pencocokan data hadits dilakukan, tahap berikutnya adalah menyiapkan data untuk pelatihan model. Data yang diperoleh melalui proses kontribusi kemudian dibagi menjadi tiga *subset*, yaitu data latih, data validasi, dan data uji. Dari total 7.000 data, pembagian dilakukan dengan proporsi 70% untuk data latih, 10% untuk data validasi, dan 20% untuk data uji, tanpa menggunakan variasi pembagian lainnya. Hal ini dilakukan untuk menjaga konsistensi serta memudahkan evaluasi performa model. Dengan pembagian tersebut, diperoleh sekitar 5.040 data latih, 560 data validasi, dan 1.400 data uji. Pembagian ini bertujuan memastikan model dapat dilatih secara maksimal, divalidasi secara objektif, dan diuji untuk mengukur performa pada data yang belum pernah digunakan sebelumnya.

2.4. *Balancing Class*

Balancing Class merupakan langkah penting dalam menangani masalah ketidakseimbangan kelas pada *dataset*. Jika tidak diterapkan teknik ini, kelas yang dominan cenderung lebih banyak diprediksi, sementara kelas minoritas sering terabaikan. Dalam hal ini, teknik *balancing class* yang digunakan adalah *Random Oversampling* (ROS), di mana sampel dari kelas minoritas diperbanyak dengan melakukan duplikat secara acak pada data hadits untuk mencapai distribusi kelas yang lebih seimbang. Berlandaskan pada penelitian [18], ROS menunjukkan kinerja terbaik dalam menyeimbangkan data pendidikan dengan *F1-Score* tertinggi. Sedangkan pada penelitian [19], menerapkan ROS pada data kelulusan mahasiswa, yang menghasilkan prediksi ketepatan waktu lulus dengan akurasi 90.04% menggunakan algoritma *Random Forest*. Hal ini menunjukkan bahwa

menerapkan teknik penyeimbangan ROS dapat meningkatkan performa model dalam menangani ketidakseimbangan kelas dan menghasilkan prediksi yang lebih akurat pada berbagai jenis data.

2.5. *Text Preprocessing*

Pra-pemrosesan teks adalah langkah awal yang penting dalam pengolahan data berbasis teks, yang bertujuan untuk menyederhanakan struktur data agar dapat dikenali dan diolah secara efisien oleh algoritma pembelajaran mesin. Melalui proses ini, gangguan atau noise pada data dapat diminimalkan [6], serta membantu dalam mentransformasi teks yang awalnya tidak terstruktur menjadi lebih terstruktur [20]. Adapun tahapan-tahapan dalam *preprocessing* teks dijelaskan sebagai berikut:

1. *Case Folding*: Dalam proses case folding akan mengubah setiap kata menjadi huruf kecil untuk menjadikan teks seragam.
2. *Cleaning*: Proses ini meliputi penghapusan tanda baca, angka, dan karakter yang tidak relevan.
3. *Stopword Removal*: tahapan ini dilakukan untuk menyaring dan menghapus kata-kata umum yang tidak memiliki kontribusi berarti terhadap pemahaman konteks informasi.
4. *Stemming*: Stemming akan mengubah setiap kata menjadi bentuk awal.
5. *Tokenizing*: Proses untuk memecah teks menjadi kata atau frasa.
6. *Sequence*: Proses mengubah setiap hasil tokenisasi menjadi angka.
7. *Padding*: Proses menyesuaikan panjang input teks agar memiliki panjang yang sama.

Tahapan tokenisasi, *sequene* dan *padding* hanya dilakukan pada metode LSTM, dikarenakan model ini mengharuskan input teks memiliki panjang yang konsisten untuk mendukung pemrosesan yang terstruktur dalam urutan. Setelah melalui tahapan-tahapan tersebut, data teks akan menjadi lebih bersih, terstruktur, dan siap untuk melewati proses vektorisasi.

2.6. *Vectorizer*

Pemrosesan ekstraksi fitur, yang dikenal juga sebagai representasi kata, sangat penting dalam mengonversi teks menjadi bentuk numerik, sehinggadapatdiprosesdan dimengertioleh algoritma pembelajaran mesin.. Dalam penelitian ini, pendekatan *Term Frequency-Inverse Document Frequency* (TF-IDF) dimanfaatkan untuk mengonversi kata-kata ke dalam bobot angka berdasarkan frekuensi kemunculannya di suatu dokumen serta kelangkaannya di seluruh kumpulan dokumen. Teknik ini memungkinkan model untuk mengenali kata-kata yang relevan dalam konteks klasifikasi teks. Penelitian [20] menyatakan bahwa, TF-IDF terbukti memberikan kinerja yang sangat baik dalam mengklasifikasikan teks *multi-label*. Di sisi lain, pada algoritma *Deep Learning*, proses dimulai dengan tahapan *Tokenizer*, *Sequences*, dan *Padding* dalam *Preprocessing*, yang kemudian dilanjutkan dengan *Word Embedding* bawaan dari LSTM pada tahap pemodelan. Adapun rumus pada TF-IDF dapat ditunjukkan pada persamaan 1-3.

$$TF\ (i,j) = \frac{Jumlah\ kemunculan\ kata\ i\ dalam\ dokumen\ j}{Total\ katta\ dalam\ dokumen\ j} \tag{1}$$

$$IDF(i,j) = log\ \frac{Jumlah\ dokumen}{Jumlah\ dokumen\ yang\ mengandung\ kata\ i} \tag{2}$$

$$TF - IDF = TF\ x\ IDF \tag{3}$$

Term Frequency (TF) merepresentasikan tingkat kemunculan kata tertentu di dalam satu dokumen, sedangkan *Document Frequency* (DF) menghitung dokumen yang mengandung kata tersebut. Nilai TF-IDF diperoleh dengan mengalikan keduanya, sehingga kata yang dominan dalam satu dokumen tapi amat jarang muncul di dokumen lainnya akan mempunyai bobot TF-IDF yang besar dan dianggap relevan.

2.7. *Random Forest* (RF)

RF merupakan algoritma *ensemble learning* yang tergolong dalam metode *supervised learning* dan dikembangkan oleh Leo Breiman [14]. Algoritma ini membangun dan menggabungkan banyak pohon keputusan (*decision tree*) dalam melakukan klasifikasi. Algoritma ini menerapkan teknik bagging sebagai pendekatan utama, yaitu pelatihan beberapa pohon menggunakan sampel data acak yang dipilih kembali (*bootstrap sampling*). Pendekatan ini membuat RF lebih tangguh terhadap *overfitting*. Dalam penelitian ini, RF akan di evaluasi dengan beberapa model, yaitu dengan parameter *Default* dari RF itu sendiri dan dengan *tuning hyperparameter* dengan menggunakan *RandomSearchCV*. Beberapa *hyperparameter* yang digunakan dalam *tuning*, yaitu *n_estimators*, *max_features*, *max_depth*, dan *criterion*. Beberapa penelitian sebelumnya telah menunjukkan efektivitas RF dalam tugas klasifikasi teks, termasuk di domain keagamaan. Seperti pada penelitian [21], yang menggunakan RF untuk klasifikasi multi-label ayat Al-Qur'an dalam kategori tematik menunjukkan bahwa RF mampu menghasilkan akurasi tinggi dan efektif menangani data teks yang kompleks. Selain itu, penelitian [22] menggunakan RF untuk klasifikasi hadis Shahih Al-Bukhari ke dalam kategori seperti

anjuran, larangan, dan informasi, dan melaporkan hasil yang kompetitif dibandingkan dengan metode lain. Keunggulan RF dalam menangani data yang kompleks dan beragam serta kemampuannya dalam memberikan interpretasi pentingnya fitur menjadikannya pilihan yang relevan dalam penelitian ini, terutama dalam menangani data hadits yang kaya akan konteks dan makna.

2.8. *Long Short-Term Memory* (LSTM)

LSTM merupakan algoritma pada pembelajaran mendalam yang dibangun dari struktur RNN. Keunggulan utama LSTM terletak pada keberadaan sel memori (*memory cell*), yang memungkinkan model untuk mengingat informasi dalam rentang waktu yang panjang, sehingga efektif dalam menangani dependensi jangka panjang pada data berurutan. Algoritma ini banyak diterapkan di ranah pemrosesan bahasa alami (*Natural Language Processing*/NLP), seperti dalam tugas penerjemahan teks, pengenalan suara, serta klasifikasi baik pada data teks maupun citra [23]. Dalam struktur LSTM, tiap unit memorinya dibekali dengan tiga komponen pengatur aliran informasi utama, yakni gerbang masuk, gerbang lupa, dan gerbang keluar, yang bekerja sama untuk menjaga serta mengelola informasi sepanjang urutan data. Sama halnya dengan metode RF, LSTM juga dievaluasi menggunakan dua model. Model pertama menggunakan parameter *default*, di mana jumlah unit LSTM (lstm_units) diset sebesar 100 *neuron*. Sedangkan model kedua menggunakan *hyperparameter tuning* yang terdiri dari *lstm_units*, *dropout*, *learning rate*, dan *dense_units* untuk mencari konfigurasi yang optimal. Pada LSTM ini di set dengan *batch_size* sebesar 32 dan *epoch* sebanyak 10, setiap model pada LSTM akan menggunakan *Early Stopping* untuk menghindari *overfitting* dan mempercepat proses *training*. Penelitian terkini mendukung penggunaan LSTM untuk klasifikasi teks, khususnya dalam konteks keagamaan. Penelitian [9], [24], mengaplikasikan LSTM untuk klasifikasi teks Al-Qur'an dengan performa yang baik dalam mengelola konteks panjang pada ayat-ayat yang kompleks.

2.9. Pemodelan dan Validasi

Setelah melalui tahapan *text preprocessing* dan *vectorization*, proses dilanjutkan dengan pelatihan model dengan menggunakan 5.060 data hadits. Dalam hal ini, setiap proses dilakukan dengan berbagai model, baik itu dengan parameter *default* maupun dengan *tuning hyperparameter*. Kemudian, dilakukan proses validasi untuk mengevaluasi performa berbagai model yang telah dilatih. Data validasi diambil sebesar 10% dari data pelatihan. Evaluasi pada model dilakukan dengan metrik *Accuracy* dan *F1-Score*. Selanjutnya, model akan dievaluasi dengan data uji untuk mengukur performanya pada data yang belum pernah diproses sebelumnya.

2.10. *Testing* dan Evaluasi

Tahapan ini melibatkan pengujian model menggunakan data uji sebanyak 1.400 hadits untuk mengukur performa klasifikasi terhadap data yang tidak dikenali. Selanjutnya, performa model dievaluasi menggunakan tiga metrik utama, yaitu *accuracy*, *F1-Score*, dan *Hamming Loss* (HL). Evaluasi berbasis *accuracy* dan *F1-score* difokuskan pada perbandingan performa antar model dalam setiap kategori klasifikasi, untuk melihat model mana yang memberikan hasil terbaik pada masing-masing kategori. Sementara itu, evaluasi HL dilakukan untuk menilai tingkat kesalahan klasifikasi secara keseluruhan pada setiap kategori, dengan mempertimbangkan model yang menggunakan skema *preprocessing* yang sama. Rumus evaluasi ditunjukkan pada persamaan 4-6 [25].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$F1 - Score = 2x\frac{Precision\ x\ Recall}{Precision + Recall} \tag{2}$$

$$Hamming\ Loss = \frac{Jumlah\ label\ yang\ salah}{Jumlah\ data\ x\ Jumlah\ kategori} \tag{3}$$

3. HASIL DAN PEMBAHASAN

Penelitian ini memanfaatkan bahasa pemrograman *Python* dalam memproses data terjemahan Hadits Bukhari. Terdapat dua variasi proses yang digunakan, yaitu skenario Awal dan Optimal. Pada variasi pertama (Awal), tahapan klasifikasi dilakukan tanpa menggunakan teknik *balancing* dan hanya mengguanakan parameter *default*. Sementara itu, pada variasi kedua (Optimal), proses klasifikasi di evaluasi dengan menggunakan teknik balancing, serta mencakup penggunaan parameter default dan hyperparameter untuk mengoptimalkan hasil klasifikasi.

3.1. Model Awal

Model awal ini, percobaan dilakukan tanpa teknik *balancing* pada masing-masing metode, yaitu RF dan LSTM, serta tidak ada pengaturan parameter khusus. Hasil dari percobaan tersebut disajikan pada Tabel 4:

Tabel 4. Hasil Model Awal Metode RF pada Pelatihan dan Validasi

Kategori	ST	SP	*Train*		*Validation*	
			Accuracy	*F1-Score*	*Accuracy*	*F1-Score*
Anjuran	*No*	*Yes*	99.98%	99.97%	81.43%	61.68%
Larangan	*No*	*No*	100%	100%	93.39%	78.72%
Informasi	*No*	*Yes*	97.60%	78.87%	96.43%	49.09%

Pada Tabel 4, menyajikan hasil pemodelan awal menggunakan metode *Random Forest* pada data *training* dan data validasi. Berdasarkan tabel tersebut, terlihat bahwa pemodelan dengan menggunakan parameter *default* menghasilkan *accuracy* dan *F1-Score* yang tinggi pada data pelatihan. Namun, ketika pemodelan di uji menggunakan data validasi, performa model mengalami penurunan. Untuk mengevaluasi lebih lanjut generalisasi model, metode *Random Forest* (RF) juga diuji menggunakan data uji (*test*). Hasil pengujian tersebut disajikan pada Tabel 4 sebagai bagian dari analisis pemodelan awal menggunakan algoritma RF.

Tabel 5. Hasil Model Awal Metode RF pada Data Uji

Kategori	*Data Test*	
	Accuracy	*F1-Score*
Anjuran	84.93%	59.78%
Larangan	93.21%	83.38%
Informasi	90.14%	47.41%
Rata-Rata	89.43%	63.52%

Didasarkan pada Tabel 5, metode RF menunjukkan akurasi tinggi pada data uji untuk ketiga kelas, dengan rentang antara 84% hingga 93% dan rata-rata sebesar 89.43%. Namun, nilai *F1-Score* pada kategori anjuran dan informasi relatif rendah. Selanjutnya disajikan hasil pemodelan awal menggunakan algoritma LSTM pada tabel 6.

Tabel 6. Hasil Model Awal Metode LSTM pada Pelatihan dan Validasi

Kategori	ST	SP	*Train*		*Validation*	
			Accuracy	*F1-Score*	*Accuracy*	*F1-Score*
Anjuran	*No*	*No*	97.58%	96.09%	81.61%	68.53%
Larangan	*Yes*	*No*	97.12%	93.15%	94.64%	81.10%
Informasi	*Yes*	*No*	97.58%	81.94%	96.96%	65.22%

Tabel 6 menunjukkan hasil dari model awal pada meode LSTM menggunakan parameter *default* (*units LSTM* = 100, *batch size* = 32, dan *epoch* = 10). Nilai *accuracy* dan *F1-Score* pada metode LSTM cenderung lebih rendah dibandingkan dengan RF, baik pada data pelatihan maupun data validasi. Hal ini menunjukkan bahwa pada kondisi model awal, performa LSTM belum optimal dalam memproses data hadits dengan parameter *default*. Setelah model diperoleh dari proses pelatihan, model tersebut akan di uji lebih lanjut menggunakan data uji. Tabel 7 merupakan Tabel yang menyajikan hasil evaluasi model awal pada data uji.

Tabel 7. Hasil Model Awal LSTM pada Data *Test*

Kategori	*Data Test*	
	Accuracy	*F1-Score*
Anjuran	83.93%	66.41%
Larangan	92.64%	82.63%
Informasi	90.57%	59.12%
Rata-Rata	89.05%	69.39%

Tabel 7 menyajikan hasil akurasi pada data uji yang diperoleh melalui pemodelan menggunakan metode LSTM. Berdasarkan hasil tersebut, akurasi tertinggi dicapai pada kategori larangan sebesar 92.64% dan informasi sebesar 90.57%. Namun, pada kategori anjuran, metode LSTM masih menunjukkan akurasi yang sedikit rendah.

3.2. Model Optimal

Dalam proses pengembangan model yang optimal, akan dilakukan eksplorasi dengan proses *balancing* untuk meningkatkan kualitas data latih. Model tidak hanya dilatih menggunakan parameter *default*, tetapi juga dilakukan optimasi *tuning hyperparameter* menggunakan *RandomizedSearchCV*. *RandomizedSearchCV* digunakan karena teknik *tuning* yang memiliki waktu yang lebih singkat dalam proses optimasi [19]. Parameter *default* tetap digunakan pada data hadits untuk membandingkan performa model antara kondisi awal dan

kondisi setelah dilakukan optimasi serta penyeimbangan data. Tabel 8 merupakan hasil dari model optimal pada metode *Random Forest*.

Tabel 8. Hasil Optimal RF pada Pelatihan dan Validasi

Kategori	Parameter Optimal	BL	ST	SP	*Train*		*Validation*	
					Accuracy	*F1-Score*	*Accuracy*	*F1-Score*
Anjuran	*Default*	*Yes*	*No*	*No*	99.99%	99.99%	82.86%	71.23%
Larangan	*Default*	*Yes*	*No*	*No*	100%	100%	93.39%	80.88%
Informasi	*n_estimator* = 200, *max_features* = log2, *max_depth* = 50, dan *criterion* = *gini*	*Yes*	*No*	*Yes*	99.99%	99.99%	95.89%	61.85%

Keterangan: BL = *Balancing*, ST = *Stemming*, SP = *Stopwords Removal*. Ketiga elemen ini merupakan bagian dari tahapan praproses data yang digunakan untuk meningkatkan performa model. Hasil optimal metode *Random Forest* yang ditampilkan pada Tabel 8 menunjukkan peningkatan performa yang sangat signifikan. Menariknya, parameter *default* pada *Random Forest* terbukti sudah cukup efektif dalam proses pelatihan data hadits yang sudah dilengkapi, terutama pada kategori anjuran dan larangan hingga mencapai akurasi 100%. Sementara itu, performa terbaik pada kategori informasi justru diperoleh melalui optimasi *hyperparameter* menggunakan *RandomizedSearchCV*.

Selanjutnya, hasil evaluasi pada data uji untuk menilai sejauh mana performa model yang telah dilatih dapat digeneralisasikan terhadap data yang belum dikenali sebelumnya.

Tabel 9. Hasil Optimal RF pada Data Uji

Kategori	*Data Test*	
	Accuracy	*F1-Score*
Anjuran	85.86%	69.03%
Larangan	93.21%	84.45%
Informasi	89.36%	58.14%
Rata-Rata	89.48%	70.54%

Berdasarkan Tabel 9, hasil optimal pada metode RF pada data uji menunjukkan performa yang bervariasi antar kategori. Kategori larangan memiliki performa tertinggi dengan akurasi sebesar 93.21% dan *F1-Score* sebesar 84.45%, diikuti oleh kategori informasi dengan akurasi 89.36%, namun *F1-Score*-nya relatif rendah, yaitu 58.14%. Sementara itu, kategori anjuran memperoleh akurasi sebesar 85.86% dan *F1-Score* sebesar 69.03%. Hal ini menunjukkan bahwa meskipun akurasi pada semua kategori tergolong tinggi, nilai *F1-Score*, terutama pada kategori informasi, masih perlu ditingkatkan.

Untuk melihat performa metode lain, dilakukan pelatihan model menggunakan arsitektur pada metode LSTM. Proses ini dilakukan untuk medapatkan hasil yang optimal dibandingkan dengan model awal yang diperoleh dari LSTM sebelumnya. Tabel 10 menyajikan hasil evaluasi model LSTM pada data pelatihan dan data validasi pada model yang optimal.

Tabel 10. Hasil Optimal LSTM pada Pelatihan dan Validasi

Kategori	Parameter Optimal	BL	ST	SP	*Train*		*Validation*	
					Accuracy	*F1-Score*	*Accuracy*	*F1-Score*
Anjuran	*Default* (*units LSTM* = 100, *epoch* = 10, dan *bacth size* = 32	*Yes*	*No*	*No*	94.15%	94.15%	81.25%	73.49%
Larangan	*units* LSTM = 128, *dropout* = 0.3, *units dense* = 256, *learning rate* = 0.01, *batch size* = 32, dan *epoch* = 10	*No*	*Yes*	*No*	98.12%	95.52%	94.11%	83.23%
Informasi	*Default* (*units LSTM* = 100, *epoch* = 10, dan *bacth size* = 32	*Yes*	*No*	*No*	99.91%	99.91%	96.25%	62.83%

Pada Tabel 10, hasil dari kategori informasi menunjukkan akurasi tertinggi dibandingkan kategori lainnya, yaitu sebesar 99.91% pada *training* dan 96.2% pada validasi. Ini mengindikasikan bahwa metode LSTM memiliki performa yang sangat baik dalam proses pembelajaran pada kategori informasi, baik dalam

proses pelatihan model maupun saat evaluasi menggunakan data validasi. Namun, jika dilihat dari *F1-Score* justru kategori informasi memiliki nilai yang sangat rendah dibandingkan kategori anjuran dan larangan.

Setelah model LSTM menunjukkan performa yang sangat baik pada data pelatihan dan validasi, evaluasi selanjutnya dilakukan pada data uji untuk melihat kemampuan model untuk mengenali pola pada data yang baru pertama kali dihadapi. Tabel 11 menyajikan hasil evaluasi metode LSTM pada data uji untuk setiap kategori.

Tabel 11. Hasil Optimal LSTM pada Data Uji

Kategori	*Data Test*	
	Accuracy	*F1-Score*
Anjuran	82.64%	69.99%
Larangan	92.71%	84.04%
Informasi	90.21%	64.27%
Rata-Rata	88.52%	72.77%

Tabel 11 menunjukkan kategori larangan meemiliki akurasi tertinggi pada data uji, yaitu sebesar 92,71%, dan juga memiliki *F1-Score* paling tinggi di antara semua kelas, yaitu 84,04%. Temuan ini menunjukkan bahwa model LSTM dapat mendeteksi pola pada kategori larangan secara akurat, meskipun diuji dengan data yang belum pernah ada sebelumnya. Sementara itu, kategori informasi memang mencatat akurasi yang lebih tinggi dibanding anjuran, namun dari sisi *F1-Score*, justru anjuran yang lebih unggul. Ini mengindikasikan bahwa prediksi model untuk kategori anjuran lebih seimbang antara presisi dan *recall*, sedangkan pada kategori informasi, meskipun akurasinya tinggi, ketepatan dalam mengenali semua label belum optimal.

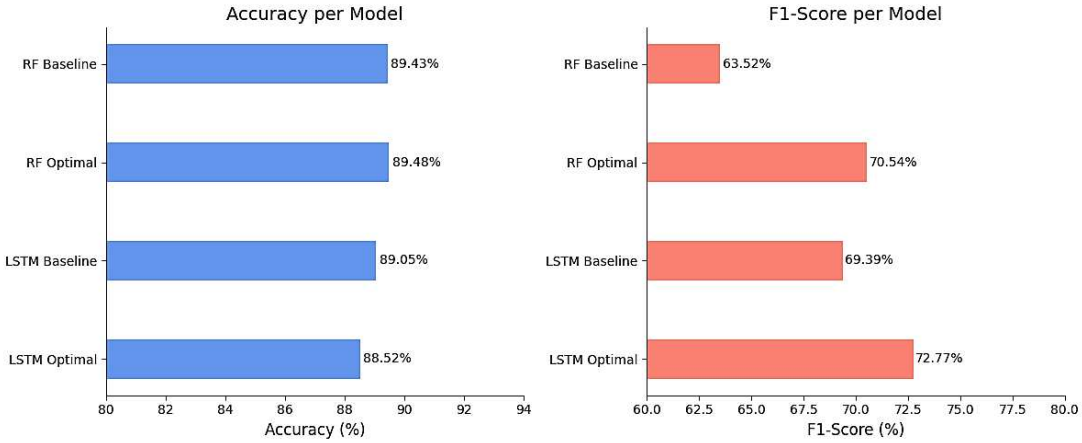
3.3. Perbandingan Performa RF dan LSTM

Berdasarkan hasil evaluasi pada tahap awal dan optimasi sebelumnya, bagian ini akan membahas perbandingan performa antara RF dan LSTM secara menyeluruh dalam mengklasifikasikan ketiga kategori. Perbandingan ini difokuskan melalui hasil evaluasi pada data uji, dengan mengacu pada metrik akurasi dan *F1-Score* dari masing-masing kategori (anjuran, larangan, dan informasi). Tujuan dari analisis ini adalah untuk mengetahui keunggulan masing-masing metode, serta melihat konsistensi performa model dalam menangani klasifikasi teks hadits pada setiap label yang diuji. Berikut disajikan hasil rata-rata dari performa setiap metode.

Tabel 12. Perbandingan hasil rata-rata performa klasifikasi RF dan LSTM

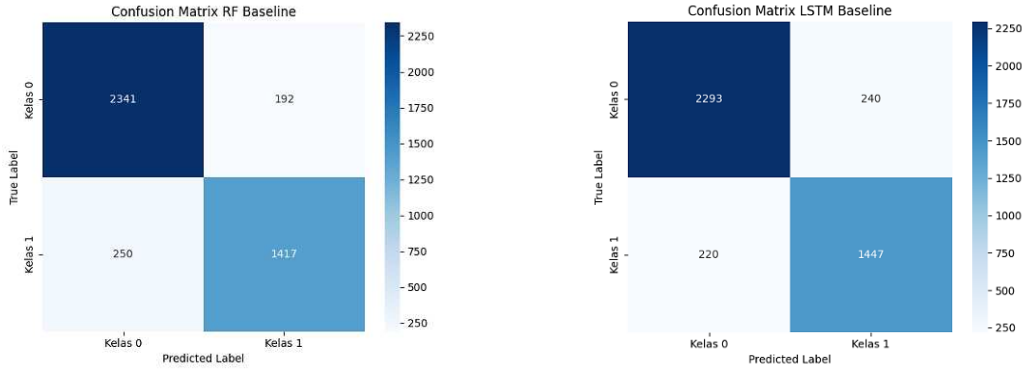
Model	Data *Test*	
	Accuracy	*F1-Score*
RF Awal	89.43%	63.52%
RF Optimal	89.48%	70.54%
LSTM Awal	89.05%	69.39%
LSTM Optimal	88.52%	72.77%

Berdasarkan analisis pada Tabel 12, terlihat bahwa akurasi pada metode RF, dalam kondisi awal maupun optimal, lebih unggul dibandingkan LSTM. Tetapi, jika kita melihat dari sisi *F1-Score* LSTM optimal lebih unggul dengan rata-rata 72.77%. Untuk mendapatkan gambaran yang lebih menyeluruh terkait performa model, ditampilkan pula visualisasi berupa grafik dari masing-masing metode, baik pada metode *Random Forest* maupun LSTM dalam kondisi awal dan optimal pada Gambar 2.

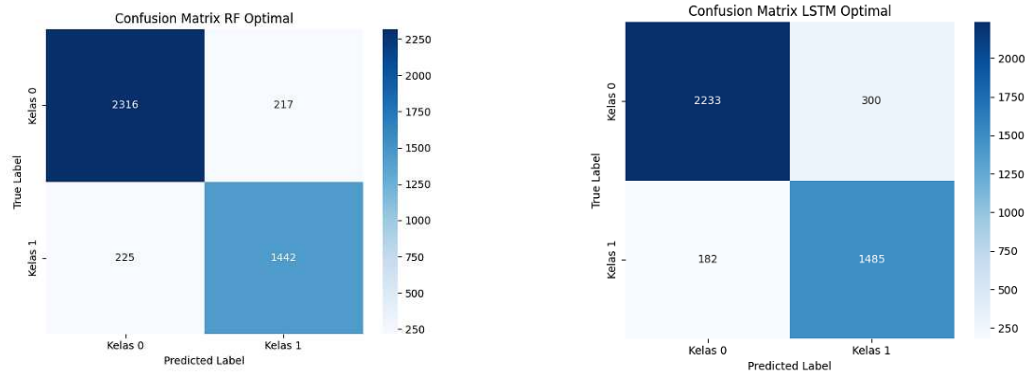


Gambar 2. Grafik Perbandingan RF dan LSTM pada model Awal dan Optimal

Gambar 3 menampilkan perbandingan performa antara model RF dan LSTM dalam dua skenario, yaitu skenario awal dan optimal melalui metrik akurasi dan *F1-Score*. Secara umum, grafik akurasi menunjukkan bahwa keempat model memiliki nilai yang cukup berdekatan, dengan RF optimal mencatat akurasi tertinggi sebesar 89,48%, diikuti oleh RF model awal (89,43%), LSTM model awal (89,05%), dan LSTM optimal (88,52%). Perbedaan yang lebih mencolok tampak pada metrik *F1-Score* pada kedua model dengan mengalami peningkatan yang signifikan. Peningkatan terbesar terlihat pada model LSTM, dari 69,39% (model awal) menjadi 72,77% (optimal), sementara RF meningkat dari 63,52% menjadi 70,54%. Hasil ini menunjukkan bahwa skenario optimal dengan menambahkan beberapa variasi model seperti *balancing class*, dan *hyperpameter tuning* mampu meningkatkan kemampuan model dalam mengklasifikasikan data secara lebih seimbang ke dalam kategori anjuran, larangan, dan informasi. Berikut merupakan confusion matrix pada klasifikasi hadits Bukhari dengan metode RF dan LSTM.



Gambar 3. Confusion Matrix Model Awal Pada RF dan LSTM



Gambar 4. *Confusion Matrix* Optimal Pada RF dan LSTM

Gambar 3 dan Gambar 4 memperlihatkan *confusion matrix* pada model RF dan LSTM pada skenario awal dan optimal. Distribusi ini memiliki jumlah keseluruhan 4.200 data karena proses klasifikasi dilakukan secara independen. Pada skenario awal, RF berhasil mengklasifikasikan 2.341 data Kelas 0 dan 1.417 data Kelas 1 dengan benar, sedangkan LSTM mengklasifikasikan 2.293 data Kelas 0 dan 1.447 data Kelas 1. Sedangkan pada skenario optimal performa kedua model meningkat, terutama pada model LSTM yang mencatat peningkatan klasifikasi benar pada Kelas 1 menjadi 1.485 data, sementara RF optimal mencatat 1.422 data Kelas 1. Sehingga, dalam skenario ini kesalahan klasifikasi juga cenderung menurun pada kedua mode. Secara keseluruhan, visualisasi ini menunjukkan bahwa dengan beberapa variasi yang ditambahkan pada proses optimal mampu meningkatkan ketepatan klasifikasi, terutama dalam menyeimbangkan prediksi antar kelas, dengan LSTM unggul dalam klasifikasi Kelas 1 dan RF menunjukkan stabilitas di kedua kelas (0 dan 1). Tabel 13 merupakan hasil prediksi dalam klasifikasi hadits pada metode RF.

Tabel 13. Hasil Prediksi Klasifikasi Hadits

Hadist	*True Label*			Metode	*Prediction*		
	A	L	I		PA	PL	PI
Telah menceritakan kepada kami [Al Humaidi] telah menceritakan kepada kami [Sufyan] telah menceritakan kepada kami [Hisyam] bahwa dia mendengar [Fathimah binti Mundzir] berkata; saya mendengar [Asma'] berkata; seorang wanita	0	1	1	RF	0	1	1
				LSTM	0	1	1

Hadist	*True Label*			Metode	*Prediction*		
	A	L	I		PA	PL	PI
bertanya kepada Nabi shallallahu 'alaihi wasallam katanya; "Wahai Rasulullah, sesungguhnya puteriku menderita penyakit gatal (cacar) hingga rambutnya rontok, sementara saya hendak menikahkannya, apakah saya boleh menyambung rambutnya? Beliau bersabda: "Sesungguhnya Allah melaknat orang yang menyambung rambutnya dan yang minta disambung."							
Telah menceritakan kepada kami [Abu Al Yaman] telah mengabarkan kepada kami [Syu'aib] dari [Az Zuhri] telah menceritakan kepada kami [Abu Salamah bin Abdurrahman] bahwa [Abu Hurairah] radliallahu 'anhu berkata; "Rasulullah shallallahu 'alaihi wasallam pernah mencium Al Hasan bin Ali sedangkan disamping beliau ada Al Aqra' bin Habis At Tamimi sedang duduk, lalu Aqra' berkata; "Sesungguhnya aku memiliki sepuluh orang anak, namun aku tidak pernah mencium mereka sekali pun, maka Rasulullah shallallahu 'alaihi wasallam memandangnya dan bersabda: "Barangsiapa tidak mengasihi maka ia tidak akan dikasihi."	1	0	1	RF	0	0	1
				LSTM	0	1	1

Keterangan: A (Anjuran), L (Larangan), I (Informasi), PA (Prediksi Anjuran), PL (Prediksi Larangan), dan PI (Prediksi Informasi). Pada tabel 13, ditampilkan hasil prediksi klasifikasi hadits menggunakan metode *Random Forest* (RF) dan *Long Short-Term Memory* (LSTM). Terlihat bahwa pada hadits pertama metode RF dan LSTM memprediksi semua kategori dengan tepat. Pada hadits kedua RF memprediksi 2 kategori dengan benar, yaitu pada kategori larangan dan informasi, sedangkan LSTM memprediksi dengan benar pada kategori anjuran dan informasi. Hal ini menunjukkan bahwa masing-masing model memiliki kekuatan yang berbeda dalam menangkap karakteristik dari setiap kategori. Selain itu, kita juga dapat menilai tingkat kesalahan klasifikasi secara keseluruhan dengan penggunaan *preprocessing* yang sama setiap kategori dengan *Hamming Loss* (HL). Pada metode RF, diperoleh HL 0.1060 (89.40%) pada skenario awal, 0.1048 (89.52%) pada optimal tanpa menggunakan *stopwords* dan *stemming*, sedangkan LSTM 0.1055 (89.45%) dan 0.1048 (89.52%) pada optimal tanpa menggunakan *stopwords* dan *stemming*. Hal ini menunjukkan bahwa penggunaan *stopwords* dan *stemming* tidak mengurangi kesalahan klasifikasi pada data hadits Bukhari lengkap.

Hasil tersebut mengindikasikan bahwa kedua metode, RF dan LSTM memiliki keunggulan masing-masing dalam mengidentifikasi kategori hadits, tergantung pada karakteristik fitur yang dominan dalam tiap kategori. Temuan ini sejalan dengan penelitian [6] yang menggunakan CRNN untuk klasifikasi hadits, di mana model tersebut menunjukkan keunggulan dalam menangkap pola spasial dan sekuensial pada teks hadits, serupa dengan kekuatan LSTM dalam memahami konteks urutan kata. RF cenderung lebih efektif dalam menangkap pola-pola yang lebih jelas dan terstruktur pada kategori larangan dan informasi, sementara LSTM lebih mampu mengenali konteks dan hubungan urutan kata yang berperan penting dalam kategori anjuran. Selain itu, nilai HL yang sama pada kedua metode, baik dengan maupun tanpa penerapan *stopwords* dan *stemming*, menunjukkan bahwa proses *preprocessing* tersebut tidak memberikan dampak signifikan terhadap performa model. Hal ini bisa disebabkan oleh karakteristik khusus teks hadits yang mengandung istilah-istilah penting dan makna kontekstual yang mungkin hilang atau berubah saat dilakukan penghapusan *stopwords* dan *stemming*. Oleh karena itu, teknik *preprocessing* yang lebih selektif atau penggunaan teknik representasi teks yang mempertahankan konteks secara lebih utuh dapat menjadi fokus pengembangan lebih lanjut untuk meningkatkan akurasi klasifikasi hadits.

4. KESIMPULAN

Berdasarkan hasil penelitian, model *Random Forest* (RF) memiliki sedikit keunggulan dalam akurasi dibandingkan dengan model *Long Short-Term Memory* (LSTM). Meskipun demikian, LSTM lebih unggul dalam menciptakan keseimbangan antara presisi dan *recall*, seperti yang terlihat pada nilai *F1-Score* tertinggi sebesar 72,77%, yang menandakan kemampuannya dalam mengatasi variasi kategori pada klasifikasi multi-label. Hal ini memperlihatkan bahwa LSTM lebih andal dalam menangani variasi kategori pada klasifikasi multi-label. Di sisi lain, nilai *Hamming Loss* yang sama pada kedua metode mengindikasikan bahwa rata-rata jumlah kesalahan label per *instance* relatif serupa. Penggunaan *stopwords removal* dan *stemming* dalam proses *preprocessing* tidak selalu memberikan dampak positif terhadap hasil klasifikasi hadits Bukhari. Sebaliknya, penerapan teknik *balancing class* terbukti mampu meningkatkan performa klasifikasi. Selain itu, kontribusi data pada hadits Bukhari juga berpengaruh positif terhadap hasil klasifikasi, di mana akurasi tertinggi sebesar 89,48% berhasil dicapai oleh metode RF, yang lebih tinggi dibandingkan penelitian sebelumnya [6] dengan akurasi sebesar 80,79%.

REFERENSI

[1] M. A. H. Muhammad, M. I. Afkarina, S. S. Shalsabila, and S. Fikri, "Hadist Ditinjau Dari Kualitas Sanad Dan Matan (Hadist Shohih, Hasan, Dhoif)," *Jurnal Kajian Islam dan Sosial Keagamaan*, vol. 1, no. 4, pp. 396–401, 2024, Accessed: Jun. 03, 2025. [Online]. Available: https://jurnal.ittc.web.id/index.php/jkis/article/view/1103

[2] R. Kustiawan, A. Adiwijaya, and M. D. Purbolaksono, "A Multi-label Classification on Topic of Hadith Verses in Indonesian Translation using CART and Bagging," *Jurnal Media Informatika Budidarma*, vol. 6, no. 2, pp. 868–875, Apr. 2022, doi: 10.30865/mib.v6i2.3787.

[3] H. M. Evan, "Analisis Deskriptif Kitab Shahih Al-Bukhari," *JIQTA: Jurnal Ilmu Al-Qur'an dan Tafsir*, vol. 1, no. 1, pp. 19–34, 2022, doi: https://doi.org/10.36769/jiqta.v1i1.187.

[4] R. S. Mutaqin, Z. Nurpadilah, and H. Z. Muttaqin, "Perawi Mudallis dalam Shahih Bukhari: Studi al-Jarh wa al-Ta'dil pada 'Umar bin 'Ali bin 'Atha' bin Muqaddam," *Riwayah : Jurnal Studi Hadis*, vol. 7, no. 2, pp. 241–272, Dec. 2021, doi: 10.21043/riwayah.v7i2.10651.

[5] N. Fatiara, N. H. Safaat, S. Agustian, and I. Afrianty, "Komparasi Metode K-Nearest Neighbors Dan Long Short Term Memory," *ZONASI: Jurnal Sistem Informasi*, vol. 6, no. 2, pp. 332–345, 2024.

[6] M. Y. A. Bakar and Adiwijaya, "Klasifikasi Teks Hadis Bukhari Terjemahan Indonesia Menggunakan Recurrent Convolutional Neural Network (CRNN)," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 8, no. 5, pp. 907–918, 2021, doi: 10.25126/jtiik.202183750.

[7] D. C. Hidayati, S. Al Faraby, and A. Adiwijaya, "Klasifikasi Topik Multi Label pada Hadis Shahih Bukhari Menggunakan K-Nearest Neighbor dan Latent Semantic Analysis," *JURIKOM (Jurnal Riset Komputer)*, vol. 7, no. 1, p. 140, Feb. 2020, doi: 10.30865/jurikom.v7i1.2013.

[8] F. Diba, M. S. Lydia, and P. Sihombing, "Analisis Random Forest Menggunakan Principal Component Analysis Pada Data Berdimensi Tinggi," *Indonesian Journal of Computer Science*, vol. 12, no. 4, pp. 2152–2160, 2023, doi: https://doi.org/10.33022/ijcs.v12i4.3329.

[9] D. P. Aftari, N. H. Safaat, S. Agustian, and I. Afrianty, "Perbandingan Performa Klasifikasi Terjemahan Al-Qur'an Menggunakan Metode Random Forest dan Long Short Term Memory," *Journal of Computer System and Informatics (JoSYC)*, vol. 5, no. 3, pp. 567–577, 2024, doi: 10.47065/josyc.v5i3.5156.

[10] S. Ningsih *et al.*, "Pengaruh Penyeimbangan Data Pada Klasifikasi Terjemahan Al-Quran Dengan Metode Naïve Bayes dan Long Short Term Memory," *Journal of Computer System and Informatics (JoSYC)*, vol. 5, no. 3, pp. 626–635, 2024, doi: 10.47065/josyc.v5i3.5181.

[11] P. Pangestu, R. Novita, S. Informasi, and U. Sultan Syarif Kasim Riau, "Systematic Literature Review: Perbandingan Algoritma Klasifikasi," *Jurnal Inovtek Polbeng - Seri Informatika*, vol. 8, no. 2, pp. 431–440, 2023, doi: https://doi.org/10.35314/isi.v8i2.3698.

[12] G. A. B. Suryanegara, Adiwijaya, and M. D. Purbolaksono, "Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi Pasien Penderita Diabetes Menggunakan Metode Normalisasi," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 114–122, Feb. 2021, doi: 10.29207/resti.v5i1.2880.

[13] H. N. Harish, S. Al Faraby, and M. Dwifebri, "Klasifikasi Multi Label Pada Hadis Bukhari Terjemahan Bahasa Indonesia menggunakan Random Forest, Mutual Information, dan Chi-Square," in *e-Proceeding of Engineering*, 2021, pp. 10583–10593. Accessed: Jun. 03, 2025. [Online]. Available: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/15675

[14] Y. Karaman, F. Akdeniz, B. K. Savaş, and Y. Becerikli, "A Comparative Analysis of SVM, LSTM and CNN-RNN Models for the BBC News Classification," in *Lecture Notes in Networks and Systems*, Springer Science and Business Media Deutschland GmbH, 2023, pp. 473–483. doi: 10.1007/978-3-031-26852-6_44.

[15] R. A. Pramunendar, D. P. Prabowo, and R. A. Megantara, "Metode Recurrent Neural Network (Rnn) Dengan Arsitektur Lstm Untuk Analisis Sentimen Opini Publik Terkait Vaksin Covid-19," *JURNAL INFORMATIKA UPGRIS*, vol. 8, no. 1, pp. 42–45, 2022.

[16] G. Budiprasetyo, M. Hani'ah, and D. Z. Aflah, "Prediksi Harga Saham Syariah Menggunakan Algoritma Long Short-Term Memory (LSTM)," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 8, no. 3, pp. 164–172, Jan. 2023, doi: 10.25077/teknosi.v8i3.2022.164-172.

[17] R. Saputra, N. S. Harahap, Novriyanto, and M. Affandes, "Penerapan Merger Rtriever Pada Question Answering System Hadits," *SATIN - Sains dan Teknologi Informasi*, vol. 10, no. 1, pp. 24–35, 2024, doi: https://doi.org/10.33372/stn.v10i1.1117.

[18] T. Wongvorachan, S. He, and O. Bulut, "A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining," *Information (Switzerland)*, vol. 14, no. 1, pp. 1–15, Jan. 2023, doi: 10.3390/info14010054.

[19] S. Diantika, H. Nalatissifa, N. Maulidah, R. Supriyadi, and A. Fauzi, "Penerapan Teknik Random Oversampling Untuk Memprediksi Ketepatan Waktu Lulus Menggunakan Algoritma Random Forest," *Computer Science (CO-SCIENCE*, vol. 4, no. 1, pp. 11–18, 2024, doi: https://doi.org/10.31294/coscience.v4i1.1996.

[20] L. Efrizoni, S. Defit, M. Tajuddin, and A. Anggrawan, "Komparasi Ekstraksi Fitur dalam Klasifikasi Teks Multilabel Menggunakan Algoritma Machine Learning," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 3, pp. 653–666, Jul. 2022, doi: 10.30812/matrik.v21i3.1851.

[21] R. A. Mu'allim and K. M. Lhaksmana, "Klasifikasi Multi-Label Ayat-Ayat Al-Qur'an Menggunakan Random Forest dan Word Centrality," *Jurnal Penelitian Informatika*, vol. 2, no. 2, pp. 9–14, 2024, doi: 10.25124/logic.v2i1.8808.

[22] M. F. Afianto, Adiwijaya, and S. Al Faraby, "Kategorisasi Teks pada Hadits Sahih Al-Bukhari menggunakan Random Forest Text Categorization on Hadith Sahih Al-Bukhari using Random Forest," in *e-Proceeding of Engineering*, 2017, pp. 4874–4881. Accessed: Jun. 03, 2025. [Online]. Available: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/5416

[23] Y. Romadhoni, K. Fahmi, and H. Holle, "Analisis Sentimen Terhadap PERMENDIKBUD No.30 pada Media Sosial Twitter Menggunakan Metode Naive Bayes dan LSTM," *Jurnal Informatika: Jurnal pengembangan IT (JPIT)*, vol. 7, no. 2, pp. 118–124, 2022, doi: https://doi.org/10.30591/jpit.v7i2.3191.

[24] I. Akbar, M. Faisal, and T. Chamidy, "Multi-label classification of Indonesian qur'an translation using long short-term memory model," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, vol. 9, no. 2, pp. 119–128, 2024, doi: https://doi.org/10.22219/kinetik.v9i2.1901.

[25] M. R. Athallah and K. M. Lhaksmana, "Hadith Text Classification Based on Topic Using Convolutional Neural Network (CNN) and TF-IDF," *Journal of Renewable Energy, Electrical, and Computer Engineering*, vol. 5, no. 1, pp. 30–36, Mar. 2025, doi: 10.29103/jreece.v5i1.20354.