**PEMODELAN TOPIK TERJEMAH AL-QUR’AN BERBAHASA INDONESIA DENGAN MENGGUNAKAN BERTOPIC**

# PENDAHULUAN

Al-quran adalah salah kitab suci orang muslim dengan banyak surat 114 surat dengan jumlah ayat 6127 ayat. Sebagai kitab pedoman muslim tentunya banyak makna yang bisa diambil dari apa yang ada dalam seitan ayatnya. Atas dasar itu penulis ingin mengclusterkan ayat-ayat alquran dengan beberapa algoritma modeling text mining, algoritma yang akan digunakan untuk pemodelan yaitu Bertopic Model

Bertopic modeling adalah Teknik pemodelan topik yang memanfaatkan transormator dan c-ItF-IDF untuk membuat cluster padat yang memungkinkan topik mudah diinterpretasikan sambil memperthankan kata-kata penting dalam deskripsi topik.

Pada pemodelan kali ini selain untuk pemenuhan tugas ujian akhir semerter, sekaligus untuk menambah pemahaman penulis terhadap terjemah ayat-ayat al-qur’an, juga agar menjadi referensi tambahan penerapan topik modeling dengan model BERTopik dalam menclusterkan ayat-ayat alquran dalam terjemah Bahasa indonesia.

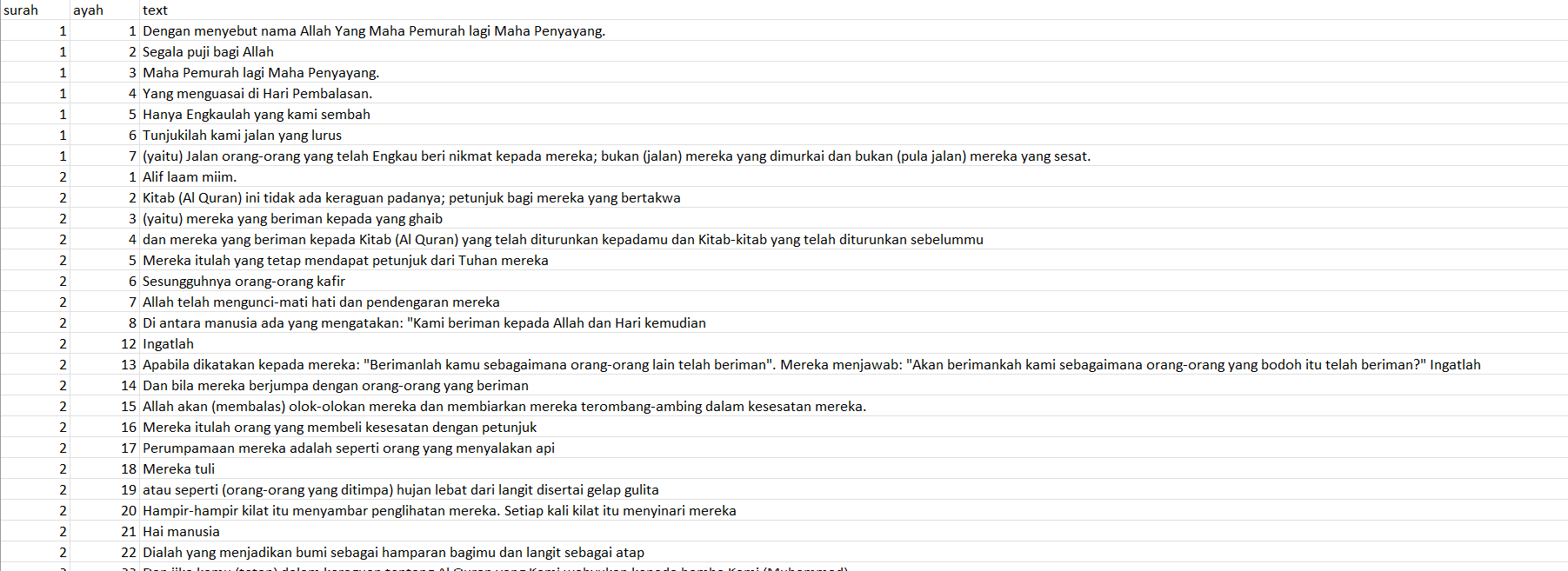
# METODOLOGI

# Pengumpulan data

Dalam penelitian ini dataset yang digunakan adalah Al-Qur’an terjemah bahasa indonesia yang didapat dari quranenc.com [1].

Selanjutnya setelah pengambilan data, ada beberapa kolom data yang disesuaikan seperti hanya kolom text terjemah bahasa indonesia saja yang digunakan untuk penomoran surat dan ayat tidak diikut sertakan. Setelah pemilihan kolom text dipilih maka dilakukan preprocesing terhadap data.

Berikut capture dataset yang didapat:



**Gambar 1.1** Capture Dataset Al-Qur’an

# Untuk pemodelan yang digunakan yaitu pemodelan BERTopik dengan menggunakan pemograman python dan tools googlecolab.

# 2.2 Proses dan Preprocesing Teks

**2.2.1 Procesing data**

Berikut scrypt pegolahan data pada python.

|  |
| --- |
| import pandas as pd  import numpy as np  from bertopic import BERTopic  import re,string  # 1 select data  df = pd.read\_csv('alquran\_terjemah\_indonesian.csv')  text = df['text']  # 2. PREPROCESSING  # Drop nilai yang kosong  text = text.fillna('')  # 3. Fungsin untuk prepocessing  def clean\_text(text):      text = text.lower()  # Lower      text = re.sub(r'\d+', '', text)  # Menghapus angka      text = re.sub(r'\W', ' ', text)  # Menghapus karakter non-alfanumerik      text = re.sub(r'\s+', ' ', text)  # Menghapus spasi berlebih      # text = re(str.maketrans("","",string.punctuation))      return text.strip()  # 4 panggil fungsi clean text dan simpan divariabel docs  docs = text.apply(clean\_text)  # 5. Deklarasikan variabel untuk Membuat model BERTopic  model = BERTopic(verbose=True)  # 6.Melakukan fit dan transformasi dokumen  topics, probabilities = model.fit\_transform(docs.tolist())  # 7. Fungsi untuk get topic yang dihasilkan  model.get\_topic\_info()  # 8 Menampilkan topik array ke 0 indeks 1  model.get\_topic(0)  # 9 visualisasi data  # visualisasi topic (intertopic Distance Map)  model.visualize\_topics()  # 10 model.visualize\_barchart() (topic wors score)  model.visualize\_barchart()  # 11 model visualisai hirarki  model.visualize\_hierarchy()  # 12 Ctf Idf score  model.visualize\_term\_rank()  #13.  similiarity map  model.visualize\_heatmap() |

# 2.2.3 Preprocesing Teks

Proses dalam preprocessing data, antara lain sebagai berikut:

1. Remove symbol dan Number

Langkah yang sering dilakukan adalah menghapus simbol dan angka yang terdapat dalam data. Hal ini dilakukan karena simbol dan angka biasanya tidak memiliki makna yang signifikan dalam proses analisis dan dapat mengganggu pemodelan topik atau analisis lainnya. Untuk menghapus simbol dan angka, kita dapat menggunakan fungsi seperti re sub () atau string punctuation pada Python.

berikut script yang digunakan dalam pengolahan data

|  |
| --- |
| text = text.lower()  # Lower      text = re.sub(r'\d+', '', text)  # Menghapus angka      text = re.sub(r'\W', ' ', text)  # Menghapus karakter non-alfanumerik      text = re.sub(r'\s+', ' ', text)  # Menghapus spasi berlebih |

1. Lower case

Lower case adalah proses mengubah semua teks menjadi huruf kecil (lower case). Proses ini dilakukan untuk menghindari perbedaan bentuk yang mungkin terjadi akibat adanya huruf besar.

|  |
| --- |
| text = text.lower()  # Lower |

# Tokenezing

Pada tahap preprocessing berikutnya, terdapat tokenisasi, yaitu proses membagi teks menjadi sejumlah token atau kata-kata

1. Model Generation and Visualization (Model BERTOPIC)

Tahap selanjutnya adalah pemodelan topik menggunakan BERTopic. BERTopic adalah salah satu algoritma pemodelan topik yang menggunakan metode embedding untuk

mengidentifikasi topik-topik yang terdapat dalam data.

|  |
| --- |
| topics, probabilities = model.fit\_transform(docs.tolist()) |

# HASIL DAN PEMBAHASAN

Pada penelitian ini menggunakan software google Colab dengan meng-install beberapa library dari pandas, numpy dan BERTopik. Dataset yang digunakan merupakan terjemahan dari Al-Qu’an yang unduh melalui website quranenc.com, dengan memilih bahasa Indonesia pada website tersebut kemudian didownload dataset berbentuk csv. Setelah dataset diunduh langkah selanjutnya adalah *preprocessing* yaitu membersihkan data dari karakter, angka dan simbol yang tidak diperlukan dan tidak bermakna, setelah preprocesing data dijadikan list untuk proses pengolahan data selanjutnya, berikut gambaran data sebelum dan sesudah preprocesing dan menjadi list.

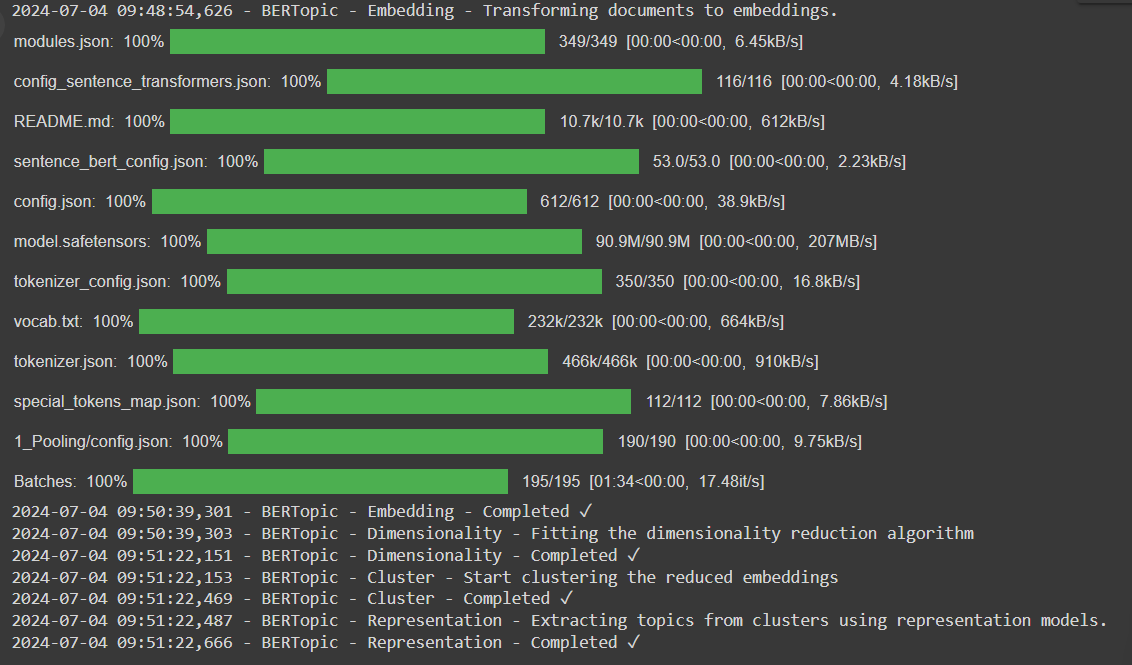
|  |
| --- |
| 0 Dengan menyebut nama Allah Yang Maha Pemurah l...  1 Segala puji bagi Allah  2 Maha Pemurah lagi Maha Penyayang.  3 Yang menguasai di Hari Pembalasan.  4 Hanya Engkaulah yang kami sembah  Name: text, dtype: object |

**Gambar 1.1** dataset sebelum preprocessing dan datalist

|  |
| --- |
| ['dengan menyebut nama allah yang maha pemurah lagi maha penyayang',  'segala puji bagi allah',  'maha pemurah lagi maha penyayang',  'yang menguasai di hari pembalasan',  'hanya engkaulah yang kami sembah'] |

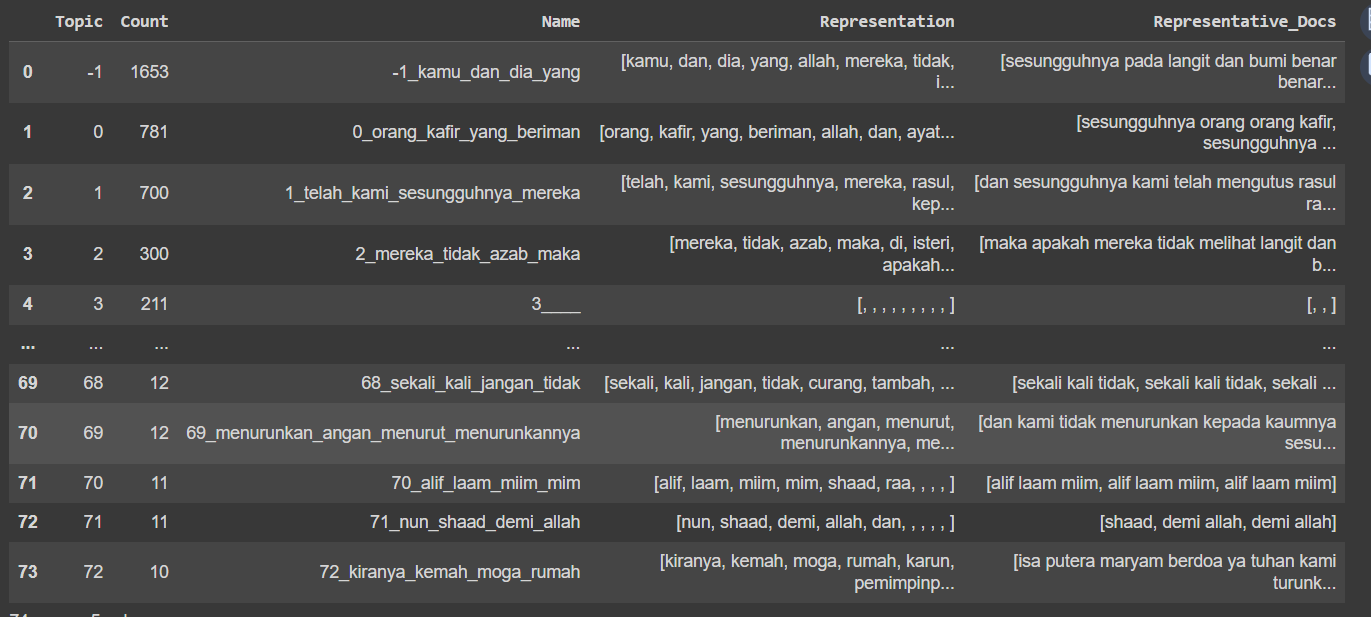
**Gambar 1.2** Dataset setelah preprocessing dan list

Setelah selesai preprocessing data, selanjutnya data akan diimplementasikan juga disesuaikan (*fit transformation*) dan embeding pada model bahasa BERTopic dengan menggunakan bahasa Indonesia. Berikut proses dan output dari dataset yang dikelompokan dalam kelompok-kelompok topik.



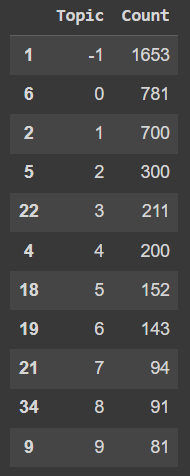
**Gambar 1.3** Proses fit dan transformasi, *embedding* dokumen BERTopic

Setelah proses Proses fit dan transformasi *embedding* dokumen, maka akan terbentuk beberapa topik juga representasi dari bertopic terhadap dataset terjemah alqur’an bahasa indonesia. Berikut capture output pengelompokan beberapa topic.



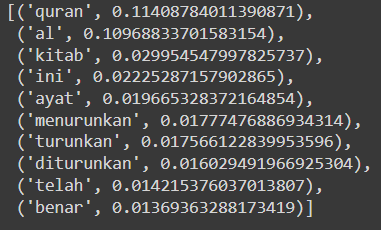
**Gambar 1.4** Model get topic informasi

Selanjutnya kita bisa memilih topic mana yang akan kita tampilkan, untuk topik -1 artinya dokumen nilai outliernya dikurangi atau dihilangkan oleh Bertopic sebanyak 1.653, kemudian dengan menggunakan soft-clustering seperti yang dilakukan oleh HDBSCAN untuk menemukan topik yang paling cocok untuk setiap dokumen outlier [5]. Dan pengelompokan secara default dari bertopic itu sendiri sebanyak 72 topik. Berikut bisa dilihat juga ukuran topic menurun.



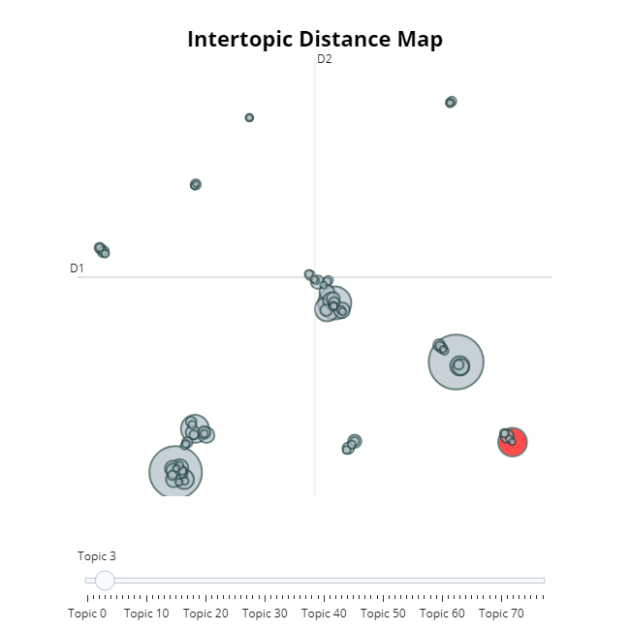
**Gambar 1.5** ukuran topic dalam urutan menurun

Dari 10 topic diatas ada beberapa topik yang dapat kita pilih dari banyaknya kata yang dikelompokan, untuk -1 seperti topic information diatas adalah outlier. berikut ini bisa memilih topik diatas dengan mengambil no topik, berikut salah satu contoh topik no 4 yang dipilih outputnya seperti ini.



**Gambar 1.6** pengelompokan topik no 4

Pada gambar 1.6 pengelompokan topik dengan dikut nili tf-idf nya,dan dari pengelompokan topik diatas seperti ‘Qur’an’, ’kitab’, ’ini’,’ayat’,’menurunkan’, ’turunkan’, ’telah ’, ’benar’ topik ini berfokus pada al-quran kitab yang benar.

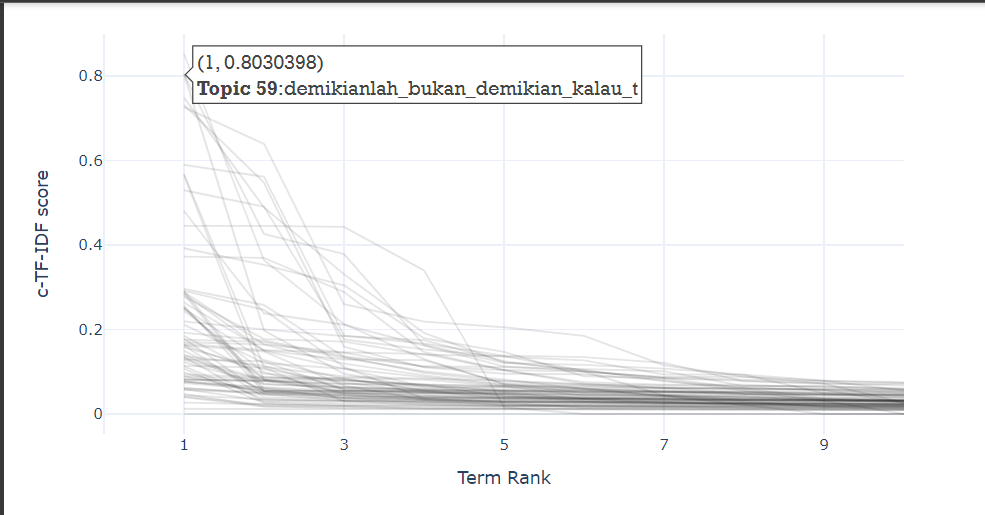


**Gambar 1.7** Intertopic distance map

Gambar 1.7 merupakan visualisasi dari pengelompokan topik pada dataset, bisa dilihat dari gambar 1.7 topik yang dibuat sebanyak 72 topik dan cendrung terkelompok kebeberapa kelompok.

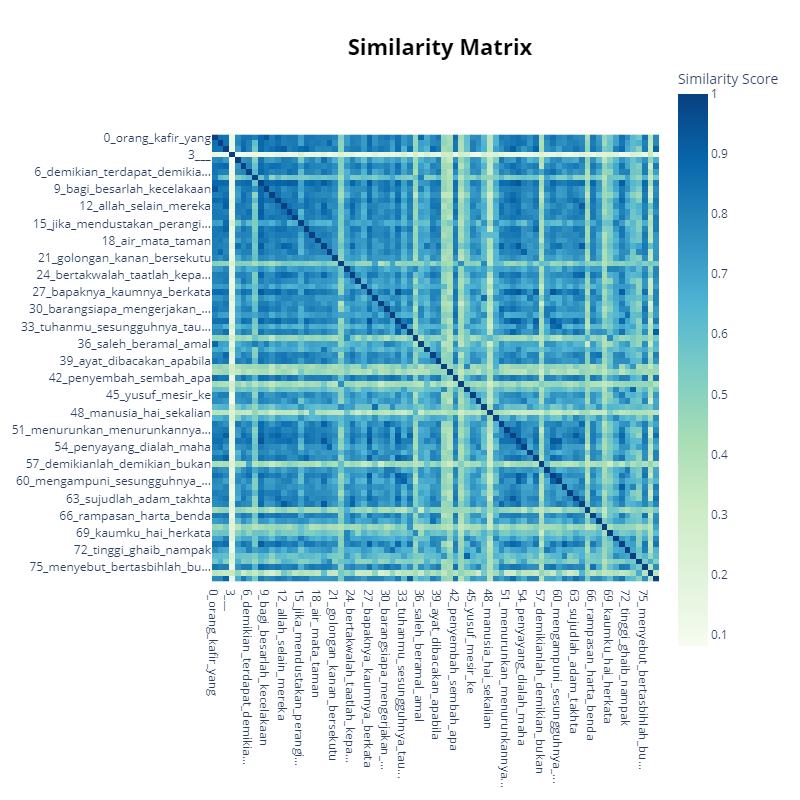
**Gambar 1.8** Topic Word Score

Pada gambar 1.8 merupakan visualisasi beberapa topik lengkat dengan range scorenya, dan ada 7 topik yang bisa tergambar dalam gambar 1.8 dengan nilai range berbeda-beda.



**Gambar 1.9** Term Rank

Pada grafik term rank dengan mengarahkan kursor kita bisa melihat langsung dengan nilai graig c-TF-IDF nya. Seperti contoh pada gambar 1.9 terlihat topic 59 dengan nilai c-TF\_IDF sebesar 0.8. diagram diatas membantu kita memutuskan apakah kita dapat memangkas jumlah istilah yang ingin kita bedakan. Misalnya, kita mungkin menganggap bahwa hanya 11 istilah teratas yang cukup penting untuk kita pertimbangkan[6].



**Gambar 1.8** Similiarity matrik

Matrik pada gambar 1.8 digunakan untuk melihat seberapa besar kemiripan setiap topik dan membantuk kita untuk mengevaluasi setiap topik.

# KESIMPULAN DAN SARAN

Setelah dilakukan pemodelan topik terhadap dataset Al-Quran sebanyak 6236 ayat, ditemukan beberapa topik utama dari masing-masing topik yang dapat dilihat melalui visualisasi topik. Dengan menggunakan BERTopic secara otomatis pada pemodelan ini dikelompokan menjadi 72 topik, kita dapat dengan mudah memodelkan topik-topik yang terdapat dalam Al-Quran dan mengekstrak informasi yang bermanfaat dari teks tersebut.

Berdasarkan penelitian ini hasil pemodelan topik menghasilkan 10 topik utama. Salah satu contoh topik ke 4 yang terwakilkan dengan kata, ‘Qur’an’, ’kitab’, ’ini’,’ayat’,’menurunkan’, ’turunkan’, ’telah ’, ’benar’.

Dari pemodelan BERTopik dengan nilai outlier data cukup tinggi sebanyak 1653 artinya 20% dari data terdapat outlier, hal ini bisa dari preprocessing yang bisa ditingkatkan lagi, juga bisa mengevuluasi data yang sekiranya kurang baik dari visualisasi data

# DAFTAR PUSTAKA

[1] <https://quranenc.com/id/browse/indonesian_complex/1>

[3] <https://towardsdatascience.com/topics-per-class-using-bertopic-252314f2640>

[5] <https://maartengr.github.io/BERTopic/api/bertopic.html#bertopic._bertopic.BERTopic.reduce_outliers>

[6] https://cees-roele.medium.com/a-term-score-matrix-for-bertopic-821e78e198ee

[7] <https://www.kaggle.com/code/yhirakawa/bertopic-visualization-of-topic-modeling#Bertopic>

[8] https://maartengr.github.io/BERTopic/getting\_started/clustering/clustering.html#agglomerative-clustering