

ABSTRAK

Nama : Maulidah Tsaniatuluzzma
NIM : 1217010042
Judul Skripsi : Pengaruh Hyperparameter Dalam FastText Terhadap *Semantic Similarity* Kata Menggunakan Dataset Al-Qur'an Bahasa Arab

Penelitian ini menganalisis pengaruh hyperparameter FastText terhadap kemiripan semantik kata pada dataset Al-Qur'an berbahasa Arab. Bahasa Arab Al-Qur'an yang kompleks memerlukan metode representasi kata yang akurat. Metodologi mencakup pengumpulan dataset 6.236 ayat dari Tanzil.net, diikuti pra-pemrosesan seperti tokenisasi, pembersihan teks, penghapusan *stopwords*, normalisasi, dan *lemmatisasi*. Model FastText dilatih menggunakan arsitektur Skip-gram, yang efektif untuk kata jarang muncul. Hyperparameter yang diuji meliputi dimensi vektor, *window size*, *learning rate*, *minimum count*, dan epoch. Evaluasi *semantic Similarity* menggunakan *cosine Similarity* menunjukkan konfigurasi optimal adalah dimensi *embedding* 300, 10 epoch, dan *window size* 5, dengan nilai similaritas mencapai 0.999. Visualisasi PCA memvalidasi kemampuan model FastText mengelompokkan kata berdasarkan makna dan konteks dalam Al-Qur'an, termasuk hubungan sinonim.

Kata Kunci: FastText, Hyperparameter, *Semantic Similarity*, Al-Qur'an Bahasa Arab, *Word Embedding*.

ABSTRACT

Nama : Maulidah Tsaniatuluzzma
NIM : 1217010042
Judul Skripsi : The Effect of Hyperparameters in FastText on Semantic Similarity of Words Using the Arabic Qur'an Dataset.

This study analyzes the impact of FastText hyperparameters on word semantic Similarity using an Arabic Quranic dataset. The complex nature of Quranic Arabic necessitates accurate word representation methods. The methodology involved collecting a 6,236-verse dataset from Tanzil.net, followed by pre-processing steps such as tokenization, text cleaning, stopwords removal, normalization, and lemmatization. The FastText model was trained using the Skip-gram architecture, which is effective for rare words. Tested hyperparameters included vector dimension, window size, learning rate, minimum count, and epochs. Semantic Similarity evaluation using cosine Similarity showed optimal configuration with an embedding dimension of 300, 10 epochs, and a window size of 5, achieving a Similarity value of 0.999. PCA visualization validated FastText's ability to cluster words based on meaning and context within the Quran, including synonym relationships.

Keywords: FastText, Hyperparameters, Semantic Similarity, Arabic Quran, Word Embedding.

KATA PENGANTAR

“Data adalah cerminan, dan algoritma adalah penerjemah. Kalam ilahi dan logika, kini membuat cerita.”

Bismillahirrahmanirrahiim

Segala puji bagi Allah SWT, Tuhan yang Maha Esa, yang mencipta langit dan bumi dalam kuasa dan cinta. Dia yang menggilir siang dan malam penuh makna, memberi ketenangan dalam lelah, cahaya di tengah gulita. Segala nikmat-Nya tak terhingga, bukti kasih-Nya yang nyata. Shalawat dan salam tercurah pada Nabi mulia, pembawa cahaya dari zaman ke zaman, penuntun jiwa dari gelap menuju terang, pelita hati dalam gundah yang tak pernah padam.

Adapun skripsi ini berjudul: **“Pengaruh Hyperparameter dalam FastText terhadap *Semantic Similarity* Kata Menggunakan Dataset Al-Qur’an Bahasa Arab”**, sebagai syarat meraih gelar Sarjana Jurusan Matematika Fakultas Sains dan Teknologi UIN Sunan Gunung Djati Bandung. Tanpa pertolongan-Nya, langkah ini tak akan sampai pada akhirnya, sebab ragu dan lelah kerap datang menyapa. Namun dukungan dari orang-orang tercinta menjadi penguat jiwa kala semangat mulai mereda. Saran dan masukan yang tulus terasa begitu bermakna, menjadi cahaya dalam pencarian makna yang tak bisa diukur dengan kata-kata. Untuk itu, penulis mengucapkan rasa syukur dan terima kasih yang sebesar-besarnya kepada :

1. Kedua orang tua, Ambu dan Abah. Yang tidak pernah lelah memberikan kekuatan untuk penulis, memberikan nasihat untuk melanjutkan perjalanan hidup dengan segala fasenya yang berubah-ubah . Semoga Allah berikan istana terindah yang penuh cahaya, serta kebahagiaan yang tiada tandingannya.
2. Seluruh anggota keluarga “Abah” tercinta, keluarga kecil dengan 13 bersaudara, yang tidak dapat penulis sebutkan satu persatu namanya, yang senantiasa selalu memberikan doa, dukungan, dan kasih sayang yang luar biasa dan tiada habisnya. Semoga Allah SWT selalu melimpahkan rahmat dan keberkahan dunia dan akhirat atas kebaikan yang telah diberikan kepada penulis dengan penuh cinta.

3. Muhammad Azka Muqorobin, selaku adik pertama penulis yang senantiasa membantu dan mendukung segala harapan dan keinginan penulis dengan hal yang ia bantu wujudkan satu-persatu. Kelak, semoga menjadi manusia bermanfaat bagi sesama. Semoga cepat menjadi sarjana meraih gelar S.Ag. nya.
4. Muhammad Irsyadul Yaqin, S.Hum., selaku kakak kedua penulis. Seperti namanya, ia selalu meyakinkan bahwa hal hebat itu mampu dan pasti penulis gapai meski banyak rintangan yang menghadang. Terima kasih sudah menjadi teman baik dan karib yang semenyenangkan itu selama bertahan di dunia per-Bandung-an. Sehat dan bahagia selalu, penyemangat hebatku.
5. Dewi Kamal Muttamimah, S.Hub.Int., selaku kakak ketiga penulis yang selalu membantu segala hal, termasuk mencari dan membantu menemukan kebahagiaan. Tanpa jasa dan ketulusan hati beliau, penulis tidak akan menjadi salah satu mahasiswi di UIN Sunan Gunung Djati ini dan bertahan hingga mencapai tahap kelulusan. Semoga segala keberkahan selalu Allah limpahkan.
6. Ustadz dan Umi Pondok Pesantren Miftahul Huda Al-Faqih 2, yang menjadi kekuatan bertahan kepada penulis di perantauan, untuk membuktikan kemampuan diri agar tidak berpikir negatif yang berlebihan.
7. Bapak Asep Solih Awaluddin, M.Si. selaku ketua Jurusan Matematika Sains dan Teknologi.
8. Bapak Dr. Arief Fatchul Huda, S.Si., M.Kom. selaku dosen pembimbing I dan Bapak Dr. Aep Saepuloh, M.Si. selaku dosen pembimbing II yang senantiasa membimbing, memberikan arahan, saran, dan motivasi kepada penulis dalam penyusunan skripsi ini.
9. Seluruh dosen dan staf di Jurusan Matematika yang telah berbagi ilmunya sehingga dapat mempermudah penulis dalam menyelesaikan skripsi ini.
10. Sobat Qadarullah, dan manusia-manusia hebat khususnya NIM akhir 20 dan 29, yang setia menemani setiap langkah, yang memberi ketenangan di saat penulis merasa gundah.

11. Manusia amatir yang kehebatannya seperti petir, merambat cepat padahal batang hidung paling hebat. Mahasiswa alumni dengan NIM akhir 43, yang membantu penulis dari segala arah. Semoga menjadi amal jariyah dan semoga Allah berkahi setiap langkahnya serta keluarganya yang turut memberi dukungan dan doa.

12. Dan semua pihak yang terlibat yang tidak dapat penulis sebutkan satu persatu. Semoga Allah SWT membalas dengan balasan yang setimpal atas segala kebbaikannya yang telah diberikan kepada penulis.

Penulis menyadari akan keterbatasan ilmu yang dimiliki bahwa penulisan skripsi ini masih terdapat kekurangan. Maka dari itu, penulis sangat mengharapkan kritik dan saran yang membangun dari pembaca sebagai bahan perbaikan di masa mendatang. Akhir kata penulis berharap skripsi ini dapat memberikan manfaat bagi pembaca pada umumnya dan penulis pada khususnya. Atas segala perhatiannya, penulis mengucapkan terima kasih.

Bandung, Agustus 2025

Penulis



DAFTAR ISI

LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
PERNYATAAN KEASLIAN SKRIPSI.....	iv
PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI.....	v
ABSTRAK	vi
ABSTRACT	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xv
DAFTAR SIMBOL	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	5
1.3 Batasan Masalah	5
1.4 Tujuan Penelitian	6
1.5 Manfaat Penelitian	6
1.6 Metode Penelitian	7
1.7 Sistematika Penulisan	7
BAB II LANDASAN TEORI	9
2.1 <i>Natural Language Processing</i> (NLP)	9
2.2 Karakteristik Bahasa Arab dalam NLP	10
2.3 Teknik <i>Pre-Processing</i>	12
2.3.1 <i>Tokenization</i> (Tokenisasi)	12
2.3.2 <i>Text Cleaning</i> (Pembersihan Teks)	12

2.3.3	<i>Stopword Removal</i> (Penghapusan Kata Umum)	13
2.3.4	<i>Text Normalization</i> (Normalisasi Teks)	13
2.3.5	<i>Lemmatization</i>	14
2.4	<i>Word Embedding</i>	14
2.4.1	Word2Vec	15
2.4.2	FastText	19
2.5	Hyperparameter	20
2.6	<i>N-gram</i> dalam Pemrosesan Teks	21
2.7	<i>Semantic Similarity</i>	23
2.8	<i>Python</i>	24
BAB III METODE PENELITIAN		26
3.1	Dataset	27
3.2	<i>Pre-Processing</i>	27
3.2.1	<i>Tokenization</i> (Tokenisasi)	28
3.2.2	<i>Text Cleaning</i> (Pembersihan Teks)	28
3.2.3	<i>Stopword Removal</i>	29
3.2.4	<i>Text Normalization</i>	29
3.2.5	<i>Lemmatization</i>	29
3.3	Pelatihan Model FastText	30
3.4	Arsitektur Model Skip-gram	32
3.5	Evaluasi <i>Semantic Similarity</i>	35
BAB IV EKSPERIMEN DAN ANALISIS		37
4.1	Validasi Metode	37
4.1.1	Analisis Semantik Kontekstual	37
4.1.2	Analisis Semantik Skalar	48
4.1.3	Visual Progres Kinerja Model	55
4.2	Penerapan Model pada Korpus Utama	58
4.3	<i>Pre-Processing</i>	60

4.4 Training Data	60
BAB V PENUTUP	85
5.1 Kesimpulan	85
5.2 Saran	86
DAFTAR PUSTAKA.....	87
RIWAYAT HIDUP	90
LAMPIRAN.....	92



DAFTAR GAMBAR

Gambar 2. 1 Arsitektur CBoW	16
Gambar 2. 2 Arsitektur Skip-gram	18
Gambar 3. 1 Diagram Alur Proses Penelitian	26
Gambar 3. 2 Dataset Al-Qur'an	27
Gambar 3. 3 Diagram Alur Tahapan Pre-Processing	28
Gambar 3. 4 Algoritma FastText	31
Gambar 3. 5 Fase Pra-pelatihan Skip-gram	32
Gambar 3. 6 Proses inti Skip-gram	33
Gambar 3. 7 Diagram Alur Perhitungan Cosine Similarity	36
Gambar 4. 1 Ilustrasi Similarity Kata Ma'rifat 20%, 40%, 60%, dan 80%.....	40
Gambar 4. 2 Ilustrasi Similarity Variasi Panjang Kalimat	41
Gambar 4. 3 Ilustrasi Similarity Konteks Tematik.....	43
Gambar 4. 4 Ilustrasi Similarity Kata Spesifik “والدة”	47
Gambar 4. 5 Ilustrasi Similarity Variasi Sinonim	48
Gambar 4. 6 Ilustrasi Similarity Variasi Jumlah Kalimat	52
Gambar 4. 7 Ilustrasi Similarity Variasi Jumlah Kalimat Sinonim	55
Gambar 4. 8 Visualisasi Analisis Semantik Kontekstual	56
Gambar 4. 9 Visualisasi Analisis Semantik Skalar	56
Gambar 4. 10 Diagram Awal Alur Penelitian.....	58
Gambar 4. 11 Dataset Al-Qur'an yang digunakan.....	59
Gambar 4. 12 Lanjutan Dataset yang digunakan	59
Gambar 4. 13 Hasil Pre-Processing	60
Gambar 4. 14 Diagram Rata-rata Nilai Similaritas berdasarkan Window size dan Dimensi	61
Gambar 4. 15 Rata-rata Nilai Similaritas Berdasarkan Epoch dan Dimensi	61
Gambar 4. 16 Ilustrasi Vektor pada Pengujian Korpus Utama (Dimensi 300, Epoch 10, Window size 5)	65
Gambar 4. 17 Ilustrasi Similarity Ilustrasi Vektor pada Pengujian Korpus Utama (Dimensi 300, Epoch 10, Window size 5).....	67
Gambar 4. 18 Ilustrasi Training FastText pada Korpus Al-Qur'an.....	72
Gambar 4. 19 Ilustrasi penyebaran Similarity dengan Kata Target خير.....	72

DAFTAR TABEL

Tabel 2. 1 Contoh N-gram.....	21
Tabel 2. 2 Contoh N-gram pada FastText	22
Tabel 4. 1 Representasi Kata Ma'rifat	38
Tabel 4. 2 Representasi Variasi Panjang Kalimat.....	41
Tabel 4. 3 Representasi Konteks Tematik	42
Tabel 4. 4 Representasi Kata Spesifik “والدة”	44
Tabel 4. 5 Representasi Variasi Sinonim.....	47
Tabel 4. 6 Representasi Variasi Jumlah Kalimat	48
Tabel 4. 7 Representasi Variasi Jumlah Kalimat Sinonim	52
Tabel 4. 8 Hasil Cosine Similarity pengujian Korpus Utama (Dimensi 300, Epoch 10, Window size 5).....	63
Tabel 4. 9 Tabel Analisis Similaritas Sinonim Kata "خير"	68



DAFTAR SIMBOL

v_w	: Vektor kata target,
v_c	: Vektor konteks, dan
η	: <i>Learning rate</i> .
G_w	: Himpunan semua n-gram (<i>subword</i>) dari kata w
\vec{Z}_g	: Vektor embedding dari n-gram g
$ G_w $: Jumlah n-gram dalam kata w
v_t	: Vektor dari kata target
v_c	: Vektor dari kata konteks positif
v_{ni}	: Vektor dari kata negatif ke- i
k	: Jumlah kata negatif yang diambil
$\sigma(x)$: Fungsi sigmoid, yaitu $\sigma(x) = \frac{1}{1+e^{-x}}$
\vec{v}_w	: Vektor input dari kata target w
\vec{v}_c	: Vektor output dari konteks positif c
\vec{v}_{nk}	: Vektor output dari kata negative ke- k
$\vec{v}_c^T \vec{v}_w$: dot product antara vektor target dan konteks positif
$\sigma(\vec{v}_{nk}^T \vec{v}_w)$: Nilai sigmoid dari dot product
v_w	: Vektor untuk kata w
G_w	: Himpunan sub-kata dari kata w
z_g	: Vektor untuk sub-kata g
$A \cdot B$: dot product dari vektor A dan B
$\ A\ $: Norma (magnitudo) dari vektor A
$\ B\ $: Norma (magnitudo) dari vektor

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemrosesan bahasa alami (*Natural Language Processing* / NLP) adalah salah satu cabang dari kecerdasan buatan yang berfokus pada pengolahan teks dan ucapan manusia, dengan tujuan agar komputer dapat memahami dan memanipulasi bahasa. Salah satu tantangan utama dalam NLP adalah merepresentasikan kata-kata dalam bentuk yang dapat dipahami oleh model komputer. *Word embedding* menjadi solusi untuk tantangan ini, di mana kata-kata direpresentasikan sebagai vektor dalam ruang dimensi tinggi [1]. Metode Word2Vec, yang diperkenalkan Mikolov et al. (2013), memelopori pendekatan ini dengan dua arsitektur utama, yaitu *Continuous Bag of Words* (CBOW) dan Skip-gram [2].

Namun, Word2Vec memiliki keterbatasan dalam menangani bahasa yang morfologinya kompleks seperti bahasa Arab, karena tidak memperhatikan struktur internal kata. Bahasa Arab, khususnya dalam teks Al-Qur'an, memiliki karakteristik unik berupa sistem akar kata, imbuhan yang rumit, dan bentuk kata yang bervariasi sehingga menuntut representasi kata yang fleksibel. Menjawab kekurangan tersebut, Bojanowski et al. (2017) mengembangkan FastText, model lanjutan dari Word2Vec yang memperhitungkan n-gram karakter sebagai sub-kata dalam pelatihan model, sehingga lebih sensitif terhadap morfologi kata dan sangat cocok untuk bahasa Arab.

Beberapa penelitian sebelumnya telah menunjukkan efektivitas penggunaan FastText dalam NLP berbahasa Arab. Belinkov dan Glass (2015) dalam studinya mengenai "*Arabic Diacritization with Recurrent Neural Networks*" mengaplikasikan *embedding* sebagai input pada model RNN untuk memulihkan harakat dalam teks Arab, menekankan pentingnya kualitas representasi kata dalam pemrosesan morfologi bahasa Arab. Zalmout dan Habash (2020) mengembangkan model *multitask adversarial learning* untuk menangani dialek dan fitur morfologis Arab secara bersamaan, penelitian ini menunjukkan bahwa pendekatan berbasis *embedding* memainkan peran krusial dalam generalisasi antar variasi bahasa Arab.

Selain itu, beberapa penelitian terdahulu telah menunjukkan potensi FastText dalam memahami bahasa yang kompleks secara morfologis. Misalnya, penelitian oleh Grave et al. (2018) memperluas FastText dengan *supervised learning* untuk meningkatkan representasi kata pada berbagai bahasa dan menunjukkan bahwa FastText unggul dalam menangani kata yang jarang muncul dan kata baru (*out-of-vocabulary*). Dalam konteks bahasa Arab, Zahran et al. (2021) menggunakan FastText untuk membangun model representasi semantik dalam pengelompokan topik pada teks keislaman dan menunjukkan hasil yang unggul dibandingkan Word2Vec dan GloVe. Sementara itu, El Mahdaouy et al. (2022) mengevaluasi performa FastText dalam klasifikasi sentimen teks Arab dan menunjukkan bahwa model ini menghasilkan akurasi yang lebih baik pada data dengan variasi bentuk kata. Penelitian-penelitian ini memperkuat bahwa FastText memiliki keunggulan dalam menangani bahasa seperti Arab yang sangat dipengaruhi oleh akar kata, imbuhan, dan struktur fleksibel.

Lebih lanjut, studi oleh Adewumi et al. (2022) berjudul “*Word2Vec: Optimal Hyperparameters and Their Impact on NLP Task*” mengevaluasi pengaruh konfigurasi hyperparameter terhadap performa model dalam berbagai tugas NLP. Hasilnya menegaskan bahwa parameter seperti dimensi vektor, *window size*, dan epoch sangat mempengaruhi kualitas *semantic Similarity*, terutama dalam konteks bahasa non-Inggris seperti Arab. Selain itu, penelitian oleh Darwish et al. (2020) melalui “*A Panoramic Survey of NLP in the Arab World*” menyoroti bahwa meskipun penelitian NLP dalam bahasa Arab meningkat, terdapat kesenjangan besar antara kebutuhan dan teknologi yang tersedia.

Dengan latar belakang tersebut, penelitian ini menggunakan FastText sebagai model *word embedding* untuk merepresentasikan kata dalam teks Al-Qur'an berbahasa Arab, dan berfokus pada analisis pengaruh variasi hyperparameter terhadap kualitas *semantic similarity*. Evaluasi dilakukan secara kuantitatif menggunakan *cosine Similarity*, untuk mengetahui sejauh mana model dapat memberikan kontribusi dalam pengembangan sistem NLP berbasis bahasa Arab, khususnya untuk kebutuhan aplikasi seperti sistem pencarian Al-Qur'an, klasifikasi tematik, hingga penerjemah kontekstual berbasis makna.

Dalam penelitian NLP berbahasa Arab, teks Al-Qur'an sering menjadi sumber data utama. Al-Qur'an memiliki struktur bahasa yang kompleks, kaya makna, dan bervariasi bentuk katanya mengikuti aturan morfologi khas. Bahasa Arab Al-Qur'an berbeda dari bahasa Arab modern, sehingga memerlukan pendekatan khusus dalam pemrosesan dan representasi katanya. Tantangan utamanya meliputi kompleksitas morfologi, pola akar kata, sistem afiks, dan makna kata yang sangat bergantung pada konteks ayat [3]. Hal ini sejalan dengan firman Allah SWT dalam Surat Yusuf ayat 2 :

إِنَّا أَنْزَلْنَاهُ قُرْآنًا عَرَبِيًّا لَعَلَّكُمْ تَعْقِلُونَ ۚ

Artinya: “Sesungguhnya Kami menurunkan Al-Qur'an berbahasa Arab, agar kamu memahaminya.” (Q.S Yusuf: 2)

Ayat ini menekankan pentingnya pemahaman bahasa Arab sebagai kunci dalam memahami kandungan Al-Qur'an. Pemahaman tersebut menjadi semakin penting ketika dikaitkan dengan perkembangan teknologi pemrosesan bahasa yang dapat memudahkan akses terhadap makna dan konteks ayat. Pentingnya bahasa Arab dalam memahami ajaran Islam juga dipertegas dalam berbagai riwayat, sebagaimana sabda Rasulullah Shallallahu 'alaihi wa sallam:

”تَعَلَّمُوا الْعَرَبِيَّةَ ، فَإِنَّهَا جُزْءٌ مِنْ دِينِكُمْ”

Artinya: “Belajarlah bahasa Arab, karena ia adalah bagian dari agamamu.” (Hadits ini diriwayatkan oleh Imam Al-Baihaqi dalam Syu'abul Iman, dan dishahihkan oleh Al-Albani dalam Silsilah Al-Ahadits Ash-Shahihah no. 1297).

Hadits ini menekankan pentingnya mempelajari bahasa Arab bagi umat Muslim karena keterkaitannya yang erat dengan pemahaman agama, termasuk Al-Qur'an. Hal ini semakin menguatkan relevansi penelitian ini dalam upaya meningkatkan pemahaman terhadap teks Al-Qur'an melalui teknologi NLP. Lebih dari itu, semangat menuntut ilmu yang mendasari penelitian ini juga didasari firman Allah dalam Surah Thaha ayat 114:

رَبِّ زِدْنِي عِلْمًا ۝ ١١٤

Artinya: “Ya Tuhanku, tambahkanlah kepadaku ilmu pengetahuan.” (Q.S Thaha: 114)

Melalui teknologi NLP dan model seperti FastText, pemeliharaan makna dan konteks Al-Qur'an dapat diupayakan secara ilmiah dan sistematis, demi memperluas akses terhadap pemahaman yang lebih mendalam. Rasulullah SAW juga bersabda:

"خَيْرُكُمْ مَنْ تَعَلَّمَ الْقُرْآنَ وَعَلَّمَهُ"

"Sebaik-baik kalian adalah yang mempelajari Al-Qur'an dan mengajarkannya." (H.R Tirmidzi No. 2907)

Dengan demikian, penelitian ini tidak hanya memiliki kontribusi akademik dalam pengembangan NLP bahasa Arab, tetapi juga bernilai ibadah dan dakwah dalam memperluas pemahaman terhadap Al-Qur'an dengan pendekatan teknologi yang ilmiah dan terstruktur.

Seiring dengan kompleksitas bahasa Arab dalam Al-Qur'an yang kaya makna, sistem akar kata, dan morfologi yang bervariasi, maka dibutuhkan pendekatan teknologi yang mampu merepresentasikan kata secara kontekstual. Di sinilah peran penting model representasi kata seperti Word2Vec dan pengembangannya, yaitu FastText, menjadi sangat relevan.

Namun, penerapan Word2Vec pada bahasa arab, khususnya dalam teks Al-Qur'an memiliki keterbatasan yang disebabkan oleh kompleksitas morfologi, struktur sintaksis yang berbeda, serta keterbatasan dataset yang tersedia, sehingga diperlukan metode representasi kata yang lebih akurat [4]. Bahasa Arab adalah bahasa flektif yang sangat kaya, di mana satu akar kata dapat menghasilkan banyak bentuk kata berbeda.

Oleh karena itu, penelitian ini menggunakan FastText, pengembangan dari Word2Vec yang tidak hanya merepresentasikan makna secara utuh tetapi juga mempertimbangkan karakter sub kata (n-gram). Pendekatan ini diharapkan lebih mampu menangkap makna kata berdasarkan bentuk morfologisnya, yang sangat relevan untuk bahasa Arab [5].

Penggunaan FastText diharapkan meningkatkan pemahaman hubungan semantik antar kata dalam teks Al-Qur'an. Kualitas representasi kata sangat dipengaruhi oleh konfigurasi hyperparameter seperti *vektor size*, *window size*, *minimum count*, *epoch*, dan metode pelatihan [6]. Pengaturan hyperparameter yang

tepat dapat secara signifikan meningkatkan kinerja model dalam memahami teks Al-Qur'an dan pada akhirnya meningkatkan akurasi tugas-tugas NLP lainnya.

Selain itu, penelitian terkait NLP dalam bahasa Arab terutama dalam pemrosesan teks Al-Qur'an, masih tergolong kurang dibandingkan dengan bahasa lain. Dengan meningkatnya permintaan terhadap aplikasi berbasis NLP yang mendukung bahasa arab, seperti mesin pencari berbasis Al-Qur'an, dan sistem terjemahan otomatis, penelitian ini diharapkan dapat berkontribusi dalam mengembangkan model representasi kata yang lebih baik untuk bahasa Arab. Temuan ini dapat menjadi acuan bagi pengembang sistem NLP dan peneliti, dalam mengoptimalkan pemrosesan bahasa Arab [6].

Dengan tetap merujuk pada kerangka Word2Vec, penelitian ini bertujuan untuk mengevaluasi bagaimana penerapan dan evaluasi FastText dengan berbagai konfigurasi hyperparameter dapat mempengaruhi hasil pemrosesan bahasa alami untuk dataset bahasa Arab, khususnya dalam teks Al-Qur'an.

1.2 Rumusan Masalah

Adapun rumusan masalah yang akan dikaji pada penelitian ini sebagai berikut:

1. Bagaimana pengaruh konfigurasi hyperparameter dalam FastText terhadap kualitas representasi kata pada pemrosesan bahasa alami menggunakan dataset Al-Qur'an bahasa Arab?
2. Bagaimana efektivitas FastText dalam merepresentasikan relasi semantik antar kata dalam teks Al-Qur'an, serta dampaknya terhadap akurasi evaluasi *semantic similarity*?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini, sebagai berikut:

1. Penelitian ini menggunakan FastText sebagai model word embedding, yang dikembangkan dari Word2Vec.
2. Penggunaan Hyperparameter berfokus pada utama *vektor size*, *window size*, *learning rate*, *minimum count*, dan jumlah iterasi (epoch).
3. Penelitian ini menggunakan dataset Al-Qur'an bahasa Arab.

4. Evaluasi yang dilakukan berbentuk representasi kata dalam ruang vektor dan hubungan *semantic similarity*.
5. Penelitian ini hanya memfokuskan analisis *semantic similarity* pada kata target خير.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Menganalisis pengaruh hyperparameter FastText (*vector size*, *windows size*, jumlah iterasi (*epoch*), *minimum count*, dan metode pelatihan) terhadap kualitas representasi kata dalam pemrosesan teks Al-Qur'an bahasa Arab, serta menentukan konfigurasi yang optimal untuk meningkatkan akurasi *semantic similarity*.
2. Mengevaluasi efektivitas FastText dalam merepresentasikan relasi semantik antar kata dalam teks Al-Qur'an dan dampaknya terhadap hasil pengukuran *semantic similarity*.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Memberikan kontribusi ilmiah dalam bidang pemrosesan bahasa alami (*Natural Language Processing*), khususnya dalam pengembangan representasi kata berbasis FastText untuk bahasa Arab Al-Qur'an. Penelitian ini dapat menjadi referensi bagi studi lanjutan terkait pemilihan hyperparameter optimal dalam *word embedding*.
2. Memberikan dasar implementasi model FastText yang lebih efektif untuk digunakan dalam berbagai aplikasi berbasis bahasa Arab, seperti sistem pencarian makna Al-Qur'an, klasifikasi tematik ayat, dan penerjemahan kontekstual.
3. Mendukung upaya pemahaman Al-Qur'an secara lebih mendalam melalui teknologi, sebagai bentuk kontribusi keilmuan dalam memahami kandungan wahyu dengan pendekatan semantik dan kontekstual.

1.6 Metode Penelitian

Metodologi dalam penelitian ini di antaranya:

1. Studi Literatur

Pada tahap studi literatur ini bertujuan untuk mengumpulkan informasi dari berbagai sumber seperti jurnal, buku, dan penelitian sebelumnya yang berkaitan dengan Word2Vec dan FastText, *word embedding*, hyperparameter, karakteristik bahasa Arab, dan evaluasi model.

2. Penelitian

Pada tahap penelitian ini penulis melakukan tahap *pre-processing* data Al-Qur'an bahasa Arab menggunakan algoritma FastText dan dilatih menggunakan berbagai kombinasi hyperparameter. Setelah pelatihan, model dievaluasi. Hasil evaluasi dianalisis untuk menentukan konfigurasi hyperparameter yang optimal dalam meningkatkan kualitas representasi kata.

1.7 Sistematika Penulisan

Pada skripsi ini terdapat lima bab sistematika penulisan di antaranya:

BAB I PENDAHULUAN

Bab pendahuluan ini berisi latar belakang masalah, rumusan masalah, tujuan penelitian, metode penelitian, dan sistematika penulisan dari masalah yang dikaji.

BAB II LANDASAN TEORI

Bab landasan teori menjelaskan tentang teori-teori yang melandasi pembahasan inti yang saling berkaitan dan sebagai penunjang dalam penulisan skripsi, seperti *Natural Language Processing (NLP)*, *Word Embedding*, *Word2Vec*, *FastText*, hyperparameter, serta metode evaluasi.

BAB III PENGARUH HYPERPARAMETER DALAM WORD2VEC TERHADAP SEMANTIC SIMILARITY KATA MENGUNAKAN DATASET AL-QUR'AN BAHASA ARAB

Pada bab ini berisi pembahasan tentang penelitian yang dilakukan dari pengambilan dataset Al-Qur'an, lalu tahap *text pre-processing*

dengan metode algoritma FastText dan dilatih menggunakan berbagai kombinasi hyperparameter. Setelah pelatihan, model dievaluasi menggunakan *cosine similarity*. Hasil evaluasi dianalisis untuk menentukan konfigurasi hyperparameter yang optimal dalam meningkatkan kualitas representasi kata dalam teks Al-Qur'an.

BAB IV EKSPERIMEN DAN ANALISIS

Bab ini berisi pemaparan mengenai analisis hasil yang telah dilakukan pada bab sebelumnya. Hasil evaluasi dari *cosine similarity* digunakan sebagai dasar dalam menilai efektivitas model yang dilatih. Dari hasil analisis, konfigurasi optimal dapat diidentifikasi untuk meningkatkan akurasi dalam pemrosesan bahasa alami berbasis bahasa Arab. Hasil ini diharapkan dapat menjadi acuan dalam penelitian lebih lanjut dan pengembangan aplikasi NLP yang lebih efektif.

BAB V PENUTUP

Bab penutup berisi hasil simpulan dari rumusan masalah yang telah dijelaskan dan berisi saran yang diperuntukkan untuk penelitian berikutnya sebagai pengembangan dari Word2Vec dan FastText.

DAFTAR PUSTAKA Bagian ini berisi daftar seluruh sumber referensi yang menjadi acuan, seperti jurnal ilmiah, buku dan penelitian sebelumnya, yang digunakan sebagai landasan teori dan penguat dalam penyusunan skripsi ini

BAB II

LANDASAN TEORI

2.1 *Natural Language Processing (NLP)*

Natural language processing adalah bidang dalam kecerdasan buatan yang memungkinkan komputer memahami dan berinteraksi dengan bahasa manusia. Bahasa dapat diartikan sebagai sekumpulan aturan atau simbol yang berfungsi untuk menyampaikan informasi. Karena, tidak semua orang memahami bahasa khusus yang digunakan oleh mesin, NLP menyediakan teknik dan alat yang memungkinkan manusia berinteraksi dengan komputer menggunakan bahasa mereka sendiri, tanpa perlu mempelajari bahasa komputer secara khusus dan mendalam [7].

NLP bekerja untuk menangkap makna dari bahasa manusia, mengidentifikasi pola linguistik, dan mengubah data linguistik menjadi representasi yang dapat dipahami oleh mesin. Beberapa tugas utama NLP meliputi analisis sintaksis, penandaan *part-of-speech*, resolusi koreferensi (menghubungkan kata ganti dengan referensinya), analisis semantik, dan generasi bahasa alami [8]. NLP digunakan dalam berbagai aplikasi termasuk penerjemahan mesin, chatbot, analisis sentimen, dan ekstraksi informasi. Dalam penelitian ini, NLP diterapkan untuk memproses bahasa Arab dan memahami hubungan antar kata menggunakan model Word2Vec.

Salah satu tantangan utama dalam NLP bahasa Arab adalah kompleksitas sistem morfologi dan sintaksisnya. Tidak seperti bahasa Inggris yang relatif sederhana dalam struktur katanya, bahasa Arab memiliki akar kata dan pola derivasi yang dapat menghasilkan berbagai bentuk kata dari satu akar yang sama. Hal ini menyebabkan NLP dalam bahasa Arab menjadi lebih sulit dibandingkan dengan bahasa lainnya [9]. Dalam penelitian ini, NLP digunakan untuk mengembangkan representasi kata yang lebih bermakna dalam bahasa Arab menggunakan pengembangan Word2Vec yang disebut dengan FastText. Dengan memilih hyperparameter yang optimal, diharapkan hasil representasi kata yang diperoleh dapat lebih baik dalam menangkap hubungan semantik dan sintaksis dalam bahasa Arab sehingga model dapat digunakan dalam berbagai tugas NLP lainnya.

2.2 Karakteristik Bahasa Arab dalam NLP

Bahasa Arab memiliki karakteristik unik yang membedakan dari bahasa lain, sehingga memerlukan pendekatan khusus dalam pemrosesan bahasa alami. Berikut beberapa aspek utama yang mempengaruhi pengolahan teks bahasa Arab dalam NLP:

1. Sistem Morfologi yang Kaya

Bahasa Arab menggunakan sistem morfologi berbasis akar (*root-based morphology*), di mana sebagian besar kata berasal dari akar tiga huruf yang membentuk berbagai pola kata. Misalnya, akar “ك - ت - ب” dapat membentuk kata “كتب” (kataba, menulis), “كاتب” (kātib, penulis), dan “مكتوب” (maktūb, tertulis). Sistem ini memungkinkan pembentukan banyak kata dari akar yang sama, sehingga menambah kompleksitas dalam NLP [10].

2. Penulisan dari Kanan ke Kiri

Bahasa Arab menggunakan sistem penulisan *Right-to-Left* (RTL) yang berbeda dengan bahasa latin yang ditulis dari kiri ke kanan. Hal ini mempengaruhi cara teks ditampilkan dan diproses dalam sistem komputer. Sebagian besar perangkat lunak NLP harus menyesuaikan tampilan dan pemrosesan teks agar mendukung penulisan RTL [11].

3. Bentuk Huruf yang Berbeda Sesuai Posisi dalam Kata

Huruf dalam bahasa Arab memiliki bentuk yang berubah tergantung pada posisinya dalam kata (awal, tengah, akhir, atau terisolasi). Misalnya huruf “ع” dalam kata “علم” memiliki bentuk berbeda dibandingkan dengan “ع” pada kata “سعيد”. Perubahan ini dapat mempersulit tokenisasi dan pencocokan kata dalam NLP [12].

4. Ketidadaan Huruf Kapital dan Variasi Penulisan

Tidak seperti bahasa latin yang menggunakan huruf kapital untuk membedakan nama atau awal kalimat, bahasa arab tidak memiliki huruf kapital. Hal ini menyebabkan kesulitan dalam deteksi entitas bernama (*Named Entity Recognition/NER*) karena tidak ada petunjuk visual yang membedakan nama orang, tempat, atau organisasi dari kata-kata umum [13].

5. Prefiks dan Sufiks yang Kompleks

Bahasa Arab memiliki berbagai prefiks dan sufiks yang melekat pada kata untuk menunjukkan waktu, kepemilikan, atau bentuk gramatikal lainnya. Sebagai contoh, kata “وكتبوا” terdiri dari “و” (dan), “كتب” (menulis), dan “وا” (mereka). Proses segmentasi kata menjadi unit dasar sangat penting dalam NLP untuk memahami struktur kata dengan benar [14].

6. Penggunaan Diakritik (Harakat) dan Variasi dalam Teks

Dalam penulisan bahasa Arab, diakritik (harakat) seperti fathah, kasrah dan dhammah digunakan untuk menentukan vokal dalam kata. Namun, dalam banyak teks modern, diakritik sering dihilangkan, menyebabkan ambiguitas dalam membaca dan memahami kata. Sebagai contoh, kata “علم” dapat dibaca sebagai “ilm” atau “allam” tergantung pada diakritiknya. Sistem NLP harus mampu mengatasi ambiguitas ini dengan pendekatan berbasis konteks [15].

7. Ambiguitas Leksikal dan Konteks yang Kuat

Bahasa Arab memiliki banyak kata yang memiliki lebih dari satu arti tergantung pada konteksnya. Misalnya kata “عين” bisa berarti “mata”, “sumber air”, atau “mata-mata” tergantung pada penggunaannya dalam kalimat. Oleh karena itu, model NLP untuk bahasa Arab harus mampu mempertimbangkan konteks secara mendalam untuk menghasilkan pemahaman yang akurat [16].

8. Pengaruh Dialek dalam NLP

Bahasa Arab memiliki banyak dialek yang berbeda secara signifikan dari bahasa Arab Standar Modern (*Modern Standard Arabic/MSA*). Misalnya, kata “bagaimana” dalam MSA adalah “كيف”, sementara dalam dialek Mesir menjadi “إزاي”. NLP dalam bahasa Arab sering kali harus mempertimbangkan perbedaan ini agar dapat menangani teks dalam berbagai variasi bahasa Arab dengan baik [17].

2.3 Teknik *Pre-Processing*

Dalam pengolahan teks bahasa Arab menggunakan NLP, tahap *pre-processing* sangat penting untuk memastikan bahwa data yang digunakan bersih, terstruktur, dan siap diproses oleh model. *Pre-processing* bertujuan untuk menghilangkan informasi yang tidak relevan, menstandarisasi teks, serta menyiapkan data agar dapat diolah oleh algoritma NLP dengan lebih efektif. Komponen dari teknik *pre-processing* terdiri dari beberapa tahapan, yaitu tokenisasi, *teks cleaning*, *stop word removal*, *stemming* dan *lemmatization*.

2.3.1 *Tokenization* (Tokenisasi)

Tokenisasi adalah proses membagi teks menjadi unit-unit kecil yang disebut token, biasanya dalam bentuk kata atau sub kata. Bahasa Arab memiliki tantangan dalam tokenisasi karena kata-kata dalam bahasa ini sering kali memiliki imbuhan awalan (*prefix*), akhiran (*suffix*), atau kata sambung yang melekat pada kata utama.

Contoh tokenisasi dalam bahasa Arab pada kalimat “ذهب الولد الى المدرسة”, setelah di tokenisasi menjadi [“ذهب”, “الولد”, “الى”, dan “المدرسة”]. Untuk menangani tantangan ini, algoritma tokenisasi seperti frasa tokenizer atau pendekatan berbasis model *deep learning* sering digunakan untuk bahasa Arab [18].

2.3.2 *Text Cleaning* (Pembersihan Teks)

Text cleaning atau pembersihan teks adalah tahapan penting dalam *pre-processing* yang bertujuan untuk menghilangkan karakter atau elemen yang tidak relevan dalam teks, seperti tanda baca, angka, simbol, dan diakritik. Dalam bahasa Arab, *text cleaning* menjadi lebih kompleks dibandingkan dengan bahasa lain karena adanya fitur khusus seperti bentuk huruf yang berubah berdasarkan posisi, diakritik, serta karakter tambahan yang dapat muncul dalam berbagai format tulisan [10]. Langkah-langkah dari *text cleaning* sebagai berikut:

1. Penghapusan tanda baca dan simbol

Teks yang berasal dari sumber seperti situs web, media sosial, atau dokumen digital sering mengandung tanda baca dan simbol yang tidak diperlukan untuk analisis NLP. Misalnya, tanda baca seperti titik (.), koma (,), tanda seru (!), atau tanda tanya (?) dapat dihapus untuk mendapatkan teks yang lebih bersih.

2. Penghapusan angka

Dalam teks berbahasa Arab, sering kali muncul dalam dua format, angka Arab Timur (٠ ١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩) dan angka Arab modern (0 1 2 3 4 5 6 7 8 9). Penghapusan angka ini penting terutama jika analisis NLP hanya berfokus pada kata-kata dan tidak memerlukan data numerik.

3. Penghapusan diakritik

Diakritik dalam bahasa Arab sering digunakan dalam teks keagamaan atau pendidikan, tetapi sering kali dihilangkan dalam teks sehari-hari. Penghapusan diakritik membantu mengurangi kompleksitas dalam analisis NLP karena banyak kata dalam bahasa arab memiliki arti yang tetap meskipun tanpa diakritik.

4. Normalisasi huruf arab

Beberapa huruf arab memiliki variasi ejaan yang berbeda dalam teks digital. Misalnya, huruf alif (ا) dapat muncul sebagai (ا, إ, ؤ), dan huruf ya (ي) dapat muncul sebagai (ى). Normalisasi bertujuan untuk menyamakan bentuk huruf sehingga konsistensi dalam analisis tetap terjaga.

2.3.3 *Stopword Removal* (Penghapusan Kata Umum)

Stopword adalah kata-kata yang sering muncul dalam sebuah bahasa tetapi tidak memberikan makna informasi bermakna bagi analisis, seperti kata hubung (*conjunctions*), kata depan (*preposition*), dan kata ganti (*pronouns*). Dalam bahasa Arab, *stoword* mencakup kata-kata seperti, kata hubung (و, أو, لكن), kata depan (في, من, على), dan kata ganti (أنا, هو, هي). *Stopword* sering kali dihapus dalam analisis NLP karena tidak memberikan nilai informatif dalam banyak aplikasi, seperti *text clasification*, *sentiment analysis*, dan *information retrieval*.

2.3.4 *Text Normalization* (Normalisasi Teks)

Text normalization (normalisasi teks) adalah proses standarisasi teks dengan menghilangkan variasi dalam penulisan kata yang tidak mempengaruhi maknanya. Dalam bahasa Arab, normalisasi sangat penting karena adanya banyak bentuk tulisan untuk huruf atau kata yang sama. Tujuan dari normalisasi teks, yaitu menyederhanakan teks untuk mengurangi kompleksitas pemrosesan, meningkatkan konsistensi data sebelum dianalisis oleh model NLP, dan menghilangkan variasi ejaan yang tidak diperlukan.

2.3.5 Lemmatization

Lemmatization adalah teknik yang digunakan untuk mengubah kata ke bentuk dasarnya atau bentuk lema (*dictionary form*). Proses ini mempertimbangkan struktur morfologi, konteks gramatikal, dan aturan bahasa untuk menentukan bentuk kata yang paling representatif secara semantik.

Dalam bahasa Arab, *lemmatization* menjadi sangat penting karena karakteristik morfologinya yang kompleks, di mana satu akar kata dapat membentuk berbagai turunan kata dengan makna yang masih berkaitan. Teknik ini membantu menyatukan variasi bentuk kata menjadi satu representasi dasar, sehingga model dapat mengenali hubungan semantik secara lebih akurat.

Contohnya:

- "يكتبون" → "كتب"
- "كتبت" → "كتب"
- "مكتوب" → "كتب"
- "كاتب" → "كتب"

Semua kata di atas berasal dari akar kata **كتب** yang berarti "menulis". Dengan *lemmatization*, perbedaan bentuk kata yang disebabkan oleh perubahan waktu, jenis pelaku, atau struktur gramatikal dapat dinormalkan ke bentuk inti yang sama. Ini memungkinkan sistem NLP untuk memahami konteks makna secara lebih dalam dan konsisten.

2.4 Word Embedding

Konsep *Word Embedding* (WE) pertama kali diperkenalkan oleh Hinton pada tahun 1986 melalui jurnal berjudul "*Learning Distributed Representations of Concepts*". WE, yang juga dikenal sebagai *word representation*, mencakup berbagai model bahasa dan metode pemilihan fitur untuk merepresentasikan suatu objek atau data. Tujuan utamanya adalah mengubah kata atau frasa dalam teks menjadi vektor berdimensi rendah [19].

Seiring kemajuan teknologi, *Word Embedding* telah menjadi metode yang populer untuk merepresentasikan dokumen dan *query* dalam sistem *Information Retrieval* (IR). Metode ini memberikan representasi yang lebih mendalam dan kaya, sehingga mampu menangkap hubungan antar kata berdasarkan makna semantiknya [20]. *Word Embedding* tidak hanya mengkodekan informasi semantik yang

berkaitan dengan makna kata, tetapi juga informasi sintaksis yang mencerminkan peran struktural kata tersebut [6]. Sintaksis merujuk pada struktur tata bahasa yang mengatur bagaimana kata-kata disusun untuk membentuk kalimat yang bermakna, seperti subjek, predikat, dan objek dalam sebuah kalimat. Sementara itu, semantik berfokus pada makna atau kalimat, mencakup hubungan antar kata, seperti sinonim, antonim, atau asosiasi makna. Kombinasi antara sintaksis dan semantik memungkinkan model NLP untuk menangkap informasi penting dari teks, yang kemudian digunakan dalam berbagai metode representasi kata.

Word embedding adalah alat yang sangat berguna dalam berbagai tugas NLP, terutama yang memerlukan teks sebagai fitur utama. Ada beberapa jenis model untuk membuat representasi kata ini, masing-masing dengan kelebihan dan kekurangannya. Karena *word embedding* dapat dianggap sebagai representasi fitur dan teks, proses ini sering dilakukan sebagai bagian dari tahap prapemrosesan dalam tugas NLP lanjutan [20].

2.4.1 Word2Vec

Salah satu teknik paling populer dalam *word embedding* adalah Word2Vec, yang dikembangkan oleh Mikolov pada tahun 2013. Word2Vec mempresentasikan kata sebagai vektor yang dapat mencerminkan makna semantik dari kata tersebut. Model ini merupakan aplikasi pembelajaran tanpa pengawasan (*unsupervised learning*) yang menggunakan jaringan saraf sederhana dengan satu *hidden layer* dan *fully connected layer*. Dimensi matriks bobot pada setiap layer ditentukan oleh jumlah kata dalam korpus yang dikalikan dengan jumlah neuron pada *hidden layer*.

Setelah model dilatih, matriks bobot pada *hidden layer* digunakan untuk mengubah kata menjadi vektor. Matriks ini berfungsi seperti tabel pencarian (*lookup table*), di mana setiap baris mempresentasikan sebuah kata, dan setiap kolom mencerminkan komponen vektor dari kata tersebut. Pendekatan ini memungkinkan Word2Vec untuk menangkap hubungan semantik di antara kata-kata dalam korpusnya [9].

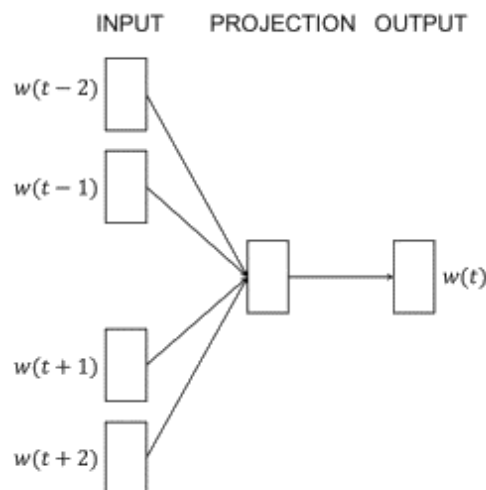
Word2Vec memiliki dua arsitektur utama, yaitu Continuous Bag of Words (CBOW) dan Skip-gram.

1. *Continuous Bag of Words (CBoW)*

Continuous Bag of Words (CBOW) memprediksi kata target berdasarkan kata-kata konteks di sekitarnya. CBOW lebih cepat dalam pelatihan dan cocok untuk data set besar, karena ia memanfaatkan konteks untuk memprediksi kata yang hilang. CBoW bekerja dengan menggabungkan semua kata konteks dalam satu vektor yang mewakili keseluruhan konteks tersebut [9].

Salah satu keunggulan utama dari CBoW adalah kecepatannya dalam pelatihan, yang membuatnya sangat cocok untuk digunakan pada dataset yang besar. Karena CBow menggunakan informasi dari beberapa kata konteks secara bersamaan, ia dapat memproses data dengan lebih efisien dibandingkan dengan metode lainnya. Pendekatan ini juga memiliki keunggulan dalam menangkap hubungan semantik antara kata target dan konteksnya, meskipun mungkin kurang efektif dalam menangani kata-kata yang jarang muncul di dataset.

Misalnya, jika memiliki kalimat “Saya sedang makan apel merah”, dan ingin memprediksi kata “makan” sebagai kata target, CBoW akan menggunakan kata-kata konteks seperti “Saya”, “sedang”, “apel”, dan “merah” untuk menghasilkan vektor representasi yang mencakup informasi dari semua kata tersebut. Dengan memanfaatkan rata-rata vektor konteks ini, model kemudian memprediksi bahwa kata target yang paling sesuai adalah makan.



Gambar 2. 1 Arsitektur CBoW

Selain itu, CBoW cenderung lebih stabil dalam pelatihan karena sifatnya yang memanfaatkan beberapa kata konteks sekaligus. Hal ini membantu mengurangi

pengaruh dari kata-kata yang jarang muncul atau konteks yang tidak lengkap, sehingga hasil pelatihan menjadi lebih andal. Karena kemampuannya ini, CBoW sering digunakan dalam berbagai aplikasi pemrosesan bahasa alami, termasuk pembuatan *embedding* kata, klasifikasi teks, dan analisis sentimen.

Meskipun demikian, ada beberapa keterbatasan pada CBoW, terutama dalam menangkap hubungan yang lebih kompleks atau non-linear antara kata target dan konteksnya. Untuk mengatasi hal ini, arsitektur lain seperti skip-gram dapat digunakan sebagai pelengkap dalam model Word2Vec. Secara keseluruhan, CBoW tetap menjadi salah satu pendekatan yang sangat efisien dan populer dalam pemrosesan data teks skala besar.

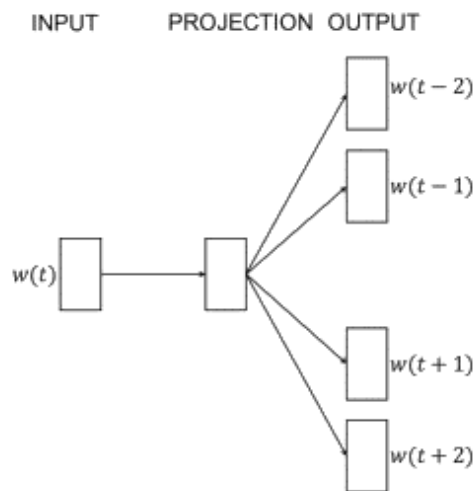
2. Skip-gram

Skip-gram memprediksi kata-kata konteks berdasarkan kata target. Skip-gram lebih efektif dalam menangkap makna kata yang jarang ditemui dan menghasilkan representasi yang lebih baik untuk kata-kata tersebut, terutama dalam korpus yang tidak seimbang. Yang berarti dataset yang digunakan dalam pelatihan tidak merata [1]. Skip-gram memanfaatkan kata target untuk memprediksi kata-kata konteks di sekitarnya dalam rentang jendela tertentu. Hal ini dilakukan dengan tujuan agar representasi kata dapat mencerminkan hubungan semantik dengan kata-kata lain yang berada dalam konteks yang sama [9].

Salah satu keunggulan Skip-gram adalah kemampuannya untuk menghasilkan representasi vektor yang berkualitas tinggi untuk kata-kata yang jarang ditemui dalam korpus. Hal ini disebabkan oleh cara kerja skip-gram, yang memberikan fokus lebih pada pasangan kata target dan kata konteks tertentu dalam jendela yang telah ditentukan. Jendela ini dapat diatur untuk mencakup beberapa kata sebelum dan sesudah kata target, memungkinkan model untuk memahami hubungan semantik antara kata target dan lingkungan kata-kata di sekitarnya. Sebagai contoh, jika kata target adalah “makan”, skip-gram dapat memprediksi kata-kata konteks seperti “saya”, “sedang”, “apel”, dan “merah”, yang berada dalam jangkauan jendela konteks tersebut.

Pendekatan ini juga sangat fleksibel dalam menangani korpus yang tidak seimbang, di mana beberapa kata muncul jauh lebih sering daripada yang lain. Dalam kasus seperti ini, skip-gram tetap mampu menghasilkan representasi yang

baik karena setiap pasangan kata target dan konteks dipelajari secara individu, tanpa mengandalkan rata-rata dari seluruh konteks. Dengan cara ini, skip-gram mampu menangkap makna dan hubungan yang mendalam, bahkan untuk kata-kata yang jarang muncul.



Gambar 2. 2 Arsitektur Skip-gram.

Selain itu, skip-gram sering dilatih dengan algoritma optimasi seperti Negative Sampling atau Hierarchical Softmax, yang dirancang untuk meningkatkan efisiensi dan kecepatan pelatihan. Negative Sampling misalnya, memilih subset kata negatif secara acak untuk mengurangi jumlah komputasi yang diperlukan, sementara Hierarchical Softmax menggunakan struktur pohon biner untuk mempercepat perhitungan probabilitas kata-kata konteks. Kombinasi dari pendekatan ini memungkinkan skip-gram bekerja dengan baik pada dataset yang besar dan kompleks.

Dengan kemampuan menangkap hubungan semantik dan menghasilkan representasi yang akurat, skip-gram juga menjadi pilihan yang sangat baik untuk berbagai aplikasi pemrosesan bahasa alami, seperti analisis sentimen, klasifikasi teks, dan pemodelan topik. Selain itu, pendekatan ini juga sering digunakan untuk membangun *embedding* kata, yang menjadi dasar bagi banyak teknik pembelajaran mendalam dalam pemrosesan bahasa alami.

2.4.2 FastText

FastText adalah model pembelajaran representasi kata yang dikembangkan oleh Facebook AI Research sebagai perbaikan atas model Word2Vec. Model ini pertama kali diperkenalkan oleh Joulin et.al pada tahun 2016 sebagai alternatif yang lebih efisien dan akurat untuk pemodelan teks, terutama pada bahasa dengan morfologi kompleks seperti bahasa Arab [21]. FastText memperluas pendekatan Word2Vec dengan memperhitungkan struktur internal kata melalui penggunaan sub-kata (*character n-gram*), yang menjadikannya lebih fleksibel dalam menangani kata baru yang belum pernah dilihat sebelumnya (*out-of-vocabulary*) [1].

Seperti Word2Vec, FastText menggunakan dua arsitektur utama dalam pembentukan representasi kata, yaitu *Continous Bag of Words* (CBoW) dan Skip-gram. Namun, perbedaan utama FastText terletak pada cara model ini memproses kata menjadi serangkaian n-gram karakter sebelum menghasilkan vektor akhir [22]. Alih-alih menganggap kata sebagai satu unit, FastText memecah kata menjadi sekumpulan n-gram karakter yang tumpang tindih, kemudian menghitung representasi kata berdasarkan kombinasi dari vektor n-gram ini [23].

Sebagai contoh, kata “كتبوا” (yang berarti menulis) akan diuraikan menjadi potongan sub-kata seperti “كتب”, “تبا”, dan “وا”. Setiap n-gram ini memiliki vektornya sendiri dan representasi akhir kata dibentuk dengan menjumlahkan vektor-vektor sub-kata ini bersama dengan vektor kata itu sendiri [24]. Dengan cara ini, FastText mampu memahami pola morfologi seperti prefiks, sufiks, dan akar kata, yang sangat penting dalam bahasa yang memiliki struktur morfologis kompleks [25].

FastText memiliki beberapa keunggulan utama dibandingkan model representasi kata lainnya, yaitu:

1. Mampu Memahami Morfologi Bahasa

Dengan memperhitungkan sub-kata, FastText dapat lebih baik dalam memahami makna kata dalam bahasa yang kaya morfologi seperti bahasa Arab. Ini berbeda dengan Word2Vec yang hanya memperlakukan kata sebagai unit tunggal tanpa memperhatikan komponen internalnya [26].

2. Menangani Kata *Out-of-Vocabulary*

Karena FastText menggunakan sub-kata, ia dapat menghasilkan representasi untuk kata yang belum pernah muncul dalam data pelatihan, sehingga lebih robust dalam aplikasi dunia nyata di mana kata baru sering muncul [27].

3. Akurasi yang Lebih Tinggi pada Bahasa Kompleks

Model ini lebih akurat dalam menangani bahasa dengan morfologi yang kaya dan struktur tata bahasa yang kompleks, menjadikannya pilihan yang lebih baik untuk banyak tugas NLP dalam konteks non-Latin [28].

2.5 Hyperparameter

Hyperparameter adalah parameter yang ditentukan sebelum proses pelatihan model dan memainkan peran penting dalam menentukan performa model. Algoritma ini bertujuan menghasilkan representasi vektor untuk kata-kata dalam teks, di mana kualitas representasi tersebut sangat bergantung pada pemilihan hyperparameter. Dengan memahami setiap hyperparameter secara mendalam, kita dapat mengoptimalkan model untuk berbagai aplikasi, seperti analisis teks, klasifikasi dokumen dan rekomendasi [6].

Salah satu hyperparameter utama adalah dimensi vektor, yang menentukan jumlah elemen dalam representasi vektor dari setiap kata. Dimensi yang lebih tinggi dapat menangkap informasi semantik dan sintaksis yang lebih kompleks, tetapi juga meningkatkan risiko overfitting jika data pelatihan tidak memadai. Sebaliknya, dimensi yang terlalu rendah dapat mempercepat pelatihan, tetapi cenderung kehilangan informasi penting. Biasanya, dimensi antara 100 hingga 300 digunakan untuk mencapai keseimbangan antara kompleksitas dan representasi yang akurat [23].

Ukuran jendela (*window size*) adalah parameter lain yang menentukan jumlah kata yang dipertimbangkan di sekitar kata target yang digunakan untuk konteks saat pelatihan. Jendela yang lebih besar membantu menangkap hubungan semantik dalam konteks yang lebih luas, seperti asosiasi antar kata dalam kalimat panjang. Namun, ini juga dapat menambah noise pada model. Sebaliknya, jendela kecil lebih cocok untuk menangkap hubungan sintaksis lokal, seperti struktur gramatikal [29].

Selain itu, minimum count adalah parameter yang menetapkan ambang batas frekuensi kemunculan kata agar dimasukkan ke dalam model. Dengan mengabaikan kata-kata yang jarang muncul, model menjadi lebih efisien dan terhindar dari noise yang tidak relevan. Namun, jika ambang batas terlalu tinggi, kata-kata penting yang jarang muncul juga dapat terabaikan, sehingga berpotensi mengurangi kualitas model [29].

Hyperparameter lain seperti learning rate dan jumlah iterasi (epoch) juga memengaruhi performa model. Learning rate mengatur seberapa besar langkah pembaruan bobot selama pelatihan. Nilai yang terlalu tinggi dapat membuat model tidak stabil, sedangkan nilai yang terlalu rendah memperlambat proses pelatihan. Sementara itu, jumlah iterasi menentukan berapa kali model melihat dataset selama pelatihan. Iterasi, yang terlalu sedikit dapat membuat model underfit, sedangkan terlalu banyak dapat menyebabkan overfitting [2].

Pemilihan hyperparameter harus disesuaikan dengan ukuran dataset, tujuan aplikasi, dan sumber daya komputasi yang tersedia. Dengan pengaturan yang tepat, FastText dapat menghasilkan representasi kata yang kaya dan dapat diandalkan untuk berbagai tugas pemrosesan bahasa alami.

2.6 *N-gram* dalam Pemrosesan Teks

Dalam ranah pemrosesan bahasa alami (*Natural Language Processing*), *n-gram* didefinisikan sebagai urutan n item yang berurutan dari suatu sampel teks atau ucapan. Item-item ini dapat bervariasi mulai dari fonem, suku kata, karakter, hingga kata, bergantung pada konteks dan tujuan aplikasi. Konsep *n-gram* memegang peranan esensial dalam berbagai aplikasi NLP karena kemampuannya dalam menangkap konteks lokal serta pola-pola repetitif yang terdapat dalam data linguistik. Klasifikasi *n-gram* secara umum didasarkan pada jumlah item (n) yang membentuknya:

Tabel 2. 1 Contoh N-gram

Jumlah N	Contoh
Unigram (N=1)	“Al-Quran”, “adalah”, “petunjuk”
Bigram (N=2)	“Al-Qur’an adalah”, “adalah petunjuk”
Trigram (N=3)	“Al-Quran adalah petunjuk”

Semakin tinggi nilai n , semakin banyak konteks yang dipertimbangkan dalam analisis teks. Namun, nilai n yang terlalu besar dapat meningkatkan kompleksitas komputasi dan memerlukan lebih banyak data pelatihan untuk mencapai hasil optimal. Peran n -gram sangat signifikan dalam beragam aplikasi NLP, meliputi:

1. *Pemodelan Bahasa (Language Modeling)*

N -gram merupakan fondasi dari model bahasa probabilistik, yang digunakan untuk memprediksi kemunculan kata berikutnya dalam suatu urutan atau untuk mengestimasi probabilitas suatu sekuens kata. Ini krusial dalam tugas-tugas seperti pengenalan suara, terjemahan mesin, dan koreksi ejaan.

2. *Ekstraksi Fitur*

N -gram dapat dimanfaatkan sebagai fitur representatif dalam berbagai model pembelajaran mesin. Sebagai ilustrasi, dalam tugas klasifikasi teks, keberagaman bigram tertentu dapat menjadi indikator kuat kategori dari suatu dokumen teks.

3. *Analisis Teks*

Digunakan untuk mengidentifikasi pola frasa, kolokasi (pasangan kata yang sering muncul bersama), atau bahkan untuk menganalisis dan mendeteksi gaya penulisan tertentu.

Berbeda dengan model *word embedding* konvensional seperti Word2Vec yang utamanya berfokus pada n -gram kata, FastText mengadopsi pendekatan yang lebih granular dengan memanfaatkan n -gram karakter. Ini berarti bahwa, memperlakukan “kata” sebagai unit terkecil, FastText menguraikan setiap kata menjadi serangkaian n -gram yang terbentuk dari karakter-karakter yang berurutan [30].

Sebagai contoh, jika mempertimbangkan kata “islam” dengan n -gram karakter berukuran 3 (trigram) dan 4 (quaddigram), FastText akan menghasilkan:

Tabel 2. 2 Contoh N -gram pada FastText

Jumlah N	Bentuk n -gram
N-gram 3 (trigram)	[“<is”, “isl”, “sla”, “lam”, “am>”]
N-gram 4 (quaddigram)	[“<isl”, “isla”, “slam”, “lam>”]

Setiap n-gram karakter ini akan memiliki representasi vektornya sendiri. Representasi vektor final untuk suatu kata kemudian dibentuk melalui proses penjumlahan vektor dari seluruh n-gram karakter penyusunan yang telah dilatih, ditambah dengan vektor dari kata utuhnya. Pendekatan n-gram karakter ini memberikan keunggulan substansi bagi FastText, khususnya untuk bahasa-bahasa dengan morfologi yang kompleks dan kaya seperti bahasa Arab [22].

Dengan demikian, pemanfaatan n-gram karakter merupakan fitur krusial yang menempatkan FastText sebagai pilihan model yang kuat untuk pemrosesan teks dalam konteks bahasa dengan struktur morfologi yang kompleks.

2.7 *Semantic Similarity*

Semantic Similarity adalah pendekatan untuk mengukur seberapa mirip makna antara dua kata atau lebih dalam konteks tertentu. Ini adalah aspek penting dalam *Natural Language Processing* (NLP) karena banyak tugas, seperti penilaian otomatis esai, deteksi plagiarisme, mesin pencari semantik, dan analisis sentimen, sangat bergantung pada kemampuan model untuk memahami kemiripan semantik antar kata atau kalimat [31].

Dalam konteks NLP, pengukuran *semantic similarity* dilakukan dengan membandingkan representasi vektor dari kata atau kalimat untuk melihat seberapa dekat makna mereka. Model pembelajaran kata seperti Word2Vec dan FastText telah banyak digunakan untuk tugas ini karena kemampuannya untuk menangkap hubungan semantik berdasarkan konteks penggunaan kata [1]. Namun, FastText memiliki keunggulan tambahan dalam memahami bahasa dengan morfologi kompleks, seperti bahasa Arab, dengan menggunakan sub-kata dalam representasinya [22].

Sebagai contoh, dalam bahasa Arab, kata-kata seperti "كتب" (menulis), "كاتب" (penulis), dan "مكتوب" (tertulis) semuanya berasal dari akar yang sama "كتب". Dengan memperhitungkan sub-kata, FastText mampu mengenali kemiripan ini meskipun bentuk kata bervariasi secara morfologis [24]. Hal ini sangat penting untuk memahami konteks dalam bahasa yang memiliki struktur tata bahasa yang rumit [25].

Beberapa metode yang umum digunakan untuk mengukur *semantic Similarity* antara dua kata atau lebih meliputi:

1. Cosine *Similarity*

Mengukur kesamaan antara dua vektor dengan melihat sudut antar vektor tersebut. Semakin kecil sudutnya, semakin besar kemiripannya [32].

2. WordSim Evaluation

Menggunakan pasangan kata yang telah diberi skor *Similarity* secara manual, kemudian membandingkan hasilnya dengan skor yang dihasilkan model untuk mengukur akurasi.

3. Word Analogy

Menguji apakah model dapat memahami hubungan semantik dengan melihat pola hubungan antar kata. Ini membantu mengukur seberapa baik model dapat menangkap hubungan semantik kompleks.

Bahasa arab memiliki morfologi yang sangat kaya, dengan struktur akar yang sering digunakan untuk membentuk berbagai kata yang memiliki makna terkait. Dengan menggunakan sub-kata, FastText dapat memahami hubungan ini dengan baik dibandingkan model Word2Vec [33].

Misalnya, untuk pasangan kata “كتب” (menulis) dan “كاتب” (penulis) atau “كتاب” (buku) dan “مكتوب” (tertulis), FastText akan menghasilkan representasi yang lebih dekat karena model ini menguraikan kata menjadi komponen-komponen yang lebih kecil, memungkinkan pemahaman yang lebih dalam terhadap hubungan semantik [27].

2.8 *Python*

Python merupakan bahasa pemrograman tingkat tinggi yang bersifat interpreted, general-purpose, dan mendukung berbagai paradigma pemrograman seperti imperatif, fungsional, dan berorientasi objek. Bahasa ini diciptakan oleh Guido van Rossum dan pertama kali dirilis pada tahun 1991. Python dirancang untuk memiliki sintaks yang sederhana dan mudah dipahami, sehingga sangat cocok digunakan dalam penelitian ilmiah dan pengembangan aplikasi berbasis data.

Salah satu keunggulan python adalah ketersediaan pustaka (library) yang sangat luas, yang mendukung berbagai bidang, termasuk analisis data, machine learning, pemrosesan bahasa alami (*Natural Language Processing*), dan visualisasi. Contoh pustaka yang populer dalam penelitian NLP antara lain NLTK,

spaCy, dan gensim, sedangkan untuk machine learning tersedia pustaka seperti scikit-learn, TensorFlow dan PyTorch.

Python juga memiliki komunitas pengembang yang besar dan aktif, yang secara konsisten berkontribusi terhadap pengembangan pustaka open-source, dokumentasi, serta forum diskusi. Hal ini membuat python terus relevan dan berkembang pesat dalam dunia akademik maupun industri.

Selain itu, python kompatibel dengan berbagai platform sistem operasi seperti Windows, Linux, dan macOS, serta dapat diintegrasikan dengan bahasa lain seperti C/C++ untuk kebutuhan komputasi performa tinggi.

Karena fleksibilitas dan kemudahan penggunaannya, python dipilih dalam penelitian ini sebagai alat bantu utama dalam proses preprocessing teks, pelatihan model FastText, serta evaluasi semantic *Similarity*, karena menyediakan pustaka dan modul yang memadai untuk seluruh tahapan tersebut [34].



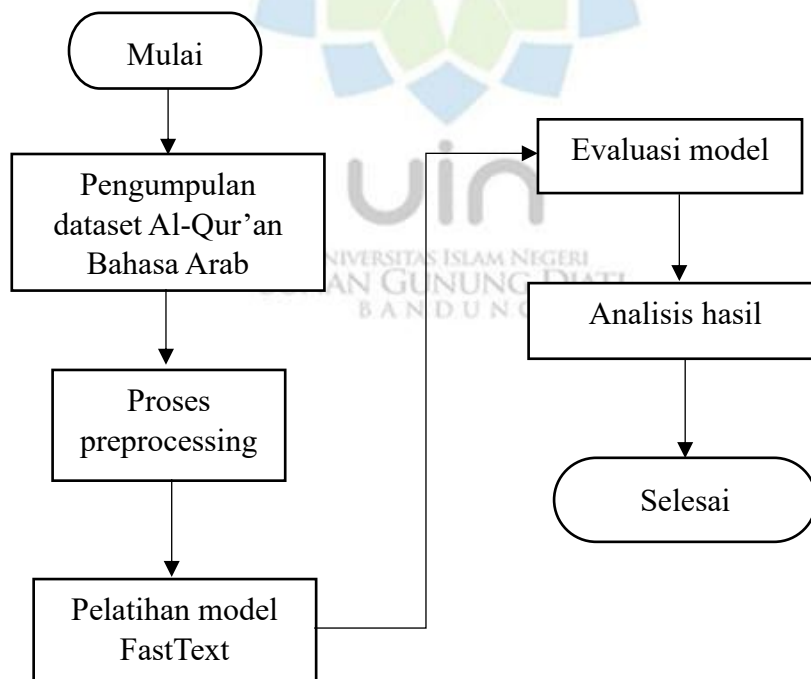
BAB III

METODE PENELITIAN

Bab ini berisi penjelasan langkah-langkah penelitian yang dilakukan mulai dari pengumpulan dataset Al-Qur'an bahasa Arab. Dilanjut dengan tahap *pre-processing* meliputi pembersihan teks, tokenisasi, normalisasi huruf, penghapusan *stopword*, dan penghapusan diakritik agar teks siap diproses oleh model. Setelah dataset siap, dilakukan pelatihan model menggunakan berbagai kombinasi hyperparameter untuk melihat pengaruhnya terhadap representasi kata.

Setelah model dilatih, dilakukan tahap evaluasi menggunakan *cosine Similarity*, dan korelasi terhadap dataset evaluasi untuk mengukur kemampuan model dalam menangkap makna kata. Selanjutnya, hasil *Similarity* dianalisis dan divisualisasikan untuk melihat semantik antar kata.

Berikut merupakan diagram alur dari proses penelitian:



Gambar 3. 1 Diagram Alur Proses Penelitian

3.1 Dataset

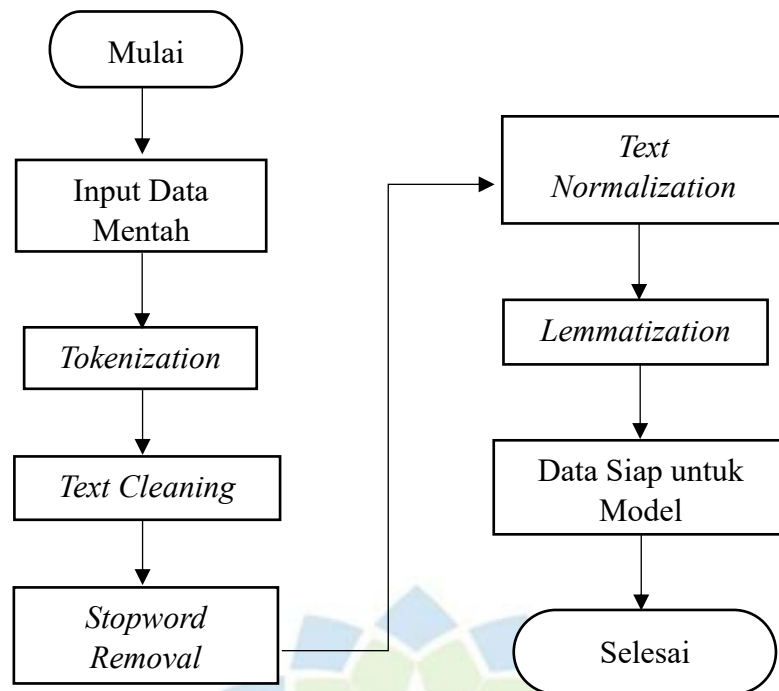
Pada penelitian ini, dataset yang digunakan adalah teks Al-Qur'an dalam bahasa arab yang diambil dari situs resmi Al-Qur'an digital Tanzil.net. Dataset ini dipilih karena memiliki struktur bahasa yang kompleks, kaya akan variasi morfologi dan semantik. Dataset terdiri dari 6.236 ayat yang dibagi berdasarkan surah dan ayat. Bahasa Arab Al-Qur'an memiliki kompleksitas bentuk kata yang tinggi, sehingga cocok untuk menguji kemampuan model embedding seperti FastText yang mempertimbangkan struktur sub-kata.

1 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ
2 اَلْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ
3 اَلرَّحْمَنِ الرَّحِيمِ
4 مُلِكِ يَوْمِ الدِّينِ
5 اِنَّكَ نَعْبُدُكَ وَاِنتَاكَ نَسْتَعِيْنُ
6 اَهْدِنَا الصِّرَاطَ الْمُسْتَقِيْمَ
7 صِرَاطَ الَّذِيْنَ اَنْعَمْتَ عَلَيْهِمْ غَيْرِ الْمَغْضُوْبِ عَلَيْهِمْ وَلَا الضَّالِّيْنَ
1 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ اَلَمْ
2 ذٰلِكَ الْكِتٰبُ لَا رَيْبَ فِيْهِ هُدًى لِّلْمُتَّقِيْنَ
3 الَّذِيْنَ يُؤْمِنُوْنَ بِالْغَيْبِ وَيُقِيْمُوْنَ الصَّلٰوةَ وَمِمَّا رَزَقْنٰهُمْ يُنْفِقُوْنَ
4 وَاَلَّذِيْنَ يُؤْمِنُوْنَ بِمَا اُنْزِلَ اِلَيْكَ وَمَا اُنْزِلَ مِنْ قَبْلِكَ وَيٰۤاٰءِ اٰخِرَةُ هُمْ يُوْقِنُوْنَ
5 اُوْلٰٓئِكَ عَلٰى هٰٓؤُلٰٓءِ مِنْ رَّبِّهِمْ ۖ وَاُوْلٰٓئِكَ هُمُ الْمُفْلِحُوْنَ
6 اِنَّ الَّذِيْنَ كَفَرُوْا سَوَآءٌ عَلَيْهِمْ ءَاَنذَرْتَهُمْ اَمْ لَمْ تُنْذِرْهُمْ لَا يُؤْمِنُوْنَ
7 خَتَمَ اللّٰهُ عَلٰى قُلُوْبِهِمْ وَعَلٰى سَمْعِهِمْ ۖ وَعَلٰى اَبْصَارِهِمْ غَشْوَةٌ وَلَهُمْ عَذَابٌ عَظِيْمٌ
8 وَمِنَ النَّاسِ مَنْ يَقُوْلُ ءَامَنَّا بِاللّٰهِ وَيَاۤٔلَيْوَمَ اَلْاٰخِرَةِ وَمَا هُمْ بِمُؤْمِنِيْنَ
9 يُخٰدِعُوْنَ اللّٰهَ وَالَّذِيْنَ ءَامَنُوْا وَمَا يَخٰدِعُوْنَ اِلَّا اَنْفُسَهُمْ وَمَا يَشْعُرُوْنَ
10 فِى قُلُوْبِهِمْ مَّرِيْضٌ فَرَادٰهُمُ اللّٰهُ مَرِيْضًا ۖ وَلَهُمْ عَذَابٌ اَلِيْمٌۢ بِمَا كَانُوْا يَكْذِبُوْنَ
11 وَاِذَا قِيْلَ لَهُمْ لَا تُفْسِدُوْا فِى الْاَرْضِ قَالُوْا اِنَّمَا نَحْنُ مُصْلِحُوْنَ
12 اَلَا اِنَّهُمْ هُمُ الْمُفْسِدُوْنَ وَلٰكِنْ لَا يَشْعُرُوْنَ

Gambar 3. 2 Dataset Al-Qur'an

3.2 Pre-Processing

Pre-processing adalah tahap awal yang krusial dalam pengolahan data teks, terutama untuk bahasa dengan struktur morfologis kompleks seperti bahasa Arab. Proses ini bertujuan untuk membersihkan teks dan menyusunnya dalam format yang lebih mudah diproses oleh model pembelajaran mesin. *Pre-processing* yang baik dapat secara signifikan meningkatkan akurasi dan efisiensi model dalam memahami dan memproses teks. Berikut adalah langkah-langkah utama yang biasanya dilakukan dalam *pre-processing* teks berbahasa Arab:



Gambar 3. 3 Diagram Alur Tahapan Pre-Processing

3.2.1 *Tokenization* (Tokenisasi)

Tokenisasi dilakukan untuk memecah teks mentah menjadi unit-unit kata atau token yang lebih kecil. Dalam bahasa Arab, proses ini cukup menantang karena adanya prefiks, sufiks, dan partikel yang sering melekat pada kata utama. Tokenisasi yang baik diperlukan untuk memastikan bahwa setiap kata atau bentuk morfologis dikenali dengan benar oleh model FastText.

Sebelum tokenisasi: "الطلاب يدرسون في الجامعة! 1234".

Sesudah tokenisasi: "الطلاب", "يدرسون", "في", "الجامعة!", "1234".

3.2.2 *Text Cleaning* (Pembersihan Teks)

Pembersihan teks melibatkan penghapusan karakter yang tidak relevan, seperti angka, tanda baca, simbol non-Arab, dan diakritik (harakat) seperti fathah, kasrah, dhammah, dan sukun. Penghapusan diakritik dianggap penting dalam penelitian ini untuk memastikan bahwa model tidak terpengaruh oleh variasi pengucapan, sehingga dapat fokus pada makna kata secara lebih konsisten.

Sebelum pembersihan teks: "الطلاب", "يدرسون", "في", "الجامعة!", "1234".

Sesudah pembersihan teks: "الطلاب", "يدرسون", "في", "الجامعة".

3.2.3 Stopword Removal

Setelah teks dibersihkan, langkah berikutnya adalah menghapus *stopword*, yaitu kata-kata yang sering muncul tetapi memiliki makna semantik yang rendah, seperti "على", "من", "في", "و". Menghapus *stopword* pada tahap ini membantu mengurangi ukuran teks yang perlu diproses, sehingga meningkatkan efisiensi *pre-processing* secara keseluruhan. Pada tahap ini, daftar *stopword* disusun secara manual berdasarkan korpus bahasa Arab yang digunakan dalam penelitian ini.

Sebelum *stopword removal*: "الطلاب", "يدرسون", "في", "الجامعة"

Sesudah *stopword removal*: "الطلاب", "يدرسون", "الجامعة"

3.2.4 Text Normalization

Normalisasi dilakukan untuk menyatukan variasi bentuk huruf yang sering muncul dalam teks Arab. Ini penting untuk mengurangi variasi kata yang disebabkan oleh perbedaan penulisan tetapi memiliki makna yang sama. Pada penelitian ini, beberapa aturan normalisasi yang diterapkan meliputi:

- a. "ا" → "أ"
- b. "ا" → "إ"
- c. "ي" → "ى"
- d. "ه" → "ة" (opsional, tergantung konteks)
- e. "و" → "ؤ"

Sebelum normalisasi teks: "الطلاب", "يدرسون", "الجامعة"

Setelah normalisasi teks: "الطلاب", "يدرسون", "الجامعة"

3.2.5 Lemmatization

Lemmatization dilakukan untuk mengembalikan kata ke bentuk dasarnya (lemma), sehingga kata-kata yang memiliki makna sama tetapi berbeda bentuk morfologis dapat dinormalkan. Proses ini penting dalam pemrosesan teks bahasa Arab yang kaya akan variasi morfologi.

Sebelum *lemmatization*: "الطلاب", "يدرسون", "الجامعة"

Setelah *lemmatization*: "طالب", "درس", "جامعة"

3.3 Pelatihan Model FastText

Model yang digunakan adalah FastText, yang merupakan pengembangan dari Word2Vec. Pelatihan dilakukan dengan mempertimbangkan sub-kata (n-gram karakter), menjadikannya lebih efektif dalam menangani variasi morfologi kata arab. Pada tahap pelatihan ini, konfigurasi hyperparameter memainkan peran penting dalam menentukan kualitas representasi kata yang dihasilkan. Pemilihan hyperparameter yang tepat akan meningkatkan akurasi model dalam memahami konteks kata dan hubungan semantik antar kata.

Beberapa hyperparameter utama yang disesuaikan dalam model FastText pada penelitian ini meliputi:

a. Dimensi Vektor (*vektor size*)

Pada penelitian ini, dimensi 200-300 yang digunakan untuk menangkap lebih banyak fitur semantik.

b. *Window size*

Pada penelitian ini, digunakan nilai 5-7 untuk mempertimbangkan kata di kiri dan kanan target.

c. *Learning Rate*

Menentukan kecepatan pembaruan bobot model selama pelatihan, diatur pada 0.05 untuk menjaga stabilitas pembaruan.

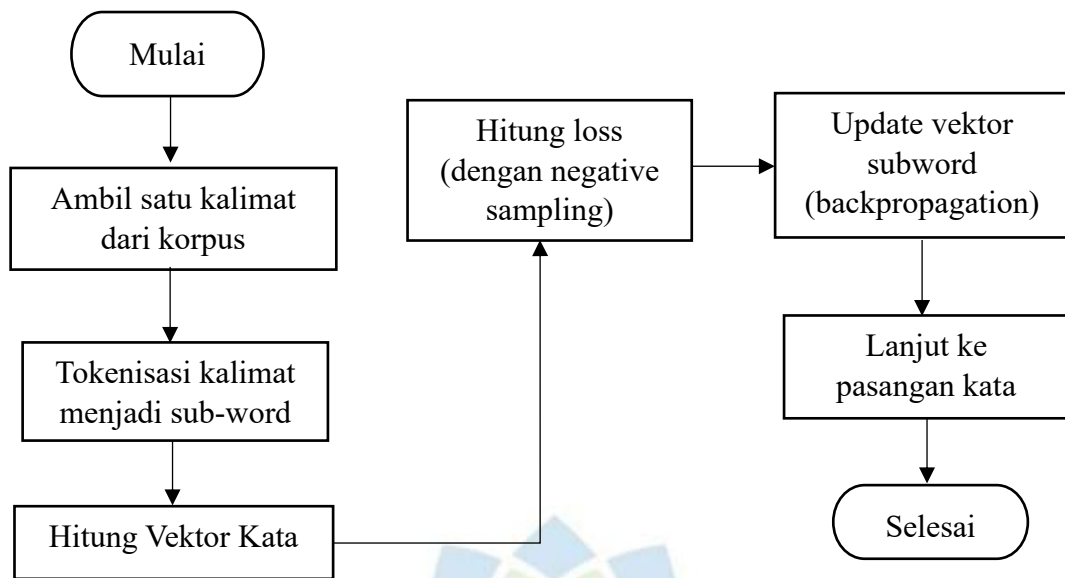
d. *Minimum Count*

Frekuensi minimum kata untuk di masukan ke dalam kamus model, diatur dalam 1 untuk menjaga stabilitas pembaruan.

e. Jumlah Iterasi (Epoch)

Pada penelitian ini, model dilatih selama 10-15 epoch untuk mencapai konvergensi yang baik.

Proses pembentukan vektor ini mencakup beberapa tahap dimulai dari pembacaan korpus, tokenisasi, pembentukan n-gram, hingga akhirnya pelatihan model.



Gambar 3. 4 Algoritma FastText

Diagram alur ini merepresentasikan proses pelatihan *embedding* kata berbasis *subword* n-gram, yang menjadi inti dari model FastText. Proses dimulai dengan mengambil satu kalimat dari korpus dan melakukan tokenisasi untuk memisahkan kata-katanya. Setiap kata dalam kalimat diperlakukan sebagai target word, sementara kata-kata di sekelilingnya dipilih sebagai *context word* berdasarkan ukuran *context windows*. Target word kemudian dipecah menjadi n-gram karakter, dan vektor representasi target dihitung sebagai penjumlahan dari vektor n-gram penyusunnya. Untuk setiap *context word*, model menghitung probabilitas kemunculannya berdasarkan target word menggunakan pendekatan softmax yang diaproksimasi dengan negative sampling.

Fungsi *loss* dihitung untuk setiap pasangan (target, konteks), dan bobot vektor n-gram diperbarui melalui backpropagation. Proses ini diulang untuk setiap kata dalam kalimat, kemudian dilanjutkan ke kalimat berikutnya hingga seluruh korpus selesai. Dengan menggunakan representasi *subword*, model dapat menghasilkan *embedding* yang robust terhadap kata langka atau belum pernah terlihat (*out-of-vocabulary*), sekaligus menangkap informasi morfologis yang tidak dimiliki oleh model berbasis kata tunggal seperti Word2Vec.

3.4 Arsitektur Model Skip-gram

Pada penelitian ini, arsitektur skip-gram dipilih untuk melatih model, karena cenderung lebih baik dalam menangkap hubungan semantik untuk kata jarang muncul, yang umum dalam teks bahasa Arab. Skip-gram bekerja dengan memprediksi konteks dari kata target. Keuntungan dari skip-gram, yaitu memiliki kemampuan untuk belajar dari kata yang jarang muncul dan memiliki representasi kata yang lebih akurat dalam korpus besar.

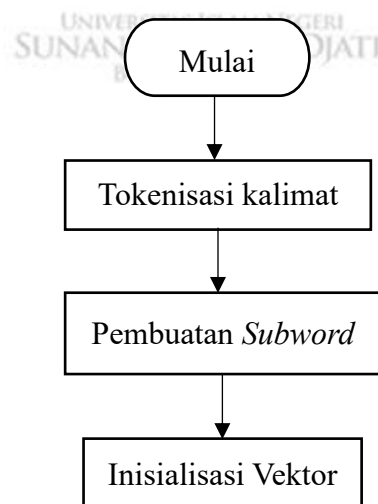
Persamaan dasar untuk pembaruan bobot dalam pendekatan skip-gram adalah:

$$v_w = v_w + \eta(v_c - v_w) \quad (3.1)$$

di mana:

v_w : Vektor kata target,
 v_c : Vektor konteks, dan
 η : *Learning rate*.

Dalam penelitian ini, secara garis besar skip-gram terbagi menjadi tiga fase utama: tahap pra-pelatihan skip-gram, tahap proses inti skip-gram dan pembaruan vektor final. Masing-masing fase ini memiliki peran penting dalam membangun representasi kata yang kaya akan konteks dan memiliki makna semantik yang mendalam.



Gambar 3. 5 Fase Pra-pelatihan Skip-gram

Fase awal pra-pelatihan skip-gram dimulai dengan tokenisasi kalimat, yang berfungsi untuk memecah teks menjadi unit-unit kata. Selanjutnya, setiap kata yang telah ditokenisasi dipecah lagi menjadi n-gram karakter (subword). Setiap n-gram ini kemudian diinisialisasi dengan vektor awal acak oleh sistem, melalui fungsi internal seperti *initialize_weight* atau *reset_weight* yang merupakan bagian dari implementasi FastText. Apabila suatu kata w terdiri dari sejumlah *subword* n-gram g_1, g_2, \dots, g_n , maka representasi vektor kata \vec{w} dihitung sebagai rata-rata dari seluruh vektor n-gram yang membentuknya:

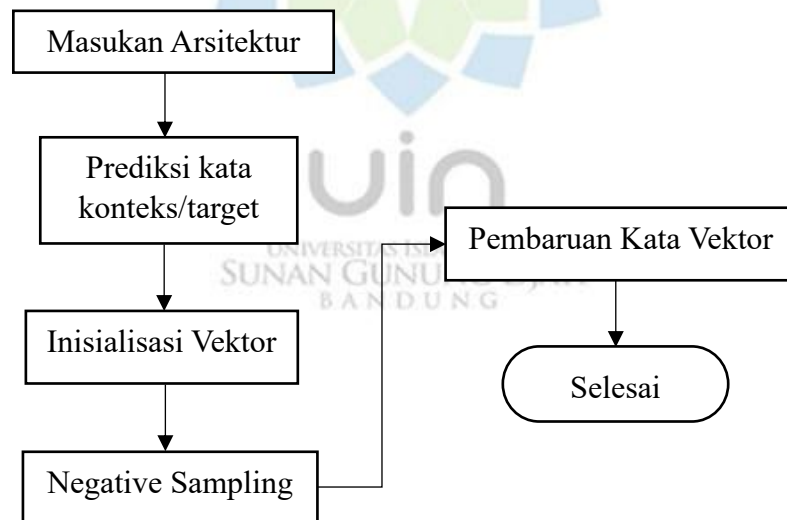
$$\vec{w} = \left(\frac{1}{|G_w|} \right) \sum_{g \in G_w} \vec{z}_g \quad (3.2)$$

di mana:

G_w : Himpunan semua n-gram (*subword*) dari kata w

\vec{z}_g : Vektor *embedding* dari n-gram g

$|G_w|$: Jumlah n-gram dalam kata w



Gambar 3. 6 Proses inti Skip-gram

Setelah melakukan tahap pra-pelatihan, model masuk ke tahap inti Skip-gram. Pada tahapan ini, model berfokus pada memprediksi kata-kata konteks di sekitar satu kata target yang dipilih dalam setiap iterasi pelatihannya. Uniknya, vektor untuk kata target dibentuk dengan menjumlahkan vektor dari semua n-gram yang membentuknya.

Untuk membuat proses ini lebih efisien dan akurat, digunakan teknik negatif sampling. Dengan metode ini, model membandingkan kata target dengan beberapa kata acak yang tidak relevan. Ini membantu model untuk membedakan konteks yang benar dari yang salah, sehingga pembelajaran menjadi lebih efektif. Fungsi *loss* pada negative sampling digunakan untuk mengukur tingkat kesalahan prediksi dengan cara memaksimalkan probabilitas pasangan kata yang benar dan meminimalkan probabilitas pasangan kata yang salah. Proses ini dihitung menggunakan rumus (3.3).

$$L = -\log(\sigma(v_c^T \cdot v_t)) - \sum_{i=1}^k \log(-\sigma(v_{ni}^T \cdot v_t)) \quad (3.3)$$

di mana:

- v_t : Vektor dari kata target
- v_c : Vektor dari kata konteks positif
- v_{ni} : Vektor dari kata negatif ke- i
- k : Jumlah kata negatif yang diambil
- $\sigma(x)$: Fungsi sigmoid, yaitu $\sigma(x) = \frac{1}{1+e^{-x}}$

Nilai *loss* tersebut menjadi dasar bagi model untuk memulai proses pembelajaran dalam memperbaiki prediksinya. Gradien dari fungsi *loss* dihitung melalui backpropagation guna menentukan arah dan besarnya penyesuaian yang dibutuhkan oleh setiap vektor, sebagaimana tercantum dalam rumus (3.4).

$$\frac{\partial L}{\partial \vec{v}_w} = (\sigma(\vec{v}_c^T \vec{v}_w) - 1) \vec{v}_c + \sum_{k=1}^k \sigma(\vec{v}_{nk}^T \vec{v}_w) \vec{v}_{nk} \quad (3.4)$$

di mana:

- \vec{v}_w : Vektor input dari kata target w
- \vec{v}_c : Vektor output dari konteks positif c
- \vec{v}_{nk} : Vektor output dari kata negative ke- k
- $\vec{v}_c^T \vec{v}_w$: Dot product antara vektor target dan konteks positif
- $\sigma(\vec{v}_{nk}^T \vec{v}_w)$: Nilai sigmoid dari dot product

Setelah gradien dihitung, pembaruan bobot vektor dilakukan secara efisien dengan menerapkan aturan *gradient descent*. Besarnya pembaruan ini diatur oleh

learning rate (η) untuk memastikan proses pembelajaran berlangsung secara stabil, sesuai dengan rumus (3.5).

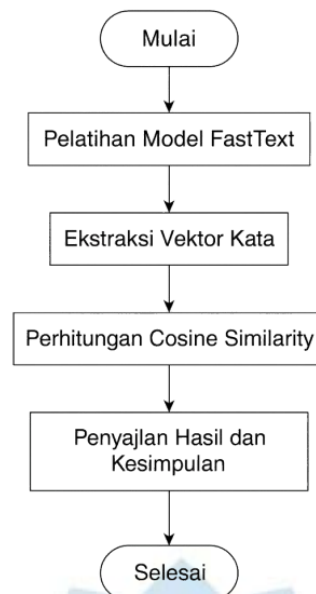
$$\vec{v}_w = \vec{v}_w - \frac{\eta(\partial L)}{\partial \vec{v}_w} \quad (3.5)$$

Proses pembelajaran yang meliputi prediksi, perhitungan loss, dan pembaruan bobot ini dilakukan secara berulang terhadap seluruh data dalam korpus. Setelah seluruh iterasi selesai, model akan menghasilkan representasi vektor akhir untuk setiap kata, di mana vektor-vektor tersebut telah memuat informasi semantik yang kaya dan dapat dimanfaatkan dalam berbagai tugas lanjutan di bidang NLP [22].

3.5 Evaluasi *Semantic Similarity*

Evaluasi *semantic similarity* merupakan tahapan untuk menilai seberapa baik model FastText yang telah dilatih mampu menangkap hubungan makna antar kata dalam bahasa Arab. Karena tujuan utama dari penelitian ini adalah mengukur pengaruh variasi hyperparameter terhadap kualitas representasi semantik, maka evaluasi dilakukan dengan pendekatan kuantitatif menggunakan *cosine similarity*.

Tujuan dari evaluasi ini untuk mengukur kemampuan model dalam mengenali kata-kata yang memiliki makna serupa atau berhubungan secara semantik, mengetahui pengaruh perubahan nilai hyperparameter terhadap performa *semantic similarity*, dan membandingkan model-model yang dilatih dengan konfigurasi hyperparameter berbeda secara objektif.



Gambar 3. 7 Diagram Alur Perhitungan *Cosine Similarity*

Pada tahap ini menggunakan metode pengukuran *cosine similarity*, di mana *cosine similarity* digunakan untuk mengukur seberapa mirip vektor kata yang dihasilkan oleh model FastText. Semakin tinggi nilai *cosine similarity* antara dua vektor (mendekati 1), semakin besar dugaan bahwa kedua kata tersebut berkaitan secara semantik. Metode ini dipilih karena fokus pada orientasi vektor, bukan panjangnya, sehingga lebih sesuai dengan karakteristik *word embeddings*.

Secara matematis, *cosine Similarity* antara dua vektor A dan B dirumuskan sebagai berikut:

$$\text{cosine similarity}(A, B) = \frac{A \cdot B}{\|A\| \times \|B\|} \quad (3.7)$$

di mana:

- $A \cdot B$: dot product dari vektor A dan B
- $\|A\|$: Norma (magnitudo) dari vektor A
- $\|B\|$: Norma (magnitudo) dari vektor B

Rumus ini menghasilkan nilai antara -1 dan 1, nilai 1 menunjukkan bahwa vektor A dan B memiliki arah yang sama (identik secara semantik). Nilai 0 menunjukkan bahwa vektor A dan B saling ortogonal atau tidak memiliki kesamaan semantik. Nilai -1 menunjukkan bahwa vektor A dan B memiliki arah yang berlawanan (berlawanan secara semantik) [32].

BAB IV

EKSPERIMEN DAN ANALISIS

Bab ini berfungsi sebagai jembatan antara landasan teori yang telah dibahas sebelumnya dan hasil eksperimen yang diperoleh dari penelitian. Bagian ini akan menguraikan secara komprehensif implementasi metodologi yang telah dirancang, dimulai dari persiapan data hingga evaluasi performa yang dihasilkan. Tujuannya adalah untuk menyajikan temuan-temuan kunci dan menganalisis dampaknya terhadap rumusan masalah penelitian.

Secara garis besar, bab ini akan menggambarkan alur kerja yang sistematis, memastikan setiap tahapan, mulai dari pengumpulan data hingga pelaporan hasil, dilakukan dengan integritas ilmiah. Bagian ini juga akan menjadi fondasi untuk pembahasan lebih lanjut di bab selanjutnya, yang menginterpretasikan implikasi dari hasil-hasil dan memberikan kesimpulan berdasarkan bukti empiris.

4.1 Validasi Metode

Sebelum diterapkan pada korpus utama, metode terlebih dahulu divalidasi untuk memastikan kinerjanya. Validasi metode dalam penelitian ini dibagi menjadi dua pendekatan utama, yaitu **analisis semantik kontekstual** dan **analisis semantik skalar**. Pendekatan kontekstual berfokus pada ketepatan pemahaman model terhadap makna berdasarkan variasi bentuk dan struktur linguistik. Sementara itu, pendekatan skalar menilai kestabilan representasi semantik yang terbentuk akibat peningkatan jumlah data. Pembagian ini bertujuan untuk membuktikan bahwa model FastText tidak hanya kuat dalam menangkap makna dari berbagai ekspresi, tetapi juga mampu mempertahankan pemahaman tersebut secara konsisten dalam skala data yang lebih besar.

4.1.1 Analisis Semantik Kontekstual

Analisis Validasi semantik kontekstual menguji kemampuan model dalam memahami makna berdasarkan variasi struktur dan lingkungan kalimat. Fokus utama dari pendekatan ini adalah sejauh mana model mampu mempertahankan relasi semantik antar kata ketika konteks atau bentuknya berubah.

1. Variasi Jumlah Kata *الامي* dan *امي*

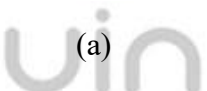
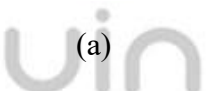
Pengujian ini mengamati pengaruh proporsi penggunaan kata ma'rifat (misalnya “*الامي*”) terhadap bentuk umum (“*امي*”) dalam korpus. Tujuannya adalah untuk melihat apakah perubahan jumlah bentuk ma'rifat memengaruhi hasil representasi vektor dan kedekatannya terhadap kata-kata bermakna serupa. Hasil kuantitatif dari pengujian ini disajikan secara rinci pada tabel 4.1.

Tabel 4. 1 Representasi Jumlah Kata *الامي* dan *امي*

Persentase Jumlah Kata <i>امي</i> dan <i>الامي</i>	Vektor kata <i>الامي</i>	<i>Cosine</i> <i>Similarity</i> Tertinggi	Contoh Hasil Kata <i>Similarity</i>
20%	[0.035940736532211304,	0.6921	الراحة
	0.1714797019958496,	0.6892	الكبيرة
	...	0.6830	العافية
	0.14011749625205994,	0.6775	التي
	-0.03808264806866646]	0.6691	السعادة
40%	[0.0029740266036242247,	0.6947	الدعوات
	0.12053225189447403,	0.6917	الواسع
	...	0.6863	التي
	0.05709918960928917,	0.6804	الكبيرة
	-0.051158104091882706]	0.6718	الأجمل
60%	[-0.014115790836513042,	0.6968	الأجمل
	0.16849441826343536,	0.6941	الحب
	...	0.6882	الكبيرة
	0.09185456484556198,	0.6813	الأمل
	0.00028409436345100403]	0.6739	السعادة
80%	[0.010114923119544983,	0.6995	الأجمل
	0.10318709164857864,	0.6967	الكبيرة
	...	0.6901	الواسع
	0.07814902067184448,	0.6835	الحب
	0.002595252124592662]	0.6758	التي

Rata-rata nilai *cosine similarity* pada skenario ma'rifat ini berada di kisaran 0.70, memperlihatkan bahwa meskipun kata-kata memiliki bentuk ma'rifat,

(a)

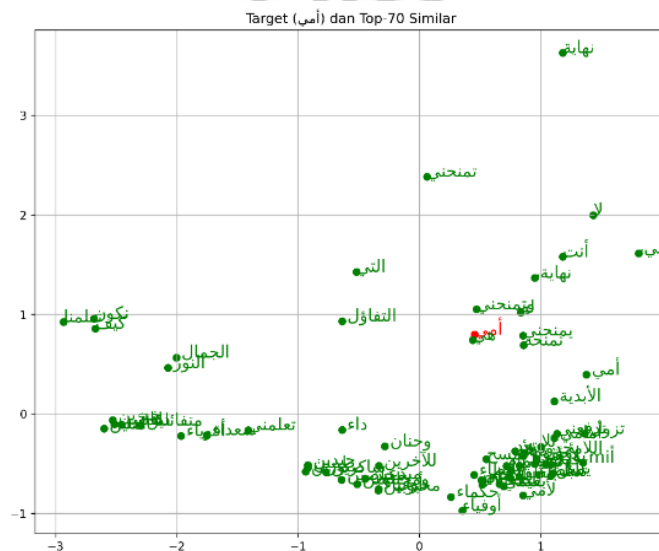


(a)

Tabel 4. 2 Representasi Variasi Panjang Kalimat

Vektor kata امي	Cosine <i>Similarity</i> tertinggi	Contoh kata Simlarity
[0.10070403665304184,	0.7771	بقيمي
-0.13881242275238037, 0.31460583209991455,	0.7623	قدمي
.....	0.7484	تماما
-0.07071464508771896, 0.016672413796186447,	0.6428	صبروا
-0.23070518672466278, 0.06786450743675232]	0.6333	صديقتي

Nilai *cosine similarity* meningkat seiring bertambahnya panjang kalimat (hingga 0.77), menunjukkan bahwa struktur kalimat yang lebih kaya memberikan konteks semantik yang lebih kuat. Visualisasi PCA mendukung hal ini: semakin panjang kalimat, semakin rapat dan fokus kluster vektor kata terbentuk. Hal ini terlihat dalam ilustrasi pada gambar 4.2, yang membuktikan bahwa model mampu menangkap nuansa makna yang lebih halus ketika diberikan informasi kontekstual yang lebih panjang.



Gambar 4. 2 Ilustrasi *Similarity* Variasi Panjang Kalimat

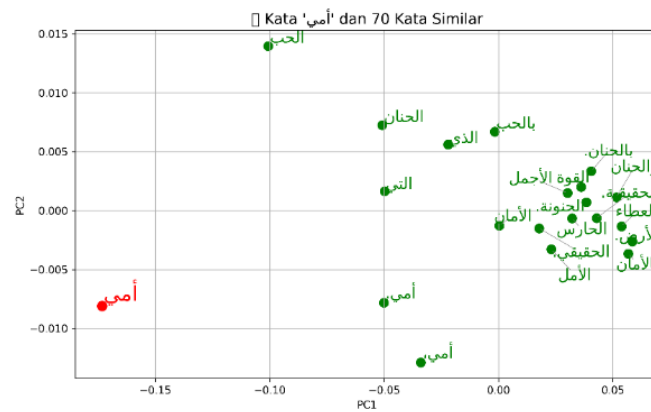
3. Konteks Tematik

Pengujian ini mengevaluasi seberapa besar pengaruh konteks tematik terhadap kemiripan makna. Kalimat-kalimat dikelompokkan berdasarkan tema seperti keluarga (25 kalimat), pekerjaan (30 kalimat), dan kasih sayang (40 kalimat). Hasil kuantitatifnya dirangkum dalam tabel 4.3.

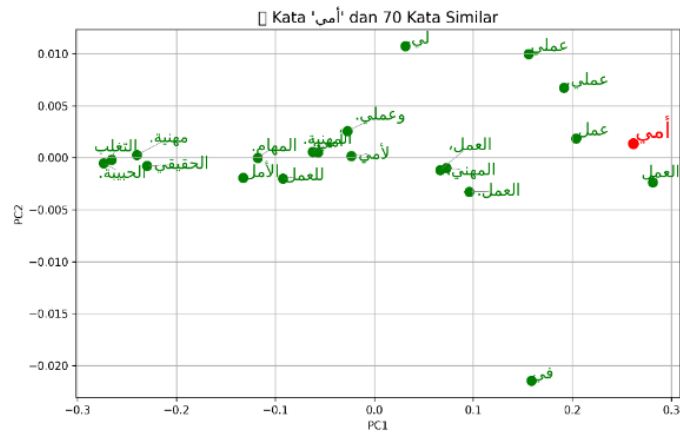
Tabel 4. 3 Representasi Konteks Tematik

Konteks	Cosine Similarity tertinggi	Kata Similarity yang muncul
Kasih sayang	0.2266	بالبهجة
	0.2046	يغطي
	0.1891	تغيب
	0.1839	تملاها
	0.1791	أتصور
Pekerjaan	0.3389	بمهنتي
	0.3314	فخورة
	0.3086	وتصير
	0.3056	أقدم
	0.2920	كبيرة
Keluarga	0.1899	بالبهجة
	0.1886	بارًا
	0.1683	لأجل
	0.1295	بأمي
	0.1253	تملاً

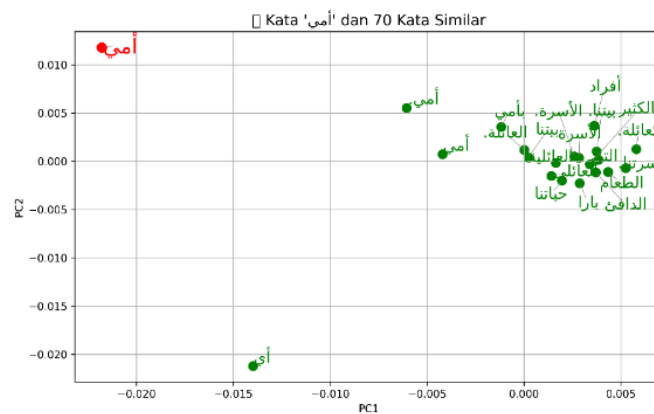
Dengan nilai *similarity* rata-rata mendekati 0.20, tema-tema yang secara semantik erat seperti kasih sayang, pekerjaan, dan keluarga memperkuat asosiasi makna antar kata meskipun jumlah kalimat berbeda. Meskipun nilai *similarity* relatif kecil, visualisasi PCA terlihat pada ilustrasi gambar 4.3 memperlihatkan kluster-kluster tematik yang jelas dan terpisah, mencerminkan bahwa konteks emosional atau fungsional dalam kalimat mampu membentuk representasi makna yang koheren.



(a)



(b)



(c)

Gambar 4. 3 Ilustrasi *Similarity* Konteks Tematik

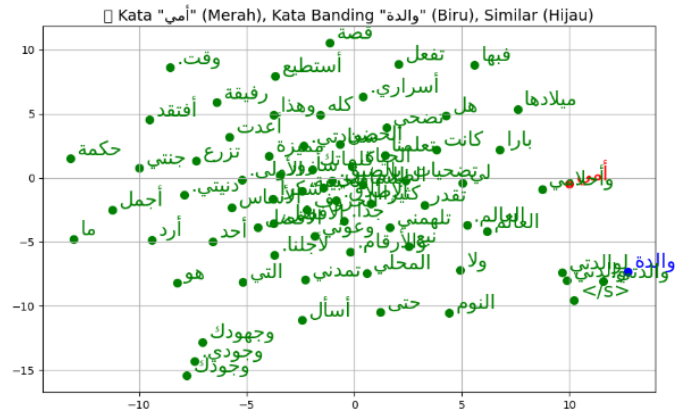
4. Variasi Jumlah Kata Spesifik *امي* والدّة

Pengujian ini menilai apakah model mengenali “والدّة” sebagai bagian dari kelompok makna yang sama dengan “امي”. Kalimat-kalimat digunakan secara konsisten dan hanya kata ini yang divariasikan. Hasil kuantitatifnya disajikan dalam tabel 4.4.

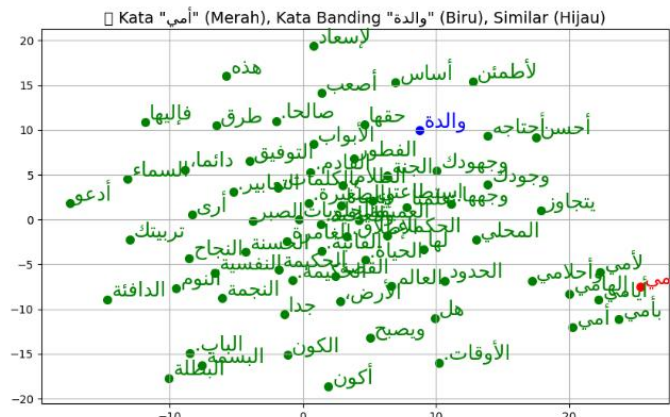
Tabel 4. 4 Representasi Jumlah Kata Spesifik *امي* والدّة

Persentase Jumlah Kata <i>امي</i> والدّة	Vektor kata <i>امي</i>	<i>Cosine</i> <i>Similarity</i> Tertinggi	Contoh Hasil Kata <i>Similarity</i>
20%	[-0.04892118275165558,	0.9792	إلهامي
	0.055234383791685104,	0.9792	الكبيرة
	...	0.9791	والدتي
	0.05323643609881401,	- 0.9791	القوة
	0.05322646722197533]	0.9791	الراحة
40%	[-0.04792971536517143,	0.9696	إلهامي
	0.07747229933738708,	0.9696	والدتي
	...	0.9696	لوالدتي
	0.07862966507673264,	- 0.9696	والراحة
	0.0041422611102461815]	0.9696	الكبيرة
50%	[-0.014620269648730755,	0.9607	واسع
	0.004293021280318499, -	0.9604	وأحلامي
	...	0.9698	هدية
	0.007938135415315628,	- 0.9597	بهديّة
	0.005756468046456575]	0.9595	الكبيرة
60%	[-0.049390535801649094,	0.9594	إلهامي
	0.05255041643977165,	0.9593	اليوم
	...	0.9593	والدتي
	0.04029661789536476,	- 0.9593	الكبيرة
	0.06471917033195496]	0.9593	الغالية
80%	[-0.08443048596382141,	0.9794	إلهامي
	0.09191451221704483,	0.9794	والدتي
	...	0.9794	الكبيرة
	0.05152143910527229,	- 0.9794	الأجمل
	0.06591468304395676]	0.9793	أيامي

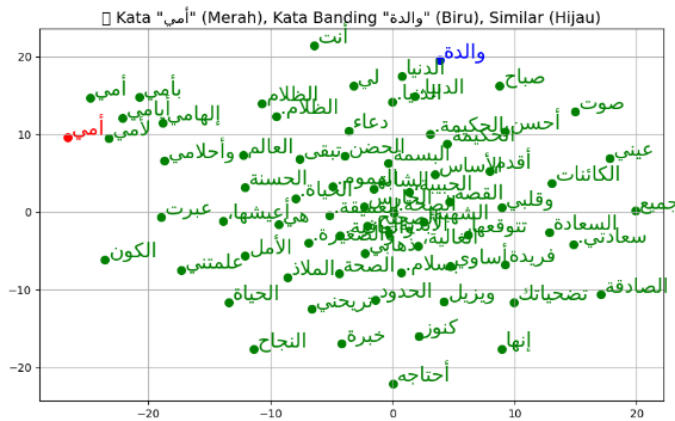
Persentase Jumlah Kata امي dan والدة	Vektor kata امي	<i>Cosine</i> <i>Similarity</i> Tertinggi	Contoh Hasil Kata <i>Similarity</i>
90%	[-0.0789184644818306,	0.9796	إلهامي
	0.09206859022378922,	0.9796	الكبيرة
	...	0.9795	والدتي
	0.026581525802612305,	0.9795	الغالية
	0.09024050086736679]	0.9795	اليوم



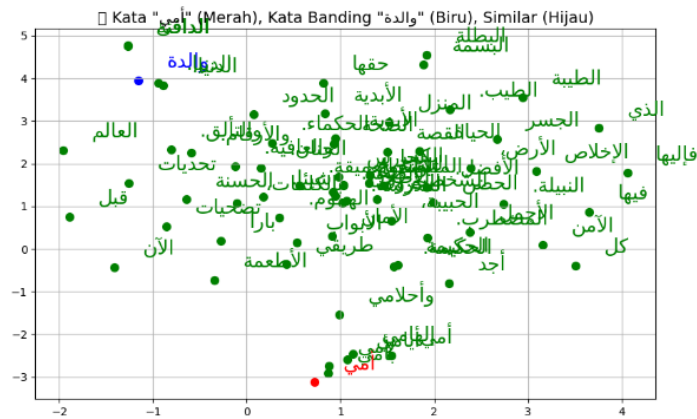
(c)



(d)



(e)



Gambar 4. 4 Ilustrasi *Similarity* Kata Spesifik “والدة”

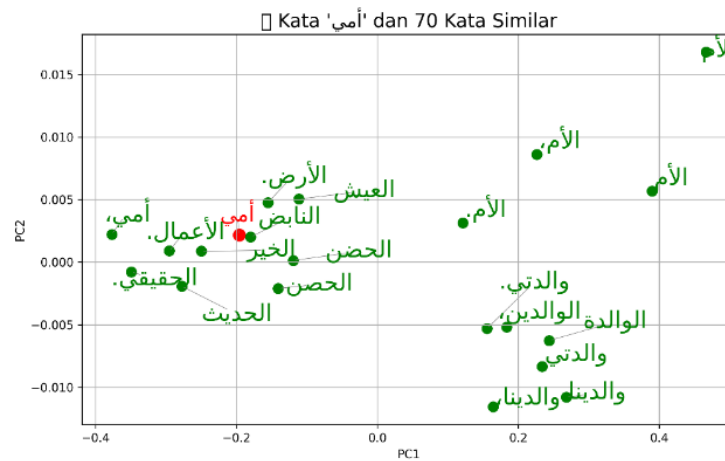
5. Variasi Sinonim

Pengujian ini menilai kemampuan model dalam mengenali beberapa sinonim sekaligus dan mengelompokkannya secara semantik yang sangat dekat dengan kata “امي”, meskipun bentuk penulisannya bervariasi seperti “والدة”, “ام”, dan “ماما”. Hasilnya terdapat pada tabel 4.5.

Tabel 4. 5 Representasi Variasi Sinonim

Vektor kata	Cosine Similarity Tertinggi	Contoh kata
[-0.0015883106971159577,	0.9987	والدة
-0.004590261727571487, -0.004187670070677996,	0.9987	وجودك
0.0018241958459839225,	0.9987	طاعة
...	0.9976	ماما
-0.001725828624330461, -0.003122132271528244, -	0.9976	ملاك
0.0036339128855615854]		

Variasi sinonim menghasilkan *similarity* tertinggi (0.99), menunjukkan kemampuan terbaik model dalam mengelompokkan kata bermakna identik. Visualisasi PCA pada gambar 4.5 mendukung ini dengan kluster vektor yang padat dan hampir tumpang tindih, membuktikan bahwa model FastText sangat andal dalam menangkap sinonimi eksplisit.



Gambar 4. 5 Ilustrasi *Similarity* Variasi Sinonim

4.1.2 Analisis Semantik Skalar

Analisis semantik skalar bertujuan untuk menguji pengaruh volume atau skala data terhadap kedalaman dan stabilitas representasi makna dalam model FastText. Dengan mengamati perubahan performa model terhadap variasi jumlah data (baik jumlah kalimat maupun variasi bentuk sinonim dalam skala besar), analisis ini memberikan gambaran seberapa tangguh dan konsisten pemahaman semantik model dalam skala yang terus bertambah.

1. Variasi Jumlah Kalimat

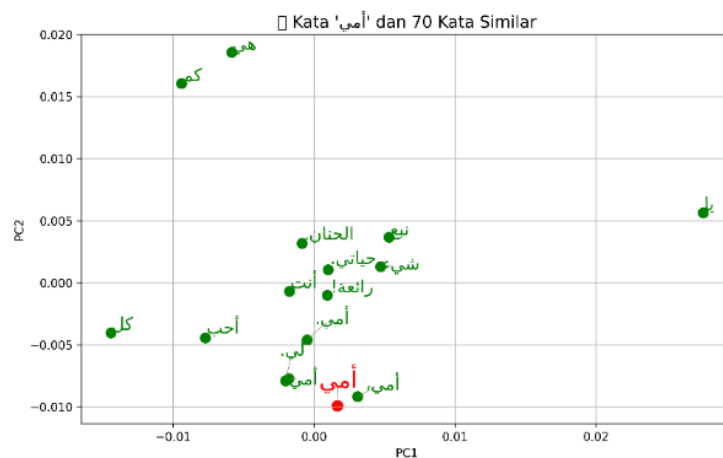
Pengujian ini bertujuan untuk mengevaluasi pengaruh kuantitas kalimat terhadap kemampuan model dalam mempelajari asosiasi makna antar kata. Semakin banyak kalimat yang tersedia dalam pelatihan, maka semakin banyak pula konteks yang dapat diobservasi oleh model. Hasil kuantitatif dari pengujian ini disajikan secara rinci pada tabel 4.6.

Tabel 4. 6 Representasi Variasi Jumlah Kalimat

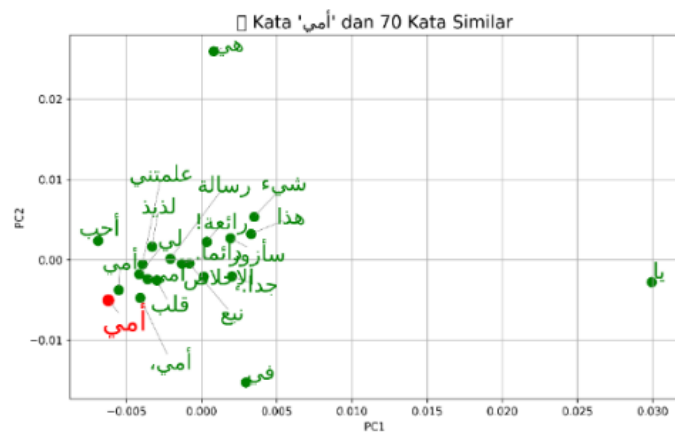
Jumlah kalimat	Top-k <i>Similarity</i> tertinggi	Kata mirip yang muncul
5	0.1025	أمي
10	0.2899	واسع
20	0.2491	واسع
50	0.9343	اليوم
100	0.9268	إلهامي

Jumlah kalimat	Top-k <i>Similarity</i> tertinggi	Kata mirip yang muncul
500	0.9998	وأحلامي
1000	0.9998	أيامي
6000	0.9998	أيامي

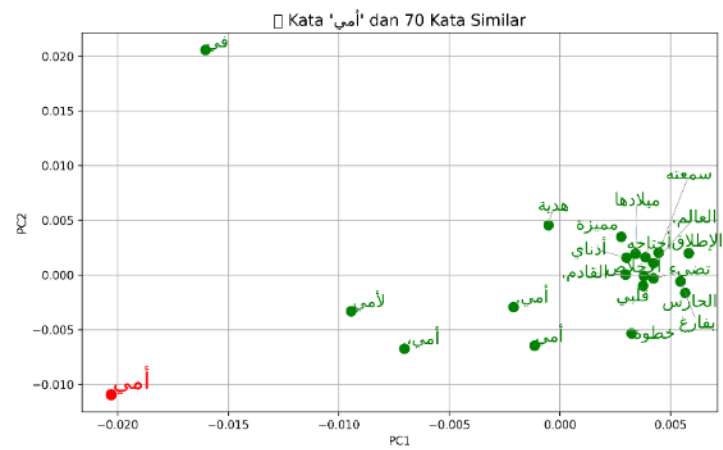
Nilai *cosine similarity* meningkat secara bertahap, dari nilai awal di bawah 0.10 hingga 0.99 ketika jumlah kalimat ditingkatkan. Visualisasi PCA pada gambar 4.6 menunjukkan bahwa dengan jumlah kalimat sedikit, distribusi vektor kata masih tersebar. Namun, ketika jumlah kalimat diperbesar, kluster semantik mulai terbentuk lebih padat dan terfokus, menandakan kestabilan makna yang meningkat. Volume kalimat yang lebih besar memperluas konteks distribusional yang dipelajari model. Dengan demikian, model FastText menunjukkan respon positif terhadap eksposur konteks yang lebih luas, yang memperkuat kekuatan asosiasi antar kata dalam ruang vektor.



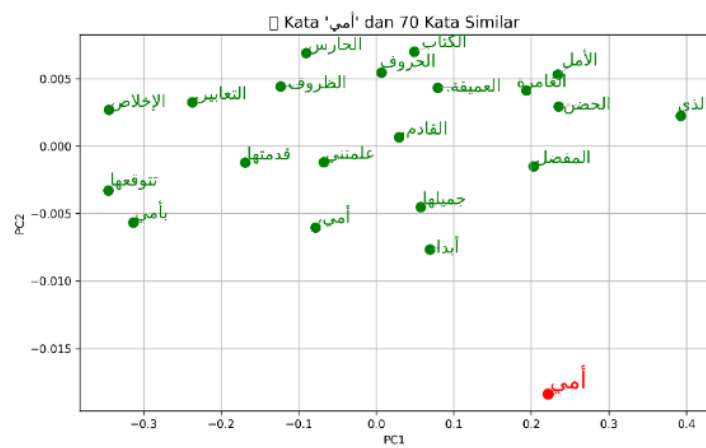
(a)



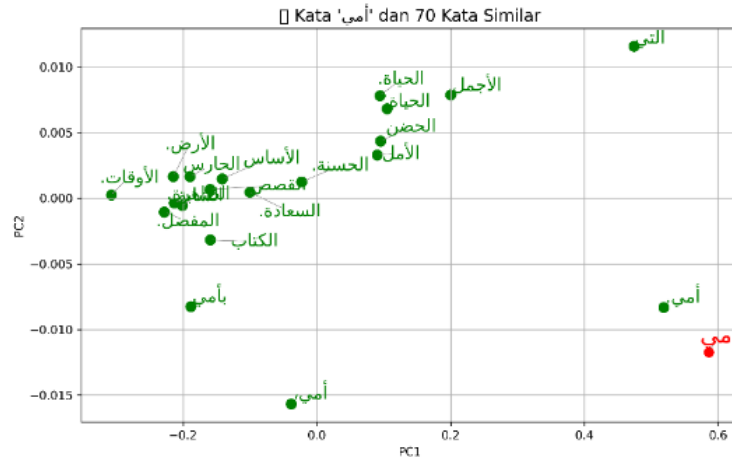
(b)



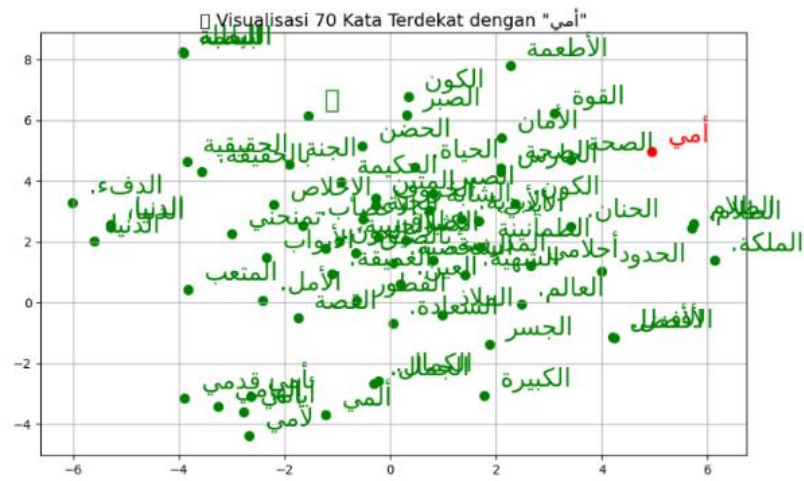
(c)



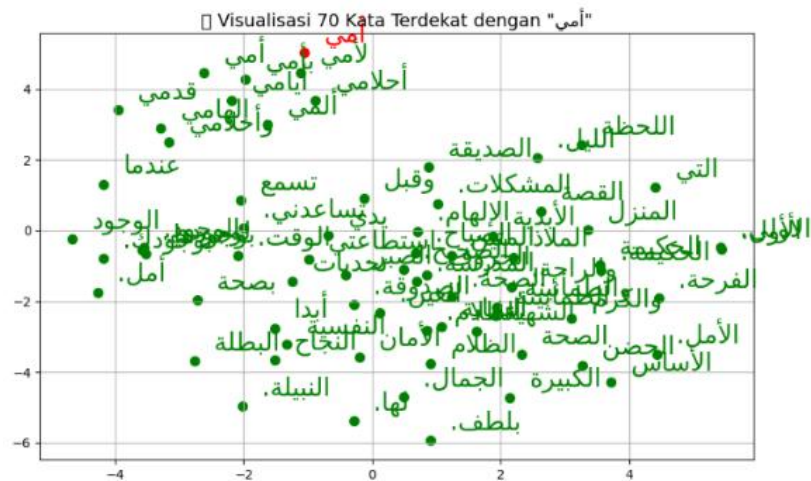
(d)



(e)



(f)



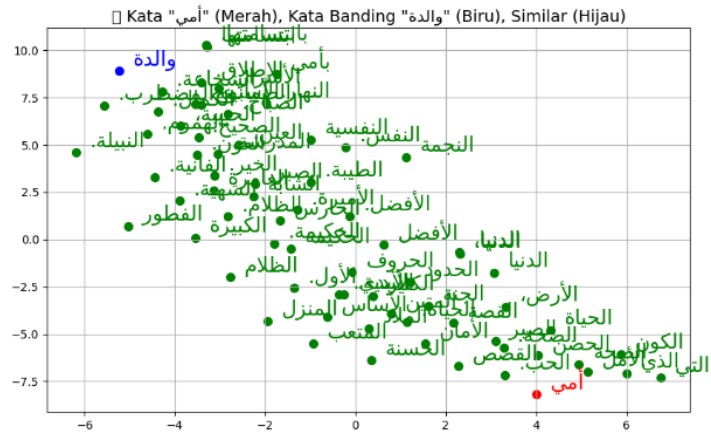
(g)



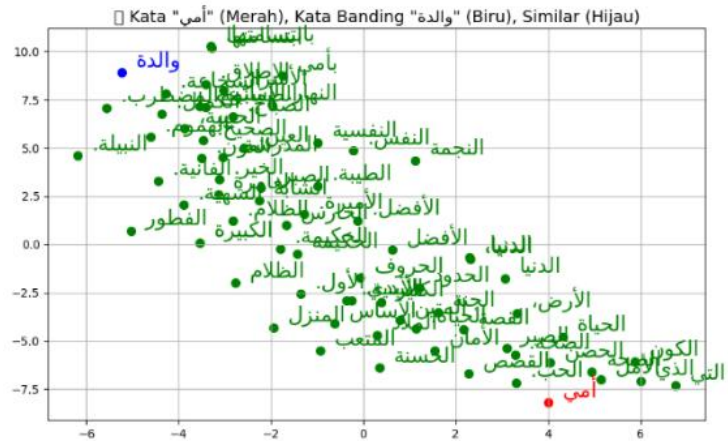
2. Variasi Jumlah Kalimat Sinonim

Tabel 4. 7 Representasi Variasi Jumlah Kalimat Sinonim

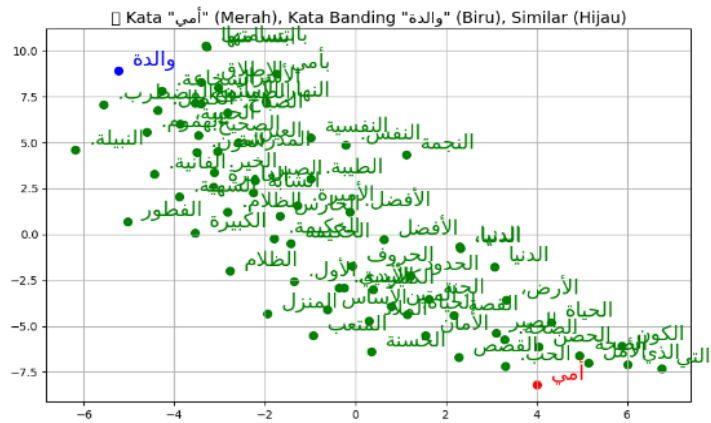
52



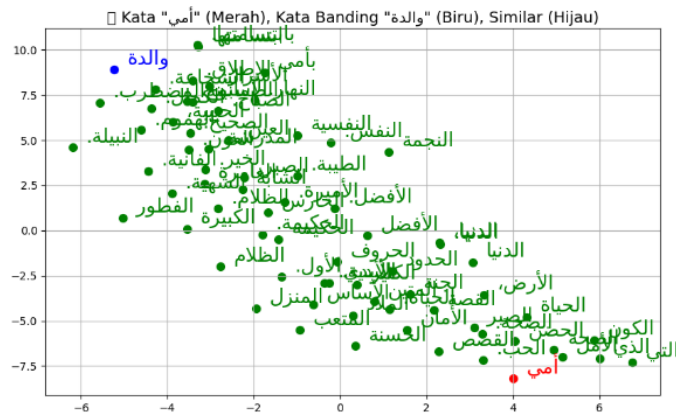
(b)



(c)



(d)



(e)

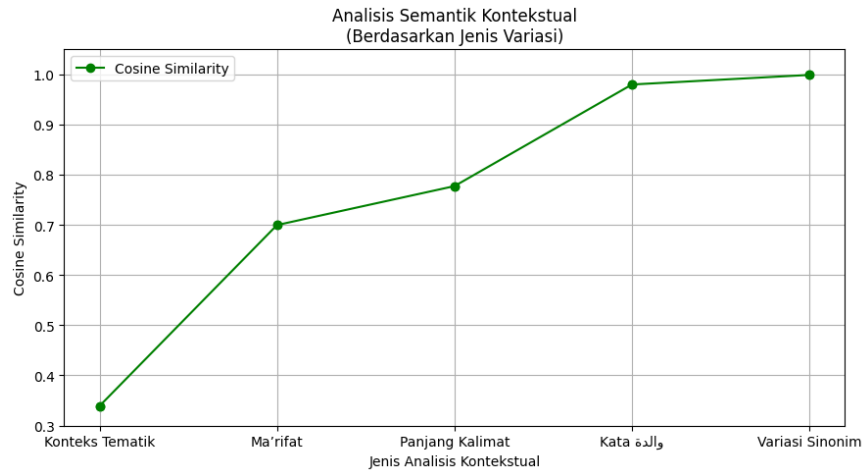
Gambar 4. 7 Ilustrasi *Similarity* Variasi Jumlah Kalimat Sinonim

4.1.3 Visual Progres Kinerja Model

Untuk memvisualisasikan seluruh temuan dari tahap analisis semantik, hasil dari berbagai skenario pengujian kunci direpresentasikan dalam dua grafik progres kinerja model. Grafik pertama merepresentasikan hasil dari sisi **kontekstual**, yang mencerminkan kemampuan model dalam memahami variasi bentuk dan makna kata dalam berbagai konteks linguistik. Grafik kedua menampilkan aspek **skalar**, yaitu seberapa besar pengaruh jumlah data terhadap stabilitas dan kedalaman pemahaman semantik yang dibentuk oleh model FastText.

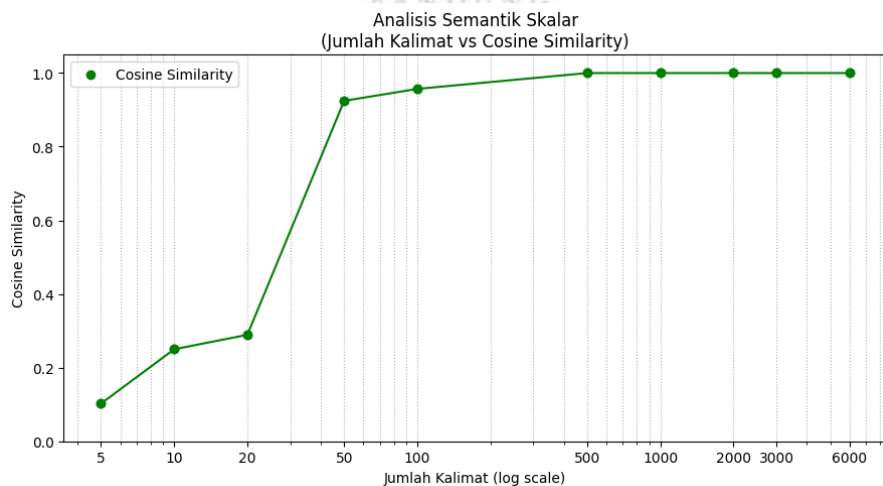
Kedua grafik ini secara bersamaan menjabarkan karakteristik FastText secara komprehensif: bahwa performa semantik tidak hanya dipengaruhi oleh banyaknya data, tetapi juga oleh bentuk morfologis, panjang kalimat, dan kejelasan hubungan leksikal antar kata. Dengan pendekatan ganda ini, evaluasi kinerja model menjadi lebih menyeluruh dan merefleksikan tantangan nyata dalam pemrosesan bahasa Arab.

Gambar 4.8 dan 4.9 menunjukkan dua jenis visualisasi yang menggambarkan kinerja model FastText dari dua pendekatan berbeda, yaitu analisis skalar dan kontekstual.



Gambar 4. 8 Visualisasi Analisis Semantik Kontekstual

Visualisasi pertama pada gambar 4.8 menyajikan analisis kontekstual, membandingkan hasil *cosine Similarity* dari berbagai kondisi konteks. Kategori “Konteks Tematik” menghasilkan *Similarity* paling rendah (sekitar 0.33), menunjukkan bahwa makna emosional dan tema seperti kasih sayang atau pekerjaan lebih sulit dipelajari oleh model secara abstrak. Sementara itu, kategori “Ma’rifat” dan “Panjang Kalimat” berada di kisaran 0.7–0.77, menunjukkan pengaruh bentuk morfologis dan panjang struktur terhadap pemahaman makna. Nilai *Similarity* tertinggi diperoleh dari kata “والدة” (0.9796) dan variasi sinonim (0.9987), membuktikan bahwa FastText sangat andal dalam mengenali hubungan leksikal eksplisit dan bentuk-bentuk sinonim langsung.



Gambar 4. 9 Visualisasi Analisis Semantik Skalar

Visualisasi kedua pada gambar 4.9 menunjukkan hubungan antara jumlah kalimat dan *cosine Similarity* dalam skala logaritmik. *Cosine Similarity* meningkat signifikan saat jumlah kalimat bertambah. Dari 5 hingga 20 kalimat, *Similarity* masih rendah (di bawah 0.3). Namun ketika jumlah data meningkat ke 50 dan seterusnya, terjadi lonjakan drastis menuju nilai hampir sempurna (0.9998), yang stabil hingga 6000 kalimat. Ini menunjukkan bahwa semakin banyak data yang diberikan, semakin kuat pemahaman semantik model. Model mampu menangkap konteks dan asosiasi makna secara lebih konsisten dan akurat dalam skala besar.

Gabungan kedua analisis ini memperlihatkan bahwa performa model FastText tidak hanya dipengaruhi oleh jumlah data (skalar), tetapi juga sangat ditentukan oleh jenis variasi semantik (kontekstual). Dalam kasus bahasa Arab yang kaya akan morfologi dan sinonim, FastText mampu menghasilkan representasi kata yang konsisten dan bermakna ketika diberikan data yang cukup dan variasi leksikal yang eksplisit.

Berdasarkan keseluruhan hasil analisis, dapat disimpulkan bahwa kombinasi antara variasi sinonim eksplisit dalam konteks keluarga dan jumlah kalimat di atas 500 memberikan hasil *cosine similarity* tertinggi dan paling stabil. Konteks yang paling kuat dalam membentuk kemiripan semantik adalah konteks keluarga dengan sinonim seperti “والدة” dan “ماما”, yang menghasilkan nilai *similarity* di atas 0.998 secara konsisten. Selain itu, model mulai menunjukkan performa optimal sejak jumlah data mencapai 500 kalimat, dengan nilai *cosine similarity* yang tetap stabil hingga 6000 kalimat. Hal ini menunjukkan bahwa baik konteks leksikal yang jelas maupun skala data yang memadai merupakan dua pilar utama keberhasilan pemodelan semantik dengan FastText.

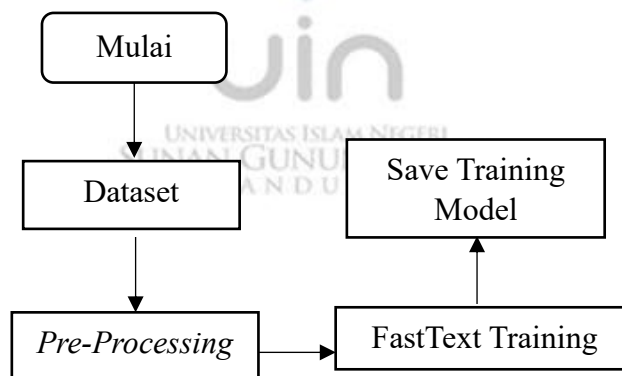
Model FastText menunjukkan performa luar biasa dalam mengenali hubungan semantik, bahkan dalam variasi bentuk kata, panjang kalimat, maupun konteks tematik. Nilai *cosine similarity* yang tinggi di seluruh skenario menunjukkan bahwa FastText sangat andal untuk pemrosesan bahasa Arab yang kompleks dan kaya konteks.

4.2 Penerapan Model pada Korpus Utama

Setelah validasi awal model FastText menggunakan data dummy yang berfokus pada kemiripan sinonim kata “امي” telah menunjukkan hasil yang menjanjikan, langkah selanjutnya dalam penelitian ini adalah menerapkan model yang telah dilatih pada korpus utama, yaitu dataset Al-Qur'an berbahasa Arab secara keseluruhan. Tahap ini merupakan inti dari analisis, di mana kualitas representasi kata yang dihasilkan oleh model akan dievaluasi secara komprehensif dalam konteks dataset yang lebih luas.

Melalui penerapan pada korpus utama ini, penelitian bertujuan untuk tidak hanya memvalidasi FastText sebagai alat yang efektif untuk bahasa Arab tetapi juga untuk memberikan wawasan mendalam tentang bagaimana hyperparameter memengaruhi representasi semantik kata dalam teks keagamaan yang kompleks seperti Al-Qur'an. Hasil dari tahap ini akan menjadi dasar untuk analisis lebih lanjut dan penarikan kesimpulan pada bab berikutnya.

Berikut merupakan diagram alur penelitian awal yang dimulai dari studi literatur, pengumpulan data penelitian untuk studi kasus, penyelesaian menggunakan algoritma FastText dan di *pre-processing* terlebih dahulu.



Gambar 4. 10 Diagram Awal Alur Penelitian

Dalam penelitian skripsi ini, data yang digunakan adalah Al-Qur'an bahasa Arab. Data ini melalui serangkaian tahapan *pre-processing* dan pelatihan FastText. Kedua proses diintegrasikan secara simultan melalui suatu algoritma yang diimplementasikan melalui platform python. Pendekatan bertahap ini diterapkan untuk meminimalkan waktu eksekusi program dan memastikan konsistensi alur kerja. Dataset Al-Qur'an berbahasa Arab dianggap sangat

relevan untuk tujuan penelitian ini karena kompleksitas struktural dan kekayaan maknanya yang beragam.

11/1 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ
 12/1 الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ
 13/1 الرَّحْمَنِ الرَّحِيمِ
 14/1 مَالِكِ يَوْمِ الدِّينِ
 15/1 إِيَّاكَ نَعْتَذِرُ وَإِيَّاكَ نَسْتَعِينُ
 16/1 آمِينَ الْمَلِكِ الْغَنِيِّ
 17/1 آمِينَ الَّذِينَ أَلْغَقْتَ عَلَيْهِمْ غَيْرَ الْمَغْضُوبِ عَلَيْهِمْ وَلَا الضَّالِّينَ
 11/2 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ
 12/2 أَلَيْسَ الْكِتَابُ لَا رَيْبَ فِيهِ هَلْ يَنْفَعُنِي
 13/2 الَّذِينَ يُؤْمِنُونَ بِالْغَيْبِ وَيُعْهِدُونَ صَلَاةَ رَبِّهِمْ وَرُزْقَاهُمْ يُنْفِقُونَ
 14/2 وَالَّذِينَ يُؤْمِنُونَ بِمَا أُنْزِلَ إِلَيْكَ وَمَا أُنْزِلَ مِنْ قَبْلِكَ وَيَآخِزُونَ هُمْ يُوَقِّتُونَ
 15/2 أُولَئِكَ عَلَى هَدًى مِنْ رَبِّهِمْ وَأُولَئِكَ هُمُ الْمُفْلِحُونَ
 16/2 إِنَّ الَّذِينَ كَفَرُوا سَوَاءٌ عَلَيْهِمْ أَأَنْذَرْتَهُمْ أَمْ لَمْ تُنْذِرْهُمْ لَا يُؤْمِنُونَ
 17/2 اخْتَلَفَ اللَّهُ عَلَى قُلُوبِهِمْ وَعَلَى سَمْعِهِمْ وَعَلَى أَبْصَارِهِمْ غِشَاوَةٌ وَلَهُمْ عَذَابٌ عَظِيمٌ
 18/2 وَفِي النَّاسِ مَنْ يَقُولُ آمَنَّا بِاللَّهِ وَبِالْيَوْمِ الْآخِرِ وَمَا هُمْ بِمُؤْمِنِينَ
 19/2 يَخْتَدِعُونَ اللَّهَ وَالَّذِينَ آمَنُوا وَمَا يُخْدَعُونَ إِلَّا أَنْفُسُهُمْ وَمَا يَشْعُرُونَ
 20/2 أَفَبِمَا نَرْسُلُ رُسُلًا هُمْ كُفَرُوا بِهِمْ وَلَهُمْ عَذَابٌ أَلِيمٌ وَمَا كَانُوا يَنْفَكُونَ
 21/2 وَأَوْدَا قَبْلَ لَهْمٍ لَا تَحْسِبُوا فِي الْأَرْضِ قَالُوا إِنَّمَا نَعْنُ مُنْجِبُونَ
 22/2 أَلَا إِنَّهُمْ هُمُ الْمُفْسِدُونَ وَلَكِنْ لَا يَشْعُرُونَ

Gambar 4. 11 Dataset Al-Qur'an yang digunakan

Pada gambar 4.11 merupakan dataset dalam penelitian ini yaitu teks Al-Qur'an tanzil yang merupakan bersumber dari Mushaf Madinah, yang dianggap sebagai salinan Al-Qur'an paling otentik (riwayat Hafs).

11/111 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ تَبَّتْ يُدَا أَيْي لَهْمٍ وَتَبَّتْ
 12/111 مَا أَغْنَى عَنْهُ مَالُهُ وَمَا كَسَبَ
 13/111 سَيَصْلَى نَارًا ذَاتَ لَهْمٍ
 14/111 وَأَمْرًا تُهْدَى حَقَالَةُ الْخَطْبِ
 15/111 فِي جِيدِهَا حَبْلٌ مِنْ مَسَدٍ
 11/112 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ قُلْ هُوَ اللَّهُ أَحَدٌ
 12/112 اللَّهُ الصَّمَدُ
 13/112 لَمْ يَلِدْ وَلَمْ يُولَدْ
 14/112 وَلَمْ يَكُنْ لَهُ كُفُوًا أَحَدٌ
 11/113 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ قُلْ أَعُوذُ بِرَبِّ الْقَلْبِ
 12/113 مِنْ شَرِّ مَا خَلَقَ
 13/113 وَمِنْ شَرِّ غَاسِقٍ إِذَا وَقَبَ
 14/113 وَمِنْ شَرِّ النَّفَّاثَاتِ فِي الْعُقَدِ
 15/113 وَمِنْ شَرِّ حَاسِدٍ إِذَا حَسَدَ
 11/114 بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ قُلْ أَعُوذُ بِرَبِّ النَّاسِ
 2/114 مَلِكِ النَّاسِ
 3/114 إِلَهِ النَّاسِ
 4/114 مِنْ شَرِّ الْوَسْوَاسِ الْخَنَّاسِ
 5/114 الَّذِي يُوَسْوِسُ فِي صُدُورِ النَّاسِ
 6/114 مِنَ الْجِنَّةِ وَالنَّاسِ

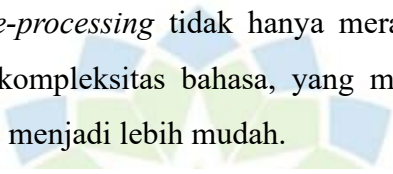
Gambar 4. 12 Lanjutan Dataset yang digunakan

Dataset yang digunakan terdapat 82.796 kata, 736.490 karakter, dengan 30 juz yang berisi 114 surat, dan 6236 ayat. Dalam dataset terdapat tanda baca seperti titik, koma, dan lainnya, semua itu tidak menjadi masalah karena yang

terpenting adalah kata yang terdapat dalam dataset memiliki karakteristik yang unik dan makna yang mendalam untuk dianalisis dalam penelitian ini.

4.3 Pre-Processing

Proses ini terdiri dari beberapa tahapan, seperti *tokenization* yang memisahkan teks menjadi satuan kata, pembersihan teks yang menghapus karakter yang tidak relevan, menghapus kata-kata yang sering muncul namun tidak memiliki makna, dan di akhiri dengan *lemmatization* yang mengembalikan kata ke bentuk dasarnya. Setelah melakukan tahapan-tahapan tersebut, data menjadi lebih terstruktur dan berfokus pada makna utama. Misalnya, kalimat panjang telah diubah menjadi barisan kata seperti *مالك يوم دين*. Hasil ini menunjukkan bahwa *pre-processing* tidak hanya merapikan data tetapi juga membantu mengurangi kompleksitas bahasa, yang membuat analisis proses pelatihan model FastText menjadi lebih mudah.



بسم له رحمن رحيم
حمد لله رب العالمين
رحمن رحيم
مالك يوم دين
اياك تعبد واياك تستعين
اهدنا صراط مستقيم
صراط انعمت عليهم مغضوب عليهم ضالين
بسم له رحمن رحيم م
كتاب ريب هدى للمتقين
يومنون بالغيب ويقيمون صلاة ومما رزقناهم ينفقون
يومنون انزل يك انزل قبلك وبالاخرة يوقنون
اولئك هدى ربهم واولئك مفلحون
ان كفروا سواء عليهم انذرتهم ام تنذرهم يومنون

Gambar 4. 13 Hasil Pre-Processing

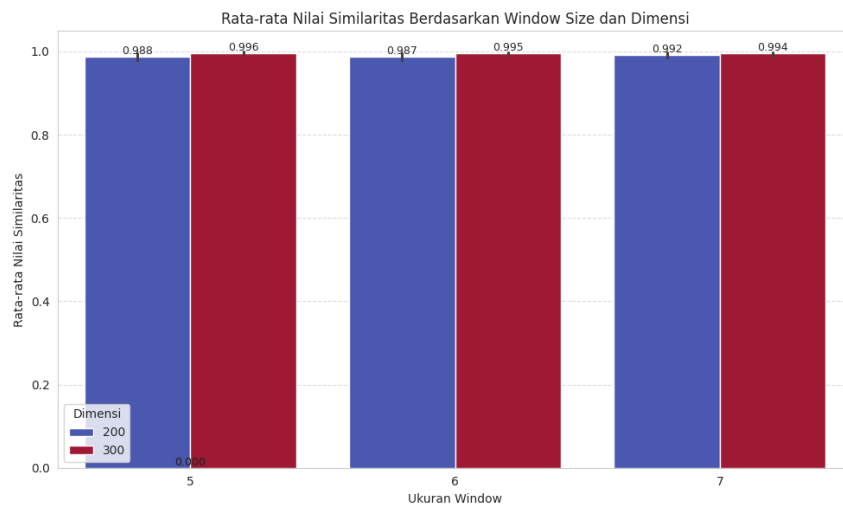
Setelah dilakukan tahap *pre-processing* pada data asli Al-Qur'an bahasa Arab berhasil mengurangi jumlah karakter dari 736.490 karakter menjadi 326.883 karakter. Ini menunjukkan bahwa data Al-Qur'an telah berhasil disederhanakan menjadi kumpulan kata inti yang lebih bersih dan relevan melalui tahap *pre-processing*.

4.4 Training Data

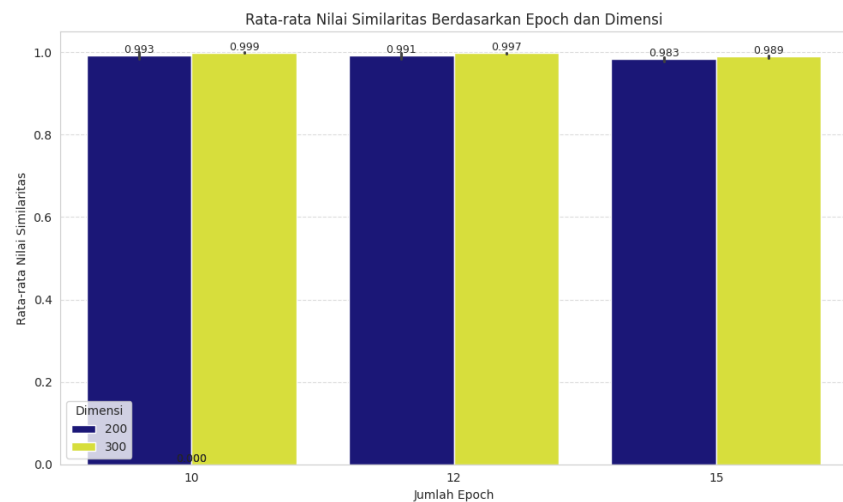
Proses ini menjelaskan secara detail proses fundamental dalam melatih model FastText menggunakan dataset Al-Qur'an berbahasa Arab yang telah melalui tahap *pre-processing*. Proses pelatihan ini dikerjakan menggunakan

Google Colab sebagai platform untuk menjalankan program dan mesin yang dijalankan adalah bahasa python. Seluruh pelatihan dilaksanakan secara lokal dengan memanfaatkan pustaka dan modul python yang sesuai. Model yang digunakan adalah FastText, sebuah pengembangan dari Word2Vec yang dipilih karena kemampuannya dalam mempertimbangkan sub-kata (*character n-gram*). Kemampuan ini sangat penting dalam menangani morfologi kata Arab.

Untuk mempermudah visualisasi hasil training data, digunakan *Principal Component Analysis* (PCA). Berikut adalah hasil dari training data yang sudah dilakukan dalam penelitian ini:



Gambar 4. 14 Diagram Rata-rata Nilai Similaritas berdasarkan *Window size* dan Dimensi



Gambar 4. 15 Rata-rata Nilai Similaritas Berdasarkan Epoch dan Dimensi

Pada Gambar 4.14 dan 4.15, hasil eksperimen performa model dievaluasi berdasarkan rata-rata nilai similaritas, dengan memvariasikan parameter kunci seperti *window size*, epoch, dan dimensi *embeddings*. Dari analisis kedua visualisasi, terlihat jelas bahwa dimensi *embeddings* merupakan faktor paling dominan yang memengaruhi nilai similaritas. Penggunaan dimensi 300 secara konsisten menghasilkan nilai similaritas yang jauh lebih tinggi (0.989-0.999) dibandingkan dimensi 200 (0.983-0.993), menunjukkan bahwa representasi vektor kata yang lebih kaya sangat krusial. Selanjutnya, terkait jumlah epoch, nilai similaritas tertinggi untuk kedua dimensi dicapai pada 10 epoch, dengan kecenderungan penurunan performa pada epoch yang lebih tinggi (12 dan 15) yang mengindikasikan kemungkinan *overfitting* atau konvergensi optimal. Sementara itu, pengaruh *window size* relatif lebih kecil pada dimensi 300, *window size* 5 sedikit unggul (0.996), namun perbedaan dengan *window size* 6 dan 7 sangat tipis.

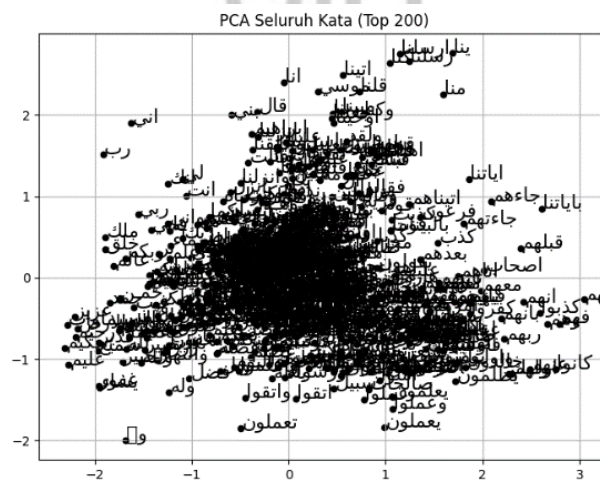
Oleh karena itu, konfigurasi optimal yang disarankan untuk mencapai rata-rata nilai similaritas tertinggi adalah dimensi *embeddings* 300, dengan 10 epoch, dan *window size* 5, yang berhasil menghasilkan nilai similaritas sangat tinggi hingga 0.999. Dalam domain NLP, nilai mendekati 1 menunjukkan bahwa vektor kata berada dalam arah yang hampir identik di ruang vektorial, sehingga model FastText dinilai berhasil merepresentasikan hubungan semantik antar kata secara optimal. Hal ini juga menjadi indikator bahwa konfigurasi hyperparameter yang digunakan mampu memaksimalkan kualitas representasi kata.

Konfigurasi optimal ini ditemukan dan divalidasi melalui pengujian berbagai hyperparameter pada korpus utama. Sebagai contoh representasi dari performa konfigurasi optimal ini, hasil pengujian *cosine Similarity* untuk lima kata utama, yaitu “علم”, “إيمان”, “إسلام”, “رزق”, dan “صلاة”, disajikan pada tabel 4.8. Tabel ini menunjukkan 5 kata terdekat (dengan nilai similaritas tertinggi) untuk setiap target, yang merefleksikan kemampuan model dalam menangkap hubungan kontekstual dan semantik dalam Al-Qur'an.

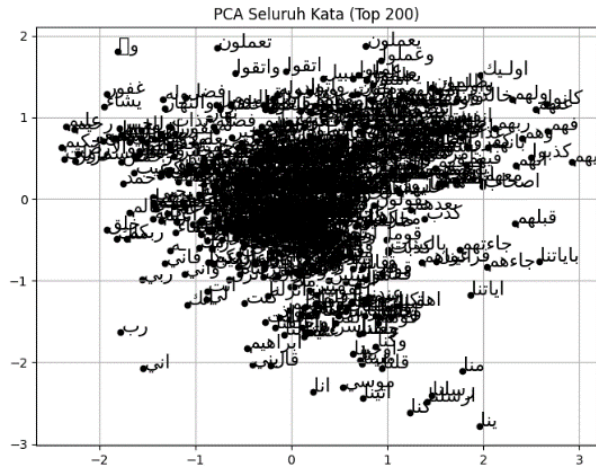
Tabel 4. 8 Hasil *Cosine Similarity* pengujian Korpus Utama (Dimensi 300, Epoch 10, *Window size* 5)

Kata Target	<i>Cosine Similarity</i> tertinggi	Contoh Kata <i>Similarity</i>
علم	0.9998	علمو
ايمان	0.9999	أرتقب
اسلام	0.9999	زام
رزق	0.9995	ألمتصدق
صلاة	0.9999	محمود

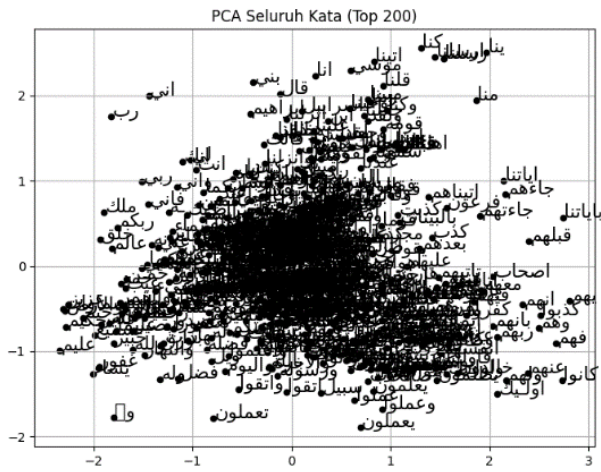
Analisis terhadap tabel ini akan memberikan gambaran konkret mengenai bagaimana model FastText mengelompokkan kata-kata berdasarkan kemiripan makna atau hubungan kontekstualnya dalam dataset Al-Qur'an. Hasil ini penting untuk memahami efektivitas model dalam merepresentasikan semantik bahasa Arab yang kompleks. Untuk memberikan pemahaman visual yang lebih mendalam mengenai struktur, sebaran, dan hubungan dalam ruang fitur, analisis visual menggunakan *Principal Component Analysis* (PCA) juga dilakukan terhadap *word embeddings* yang di hasilkan FastText terhadap korpus Al-Qur'an



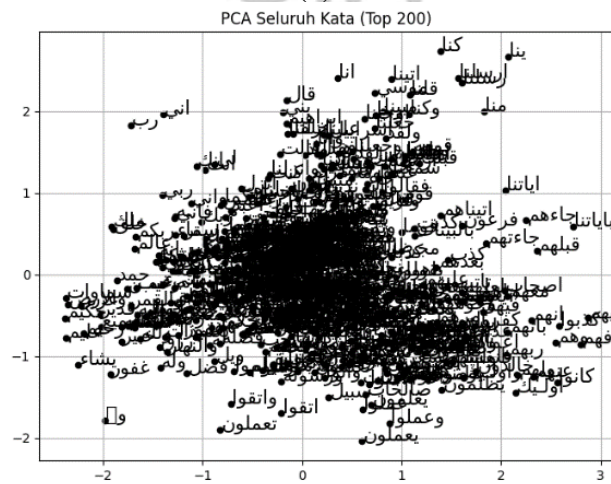
(a)



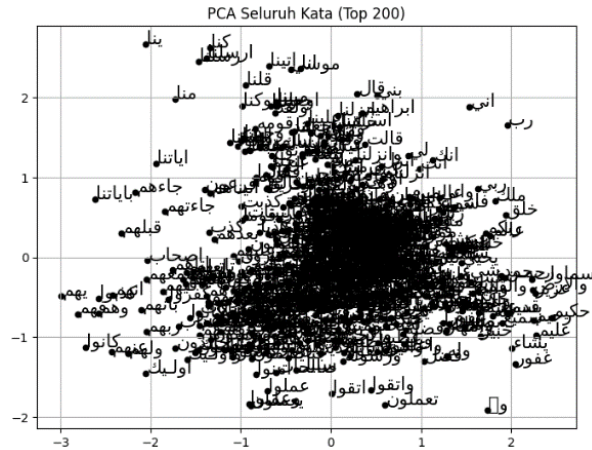
(b)



(c)



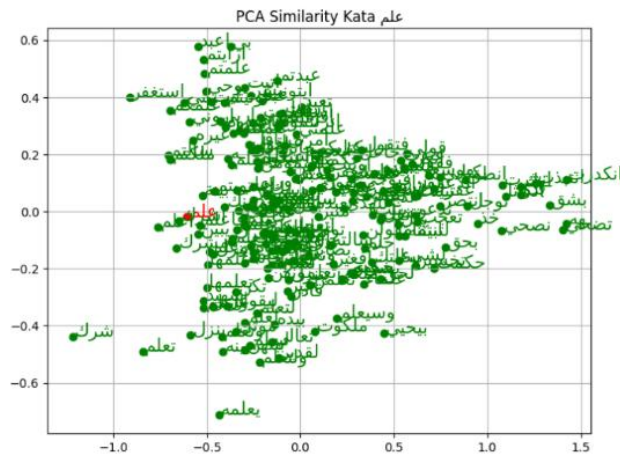
(d)



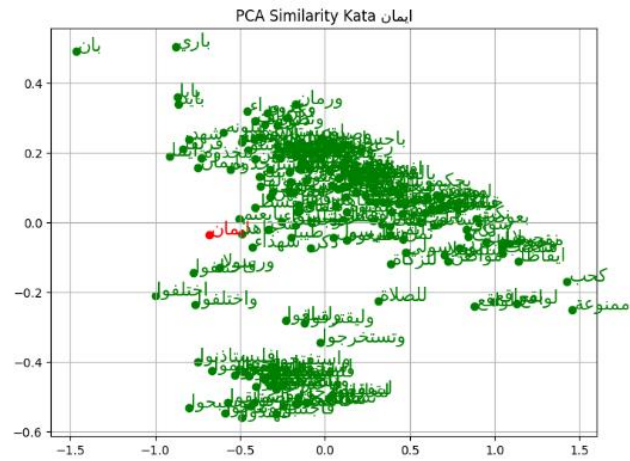
(e)

Gambar 4. 16 Ilustrasi Vektor pada Pengujian Korpus Utama (Dimensi 300, Epoch 10, *Window size* 5)

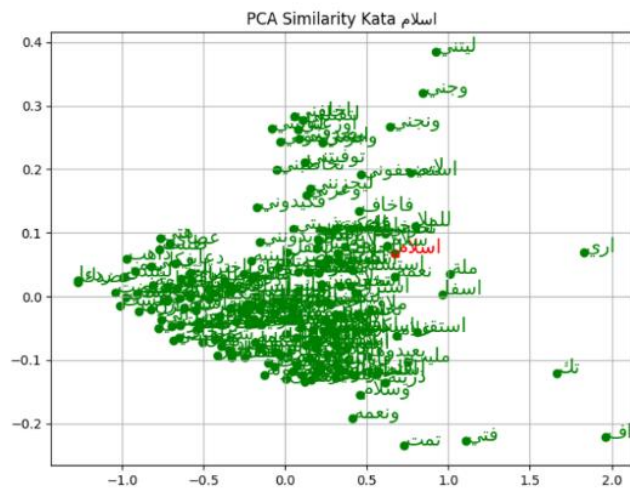
Gambar 4.16 memperlihatkan distribusi keseluruhan *word embeddings* dalam ruang vektor dua dimensi. Visualisasi ini menunjukkan adanya klaster utama yang padat, mengindikasikan bahwa sebagian besar kosa kata dalam dataset saling terkait dan terorganisir dalam ruang vektor. Meskipun ada beberapa kata yang tersebar di luar klaster utama, pola ini secara umum menegaskan kemampuan model dalam memetakan keseluruhan kosa kata ke dalam ruang vektor yang terstruktur



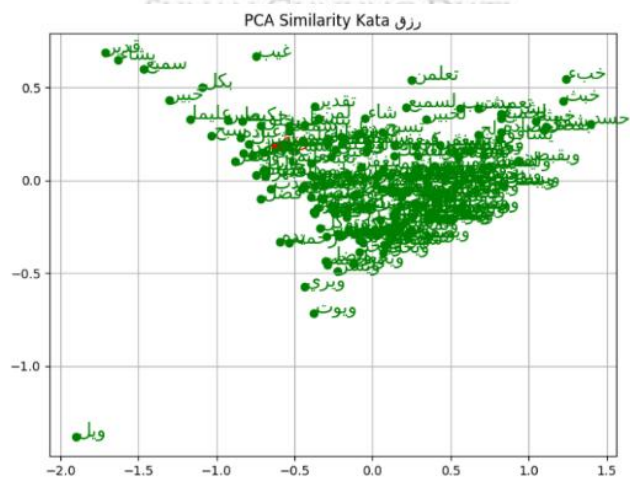
(a)



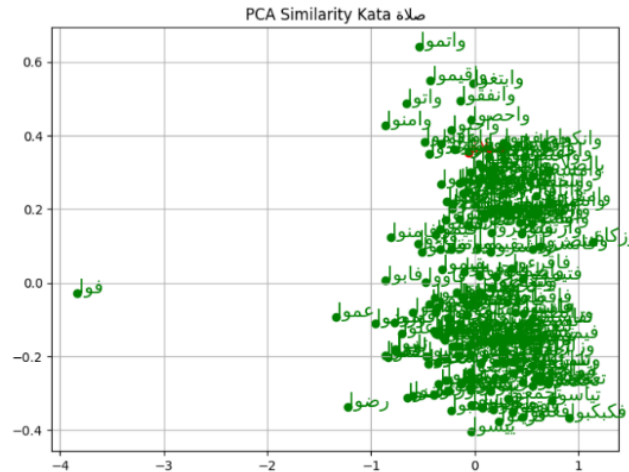
(b)



(c)



(d)



(e)

Gambar 4. 17 Ilustrasi *Similarity* Ilustrasi Vektor pada Pengujian Korpus Utama (Dimensi 300, Epoch 10, *Window size* 5)

Secara lebih spesifik, Gambar 4.17 menunjukkan konsentrasi kata-kata yang memiliki kemiripan semantik atau hubungan kontekstual yang kuat di sekitar kata target. Kerapatan dan kedekatan klaster kata-kata serupa ini secara visual memvalidasi kemampuan model FastText dalam mengidentifikasi kemiripan semantik. Hal ini membuktikan bahwa model berhasil menempatkan kata-kata yang relevan secara semantik dalam jarak vektor yang sangat dekat, menunjukkan kemampuannya membedakan dan mengelompokkan kata berdasarkan makna. Visualisasi-visualisasi ini sangat kuat memvalidasi kemampuan model FastText dalam mengidentifikasi kemiripan semantik, baik secara umum dalam korpus maupun spesifik di sekitar kata target.

Setelah mengidentifikasi konfigurasi hyperparameter optimal dan menganalisis similaritas kata-kata utama secara umum, penelitian ini akan melanjutkan dengan fokus pada pengujian yang lebih spesifik terhadap kemampuan model dalam mengenali hubungan sinonim. Berbeda dengan pengujian similaritas umum yang telah disajikan, pengujian sinonim akan melibatkan pasangan kata yang secara linguistik memang dikategorikan sebagai sinonim dalam bahasa Arab, untuk mengevaluasi akurasi model dalam membedakan kedekatan makna yang lebih presisi. Hal ini bertujuan untuk memberikan bukti lebih lanjut mengenai keunggulan FastText dalam menangani kekayaan morfologi bahasa Arab dan kompleksitas hubungan semantik di dalamnya.

Sebagai contoh, tabel 4.9 menyajikan hasil *cosine similarity* untuk kata target “خير”, beserta kata-kata serupa, nilai similaritas, dan kategori tema utama yang teridentifikasi dari korpus Al-Qur’an yang telah di *pre-processing*.

Tabel 4. 9 Tabel Analisis Similaritas Sinonim Kata "خير"

No.	Kata Serupa	Terjemahan Indonesia	Skor Similarity	Tema Utama
1	خيرا	kebaikan	0,9509	Nilai Moral & Kebaikan
2	وخير	dan kebaikan	0,9490	Nilai Moral & Kebaikan
3	خيرة	pilihan terbaik	0,9461	Nilai Moral & Kebaikan
4	بخير	dalam keadaan baik	0,9450	Nilai Moral & Kebaikan
5	ويصدكم	dan Dia menghalangi kalian	0,9438	Hubungan Sosial & Interaksi
6	ويقللكم	dan Dia mengurangi kalian	0,9415	Perbuatan & Amal
7	كبير	Maha Besar	0,9402	Nilai Moral & Kebaikan
8	تغرركم	menipumu	0,9391	Hubungan Sosial & Interaksi
9	تعمل	kamu berbuat	0,9391	Perbuatan & Amal
10	لعلكم	agar kamu	0,9384	Perintah & Larangan
11	ويجركم	dan Dia menarikmu	0,9382	Perbuatan & Amal
12	يحب	Dia mencintai	0,9376	Hubungan Sosial & Interaksi
13	واتقين	dan bertakwalah	0,9374	Perintah & Larangan
14	ويلكم	celakalah kalian	0,9370	Perintah & Larangan
15	قلوبكم	hati kalian	0,9356	Hubungan Sosial & Interaksi
16	لايمان	karena iman kalian	0,9355	Keimanan & Ketakwaan
17	ويحذركم	dan Dia memperingatkanmu	0,9352	Perintah & Larangan

No.	Kata Serupa	Terjemahan Indonesia	Skor Similarity	Tema Utama
18	ويامرکم	dan Dia memerintahkanmu	0,9346	Perintah & Larangan
19	ويمدکم	dan Dia akan membantu kalian	0,9342	Perlindungan & Pertolongan Allah
20	بدينکم	dengan agama kalian	0,9333	Keimanan & Ketakwaan
21	احسنکم	kamu berbuat baik	0,9332	Perbuatan & Amal
22	ينصرکم	dan Dia akan menolongmu	0,9329	Perlindungan & Pertolongan Allah
23	وليسنکم	dan jika Dia menyentuh kalian	0,9323	Perlindungan & Pertolongan Allah
24	مقلبکم	pergantian keadaan kalian	0,9312	Perbuatan & Amal
25	بصيرة	penglihatan batin	0,9311	Petunjuk & Hidayah
26	مومن	orang beriman	0,9307	Keimanan & Ketakwaan
27	ويزککم	dan menyucikan kalian	0,9306	Petunjuk & Hidayah

Tabel 4.9 secara spesifik menggambarkan bagaimana model memetakan kata target dengan kata-kata lain yang memiliki kemiripan semantik dan kontekstual, yang kemudian dikelompokkan berdasarkan tema utama. Analisis dari tabel ini akan fokus pada bagaimana nilai similaritas yang tinggi untuk kata-kata dalam kategori tema yang sama menunjukkan kemampuan model untuk menangkap sinonim atau kedekatan makna dalam konteks Al-Qur'an. Hal ini lebih lanjut membuktikan bahwa FastText, dengan kemampuannya memahami struktur sub-kata, sangat efektif dalam menangani keragaman kata dalam bahasa Arab dan memahami hubungan makna yang kompleks di baliknya, termasuk hubungan sinonim. Pengelompokan tema utama terdiri dari tematik utama, yaitu:

a. Nilai Moral & Kebaikan

Tema ini menggambarkan nilai-nilai luhur seperti kebaikan, keagungan, dan keberkahan yang menjadi inti dalam ajaran Islam. Kebaikan (khair) dalam konteks Al-Qur'an mencakup segala bentuk manfaat yang membawa maslahat dunia dan akhirat, serta terkait erat dengan akhlak dan hikmah. Nilai moral yang baik merupakan landasan amal dan petunjuk Allah. Kata-kata terkait, seperti خيرا (kebaikan), وخير (dan kebaikan), خيرة (pilihan terbaik), بخير (dalam keadaan baik), dan كبير (Maha Besar).

b. Hubungan Sosial dan Interaksi

Tema ini mencerminkan dinamika hubungan manusia seperti cinta, tipu daya, dan komunikasi hati. Dalam Al-Qur'an, manusia diperingatkan untuk menjaga diri dari tipu daya dunia dan setan, serta dianjurkan untuk saling mencintai dan menjaga hati dalam interaksi sosial. Kata-kata terkait, seperti ويصدكم (dan Dia menghalangi kalian), يغرنكم (mereka menipumu), تغرنكم (menipumu), يحب (Dia mencintai), dan قلوبكم (hati kalian).

c. Perbuatan dan Amal

Tema ini menunjukkan pentingnya tindakan nyata dalam Islam. Amal perbuatan, baik yang terlihat kecil maupun besar, akan dihitung dan dibalas oleh Allah. Al-Qur'an mengaitkan iman yang benar dengan amal yang saleh sebagai prasyarat keselamatan dan keberhasilan akhirat. Kata-kata terkait, seperti تعمل (kamu berbuat), احسنتم (kamu berbuat baik), ويقللكم (dan Dia mengurangi kalian), ويجركم (dan Dia menarikmu), dan متقلبكم (pergantian keadaan kalian).

d. Perintah dan Larangan

Tema ini menyoroti struktur instruksi ilahi dalam Al-Qur'an. Allah memberi perintah dan larangan sebagai bentuk kasih sayang dan tuntunan agar manusia tidak tersesat. Perintah seperti bertakwa, menjaga diri, dan menjauhi tipu daya dunia menjadi penanda jalan lurus. Kata-kata terkait, seperti واتقوا (dan bertakwalah), ويحذركم (dan Dia memperingatkanmu), ويامرکم (dan Dia memerintahkanmu), لحکم (agar kamu), dan ويلکم (celakalah kalian).

e. Keimanan dan Ketakwaan

Keimanan menjadi fondasi dasar dalam Al-Qur'an. Tanpa iman, amal menjadi sia-sia. Tema ini menyoroti pentingnya keyakinan kepada Allah, serta hubungan erat antara iman, agama, dan identitas spiritual seorang Muslim. Ketakwaan memperkuat nilai iman melalui amal yang benar. Kata-kata terkait, seperti لايمانكم (karena iman kalian), بدينكم (dengan agama kalian), dan مومن (orang beriman).

f. Perlindungan & Pertolongan Allah

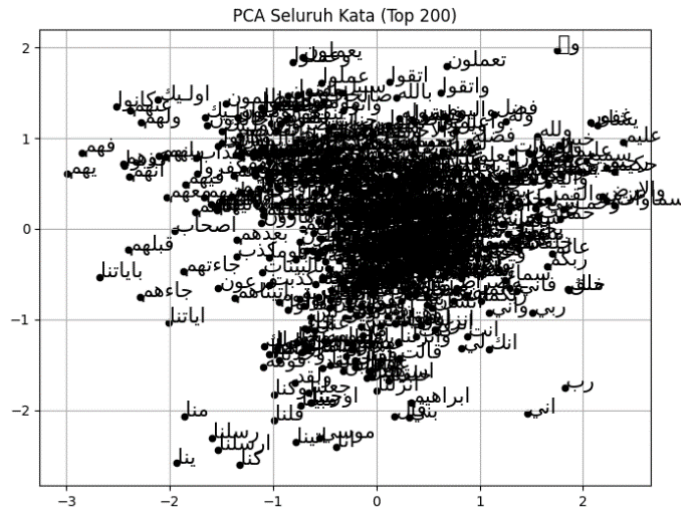
Tema ini mengandung janji pertolongan dan perlindungan dari Allah terhadap hamba-Nya yang taat dan bertawakal. Allah menjanjikan kemenangan, dukungan, dan penyelamatan dari marabahaya kepada orang-orang yang menyerahkan diri sepenuhnya kepada-Nya. Kata-kata terkait, seperti ويمدكم (dan Dia akan membantu kalian), وينصركم (dan Dia akan menolongmu), dan وليمسكنكم (dan jika Dia menyentuh kalian).

g. Petunjuk & Hidayah

Petunjuk (hidayah) adalah anugerah terbesar yang membimbing manusia dari kegelapan menuju cahaya. Tema ini menunjukkan bahwa Allah memberikan penglihatan batin, penyucian jiwa, dan ilmu sebagai sarana untuk mencapai kebenaran dan kebahagiaan hakiki. Kata-kata terkait, seperti بصيرة (penglihatan batin), dan يزيكم (dan menyucikan kalian).

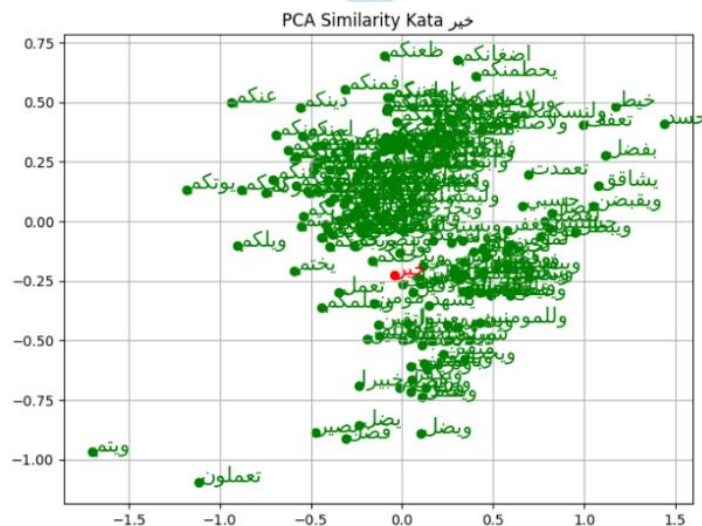
Model FastText mendemonstrasikan bahwa model mampu membangun representasi semantik yang stabil, di mana kata-kata yang paling mirip dengan "khair" (kebaikan) bukanlah kumpulan kata acak, melainkan terorganisir dalam klaster-klaster makna yang koheren, seperti nilai moral, amal perbuatan, keimanan, dan pertolongan Ilahi. Hal ini membuktikan bahwa FastText dapat menangkap nuansa semantik kompleks dalam teks Al-Qur'an berbahasa Arab, terutama dalam memahami keterhubungan makna secara tematik dan kontekstual.

Untuk memberikan pemahaman visual yang lebih mendalam mengenai struktur, sebaran, dan hubungan dalam ruang fitur, analisis visual menggunakan *Principal Component Analysis* (PCA) juga dilakukan terhadap *word embeddings* yang di hasilkan FastText terhadap korpus Al-Qur'an.



Gambar 4. 18 Ilustrasi Training FastText pada Korpus Al-Qur'an

Visualisasi pada gambar 4.18 ini menunjukkan distribusi umum *word embeddings* dari korpus Al-Qur'an. Plot ini menampilkan adanya kluster padat di bagian tengah, mengindikasikan bahwa mayoritas kata-kata memiliki keterkaitan semantik dan kontekstual yang erat. Sebaran yang relatif merata dengan beberapa *outlier* menunjukkan bahwa model berhasil memetakan kosa kata yang luas ke dalam ruang vektor yang terstruktur.



Gambar 4. 19 Ilustrasi penyebaran *Similarity* dengan Kata Target خير

Visualisasi pada gambar 4.19 menyoroti kata target “خير” dengan warna merah, dan kata-kata yang memiliki kemiripan tertinggi dengannya menggunakan warna hijau. Visualisasi ini jelas menunjukkan bahwa kata-kata serupa

mengelompok secara erat di sekitar kata target, membentuk klaster yang padat. Kerapatan klaster ini adalah bukti visual yang kuat bahwa model FastTeks berhasil menangkap hubungan semantik dan kontekstual yang signifikan dalam teks Al-Qur'an, menempatkan kata-kata yang relevan secara makna pada posisi yang berdekatan dalam ruang vektor.

Pada bagian ini akan menampilkan cuplikan ayat-ayat Al-Quran. Pemilihan ayat-ayat ini bertujuan untuk menjelaskan secara konkret bagaimana kata-kata yang diidentifikasi serupa oleh model FastText memiliki kesamaan tema atau terkait dengan konsep “خير” dalam konteks ayat tersebut.

1. Nilai Moral dan Kebaikan

Pada klaster tematik ini ditemukan kata-kata yang ontologinya berpusat pada konsep خيرا (kebaikan) serta manifestasi keagungan ilahi.

a. خيرا (kebaikan)

Arti khaira merujuk pada segala bentuk kebaikan yang membawa manfaat dan pahala, baik di kehidupan dunia maupun akhirat, serta merupakan landasan akhlak mulia. Ini menegaskan bahwa nilai kebaikan adalah sesuatu yang fundamental dan abadi. Hal ini tertuang dalam firman Allah pada surat Al-A'la ayat 17:

وَالْآخِرَةُ خَيْرٌ وَأَبْقَىٰ ١٧

Artinya: Padahal Kehidupan akhirat itu lebih baik dan lebih kekal

b. وخير (dan kebaikan)

Frasa وخير menekankan penambahan atau pelengkap dari suatu kebaikan yang berlimpah, seringkali mengacu pada hikmah atau anugerah yang diberikan Allah kepada hamba-Nya. Istilah ini menggarisbawahi bahwa kebaikan tersebut adalah sesuatu yang besar dan bermanfaat luas. Ini termuat dalam firman Allah pada penggalan surat Al-Baqarah ayat 269:

فَقَدْ أُوتِيَ خَيْرًا كَثِيرًا ۚ

Artinya: Sungguh Ia telah di anugerahi kebaikan yang banyak

c. خيرة (pilihan terbaik)

Kata خيرة secara ontologis menyoroti kehendak mutlak dan kebijaksanaan Allah dalam memilih dan menciptakan. Ini menyiratkan bahwa setiap pilihan atau ciptaan dari Allah adalah yang terbaik dan paling sempurna,

yang seharusnya diterima dengan penuh kepasrahan dan keyakinan. Hal ini dapat ditemukan dalam firman Allah pada penggalan surat Al-Qasas ayat 68:

وَرَبُّكَ يَخْلُقُ مَا يَشَاءُ وَيَخْتَارُ ۚ

Artinya: Tuhanmu menciptakan dan memilih apa yang dia kehendaki

d. بخير (dalam keadaan baik)

Ekspresi بخير menggambarkan kondisi kesejahteraan, keberuntungan, atau berada dalam kebaikan, baik secara fisik maupun spiritual. Dalam konteks Al-Qur'an, ini sering kali terkait dengan balasan atas amal saleh atau kondisi umum orang-orang yang beriman. Frasa ini tertuang dalam firman Allah pada surat Al-Baqarah ayat 184:

أَيَّامًا مَّعْدُودَاتٍ فَمَنْ كَانَ مِنْكُمْ مَّرِيضًا أَوْ عَلَى سَفَرٍ فَعِدَّةٌ مِنْ أَيَّامٍ أُخَرَ وَعَلَى الَّذِينَ يُطِيقُونَهُ فِدْيَةٌ طَعَامَ مِسْكِينٍ فَمَنْ تَطَوَّعَ خَيْرًا فَهُوَ خَيْرٌ لَهُ وَأَنْ تَصُومُوا خَيْرٌ لَكُمْ إِنْ كُنْتُمْ تَعْلَمُونَ ١٨٤

Artinya: (Yaitu) beberapa hari tertentu. Maka, siapa di antara kamu sakit atau dalam perjalanan (lalu tidak berpuasa), (wajib mengganti) sebanyak hari (yang dia tidak berpuasa itu) pada hari-hari yang lain. Bagi orang yang berat menjalankannya, wajib membayar fidyah, (yaitu) memberi makan seorang miskin. Siapa dengan kerelaan hati mengerjakan kebajikan, itu lebih baik baginya dan berpuasa itu lebih baik bagimu jika kamu mengetahui.

2. Hubungan Sosial dan Interaksi

Pada klaster tematik ini ditemukan kata-kata yang secara ontologis mencerminkan dinamika hubungan antar manusia dan interaksi dengan entitas non-manusia seperti Allah atau setan, serta peran hati sebagai pusatnya.

a. ويصدكم (dan Dia menghalangi kalian)

Frasa ini menggambarkan tindakan ilahi berupa perlindungan dan pencegahan dari pengaruh negatif yang dapat menyesatkan manusia. Secara ontologis, ini merujuk pada peran Allah sebagai pelindung yang menjauhkan hamba-Nya dari hal-hal yang berbahaya bagi keimanan atau kesejahteraan mereka. Hal ini tertuang dalam firman Allah pada surat Al-Maidah ayat 91:

إِنَّمَا يُرِيدُ الشَّيْطَانُ أَنْ يُوقَعَ بَيْنَكُمْ الْعَدَاوَةَ وَالْبَغْضَاءَ فِي الْخَمْرِ وَالْمَيْسِرِ وَيَصُدَّكُمْ عَنْ ذِكْرِ اللَّهِ وَعَنِ الصَّلَاةِ فَهَلْ أَنْتُمْ مُنْتَهُوْنَ ٩١

Artinya: Sesungguhnya setan hanya bermaksud menimbulkan permusuhan dan kebencian di antara kamu melalui minuman keras dan judi serta (bermaksud) menghalangi kamu dari mengingat Allah dan (melaksanakan) salat, maka tidakkah kamu mau berhenti?

b. تغرنكم (menipumu)

Kata ini merupakan penegasan lebih lanjut mengenai bahaya tertipu oleh kehidupan dunia beserta gemerlapnya. Secara ontologis, ini menggarisbawahi sifat ilusi dari kenikmatan duniawi yang dapat mengalihkan manusia dari tujuan akhirat mereka. Hal ini dapat ditemukan dalam firman Allah pada surat Fatir ayat 5:

يَا أَيُّهَا النَّاسُ إِنَّ وَعْدَ اللَّهِ حَقٌّ فَلَا تَغُرَّنَّكُمُ الْحَيَاةُ الدُّنْيَا وَلَا يَغُرَّنَّكُم بِاللَّهِ الْغُرُورُ ٥

Artinya: Wahai manusia, sesungguhnya janji Allah itu benar. Maka, janganlah sekali-kali kehidupan dunia memperdayakan kamu dan janganlah (setan) yang pandai menipu memperdayakan kamu tentang Allah.

c. يحب (Dia mencintai)

Verba ini secara ontologis menunjukkan sifat kasih sayang Allah yang mendalam kepada hamba-hamba-Nya yang taat dan berbuat kebaikan. Ini menyoroti hubungan mutualis antara pencipta dan ciptaan-Nya, di mana cinta Allah adalah balasan atas ketaatan dan keikhlasan. Frasa ini tertuang dalam firman Allah pada penggalan surat Al-Baqarah ayat 222:

إِنَّ اللَّهَ يُحِبُّ التَّوَّابِينَ وَيُحِبُّ الْمُتَطَهِّرِينَ ٢٢٢

Artinya: Sesungguhnya Allah menyukai orang-orang yang bertobat dan menyukai orang-orang yang menyucikan diri.

d. قلوبكم (hati kalian)

Kata ini secara ontologis merujuk pada hati sebagai pusat kesadaran spiritual, tempat iman bersemayam, dan wadah bagi petunjuk ilahi. Ini adalah inti dari diri manusia yang menjadi sasaran utama bagi hidayah maupun bisikan jahat. Hal ini disebutkan dalam firman Allah pada surat Al-Hujurat ayat 14:

﴿قَالَتِ الْأَعْرَابُ آمَنَّا قُلْ لَمْ تُؤْمِنُوا وَلَكِنْ قُولُوا أَسْلَمْنَا وَلَمَّا يَدْخُلِ الْإِيمَانُ فِي قُلُوبِكُمْ وَإِنْ تُطِيعُوا اللَّهَ وَرَسُولَهُ لَا يَلِتْكُمْ مِنْ أَعْمَالِكُمْ شَيْئًا إِنَّ اللَّهَ غَفُورٌ رَحِيمٌ ١٤﴾

Artinya: Orang-orang Arab Badui berkata, “Kami telah beriman.” Katakanlah (kepada mereka), “Kamu belum beriman, tetapi katakanlah, ‘Kami baru berislam’ karena iman (yang sebenarnya) belum masuk ke dalam hatimu. Jika kamu taat kepada Allah dan Rasul-Nya, Dia tidak akan mengurangi sedikit pun (pahala) amal perbuatanmu.” Sesungguhnya Allah Maha Pengampun lagi Maha Penyayang.

3. Perbuatan dan Amal

Pada klaster tematik ini ditemukan kata-kata yang secara ontologis menunjukkan pentingnya tindakan nyata dan amal perbuatan dalam ajaran Islam.

a. يعمل (kamu berbuat)

Verba ini menekankan prinsip akuntabilitas ilahi, bahwa setiap tindakan yang dilakukan manusia akan tercatat dan memiliki konsekuensi. Secara ontologis, ini mengacu pada gagasan tentang tanggung jawab individu atas perbuatannya di dunia. Hal ini tertuang dalam firman Allah pada surat Az-Zalzalah ayat 7:

﴿فَمَنْ يَعْمَلْ مِثْقَالَ ذَرَّةٍ خَيْرًا يَرَهُ ٧﴾

Artinya: Siapa yang mengerjakan kebaikan seberat zarrah, dia akan melihat (balasan)-nya.

b. احسنتم (kamu berbuat baik)

Ekspresi ini mencerminkan konsep ihsan, yaitu melakukan kebaikan dengan kualitas terbaik dan penuh kesadaran akan pengawasan Allah. Secara ontologis, ini menggarisbawahi nilai kebajikan yang melampaui sekadar ketaatan, menuju kesempurnaan dalam beramal. Istilah ini termuat dalam firman Allah pada penggalan surat Al-isra ayat 7:

﴿إِنْ أَحْسَنْتُمْ أَحْسَنْتُمْ لِأَنْفُسِكُمْ وَإِنْ أَسَأْتُمْ فَلَهَا﴾

Artinya: Jika berbuat baik, (berarti) kamu telah berbuat baik untuk dirimu sendiri. Jika kamu berbuat jahat, (kerugian dari kejahatan) itu kembali kepada dirimu sendiri.

c. وَيَقْتُلُكُمْ (dan Dia mengurangi kalian)

Frasa ini menggambarkan strategi atau ujian Allah dalam konteks amal, bisa jadi dalam jumlah atau kekuatan, sebagai bagian dari rencana ilahi untuk menguji atau memurnikan hamba-Nya. Secara ontologis, ini menyoroiti kekuasaan mutlak Allah dalam mengatur kondisi dan situasi, bahkan dalam detail terkecil. Hal ini dapat ditemukan dalam firman Allah pada surat Al-Anfal ayat 44:

وَإِذْ يُرِيكُمُوهُمْ إِذِ الْتَقَيْتُمْ فِي آَعْيُنِكُمْ قَلِيلًا وَيُقَلِّلُكُمْ فِي آَعْيُنِهِمْ لِيَقْضِيَ اللَّهُ أَمْرًا كَانَ مَفْعُولًا ۖ وَاللَّهُ تَرْجِعُ الْأُمُورَ ۚ ٤٤

Artinya: (Ingatlah) ketika Dia memperlihatkan mereka kepada kamu (orang-orang beriman), ketika kamu berjumpa dengan mereka (berjumlah) sedikit menurut penglihatan matamu dan Dia memperlihatkan kamu (berjumlah) sedikit dalam penglihatan mereka supaya Allah melaksanakan suatu urusan yang harus terjadi. Hanya kepada Allah segala urusan dikembalikan.

d. اَجْرَهُم (memberi pahala)

Kata ini dapat bermakna ganda, yaitu memberi pahala, menyelamatkan, atau bahkan menarik ke arah konsekuensi amal. Secara ontologis, ini mengacu pada peran Allah sebagai pemberi balasan yang adil atau penyelamat bagi mereka yang berhak. Frasa ini tertuang dalam firman Allah pada surat Az-Zumar ayat 10:

قُلْ يٰۤاَعْبَادِ الدِّينِ اٰمَنُوْا اَتَقُوْا رَبَّكُمْ ۖ لِلَّذِيْنَ اَحْسَنُوْا فِيْ هٰذِهِ الدُّنْيَا حَسَنَةٌ ۗ وَاللّٰهُ وٰسِعٌ ۭ اِنَّمَا يُؤَفِّى الصّٰبِرُوْنَ اَجْرَهُمْ بِغَيْرِ حِسَابٍ ١٠

Artinya: Katakanlah (Nabi Muhammad), “Wahai hamba-hamba-Ku yang beriman, bertakwalah kepada Tuhanmu.” Orang-orang yang berbuat baik di dunia ini akan memperoleh kebaikan. Bumi Allah itu luas. Sesungguhnya hanya orang-orang yang bersabarlah yang disempurnakan pahalanya tanpa perhitungan.

e. مُتَقَلِّبُكُمْ (pergantian keadaan kalian)

Istilah ini menggambarkan dinamika dan perubahan kondisi kehidupan manusia, serta ujian yang menyertai amal perbuatan mereka di dunia. Secara ontologis, ini menyoroiti kekuasaan Allah yang Maha Mengatur

segala perubahan dan nasib manusia. Hal ini disebutkan dalam firman Allah pada surat Muhammad ayat 19:

فَاعْلَمْ أَنَّهُ لَا إِلَهَ إِلَّا اللَّهُ وَاسْتَغْفِرْ لِذَنْبِكَ وَلِلْمُؤْمِنِينَ وَالْمُؤْمِنَاتِ وَاللَّهُ يَعْلَمُ مُتَقَلَّبَكُمْ وَمَثْوَاكُمْ ١٩

Artinya: Ketahuilah (Nabi Muhammad) bahwa tidak ada Tuhan (yang patut disembah) selain Allah serta mohonlah ampunan atas dosamu dan (dosa) orang-orang mukmin laki-laki dan perempuan. Allah mengetahui tempat kegiatan dan tempat istirahatmu.

4. Perintah dan Larangan

Pada klaster tematik ini ditemukan kata-kata yang secara ontologis menyoroti struktur instruksi ilahi yang menjadi inti dalam Al-Qur'an. Tema ini mengulas tentang bagaimana perintah dan larangan Allah diberikan sebagai manifestasi kasih sayang dan tuntunan, bertujuan membimbing manusia menuju jalan yang benar dan menghindarkan mereka dari kesesatan. FastText, melalui kemampuannya dalam memahami konteks dan hubungan semantik antar kata, berhasil mengelompokkan istilah-istilah ini, mencerminkan pemahamannya terhadap kerangka panduan ilahi yang komprehensif.

a. واتقوا (dan bertakwalah)

Perintah ini merupakan inti ajaran Al-Qur'an, menekankan keharusan menjaga diri dari segala yang dilarang Allah, dan senantiasa menyadari kehadiran-Nya. Secara ontologis, ini adalah perintah utama yang membimbing manusia menuju kesadaran spiritual tertinggi. Hal ini tertuang dalam firman Allah pada surat Ali Imran ayat 102:

يَا أَيُّهَا الَّذِينَ آمَنُوا اتَّقُوا اللَّهَ حَقَّ تَقَاتِهِ وَلَا تَمُوتُنَّ إِلَّا وَأَنْتُمْ مُسْلِمُونَ ١٠٢

Artinya: Wahai orang-orang beriman, bertakwalah kepada Allah dengan sebenar-benarnya takwa kepada-Nya dan janganlah kamu mati kecuali dalam keadaan muslim.

b. ويحذركم (dan Dia memperingatkanmu)

Frasa ini menggambarkan bentuk peringatan ilahi, sebuah tanda kasih sayang Allah agar manusia senantiasa berhati-hati terhadap konsekuensi dari perbuatan yang melanggar batas-batas-Nya. Secara ontologis, peringatan ini berfungsi sebagai rambu-rambu yang melindungi manusia

dari bahaya spiritual dan duniawi. Istilah ini termuat dalam firman Allah pada surat Ali Imran ayat 28:

لَا يَتَّخِذِ الْمُؤْمِنُونَ الْكَافِرِينَ أَوْلِيَاءَ مِنْ دُونِ الْمُؤْمِنِينَ وَمَنْ يَفْعَلْ ذَلِكَ فَلَيْسَ مِنَ اللَّهِ فِي شَيْءٍ إِلَّا أَنْ تَتَّقُوا مِنْهُمْ تُقَاتُوا ۗ وَاللَّهُ نَفْسَهُ ۖ وَاللَّهُ الْمَصِيرُ ٢٨

Artinya: Janganlah orang-orang mukmin menjadikan orang kafir sebagai para wali dengan mengesampingkan orang-orang mukmin. Siapa yang melakukan itu, hal itu sama sekali bukan dari (ajaran) Allah, kecuali untuk menjaga diri dari sesuatu yang kamu takuti dari mereka. Allah memperingatkan kamu tentang diri-Nya (siksa-Nya). Hanya kepada Allah tempat kembali.

c. ويامرکم (dan Dia memerintahkanmu)

Kata ini menunjukkan instruksi langsung dan otoritatif dari Allah kepada hamba-Nya. Secara ontologis, ini menegaskan kekuasaan mutlak Allah sebagai pembuat syariat dan penentu kebaikan serta keburukan. Hal ini dapat ditemukan dalam firman Allah pada surat Al-Baqarah ayat 268:

الشَّيْطَانُ يَعِدُكُمُ الْفَقْرَ وَيَأْمُرُكُم بِالْفَحْشَاءِ ۗ وَاللَّهُ يَعِدُكُم مَّغْفِرَةً مِنْهُ وَفَضْلًا ۗ وَاللَّهُ وَاسِعٌ عَلِيمٌ ٢٦٨

Artinya: Setan menjanjikan (menakut-nakuti) kamu kemiskinan dan menyuruh kamu berbuat keji, sedangkan Allah menjanjikan kamu ampunan dan karunia-Nya. Allah Maha luas lagi Maha Mengetahui

d. لعلکم (agar kamu)

Frasa ini sering muncul dalam konteks pengaturan syariat dan hikmah di baliknya, menjelaskan tujuan atau manfaat dari suatu perintah atau larangan ilahi. Secara ontologis, ini menggarisbawahi bahwa setiap aturan Allah memiliki tujuan kebaikan bagi manusia. Frasa ini tertuang dalam firman Allah pada surat Al-Baqarah ayat 183:

يَا أَيُّهَا الَّذِينَ آمَنُوا كُتِبَ عَلَيْكُمُ الصِّيَامُ كَمَا كُتِبَ عَلَى الَّذِينَ مِنْ قَبْلِكُمْ لَعَلَّكُمْ تَتَّقُونَ ١٨٣

Artinya: Wahai orang-orang yang beriman, diwajibkan atas kamu berpuasa sebagaimana diwajibkan atas orang sebelum kamu agar kamu bertakwa.

e. ويلکم (celakalah kalian)

Ekspresi ini merupakan bentuk ancaman atau peringatan keras bagi mereka yang menyimpang dari perintah Allah atau melanggar larangan-Nya. Secara ontologis, ini menekankan keadilan ilahi dan konsekuensi

serius bagi pelanggaran. Hal ini disebutkan dalam firman Allah pada surat Al-Qashash ayat 80:

وَقَالَ الَّذِينَ أُوتُوا الْعِلْمَ وَيَنْتَظِرُ ثَوَابُ اللَّهِ خَيْرٌ لِّمَنْ آمَنَ وَعَمِلَ صَالِحًا وَلَا يُلْقِيهَا إِلَّا الصَّابِرُونَ ٨٠

Artinya: Orang-orang yang dianugerahi ilmu berkata, “Celakalah kamu! (Ketahuilah bahwa) pahala Allah lebih baik bagi orang-orang yang beriman dan beramal salah. (Pahala yang besar) itu hanya diperoleh orang-orang yang sabar.”

5. Keimanan dan Ketakwaan

Pada klaster tematik ini ditemukan kata-kata yang secara ontologis menggarisbawahi keimanan sebagai fondasi dasar dalam ajaran Al-Qur'an. Tema ini menyoroti pentingnya keyakinan yang teguh kepada Allah, serta hubungan erat antara iman, agama, dan identitas spiritual seorang Muslim, di mana ketakwaan memperkuat nilai iman melalui amal yang benar. Model FastText berhasil mengelompokkan kosakata yang esensial bagi konsep ini, menunjukkan pemahamannya terhadap inti dari keyakinan dan praktik keagamaan dalam korpus Al-Qur'an.

a. لايمان (karena iman)

Frasa ini menekankan bahwa iman adalah landasan fundamental untuk memperoleh rahmat, petunjuk, dan balasan kebaikan dari Allah. Secara ontologis, ini mengacu pada gagasan bahwa iman bukan sekadar keyakinan pasif, melainkan sebuah prasyarat aktif yang menggerakkan hamba menuju kebaikan dan keberkahan ilahi. Hal ini tertuang dalam firman Allah pada surat Al-Hujurat ayat 17:

يَمُنُونَ عَلَيْكَ أَنْ اسْلَمُوا ۖ قُلْ لَا تَمُنُوا عَلَيَّ إِسْلَامَكُمْ بَلِ اللَّهُ يَمُنُّ عَلَيْكُمْ أَنْ هَدَيْكُمْ لِلْإِيمَانِ إِنْ كُنْتُمْ صَادِقِينَ ١٧

Artinya: Mereka merasa berjasa kepadamu dengan keislaman mereka. Katakanlah, “Janganlah merasa berjasa kepadaku dengan keislamanmu. Dengan menunjukkan kamu kepada keimanan, jika kamu orang yang benar.”

b. بدينكم (dengan agama kalian)

Ekspresi ini menggambarkan agama sebagai identitas yang melekat dan penjaga nilai-nilai hidup bagi seorang Muslim. Secara ontologis, ini

menyoroti agama sebagai sistem kepercayaan dan praktik yang komprehensif, membentuk pandangan dunia dan perilaku individu. Istilah ini termuat dalam firman Allah pada surat Al-Hujurat ayat 16:

قُلْ أَتَعْلَمُونَ اللَّهَ بِدِينِكُمْ وَاللَّهُ يَعْلَمُ مَا فِي السَّمَوَاتِ وَمَا فِي الْأَرْضِ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ ١٦

Artinya: Katakanlah (kepada mereka), “Apakah kamu akan memberi tahu Allah tentang agamamu (keyakinanmu), padahal Allah mengetahui apa yang ada dilangit dan apa yang ada di bumi serta Allah Maha Mengetahui segala sesuatu.”

c. مومن (orang beriman)

Gelar ini merujuk pada individu yang percaya dan taat sepenuhnya kepada Allah, Rasul-Nya, dan ajaran-ajaran Islam. Secara ontologis, gelar mukmin adalah status mulia yang diberikan kepada mereka yang memiliki keyakinan kokoh, yang tercermin dalam perbuatan dan ketakwaan mereka. Hal ini disebutkan dalam firman Allah pada surat Al-Anfal ayat 2:

إِنَّمَا الْمُؤْمِنُونَ الَّذِينَ إِذَا ذُكِرَ اللَّهُ وَجِلَتْ قُلُوبُهُمْ وَإِذَا تُلِيَتْ عَلَيْهِمْ آيَاتُهُ زَادَتْهُمْ إِيمَانًا وَعَلَىٰ رَبِّهِمْ يَتَوَكَّلُونَ ٢

Artinya: Sesungguhnya orang-orang mukmin adalah mereka yang jika disebut nama Allah, gemetar hatinya dan jika dibacakan ayat-ayat-Nya kepada mereka, bertambah (kuat) imannya dan hanya kepada Tuhannya mereka bertawakal.

6. Perlindungan & Pertolongan Allah

Pada klaster tematik ini ditemukan kata-kata yang secara ontologis mengandung janji pertolongan dan perlindungan dari Allah terhadap hamba-Nya yang taat dan bertawakal. Tema ini menggambarkan bahwa Allah akan memberikan kemenangan, dukungan, dan penyelamatan dari mara bahaya kepada orang-orang yang menyerahkan diri sepenuhnya kepada-Nya. FastText, dengan kemampuannya memahami konteks dan hubungan semantik antar kata, berhasil mengelompokkan istilah-istilah ini, mencerminkan pemahamannya terhadap konsep dukungan ilahi dalam korpus Al-Qur'an.

a. ويمدكم (dan Dia akan membantu kalian)

Frasa ini merupakan janji dukungan ilahi, baik secara spiritual maupun material, yang diberikan kepada mereka yang berjuang di jalan-Nya. Secara ontologis, ini mengacu pada gagasan tentang bantuan dan penguatan yang tak terbatas dari Allah kepada hamba-Nya yang membutuhkan. Hal ini tertuang dalam firman Allah pada surat Al-Anfal ayat 9:

إِذْ تَسْتَغِيثُونَ رَبَّكُمْ فَاسْتَجَابَ لَكُمْ أَنِّي مُمِدُّكُمْ بِالْفِ مِّنَ الْمَلَائِكَةِ مُرْدِفِينَ ٩

Artinya: (Ingatlah) ketika kamu memohon pertolongan kepada Tuhanmu, lalu Dia mengabulkan (-nya) bagimu (seraya berfirman), “Sesungguhnya Aku akan mendatangkan bala bantuan kepadamu berupa seribu malaikat yang datang berturut-turut.”

b. ينصركم (Dia akan menolongmu)

Kata ini secara spesifik merujuk pada pertolongan dan kemenangan yang diberikan Allah bagi mereka yang membela agama-Nya dan menegakkan kebenaran. Secara ontologis, ini menekankan bahwa pertolongan sejati hanya datang dari Allah dan diberikan kepada mereka yang berjuang untuk tujuan-Nya. Istilah ini termuat dalam firman Allah pada surat Muhammad ayat 7:

يَا أَيُّهَا الَّذِينَ آمَنُوا إِن تَنْصُرُوا اللَّهَ يَنْصُرْكُمْ وَيُثَبِّتْ أَقْدَامَكُمْ ٧

Artinya: Wahai orang-orang yang beriman, jika kamu menolong (agama) Allah, niscaya Dia akan menolongmu dan meneguhkan kedudukan.

c. وليمسنكم (dan jika Dia menyentuh kalian)

Ekspresi ini menggambarkan kekuasaan mutlak Allah dalam memberikan ujian berupa kesulitan atau kesusahan, dan sekaligus kemampuan-Nya untuk mengangkat atau memberikan kesembuhan. Secara ontologis, ini menyoroti bahwa setiap musibah atau kebaikan berasal dari Allah, dan hanya Dia yang memiliki kuasa untuk mengubah keadaan. Hal ini disebutkan dalam firman Allah pada surat Al-An‘am ayat 17:

وَأَن يَّمْسَسَكَ اللَّهُ بِضُرٍّ فَلَا كَاشِفَ لَهُ إِلَّا هُوَ وَإِن يَّمْسَسَكَ بِخَيْرٍ فَهُوَ عَلَى كُلِّ شَيْءٍ قَدِيرٌ ١٧

Artinya: Jika Allah menimpakan kemudaran kepadamu, tidak ada yang dapat menghilangkannya selain Dia; dan jika Dia memberikan kebaikan kepadamu, Dia Maha kuasa atas segala sesuatu.

7. Petunjuk & Hidayah

Pada klaster tematik ini ditemukan kata-kata yang secara ontologis mengulas tentang Petunjuk (Hidayah) sebagai karunia terbesar dari Allah. Tema ini menggambarkan bagaimana hidayah membimbing manusia dari kegelapan menuju cahaya kebenaran, serta menunjukkan bahwa Allah menganugerahkan penglihatan batin, kemampuan menyucikan jiwa, dan ilmu sebagai sarana fundamental untuk mencapai pemahaman hakiki, kebenaran, dan kebahagiaan abadi. FastText berhasil mengelompokkan kosakata yang esensial bagi konsep hidayah, mencerminkan pemahamannya terhadap kedekatan semantik antara pencerahan spiritual, pemurnian diri, dan proses pembimbingan ilahi dalam korpus Al-Qur'an.

a. بصيرة (penglihatan batin)

Kata ini merujuk pada kemampuan spiritual dan intelektual untuk memahami kebenaran secara mendalam, melampaui sekadar penglihatan fisik. Secara ontologis, bashirah adalah anugerah ilahi yang memungkinkan seseorang untuk melihat tanda-tanda kebesaran Allah dan hikmah di balik ciptaan-Nya. Hal ini tertuang dalam firman Allah pada surat Yusuf ayat 108:

قُلْ هَذِهِ سَبِيلِي أَدْعُو إِلَى اللَّهِ عَلَىٰ بَصِيرَةٍ أَنَا وَمَنِ اتَّبَعَنِي يُغْنِبَنِي اللَّهُ وَمَا أَنَا مِنَ الْمُشْرِكِينَ
١٠٨

Artinya: Katakanlah (Nabi Muhammad), “inilah jalanku, aku dan orang-orang yang mengikutiku mengajak (seluruh manusia) kepada Allah dengan bukti yang nyata. Mahasuci Allah dan aku tidak termasuk golongan orang-orang musyrik.”

b. ويزكيكم (dan menyucikan kalian)

Frasa ini menggambarkan proses penyucian jiwa, yaitu pembersihan diri dari dosa dan akhlak tercela, sebagai bagian integral dari pembinaan iman. Secara ontologis, penyucian ini merupakan bentuk rahmat dan bimbingan

ilahi yang membawa manusia menuju kesempurnaan spiritual dan moral. Istilah ini termuat dalam firman Allah pada surat Al-Baqarah ayat 151:

كَمَا أَرْسَلْنَا فِيكُمْ رَسُولًا مِّنْكُمْ يَتْلُوا عَلَيْكُمْ آيَاتِنَا وَيُزَكِّيكُمْ وَيُعَلِّمُكُمُ الْكِتَابَ وَالْحِكْمَةَ وَيُعَلِّمُكُم مَّا لَمْ تَكُونُوا تَعْلَمُونَ ۝ ١٥١

Artinya: Sebagaimana (Kami telah menyempurnakan nikmat kepadamu), Kami pun mengutus kepadamu seorang Rasul (Nabi Muhammad) dari (kalangan) kamu yang membacakan kepadamu ayat-ayat Kami, menyucikan kamu, dan mengajarkan kepadamu Kitab (Al-Qur'an) dan hikmah (sunah), serta mengajarkan apa yang belum kamu ketahui.



BAB V

PENUTUP

5.1 Kesimpulan

Penelitian ini menganalisis pengaruh hyperparameter dalam FastText terhadap *semantic Similarity* kata menggunakan dataset Al-Qur'an Bahasa Arab. Berdasarkan hasil eksperimen dan analisis, dapat ditarik beberapa kesimpulan:

1. Konfigurasi hyperparameter dalam FastText terbukti memiliki pengaruh signifikan terhadap kualitas representasi kata. Kombinasi dimensi vektor 300, epoch 10, dan window size 5 menghasilkan performa terbaik dengan nilai *cosine Similarity* mencapai 0.999. Ini menunjukkan bahwa pemilihan dimensi yang cukup besar dan jumlah epoch yang tepat penting untuk menangkap makna semantik dalam teks Al-Qur'an.
2. FastText menunjukkan keunggulan signifikan dibandingkan Word2Vec dalam menangani kompleksitas morfologi bahasa Arab, seperti yang ditemukan dalam teks Al-Qur'an. Model ini mengatasi tantangan bahasa Arab yang flektif dan kaya derivasi kata dengan mempertimbangkan struktur internal kata melalui penggunaan sub-kata (character n-gram). Pendekatan ini memungkinkan FastText untuk memahami pola morfologi (prefiks, sufiks, akar kata) dan menghasilkan representasi untuk kata-kata yang belum pernah muncul dalam data pelatihan (out-of-vocabulary). Bukti kemampuan ini terlihat dari hasil analisis *cosine Similarity* kata "خير" (kebaikan), di mana model berhasil mengelompokkan kata-kata yang relevan secara semantik ke dalam kluster tema yang koheren seperti nilai moral, perbuatan dan amal, keimanan dan ketakwaan, perintah dan larangan, perlindungan dan pertolongan Allah, serta petunjuk dan hidayah. Visualisasi PCA juga secara kuat memvalidasi kemampuan FastText dalam mengidentifikasi kemiripan semantik dan mengelompokkan kata-kata berdasarkan makna, bahkan dengan adanya variasi leksikal yang kompleks pada bahasa Arab.

5.2 Saran

Untuk pengembangan penelitian ini di masa mendatang, beberapa aspek krusial patut menjadi perhatian. Pertama, disarankan untuk melaksanakan eksplorasi hyperparameter FastText secara lebih sistematis guna mengidentifikasi konfigurasi optimal yang mungkin belum tercakup. Pendekatan optimasi hyperparameter otomatis dapat digunakan untuk efisiensi. Kedua, perbandingan komprehensif dengan model *word embedding* kontemporer, termasuk arsitektur berbasis Transformer, sangat dianjurkan untuk mengevaluasi efektivitas relatif FastText dalam pemrosesan bahasa Arab Al-Qur'an. Ketiga, pengembangan atau adopsi dataset evaluasi *gold standard* yang tervalidasi secara linguistik akan meningkatkan objektivitas dan presisi pengukuran *semantic Similarity* model. Terakhir, implementasi model yang telah dioptimalkan pada tugas-tugas *Natural Language Processing* (NLP) hilir, seperti sistem pencarian semantik atau klasifikasi teks Al-Qur'an, akan memperjelas nilai guna dan relevansi aplikatif dari temuan penelitian ini.



DAFTAR PUSTAKA

- [1] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” Oct. 2013, [Online]. Available: <http://arxiv.org/abs/1310.4546>
- [2] Y. Goldberg and O. Levy, “word2vec Explained: deriving Mikolov et al.’s negative-sampling word-embedding method,” Feb. 2014, [Online]. Available: <http://arxiv.org/abs/1402.3722>
- [3] K. Darwish *et al.*, “A Panoramic Survey of Natural Language Processing in the Arab World,” Nov. 2020, [Online]. Available: <http://arxiv.org/abs/2011.12631>
- [4] W. Aransa, “Statistical Machine Translation of the Arabic Language.” [Online]. Available: <https://theses.hal.science/tel-01316544v1>
- [5] K. Shaalan, “Nizar Y. Habash, Introduction to Arabic natural language processing (Synthesis lectures on human language technologies),” *Machine Translation*, vol. 24, no. 3–4, pp. 285–289, Dec. 2010, doi: 10.1007/s10590-011-9087-8.
- [6] T. Adewumi, F. Liwicki, and M. Liwicki, “Word2Vec: Optimal hyperparameters and their impact on natural language processing downstream tasks,” *Open Computer Science*, vol. 12, no. 1, pp. 134–141, Jan. 2022, doi: 10.1515/comp-2022-0236.
- [7] D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural language processing: state of the art, current trends and challenges,” *Multimed Tools Appl*, vol. 82, no. 3, pp. 3713–3744, Jan. 2023, doi: 10.1007/s11042-022-13428-4.
- [8] R. Mihalcea, H. Liu, and H. Lieberman, “LNCS 3878 - NLP (Natural Language Processing) for NLP (Natural Language Programming).”
- [9] R. Muñoz, A. Montoyo, and E. Métais, Eds., *Natural Language Processing and Information Systems*, vol. 6716. in *Lecture Notes in Computer Science*, vol. 6716. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-22327-3.
- [10] Nizar. Habash and Graeme. Hirst, *Arabic Natural Language Processing*. Morgan & Claypool Publishers American International Distribution Corporation [distributor], 2010.
- [11] N. Habash, “Arabic Natural Language Processing,” in *EMNLP 2022 - 2022 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, Association for Computational Linguistics (ACL), 2022, pp. 9–10. doi: 10.1145/1644879.1644881.
- [12] N. Habash and O. Rambow, “Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop,” 2005.

- [13] K. Shaalan, "A Survey of Arabic Named Entity Recognition and Classification," 2014, doi: 10.1162/COLI.
- [14] A. Farghaly and K. Shaalan, "Arabic Natural Language Processing," in *EMNLP 2022 - 2022 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, Association for Computational Linguistics (ACL), 2022, pp. 9–10. doi: 10.1145/1644879.1644881.
- [15] Y. Belinkov and J. Glass, "Arabic Diacritization with Recurrent Neural Networks," Association for Computational Linguistics, 2015. [Online]. Available: <http://www.qamus.org/transliteration.htm>.
- [16] N. Zalmout and N. Habash, "Adversarial Multitask Learning for Joint Multi-Feature and Multi-Dialect Morphological Modeling," Association for Computational Linguistics.
- [17] H. Bouamor, N. Habash, and K. Oflazer, "A Multidialectal Parallel Corpus of Arabic."
- [18] N. Chenfour and S. Abdelmoumni, "MORPHOSCRIPT DATA MODEL AND ARABIC MORPHOLOGICAL AUTOMATA," *Journal of Southwest Jiaotong University*, vol. 56, no. 6, pp. 131–145, Dec. 2021, doi: 10.35741/issn.0258-2724.56.6.11.
- [19] S. Srinivasan, Ed., *Guide to Big Data Applications*, vol. 26. in *Studies in Big Data*, vol. 26. Cham: Springer International Publishing, 2018. doi: 10.1007/978-3-319-53817-4.
- [20] K. A. Hambarde and H. Proenca, "Information Retrieval: Recent Advances and Beyond," Jan. 2023, doi: 10.1109/ACCESS.2023.3295776.
- [21] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification," Jul. 2016, [Online]. Available: <http://arxiv.org/abs/1607.01759>
- [22] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information", doi: 10.1162/tac1_a_00051/1567442/tac1_a_00051.pdf.
- [23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," Jan. 2013, [Online]. Available: <http://arxiv.org/abs/1301.3781>
- [24] R. Al-Rfou, B. Perozzi, and S. Skiena, "Polyglot: Distributed Word Representations for Multilingual NLP," Jul. 2013, [Online]. Available: <http://arxiv.org/abs/1307.1662>
- [25] W. Ling, C. Dyer, A. Black, and I. Trancoso, "Two/Too Simple Adaptations of Word2Vec for Syntax Problems." [Online]. Available: <https://github.com/wlin12/wang2vec>.
- [26] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning Word Vectors for 157 Languages." [Online]. Available: <https://fasttext.cc/>

- [27] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, "FastText.zip: Compressing text classification models," Dec. 2016, [Online]. Available: <http://arxiv.org/abs/1612.03651>
- [28] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," Sep. 2014, [Online]. Available: <http://arxiv.org/abs/1409.3215>
- [29] X. Rong, "word2vec Parameter Learning Explained," Nov. 2014, [Online]. Available: <http://arxiv.org/abs/1411.2738>
- [30] D. A. Elekes, A. Englhardt, M. Schäler, and K. Böhm, "Resources to Examine the Quality of Word Embedding Models Trained on n-Gram Data," 2018. [Online]. Available: <http://dbis.ipd.kit.edu/2568.php>
- [31] T. Pedersen, S. Patwardhan, and J. Michelizzi, "WordNet::Similarity-Measuring the Relatedness of Concepts." [Online]. Available: <http://search.cpan.org/dist/WordNet-Similarityhttp://wn-similarity.sourceforge.net>
- [32] M. Wibowo, C. Quix, N. S. Hussien, H. Yuliansyah, and F. D. Adhinata, "Similarity Identification of Large-scale Biomedical Documents using Cosine Similarity and Parallel Computing," *Knowledge Engineering and Data Science*, vol. 4, no. 2, p. 105, Feb. 2022, doi: 10.17977/um018v4i22021p105-116.
- [33] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning Word Vectors for 157 Languages." [Online]. Available: <https://fasttext.cc/>
- [34] "Natural Language Processing with Python."

