Terbit pada laman web jurnal : **http://ojsamik.amikmitragama.ac.id**

JURNAL JARINGAN SISTEM INFORMASI ROBOTIK (JSR)
Vol. 6 No. 2 TAHUN 2022 E - ISSN : 2579-373X

MEMBANGUN *SLANG DICTIONARY* UNTUK NORMALISASI TEKS MENGGUNAKAN *PRE-TRAINED FASTTEXT MODEL*

Lavenia Situmorang[1], Enjelin Hutahaean[2], Ruth Angeli Sibarani[3]
Junita Amalia[*4]
[1,2,3,4]Sistem Informasi – Fakultas Teknik Elektro dan Informatika, Institut Teknologi Del
lavesitumorang4@gmail.com[1], Enjelinhutahaean250@gmail.com[2], angelisibarani@gmail.com[3]
junita.amalia@del.ac.id[4]

Abstract

Slang word is a complex word, difficult and cannot be ignored. Slang is used by certain circles and is limited so that not everyone knows the meaning of the conversations carried out by group members. Based on previous research that has been done, namely making slang using a manual process that requires quite a lot of time to collect slang words, so that our research aims to collect slang words by applying Deep Learning, namely Natural Language Processing using the word embedding FastText method to speed up the collection process. slang words. The author implements the techniques and algorithms that have been designed in the previous stage. This stage will ensure that the processes carried out in the research can be carried out in accordance with the theories that support the research. From the combined data between YouTube comments and the Indonesian dictionary, itwas found that 421 words are slang words. These slang words are obtained by means of the process of lookingfor word similarities (similarity words) between YouTube comments and Indonesian dictionaries. In building a slang dictionary from the youtube comment dataset with a pre-trained FastText model, a preprocessing process and normalization is carried out. After the normalization process was carried out to get normal words from each slang candidate, the results of the slang dictionary were 278 rows consisting of four columns, namely the lexical column, threshold, slang candidate, and normal words using a threshold of 0.05, 0.1 and 0.2
Keywords: Slang, Pre-trained FastText, NLP, Similarity Word

Abstrak

Kata Slang merupakan kata yang kompleks, sulit dan tidak dapat diabaikan. Slang digunakan oleh kalangan tertentu dan terbatas sehingga tidak semua orang mengetahui maksud dari percakapan yang dilakukan oleh anggota kelompok. Berdasarkan penelitian terdahulu yang telah dilakukan yaitu pembuatan slang menggunakan proses manual yang memerlukan cukup banyak waktu untuk mengumpulkan kata slang, sehingga penelitian yang kami lakukan bertujuan untuk mengumpulkan kata slang dengan menerapkan Deep Learning yaitu Natural Language Processing dengan menggunakan metode word embedding FastText untuk mempercepat proses pengumpulan kata slang. Penulis melakukan implementasi teknik dan algoritma yang telah dirancang pada tahapan sebelumnya. Tahapan ini memastikan bahwa proses yang dilakukan dalam penelitian dapat dilaksanakansesuai dengan teori-teori yang mendukung penelitian. Dari gabungan data antara kata komentar youtube dan kamus Bahasa Indonesia, didapatkan 421 kata yang merupakan kata slang. Kata slang ini didapatkan dengan caraproses mencari kesamaan kata (similarity word) antara kata komentar youtube dan kamus Bahasa Indonesia. Dalam membangun kamus slang dari dataset komentar youtube dengan pre-trained FastText model dilakukan proses preprocessing dan normalisasi. Setelah dilakukan proses normalisasi untuk mendapatkan kata normal dari setiap kandidat slang maka didapatkan hasil kamus slang sebanyak 278 baris yang terdiri dari empat kolom yaitu kolom ekstaksislang, threshold, kandidat slang, dan kata normal dengan menggunakan threshold 0.05, 0.1 dan 0.2.
Kata kunci: Slang, Pre-trained FastText, NLP, Similarity Word

1. Pendahuluan

Kata slang merupakan kata yang kompleks, sulit dan tidak dapat diabaikan. Slang digunakan oleh kalangan tertentu dan terbatas sehingga tidak semua orang mengetahui maksud dari percakapan yang dilakukan oleh anggota kelompok. Slang juga disebut sebagai bahasa prokem karena kosakata pada slang selalu berubah-ubah dan temporal [1]. Menurut Kamus Besar Bahasa Indonesia, slang adalah ragam bahasa tidak resmi dan tidak baku yang sifatnya musiman, jenis kosakata dari kata slang bersifat informal yang biasanya digunakan berkomunikasi antar orang yang sudah mengenal dengan baik [2]. Kata slang banyak digunakan dalam penyampaian pendapat pada media sosial, hal ini dikarenakan banyaknya pendapat yang dituliskan dengan cara disingkat atau salah ketiksehingga dalam melakukan analisis sentimen, klasifikasi teks maupun peringkasan teks masih ditemukan kesulitan dikarenakan perlunyadilakukan representasi ulang terhadap kata agar diketahui bentuk normal dari kata slang tersebut.

Penelitian "Perbandingan Kinerja *Word Embedding Word2ve*, *Glove* dan *FastText* pada Klasifikasi Teks" yang dilakukan oleh Arliya Nurdin, dkk yaitu tantangan dalam melakukan ekstraksi teks adalah karakteristik teks yang tidak terstruktur. Ketiga metode tersebut dipilih dalam melakukan penelitian tersebut dikarenakan metode ini dapat memahami makna semantik, sintaktik, dan urutan bahkan konteks di sekitar kata. Penelitian tersebut menggunakan dataset *newsgroup* dari sekitar 18.846 artikel, 134.142 kosakata, dengan 20 topik yang dibagi menjadi 11.314 data latih dan 7.532 data uji dan *Reuters Newswire* terdiri dari 11.228 berita yang diperoleh dari kantor berita Reuters, 30.979 kosakata, dan berlabel 46 topik.Data latih dari dataset ini sebanyak 8.982 data dan data uji 2.246 data. Setelah dilakukannya perbandingan antara ketiga metode tersebut maka didapatkan hasil bahwa *FastText* lebih baik dari duametode lainnya dengan nilai F-*Measure* 0,979 untuk dataset 20 *newsgroup* dan 0,715 untuk dataset *Reuters newswire*. *Word2Vec* dan *GloVe* tidak mampu merepresentasikan vektor dari katayang tidak ada dalam korpus (*out of vocabulary*), berbeda dengan *FastText* yang dapat diandalkan untuk permasalahan *out of vocabulary* ini. Oleh sebab itu kinerja terbaik dari eksperimen pada penelitian tersebut diperoleh dengan menggunakan *word embedding FastText* [3]
.

Berdasarkan pemaparan tersebut peneliti tertarik untuk membuat *slang dictionary* untuk normalisasi teks dengan *pre-trained FastText model* karena diketahui *FastText* mampu mengatasi permasalahan *out of vocabulary*. Untukmendapatkan bentuk normal dari kata *slang* yang dikumpulkan dilakukan normalisasi kata, sehingga *slang dictionary* yang dihasilkan terdiri dari kata *slang* dan bentuk normalnya. Pada penelitian ini, peneliti berkontribusi dalam memperoleh dataset komentar youtube berbahasa Indonesia dengan teknik *scraping*. Lalu, hasil dari *scraping* dilanjutkan ke tahap *preprocessing* yaitu *casefolding*, *tokenization* dan *remove duplicate*. Kemudian melakukan proses *pre-trained FastText model* pada data komentar youtube berbahasa Indonesia dan data KBBI, agar setiap kata tersebut memiliki vektor kata. Dengan demikian maka dilakukan penelitian ini, yang berjudul "Membangun *Slang Dictionary* Untuk Normalisasi Teks menggunakan *Pre-trained FastText Model*" yang diharapkan bisa mengumpulkan kata-kata slang.

2. Tinjauan Pustaka

Pada penelitian yang dilakukan oleh Liang Wu, Fred Morstatter, Huan Liu yaitu "*SlangSD: Building and Using a Sentiment Dictionary of Slang Words for Short-Text Sentiment Classification*", dalam melakukan sentimen analisismasih sulit dilakukan karena masih banyak bentuk kata yang tidak formal dan sulit dipahami sehingga penelitian yang dilakukan oleh Liang Wu, Fred Morstatter, Huan Liu bertujuan untuk membuat sebuah kamus sentimen pertama dari kata-kata *slang*. Dalam membangun kamus sentimen dari kata *slang* bertujuan untuk memperkenalkan cara otomatis melabeli polaritas sentimen dari kata-kata *slang* dalam skala besar, dan menunjukkan kegunaan *Slang* SD dalam sistem klasifikasi sentimen yang ada dengan *real-world datasets* dari teks informal [4].

Penelitian lain yang dilakukan oleh JasonTurner dan Mehmed Kantardzic yaitu "*Twitter Query Expansion via Word2Vec Urban Dictionary Model*", diusulkan sebuah sistem baru di mana akandibangun model *Word2Vec* pada kumpulan *tweet* yang ada, lalu model tersebut digunakan untuk menemukan istilah terkait melalui kesamaan *cosinus* yang diketahui untuk istilah terkait ganja, dan akhirnya memvalidasi kata-kata ini sebagai

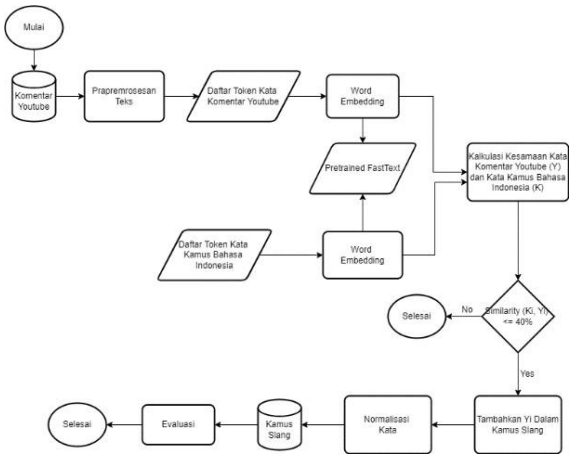
terkait ganja melalui kamus *slang* online populer *Urban Dictionary*. Kemudian akan diperiksa pengaruh dari ekspansi permintaan pencarian dengan melihat jumlah tweet, frekuensi kata, dan properti jejaring sosial [5].

3. Metodologi Penelitian

Penjelasan tahapan perancangan sistem berdasarkangambar diatas adalah sebagai berikut:

1. Pengambilan dataset dari komentar 10 akun subscriber terbanyak di Indonesia per tahun 2022, yaitu pada akun Ricis Official, Atta Halilintar, Jess No Limit, Rans Entertainment, TRANS7 OFFICIAL, Frost Diamond, Baim Paula, Indosiar, Gen Halilintar dan Deddy Corbuzier untuk dijadikan dataset yang diambil dengan cara scraping. Setelah dengan teknik *scraping* dataset dari komentar Youtubemenjadi dataset yang di *pre-processing*.
2. Pra-pemrosesan data yang dilakukan yaitu *case folding* dengan tujuan mengubah huruf kapital menjadi huruf kecil, *tokenization* pemisahan kalimat menjadi bentuk token dan *remove duplicate* dengan tujuan untuk menghapus data yang redundan.
3. Kemudian data yang telah diproses akan diterapkan ke dalam bentuk *word embedding* menggunakan *pre-trained FastText* model yang bertujuan untuk mengubah data ke dalam bentuk *vector*, kemudian didapatkan kata yang menjadi kandidat slang dengan menerapkan*similarity* <=40%.
4. Jika terdapat *similarity* <=40% maka dimasukkan ke dalam kamus slang.
5. Setelah mendapatkan kandidat slang, maka dilakukan tahap normalisasi untukmendapatkan kata normal untuk setiap kandidat slang.

Tahap akhir adalah evaluasi terhadap sistem apakah sesuai dengan yang diharapkan. Hasil evaluasi dengan menghitung nilai akurasi antara *slang dictionary* yang dibangun dengan kamus alay. Tahapan penelitian yang dilakukan ditunjukkan pada Gambar 1 berikut.



Gambar 1. Tahapan penelitian

3.1 Dataset

Dataset yang digunakan pada penelitian ini adalah dataset komentar youtube. Sumber pengumpulan data yang digunakan pada penelitian ini adalah internet. Teknik yang digunakan untuk pengumpulan data melalui internet adalah *Web scraping. Web scraping* adalah salah satu cara yang digunakan untuk memperoleh data dari sebuah situs web secara otomatis [6]. Setelah dilakukan proses *scrapping*, maka data terkumpul sebanyak 47.802 baris yang terdiri dari dua kolom yaitu kolom *name* dan *comment.*

3. 2 Text Preprocessing

Tahapan *preprocessing* data yang diterapkan untuk penelitian merupakan langkah awal untuk mengimplementasikan algoritma yang telah dipilih dalam penelitian. Tahapan *preprocessing* merupakan tahapan penting untuk membersihkan data yang digunakan dalam proses pembelajaran [7]. Pada penelitian ini, tahapan preprocessing yang dilakukan adalah sebagai berikut:

1. Tahap *case folding*, digunakan untuk melakukan penyaringan kata dari isi dokumen, yaitu mengubah semua huruf kapital menjadi huruf kecil.
2. Tahap *tokenizing*, dilakukan proses pemotongan pada sebuah kalimat menjadi kata atau biasa disebut token.

Tahap *remove duplicate*, dilakukan penghapusan data komentar *youtube* yang ganda atau lebih dari satu karena apabila sebuah kata muncul terlalu banyak tetapi memiliki kata dan penulisan yang sama persis akan menyebabkan bobot nilai suatu kata menjadi tinggi dan tidak akan mendapatkandata yang unik yang mengakibatkan data redundan.

3. 3 Word Embedding menggunakan FastText

Pada penelitian ini digunakan *Pre-Trained FastText* model yang tersedia dalam korpus yang

siap pakai. *Pre-Trained FastText* digunakan untuk mengkonversi kata atau kalimat ke dalam bentuk vektor agar dapat diproses oleh algoritma *deep learning*. *Pre-Trained FastText* ini menggunakan model CBOW untuk menghitung representasi kata. Model CBOW menggunakan konteks untuk memprediksi target kata. CBOW memiliki waktu training lebih cepat dan memiliki akurasi yang sedikit lebih baik untuk frequent words [3]. *Pre- Trained FastText Model* menggunakan *embedding dimension* berukuran 300, artinya hidden layer yang digunakan berjumlah 300. Adapun *hyperparameter* yang digunakan pada implementasi *embedding* dengan *Pre-Trained FastText* adalah *character n- grams of length* 5, *window of size 5* dan 10 *negatives*.

Berikut penjelasan untuk setiap hyperparameter yang digunakan:

1. Vector Size adalah ukuran dimensi embedding untuk membangun model dalam proses vektorisasi data komentar youtube. Penggunaan ukuran vector size disesuaikan dengan jumlah data yang digunakan. Semakin besar ukuran vector size maka semakin baik vektor yang dihasilkan, namun waktu komputasi yang dibutuhkan semakin tinggi juga. Contohnya, jika digunakan ukuran *vector size* dengan nilai 300, maka hasil representasi kata disimpan ke dalam sebuah variabel hasil vektorisasi dengan ukuran 300 x 300. Sehinggapada penggunaan ukuran *vector size* 300, satu kata/term di dalam data komentar youtube dipetakan menjadi 300 vektor yang mewakilinya.
2. *Character n-grams of length*, semua kombinasi kata atau huruf yang berdekatan kepada setiap kata yang direpresentasikan sebagai sekumpulan karakter *n-grams*.
3. *Window* adalah *hyperparameter* untuk mengatur jumlah tetangga terdekat yang digunakan untuk melakukan *training* model. Sehingga apabila kata target berada di tengah kalimat maka, *window* berperan mengatur kata konteks dan kata target pada data komentar youtube sebanyak panjang *window* yang sudah didefinisikan.
4. *Negatives* adalah jika parameter bernilai lebih dari 0 maka negative sampling digunakan.

4. Hasil dan Pembahasan

4. 1 Hasil Text Preprocessing

Text preprocessing yang dilakukan untuk memperoleh hasil seperti yang dapat dilihat pada contoh Tabel 1.

Tabel 1. Hasil *Text Preprocessing*

Indeks	Comment
0	[good, luck, bang, atta, aku, fans, beratnya, bang, atta]
1	[ikut, terharu, peduli, bapak2, yg, bekerja, keras, ajak, shopping, makan, berbagi, uang, semoga, barokah, amiin, anak, yg, baik, sekali]
2	[papata, selalu, memberikan, vlog, yang, menarik, dan, dapat, menjadi, contoh, lainnya]
….	….
44036	[alhamdulillah, bang, atta, baik, org, nya, sama, boss, padang, merdeka]
44037	[salut, gue, sama, atta, karna, kebaikannya, sehat, selalu, ya, atta, semoga, selalu, di, lindungi, oleh, tuhan]
44038	[subahanallah, bang, atta, halilintar, suksek, terus]

Berdasarkan hasil tersebut, data komentar youtube yang telah melalui tahapan *text preprocessing* telah bersih dan berisi teks komentar youtube untuk membangun *slang dictionary*.

4. 2 Hasil Kamus Slang

Hasil kamus slang untuk membangun *slang dictionary* untuk normalisasi teks menggunakan *pre-trained fasttext model* dijelaskan sebagai berikut..

Kata *slang* yang didapatkan dengan *threshold value* 0.2, 0.3 dan 0.4 adalah sebanyak 1771 kata. Dikarenakan pada penelitian yang dilakukan hanya membangun *slang dictionary* Bahasa Indonesia sehingga kata *slang* yang bahasa inggris dilakukan penghapusan manual dan dilakukan penghapusan kata *slang* yang *duplicate* dengan menggunakan remove duplicate agar kata *slang* yang didapatkan bersifat unik. Setelah dilakukannya tahap tersebut, maka didapatkan kumpulan kata *slang* sebanyak 421 kata.

Tabel 2. Hasil Kata Slang

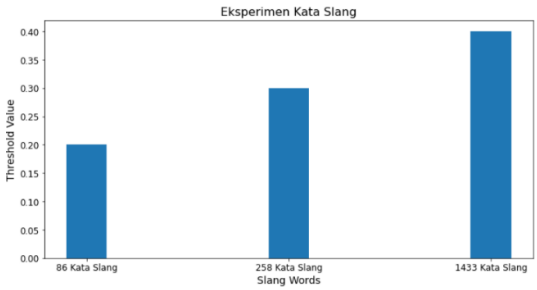
Indeks	*Slang*
0	abi
1	acnya
2	adlah
3	agaama
4	akirnya
…	…
416	yrb
417	yukmba
418	yutub
419	yyg
420	zemua

Kemudian dilakukan proses normalisasi untuk mendapatkan kata normal dari setiap kandidat *slang*. Hasil kamus *slang* yang terdiri dari empat kolom yaitu kolom ekstraksi *slang*, *threshold*, kandidat *slang*, dan kata normal. Dimana *threshold* yang digunakan yaitu 0.05, 0.1 dan 0.2 dan memiliki data sebanyak 278 baris.

4. 3 Pembahasan

Dalam membangun kamus *slang* (*slang dictionary*) digunakan data sebanyak 40000 data dari total komentar youtube dan 10000 data dari total kamus bahasa indonesia. Dari hasil evaluasi model yang dilakukan pada setiap eksperimen, maka didapatkan kumpulan kata *slang* yang disimpan sebagai kamus *slang* dengan menggunakan *threshold value* 0.2, 0.3 dan 0.4.

Dalam membangun kamus *slang* dilakukan tiga eksperimen dimana tiga eksperimen tersebut merupakan pembagian dari semua data yang digunakan berdasarkan *threshold* value 0.2, 0.3, dan
0.4. Oleh karena itu, pada gambar hasil model *embedding FastText* merupakan bar chart dari total keseluruhan kata *slang* yang didapatkan berdasarkan *threshold* value 0.2, 0.3, dan 0.4.



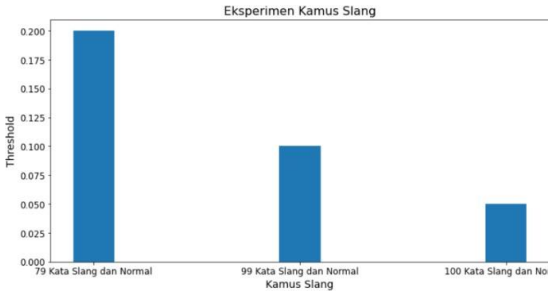
Gambar 2. *Bar chart* berdasarkan Eksperimen Kata Slang

Dari visualisasi di atas, dapat disimpulkan bahwa dapat dilihat bahwa semakin besar *threshold value* yang digunakan maka semakin banyak kata *slang* yang didapatkan.

Setelah dilakukan eksperimen I-III maka kata *slang* yang didapatkan sebanyak 1.777 kata. Lalu ketika dilakukan pemeriksaan terhadap *slang* yang telah dihasilkan, maka *slang* hasil dari *threshold* yang lebih kecil akan dihasilkan juga pada *threshold* yang lebih besar, sehingga dilakukan penghapusan kata *slang* yang *duplicate* dengan menggunakan *remove duplicate* agar kata *slang* yang didapatkan bersifat unik. Lalu dikarenakan penelitian yang dilakukan hanya membangun *slang dictionary* Bahasa Indonesia sehingga kata *slang* yang bahasa inggris dilakukan penghapusan manual. Setelah dilakukannya tahap tersebut, maka didapatkan kumpulan kata *slang* sebanyak 421 kata.

Kemudian kata *slang* yang sudah dikumpulkan akan dilakukan proses normalisasi untuk mendapatkan kata normal dari setiap kandidat *slang*. Proses pertama yang dilakukan adalah dengan mengkalkulasikan kesamaan (*similarity*)antara kandidat *slang* dan kata kamus Bahasa Indonesia berdasarkan vektor *word embedding* masing-masing. Kemudian untuk setiap kandidat *slang*, sortir (urutkan) skor kesamaan setiap kandidat *slang* dan kata kamus Bahasa Indonesia. Setelah dilakukannya proses sortir, pilih skor kesamaan > *threshold*. Jika skor kesamaan > *threshold* maka dimasukkan ke kamus *slang* yang terdiri dari kata *slang*, kata normal, *threshold* dan *similarity* dan jika skor kesamaan < *threshold* maka dihapus dari daftar kamus *slang*.

Untuk setiap kandidat *slang* memiliki kata normal dengan melihat dari *similarity* kandidat *slang* dan normal. *Threshold* yang digunakan adalah 0.05, 0.1, dan 0.2.



Gambar 3. *Bar chart* berdasarkan Eksperimen Kamus Slang

Dapat dilihat dari visualisasi diatas *threshold* 0.05 memiliki 100 kata *slang* dan normal, *threshold* 0.1 memiliki 99 kata *slang* dan normal, dan *threshold* 0.2 memiliki 79 kata *slang* dan normal.

5. Kesimpulan dan Saran

Berdasarkan uraian hasil penelitian dan pembahasan yang telah dijelaskan sebelumnya, maka dapat diambil kesimpulan langkah untuk membangun kamus *slang* adalah dataset komentar youtube dengan *pre-trained FastText* model dilakukan proses preprocessing yang mencakup *case folding*, *tokenizing* dan *remove duplicate*. Setelah dilakukannya proses *preprocessing*, dilakukan tahap membuat list *token* dan *list comprehension* untuk mendapatkan *word vector* setiap kata komentar youtube. Kemudian dilakukan tahap membuat *list* token kamus Bahasa Indonesia dan *list*

comprehension untuk mendapatkan *word vector* setiap kata kamus Bahasa Indonesia. Untuk mendapatkan vektor kata untuk setiap kata komentar youtube dan kamus bahasa indonesia diperlukan metode *pre-trained FastText model* dengan nilai *threshold* yang diterapkan untuk mendapatkan kata *slang* adalah 0.2, 0.3 dan 0.4.

Kemudian dilakukan tahap *cosine similarity* untuk menghitung kesamaan kata antar komentar youtube dan kamus Bahasa Indonesia dan *threshold value* sebagai nilai ambang dalam mendapatkan nilai *similarity word* kata komentar youtube. *Threshold value* yang digunakan untuk membangun*slang dictionary* adalah 0.05, 0.1, dan 0.2. Setelah di dapatkan hasil dari *threshold* value yang digunakan untuk membangun *slang dictionary*, maka didapatkan hasil terbaik yaitu dari *threshold* dengan nilai 0.05 dikarenakan terdapat hasil *Accuracy 0.65, Precission 1.00, Recall 1.00* dan *F-1 score* didapatkan hasil 1.00. Setelah kata slang terkumpul, maka dilakukan tahap normalisasi yang bertujuan untuk mendapatkan kata normal dari setiap kandidat slang. Tahap normalisasi yaitu melakukan tahap kalkulasi kesamaan (*similarity*) terhadap kandidat slang dan kata kamus Bahasa Indonesia berdasarkan *word embedding vector* masing-masing. Kemudian, untuk setiap kandidat slang, sortir (urutkan) skor kesamaan setiap kandidat slang dan kata kamus Bahasa Indonesia. Setelah dilakukannya proses sortir, pilih skor kesamaan > *threshold*. Jika skor kesamaan > *threshold* maka dimasukkan ke kamus *slang* yang terdiri dari kata slang, kata normal, *threshold* dan *similarity* dan jika skor kesamaan < *threshold* maka dihapus dari daftar kamus slang. Setelah didapatkankata slang dan kata normal pada tahap normalisasi, maka dilakukan tahap evaluasi terhadap kata *slang* dan kata normal yang diterapkan ke sentimen analisis.

Disarankan pada penelitian selanjutnya untuk melakukan studi lanjut untuk mengetahui varian model *pre-trained* terbaik penghasil *slang dictionary* Bahasa Indonesia dan untuk membangun*slang dictionary* multi bahasa.

Daftar Rujukan

[1] M. Rusli, M. R. Faisal, and I. Budiman, "Ekstraksi Fitur Menggunakan Model Word2Vec Untuk Analisis Sentimen Pada Komentar Facebook," *Semin. Nas. Ilmu Komput.*, vol. 2, no. January 2019, pp. 104–109, 2019.

dan T. R. I. Badan Pengembangan dan Pembinaan Bahasa, Kementerian Pendidikan, Kebudayaan, Riset, "KBBI Daring," 2016. https://kbbi.kemdikbud.go.id/entri/slang

[2] A. Nurdin, B. Anggo Seno Aji, A. Bustamin, and Z. Abidin, "Perbandingan Kinerja Word Embedding Word2Vec, Glove, Dan Fasttext Pada Klasifikasi Teks," *J. Tekno Kompak*, vol. 14, no. 2, p. 74, 2020, doi: 10.33365/jtk.v14i2.732.

[3] L. Wu, F. Morstatter, and H. Liu, "SlangSD: building, expanding and using a sentiment dictionary of slang words for short-text sentiment classification," *Lang. Resour. Eval.*, vol. 52, no. 3, pp. 839–852, 2018, doi: 10.1007/s10579-018-9416-0.

[4] J. Turner and M. Kantardzic, "Twitter query expansion via Word2Vec-Urban Dictionary model," *ACM Int. Conf. Proceeding Ser.*, pp. 43–46, 2018, doi: 10.1145/3277104.3278310.

[5] A. Josi, L. A. Abdillah, and Suryayusra, "Penerapan teknik web scraping pada mesin pencari artikel ilmiah," 2014, [Online]. Available: http://arxiv.org/abs/1410.5777

[6] A. P. Widyassari *et al.*, "Review of automatic text summarization techniques & methods," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 4, pp. 1029– 1046, 2022, doi: 10.1016/j.jksuci.2020.05.006.