BAB III
METODOLOGI PENELITIAN

3.1 Objek dan Subjek Penelitian

Penelitian ini bertujuan untuk mengetahui sejauh mana akurasi yang dapat diberikan oleh model algoritma *K-NN* jika diintergrasikan dengan *label powerset* dalam mengklasifikasikan teks hadis. Subjek pada penelitian ini adalah teks hadis sahih bukhari dengan topik anjuran, larangan dan informasi, sedangkan untuk objek yang dilakukan yaitu mengklasifikan *multi-label* berdasarkan topik anjuran, larangan dan informasi Diharapkan penelitian ini dapat menjadi bahan penelitian pada data *multi-label* dan dapat diterapkan pada kasus kasus *multi-label classification* lainya.

3.2 Alat dan Bahan Penelitian

Alat dan bahan yang digunakan pada penelitian ini berupa perangkat keras dan perangkat lunak yang dilengkapi dengan beberapa pendukung.

3.2.1 Alat penelitian

Dalam penelitian ini penulis menggunakan perangkat keras berupa laptop **Lenovo Ideapad Gaming 3i** dengan spesifikasi sebagai berikut:

1. Processor Intel® Core™ i5-10300H CPU @2.50 GHz 2.5 GHz.
2. Memory (RAM) 8,00 GB (x2)

Adapun perangkat lunak yang digunakan pada penelitian ini antara lain:

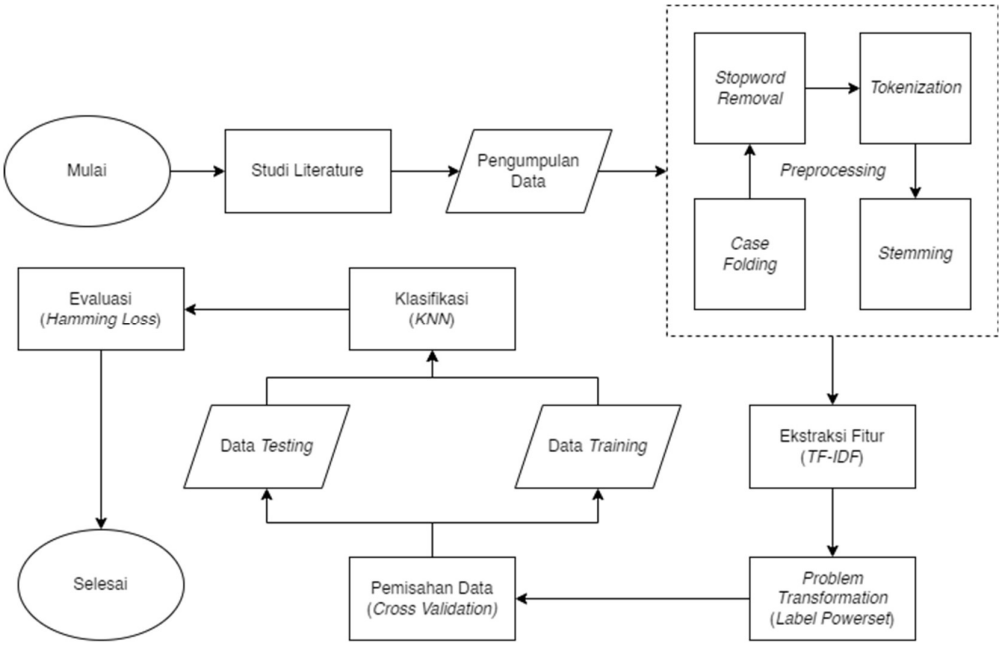
1. Python
2. Visual studio code
3. Google Collab

3.2.2 Bahan penelitian

Untuk bahan penelitian ini *dataset* yang digunakan adalah hadis shahih bukhari dari penelitian sebelumnya yang berjumlah 1064 data, dengan topik informasi sebanyak 1043 data, larangan sebanyak 98 data, dan anjuran sebanyak 230 data. Kemudian jumlah teks hadis yang memiliki lebih dari 1 topik dalam 1 teks yaitu, ada 239 teks hadis yang memiliki 2 topik dan 34 teks hadis yang memiliki 3 topik.

3.3 Diagram Alir Penelitian

Topik penelitian ini adalah melakukan klasifikasi *multi-label* pada teks hadis yang memiliki topik anjuran, larangan dan informasi. Penelitian ini dimulai dari tahap studi literatur, pengumpulan data, *preprocessing dataset*, ekstraksi fitur, *problem transformation,* pemisahan data antara data *training* dan *testing,* diakhiri dengan melakukan klasifikasi model. Tahapan penelitian digambarkan dalam bentuk *flowchart* pada Gambar 3.1 sebagai berikut.



Gambar 3. 1 *Flowchart* alur penelitian

3.3.1. Studi Literatur

Pada tahap ini peneliti melakukan studi literatur mengenai topik penelitian yang dilakukan. Sumber yang digunakan berupa jurnal dan skripsi tentang penelitian sebelumnya, *ebook,* dan juga *website* yang memiliki model serupa, untuk mempelajari model yang akan dipakai dan pencarian metode yang tepat untuk menyelesaikan model yang dibuat berdasarkan bahasan tentang topik penelitian ini. Studi literatur meliputi penjelasan tentang proses *preprocessing*, proses *feature extraction* tf-idf, *problem transformation label powerset,* pembagian data *k-fold cross*

validation, arsitektur *k-nearest neighbor,* evaluasi *hamming loss* dan literatur terkait yang bisa menjadi acuan dalam penelitian ini.

3.3.2. Pengumpulan Data

Dataset yang dikumpulkan berdasarkan penelitian sebelumnya, yaitu teks hadis shahis bukhari terjemahan bahasa indonesia yang sudah diambil sebelumnya. *Dataset* yang digunakan berjumlah 1064 teks dan 3 topik atau label sebanyak 3 topik yaitu anjuran, larangan dan informasi.

3.3.3. *Preprocessing*

Pada tahap ini *preprocessing* data dilakukan dengan 4 tahapan untuk membersihkan data dari *noise*, agar dapat diproses pada model algoritma yang dipakai. Berikut ke-4 tahapan *preprocessing* :

1. *Case Folding*

   Pada proses ini dilakukan perubahan setiap huruf teks hadist yang memiliki huruf kapital atau *uppercase* menjadi *lowercase* atau huruf kecil, dengan tujuan untuk mempercepat kemungkinan pemrosesan teks selanjutnya dalam membandingkan kata-kata yang memiliki arti atau makna yang mirip. dengan contoh kalimat pada tabel 3.1 berikut :

   Tabel 3. 1 Tahapan *case folding*

Sebelum *case folding*	Sesudah *case folding*
Kami pernah shalat Maghrib bersama Nabi ketika matahari sudah tenggelam tidak terlihat	kami pernah shalat maghrib bersama nabi ketika matahari sudah tenggelam tidak terlihat

2. *Stopword Removal*

   Teks yang sebelumnya sudah di proses *case* folding dengan mengubah setiap huruf pada teks menjadi kecil atau *lowercase* di proses pada *stopword removal* untuk dihapus kata kata yang memiliki nilai yang kurang penting dan tidak bermakna, dengan contoh dari kalimat pada tabel 3.2 berikut:

Tabel 3. 2 Tahapan *stopword removal*

Sebelum *stopword removal*	Sesudah *stopword removal*
kami pernah shalat maghrib bersama nabi ketika matahari sudah tenggelam tidak terlihat	pernah shalat maghrib bersama nabi matahari tenggelam terlihat

3. Tokenization

Teks yang sudah diproses dengan case folding, stopword removal dianggap sudah bersih dari kata tidak penting, selanjutnya diubah menjadi kumpulan – kumpulan kata per kata pada tokenization. Dengan contoh kalimat "pernah shalat maghrib sama nabi matahari tenggelam lihat" lalu menghasilkan tokenization seperti tabel 3.3 dibawah:

Tabel 3. 3 Tahapan *tokenization*

Sebelum *tokenization*	Sesudah *tokenization*
pernah shalat maghrib bersama nabi matahari tenggelam terlihat	[pernah, shalat, maghrib, bersama, nabi, matahari, tenggelam, terlihat]

4. Stemming

Proses terakhir setelah memisahkan kata menjadi kumupan kata per kata, menghilangkan kata imbuhan pada kalimat agar kalimat tersebut menjadi kata dasar, dengan contoh dari kalimat pada tabel 3.4:

Tabel 3. 4 Tahapan *stemming*

Sebelum *stemming*	Sesudah *stemming*
['pernah, shalat, maghrib, bersama, nabi, matahari, tenggelam, terlihat]	[pernah, shalat, maghrib, sama, nabi, matahari, tenggelam, lihat]

3.3.4. Ekstraksi Fitur

Pada tahapan ekstraksi firtur ini bertujuan untuk memberikan bobot pada teks yang telah melewati proses *preprocessing* dan juga untuk melakukan representasi dokumen. Pemberian bobot menggunakan metode TF-IDF untuk mengekstraksi bobot pada setiap teks dan juga merepresentasikannya ke dalam matriks. Dengan tahapan alur sebagai berikut :

1. Menghitung *term frequency*

   Menghitung secara langsung kemunculan kata pada dokumen, sebagai contoh di tabel 3.5 berikut :

   Tabel 3. 5 *Term frequency*

Term	Frequency
pernah	1
shalat	2
maghrib	1
sama	1
Nabi	2
matahari	1
tenggelam	2
Lihat	1

2. Menghitung *inverse document frequency*

   Menghitung jumlah kata yang keluar pada setiap dokumen yang ada dengan rumus persamaan (2.3), dengan contoh hasil perhitungan pada tabel 3.6 berikut :

Tabel 3. 6 *Inverse document frequency*

Term	Dok 1	Dok 2	Dok 3	DF	IDF
Lihat	1	2	0	2	0,176
Maghrib	1	2	1	3	0
Matahari	1	1	3	3	0
Nabi	2	1	0	2	0,176

3. Menghitung *term frequesncy – inverse document frequency*

   Menghitung hasil dari masing masing TF dari setiap dokumen dengan IDF yang telah dihasilkan dengan rumus persamaan (2.4), contoh hasil dari perhitungan dapat dilihat pada tabel 3.7 sebagai berikut :

Tabel 3. 7 *Term frequency* dan *inverse document frequency*

Term	Dok 1	Dok 2	Dok 3	DF	IDF	TF.IDF Dok1	TF.IDF Dok2	TF.IDF Dok3
Lihat	1	2	0	2	0,176	0,176	0,352	0
Maghrib	1	2	1	3	0	0	0	0
matahari	1	1	3	3	0	0	0	0
Nabi	2	1	0	2	0,176	0,352	0,176	0

3.3.5. *Problem Transformation*

Pada tahapan ini bertujuan untuk merubah *dataset* dari *multi-label* menjadi *multi-class*, supaya dapat dilakukan klasifikasi menggunakan model algoritma yang dipakai, dengan contoh awal *dataset* ditabel 3.8 berikut:

Tabel 3. 8 *Dataset* sebelum *label powerset*

Data	Nasihat	Larangan	Informasi
Kami pernah shalat Maghrib bersama Nabi ketika matahari sudah tenggelam tidak terlihat.	0	0	1

Perubahan dari *multi-label* menjadi *multi-class* dapat dilihat dengan contoh ditabel 3.9 sebagai berikut :

Tabel 3. 9 *Dataset* seteleh *label powerset*

Data	Class
Kami pernah shalat Maghrib bersama Nabi ketika matahari sudah tenggelam tidak terlihat.	001

3.3.6. Pembagian Data

Pada tahapan ini sebelum masuk ke tahap klasifikasi *dataset* dibagi menjadi data *testing* dan data *training* dengan menggunakan metode *k-fold cross validation*, dimana sebagai contoh nilai *k* diatur sebesar 4 agar *dataset* dapat dibagi secara merata dengan tampilan ditabel 3.10 berikut :

Tabel 3. 10 Tahapan *k-fold cross validation*

Fold ke-n	Data ke 1-266	Data ke 267-532	Data ke 533-798	Data ke 799-1064
Fold ke-1	Testing	Training	Training	Training
Fold ke-2	Training	Testing	Training	Training
Fold ke-3	Training	Training	Testing	Training
Fold ke-4	Training	Training	Training	Testing

3.3.7. Klasifikasi

Pada tahap klasifikasi setelah didapatkan pembagian data *testing* dan data *training,* data tersebut dimasukan ke dalam model algoritma *K-NN*

untuk dilakukan prediksi untuk klasifikasi *multi-label* berdasasrkan dengan tahapan dari algoritma *K-NN* sebagai berikut :

1. Menentukan nilai atau jumlah dari *k*
2. Menghitung jarak antara satu data baru dengan seluruh data *training*.
3. Hasil perhitungan diurutkan berdasarkan nilai kemiripannya.
4. Menentukan beberapa objek berdasarkan jarak terdekat sebanyak dari nilai k
5. Menentukan kelas untuk data baru berdasarkan hasil frekuensi yang paling tinggi dari nilai k.

3.3.8. Evaluasi

Setelah melakukan klasifikasi, tahap terakhir adalah mengevaluasi hasil yang telah dihitung pada proses klasifikasi dengan menggunakan *hamming loss*, untuk memperoleh performansi dari sistem yang dibangun.