

Deteksi Berita Hoax di Media Sosial Twitter dengan Ekspansi Fitur Menggunakan Glove

1st Alfiah Ramadian Jamaludin

Fakultas Informatika
Universitas Telkom
Bandung, Indonesia

alfiramadian@students.telkomuniversity.ac.id

2nd Erwin Budi Setiawan

Fakultas Informatika
Universitas Telkom
Bandung, Indonesia

erwinbudisetiawan@telkomuniversity.ac.id

Abstrak

Terdapat banyak berita *hoax* yang saat ini banyak beredar di masyarakat. Bahkan di Indonesia, khususnya di media sosial, fenomena *hoax* tidak jarang terjadi. *Hoax* bisa membuat orang resah karena informasi yang tidak diketahui kebenarannya. Untuk mengetahui informasi yang disebarluaskan, kita perlu mengklasifikasikannya untuk mengetahui apakah itu *hoax* atau tidak. Oleh karena itu dalam penelitian ini, dikembangkan sistem yang mampu mendeteksi informasi *hoax* di media sosial Twitter dengan menggunakan metode ekspansi fitur *Global Vectors for Word Representation (GloVe)*. Metode ekspansi fitur *Glove* digunakan untuk mengurangi adanya ketidakcocokan kosakata pada sebuah tweet pada Twitter. Proses klasifikasi yang digunakan beberapa metode yaitu, *Support Vector Machine (SVM)*, *Naive Bayes* dan *Recurrent Neural Network (RNN)*. Pada hasil penelitian menunjukkan bahwa sistem pendeteksi *Hoax* menggunakan ekspansi fitur memiliki akurasi sebesar 91,92% pada metode klasifikasi *SVM* dengan menggunakan korpus *GloVe Tweet+Berita* dan menggunakan Top 10.

Kata kunci: *Hoax*, *Twitter*, *Support Vector Machine (SVM)*, *Naive Bayes*, *Recurrent Neural Network (RNN)*.

Abstract

There is a lot of *hoax* news that is currently circulating in the community. Even in Indonesia, especially in social media, the phenomenon of *hoaxes* is not uncommon. *Hoaxes* can make people nervous because of information that is not known to be true. To find out the information that is disseminated, we need to classify it to find out whether it is a *hoax* or not. Therefore, in this study, a system was developed that was able to detect *hoax* information on Twitter social media using the *Global Vectors for Word Representation (GloVe)* feature expansion method. The *Glove* feature expansion method is used to reduce vocabulary mismatches in a tweet on Twitter. The classification process used several methods, namely, *Support Vector Machine (SVM)*, *Naive Bayes* and *Recurrent Neural Network (RNN)*. The results show that the *Hoax* detection system using feature expansion has an accuracy of 91.92% in the *SVM* classification method using the *GloVe Tweet+Berita* corpus and using Top 10.

Keywords: *Hoax*, *Twitter*, *Support Vector Machine (SVM)*, *Naive Bayes*, *Recurrent Neural Network (RNN)*.

I. PENDAHULUAN

Hoax (baca: / ho.aks /) adalah berita atau liputan yang mencoba menarik perhatian pembaca mengenai kebenaran lalu mencoba untuk meyakinkan pembaca. Persebaran *hoax* bergantung pada siapa yang membacanya apakah akan ikut

menyebarkan tanpa cek kembali kebenarannya atau dengan sengaja mengirimkan ulang ke semua orang di media sosial [1]. Pada kamus besar bahasa Indonesia kata *hoax* dibaca menjadi *hoaks*, yang artinya informasi bohong [2].

Seiring dengan berkembangnya teknologi, masyarakat beralih ke media sosial sebagai alat komunikasi. Tanpa disadari *hoax* atau berita palsu juga ikut berkembang di kalangan masyarakat pengguna media sosial. Media sosial merupakan kategori wacana online tempat orang membuat konten dan membagikannya [3]. Jika ada yang menyebarkan berita bohong akan mendapat hukuman selama 6 tahun dan denda sampai dengan Rp.1.000.000.000 (satu milyar rupiah) [4]. Pada penelitian, menganalisis bahwa Twitter merupakan platform media sosial paling populer dalam penyebaran berita *hoax* selama 2016-2018. Twitter, juga merupakan media sosial paling populer untuk meneliti tentang berita *hoax*. Peneliti menggunakan informasi penyebaran data dari Twitter, menganalisis, dan kemudian menentukan informasi tersebut bohong atau tidak. Untuk menentukan informasi tersebut, penelitian tentang sistem klasifikasi *hoax* telah dilakukan oleh beberapa peneliti. Salah satu studi penelitian membandingkan tiga algoritma yaitu algoritma *Naive Bayes*, algoritma *Support Vector Machine (SVM)*, dan algoritma *Decision Tree* dalam klasifikasi *hoax*. Dan hasilnya adalah bahwa algoritma *Naive Bayes* menunjukkan akurasi terbaik yaitu sebesar 91,36% [5]. Penelitian metode klasifikasi *hoax* lainnya yaitu menggunakan *SVM* dengan bobot *TF-IDF* yang menunjukkan akurasi 95,83% [6].

Teknik *Word2Vec* memiliki cara kerja dengan cara pemindaian kata yang terdapat pada korpus besar dan memiliki cara yang mirip dengan *GloVe*, namun kami menggunakan metode acak untuk membuat vektor pada setiap kata-kata dengan cara membangun neural network yang sederhana. Jika dibandingkan, metode *Word2Vec* dapat menandingi dari metode *TF-IDF*. Yaitu dapat merepresentasikan setiap kata-kata dengan menyuguhkan kecocokan arti yang sama pada setiap katanya. Setelah rilis *Word2Vec*, metode *word embeddings* menjadi topik hangat pada kalangan peneliti NLP, dan diikuti dengan *word embeddings* lainnya yang muncul. *GloVe* adalah salah satunya. (Pennington et al., 2014). Dilihat dari risetnya, metode *word embeddings GloVe* dapat menyaingi *Word2Vec* padasegi akurasi, kecepatan, kemampuan pada penyesuaian diri saat sistem [7]. Ini yang menjadi alasan mengapa metode *GloVe* dipakai pada penelitian tugas akhir ini, untuk meminimalisir kesalahan kosakata *vocabulary*. Pada penelitian ini, *GloVe* dipakai pada korpus yang berbahasa Indonesia.

Berdasarkan penelitian di atas, masih ada yang belum menggunakan fitur ekspansi di dalam sistemnya, dan pada tugas akhir ini akan dilakukan pengembangan dengan menambah fitur ekspansi dalam proses klasifikasi. Dalam penambahan pengembangan sistem kami menerapkan fitur ekspansi *GloVe* secara otomatis untuk menghitung nilai *similarity* antar kata pada korpus sehingga dapat mengurangi kemiripan antar kata. Metode klasifikasi yang di gunakan dalam membangun sistem ini adalah *Support Vector Machine (SVM)*, *Naive Bayes* dan *Recurrent Neural Network (RNN)*. Output yang diinginkan pada eksperimen ini adalah sebuah program yang dapat mendeteksi info *hoax* dengan akurasi yang tinggi pada metode klasifikasi yang digunakan.

Pada jurnal tugas akhir ini disusun sebagai berikut, bagian kedua berisi studi terkait yaitu tentang studi literatur atau teori yang berkaitan dengan penelitian tugas akhir ini. Bagian ketiga terdapat sistem yang akan di bangun yang berisikan rancangan sistem yang telah dibuat. Bagian keempat terdapat eksperimen pada bagian ini berisikan data set dan hasil eksperimen pada penelitian. Bagian kelima yaitu kesimpulan, berisikan tentang rangkuman dan intisari dari penelitian yang telah dilakukan.

II. KAJIAN TEORI

Terdapat beberapa hasil penelitian terdahulu yang dijadikan sebagai dasar tinjauan dalam pembuatan dan penelitian tugas akhir ini. Pada penelitian ini fitur ekspansi untuk sentimen analisis pada *twitter* memakai metode *SVM*, *Logistic Regression* dan *Naive Bayes* [8]. Memiliki *accuracy* paling tinggi pada metode *Logistic Regression* yaitu 98.81% sedangkan untuk *Naive Bayes* yaitu 82.4%. Kelebihan: akurasi yang didapat dari tiap metode di atas 80%. Kekurangan: perbandingan data label tidak seimbang, yaitu label positif 50.50%, label negatif 8.48% dan label netral 41.02%.

Pada penelitian ini [9], melakukan perbandingan kinerja metode *RNN* dan *Naive Bayes* dengan teknik pembobotan *TF-IDF* dan menunjukkan hasil akurasi yang paling baik yaitu sebesar 97.77% untuk metode *RNN* yang dibandingkan dengan *Naive Bayes* sebesar 80%. Kelebihan: akurasi dengan metode *RNN* mendapatkan hasil yang sangat baik yaitu 97.7%. Kekurangan: sementara akurasi dengan metode *SVM* hanya mendapatkan 63.99%, serta perbandingan label data yang tidak seimbang yaitu 3800 label positif, 800 label negatif dan 400 label netral.

Pada penelitian ini [10], di hasilkan bahwa metode *TF-IDF* dan klasifikasi *SVM* dapat digunakan untuk mendeteksi berita *hoax* di *twitter*, hasil pengujian kinerja dengan menganalisis akurasi sistem dibuat diperoleh rasio terbaik pada 90:10 dengan nilai 78.33%. Kelebihan: Akurasi testing dilakukan perkategori dataset yaitu *retweet*, *URLs*, *mention*, *hashtag*, dll. Kekurangan: Akurasi terbaik yang didapatkan hanya sebesar 78.33%.

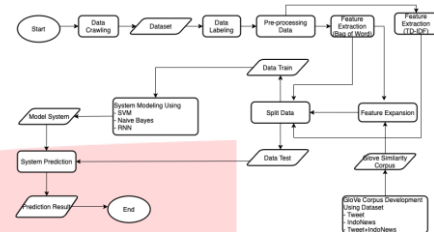
Pada penelitian ini [11], di lakukan fitur semantik dan pembobotan *TF-IDF* serta pengklasifikasian menggunakan *Naive Bayes*, *Random Forest* dan *RNN*. Dan menghasilkan akurasi 92.70% untuk metode *GloVe* dengan klasifikasi *RNN*. Kelebihan : Akurasi yang didapat dari tiap metode diatas 80% serta memiliki perbandingan label dataset

yang seimbang yaitu 50:50. Kekurangan: persebaran dataset yang digunakan tidak dijelaskan secara detail, hanya mencakup 50% saja.

III. METODE

Sistem klasifikasi *hoax* yang akan dibangun dalam penelitian ini dapat dicermati dalam Gambar 1.

A. Gambaran Sistem



Gambar 1. Gambaran Flowchart Sistem

B. Crawling dan Pelabelan Data

Pada tugas akhir ini menggunakan dataset yang berasal dari hasil *crawling Twitter* yang berbahasa Indonesia dengan menggunakan *Application Programming Integration (API) Key* yang telah diberikan oleh *Twitter Developer*. Dalam proses *crawling* data diambil *Tweet* dengan kata kunci tertentu dan *hashtag*, data tersebut di simpan dengan format CSV dan di beri label 0 untuk *non-hoax* dan 1 untuk *hoax*. Secara umum cara yang digunakan penulis untuk menentukan data *tweet hoax* atau tidak dengan cara menelusuri kembali berita pada media-media dan di lakukan secara manual, pelabelan di lakukan dengan bantuan 3 orang untuk menentukan representasi setiap yang memberi label dengan menggunakan prinsip *majority vote*, dan suara terbanyak maka akan di ambil untuk hasil label yang fix. Sebagai contoh data *tweet* yang sudah diberikan label dapat diperhatikan pada Tabel 1.

Tabel 1. Contoh Tweet dan Label dalam Dataset

| Tweet | Label |
|--|----------|
| Bill Gates Beli Aplikasi Telegram USD 600 Juta | Hoax |
| Masker dapat menghindarkan kita dari penyebaran dan penularan Covid-19 | Non Hoax |
| Vaksin Covid dapat sebab kanker serviks | Hoax |
| Masyarakat Papua percaya kerja perintah | Non Hoax |
| Novel Baswedan tak layak jadi pegawai KPK professional | Non Hoax |

C. Preprocessing

Preprocessing data merupakan tahapan yang akan di lakukan sebelum data di masukan ke dalam klasifikasi. Proses ini berfungsi untuk menghilangkan kesalahan, kesamaan bentuk kata serta mengurangi banyaknya kata dari data yang sudah ada. Terdapat 5 tahap dalam *preprocessing*, antara lain:

1. *Cleaning*: Untuk proses ini melakukan penghapusan dari karakter – karakter yang harusnya tidak diperlukan, tanda baca, url (“http://”, “www...com”), hastag (#), dan username (@username) yang terdapat pada dataset.
2. *Case folding*: Pada proses ini dilakukan perubahan seluruh kata yang ada pada dataset yang awalnya huruf kapital menjadi huruf kecil, proses ini dibantu oleh library String.
3. *Stopword Removal*: Pada tahap ini akan diterapkan eliminasi kata yang tak sesuai atau tidak berhubungan seperti yang tidak memiliki arti khusus seperti kata ganti, preposisi, dan konjungsi. Proses penghapusan *stopwords* dalam Bahasa Indonesia digunakan suatu *library* pada *python* yaitu *library* Natural Language Toolkit (NLTK).
4. *Normalisasi Kata*: Yaitu mengubah kata singkatan dan menstandarkan kata yang memiliki makna yang sama, salah penulisan (typo), kata gaul, dan kata alay (tidak baku menjadi baku) dengan bantuan *corpus* kata yang dibuat secara manual.
5. *Stemming*: Pada proses ini dilakukan perubahan kata – kata yang ada menjadi kata dasarnya, dengan cara menghapus imbuhan yang ada pada kata tersebut. Proses *stemming* menggunakan sebuah *library* khusus untuk pemrosesan Bahasa Indonesia yaitu *library* Sastrawi.
6. *Tokenization*: Pada proses ini dilakukan pemecahan kalimat menjadi kata per kata yang biasa disebut token.

D. Ekstraksi Fitur

Fitur ekstraksi merupakan tahapan untuk *menscanning* ciri yang terdapat dalam teks dokumen. Fitur ekstraksi sebagai faktor yang sangat krusial pada pengerjaan dokumen dalam mesin pencari lantaran sangat memilih keberhasilan proses *text mining*. *Text mining* proses yang dilakukan untuk mencari informasi yang terdapat dalam dokumen berdasarkan nilai ciri fitur hasil ekstraksi fitur [12]. Fitur ekstraksi serta fitur TF-IDF, pada proses ini akan melakukan *vector text* kepada setiap dokumen dengan TF-IDF. Proses ini dikenal sebagai Vektorisasi Teks. Setiap entri menandakan ada (atau tidak adanya) kata dalam ulasan. Kami menempatkan 1 jika kata tersebut ada dalam ulasan, dan 0 jika tidak ada. Misal ada vektor seperti berikut: ”bagus”, ”film”, ”bukan”, ”a”, ”lakukan”, ”suka” dan terdapat sebuah tweet seperti ”bukan sebuah film yang bagus” maka representasi vektor yang terbentuk adalah 1,1,1,0,0,0.

E. Term Frequency – Inverse Document Frequency (TF-IDF)

Setelah tahap *preprocessing*, maka dilakukan tahap *Term Frequency dengan Inverse Document Frequency* TF-IDF untuk menetapkan bobot pada data. TF-IDF merupakan metode penghitungan bobot kata yang sering dipakai saat mengambil informasi. Metode ini merupakan metode yang

hasilnya efisien dan memberikan output yang tepat, pada proses ini menghitung nilai *Term Frequency* dan *Reverse Document Frequency* (IDF) untuk setiap kata pada dokumen *corpus* [13].

$$W_{ij} = t_{fij} \rightarrow Id f_j, Id f_j = \log \frac{N}{df} \quad (1)$$

F. GloVe

GloVe merupakan *word embeddings* dari *Stanford University* untuk memrepresentasikan kata-kata. *GloVe* adalah metode pengajaran tanpa pengawasan untuk representasi kata yang melampaui model lain dalam kesetaraan kata dan identifikasi nama [14]. *Unsupervised learning* adalah cara pengumpulan data tanpa data latih sehingga data diambil dari data yang ada di banyak bagian. *GloVe* menggunakan skrip korpus *Twitter*. Kata-kata ini menghasilkan vektor dengan ratusan dimensi untuk setiap kata yang dapat ditemukan nanti di kamus. Algoritma mencakup probabilitas bahwa sebuah kata akan muncul didalam perhitungan. Model *GloVe* ditetapkan berdasarkan:

$$w^T + w_k + b_i + b_k = \log(X_{ik}) \quad (2)$$

Dijelaskan bahwa w merupakan vektor kata, sementara w_k merupakan vektor konteks kata. b_i dan b_k merupakan arah skalar pada kata utama dan konteks kata. X merupakan matriks kemunculan dimana X_{ik} menyatakan jumlah berapa waktu kata k ada pada konteks kata i . Untuk menghitung nilai X_{ik} dilakukan dengan menjumlahkan statistik keberadaan kata dalam bentuk matriks keberadaan x . Dimana setiap elemen pada matriks X_{ik} mewakili seberapa sering kata muncul dalam konteks kata j . Konteks kata adalah kumpulan kata yang berada sebelum dan sesudah kata i sebanyak *windows size* yang diberikan. Pada setiap kata dilakukan pembobotan kata dengan cara $\frac{1}{distance}$ dimana *distance* dihitung dari panjang konteks kata dikurangi posisi kata tersebut. Kemudian perhitungan nilai pada $f(X_{ik})$ yaitu dengan menggunakan persamaan dibawah ini:

$$f(X_{ik}) = \begin{cases} \left(\frac{X_{ik}}{x_{max}}\right)^\alpha, & \text{if } X_{ik} < x_{max} \\ 1; & \text{lainnya} \end{cases} \quad (3)$$

Gambar 2. Rumus

Model *GloVe* diatas merupakan fungsi pembobotan ke dalam fungsi cost yang memberikan model seperti di bawah ini :

$$J = \sum_{i,j=1}^V f(X_{ij})(w^T w_j + b_i + b_j - \log X_{ij})^2 \quad (4)$$

G. Ekspansi Fitur

Selanjutnya melakukan fitur ekspansi, fitur ini dilakukan setelah data *tweet* diterapkan metode *Word Embedding*, fungsi metode tersebut untuk mengatasi masalah ketidakcocokan pada kosakata, yaitu dengan mengidentifikasi kata-kata yang telah di dapatkan dari penggunaan

metode *GloVe*. *GloVe* merupakan metode yang berbasis hitungan, dimana *GloVe* akan menghitung seberapa sering suatu kata muncul secara bersamaan dalam sebuah korpus sehingga dapat ditentukan nilai *similarity* dari kata-kata tersebut. Kegunaan metode tersebut untuk mengidentifikasi dan mengelompokkan kemiripan kata-kata. Kemudian pada penelitian ini dilakukan perluasan fitur *Feature Expansion* pada dokumen yang telah di beri bobot nilai menggunakan TF-IDF.

Algoritma *GloVe* ini akan menghasilkan *output* berupa *list* dari *similarity words*. Sebagai contoh dapat dilihat pada Tabel 2 daftar dari kata yang memiliki *similarity* dengan kata “apbd” yang telah diurutkan sesuai dengan rankingnya.

Tabel 2. Contoh Similarity Word Dari Kata Apbd

| Rank 1 | Rank 2 | Rank 3 | Rank 4 | Rank 5 | Rank 6 | Rank 7 | Rank 8 | Rank 9 | Rank 10 |
|--------|----------|--------|-------------|--------|---------|-----------|-----------|---------|---------|
| apbn | anggaran | rupiah | dianggarkan | Bps | Belanja | pembinaan | pendanaan | triliun | dana |

Setelah didapatkan *similarity words*, akan dilanjutkan dengan proses ekspansi fitur terhadap vektor representasi yang didapat dari proses ekstraksi fitur. Proses untuk melengkapi vektor tersebut akan dilakukan dengan mengganti vektor-vektor yang bernilai 0 dengan *similar words* yang ada di daftar korpus *GloVe*. Langkah-langkah yang dilakukan dalam proses ekspansi fitur ini mengacu pada penelitian [15]. Algoritma Ekspansi fitur yang dibangun menggunakan input berupa teks vektor yang berupa matriks dan korpus *similarity* dari *GloVe* yang akan menghasilkan matriks dari teks vektor. Algoritma ekspansi fitur yang digunakan untuk variasi Top 1 *features* pada penelitian ini dapat dilihat pada algoritma 1.

Algoritma 1 *featureExpansion(textVector, korpusGloVe)* → matriks teks vektor
 //Mengaplikasikan ekspansi fitur terhadap setiap fitur pada matriks vektor dataset

Input : *textVector* adalah matriks vektor dari dataset. *korpusGloVe* adalah korpus *similarity* kata-kata yang dibangun dengan *GloVe*

```

1: for each val ∈ GBoW_vector do
2:   temp ← val.copy()
3:   for j ← 0 to val.length() do
4:     t ← cv.vocabulary_.get(similar[j])
5:     if (val[j] = 0) and (similar[j] ≠ Null) and (t ≠ Null):
6:       if (temp[t] = 1):
7:         val[j] ← 1
8:       end if
9:     end if
10:  end for
11: end for
    
```

Gambar 3. Algoritma 1

Sebagai ilustrasi dari algoritma 1, jika terdapat *tweet A* yaitu “Surat Tawaran Dana Bantuan APBN/APBD Mengatasnamakan Ditjen Dikti Kemendikbud” dan terdapat nilai vektor untuk kata “apbd” adalah 0. Maka akan dicari *similarity* dari kata “apbd” pada korpus *GloVe* yang mana dapat dilihat pada Tabel 2. Setelah itu, didapatkan bahwa kata “apbn” memiliki *similarity* dengan ranking tertinggi dari kata “apbd” dan kata “apbn” juga terdapat didalam *tweet A*, maka nilai vektor dari kata “apbd” akan dirubah nilainya menjadi 1.

H. Support Vektor Machine (SVM)

Support Vektor Machine (SVM) cara kerjanya yaitu dengan menemukan fungsi pemisah line yang dapat memisahkan dua set dari data antara dua kelas yang berbeda [16]. Metode ini disebut dengan metode *learning machine* yang memiliki prinsip *Structural Risk Minimization* (SRM) yang bertujuan menemukan *hyperplane* terbaik untuk memisahkan dua buah dari kelas pada inputan space [17]. Secara umum cara kerja SVM ini berprinsip dengan *linier classifier*, lalu kemudian dikembangkan untuk mendapat hasil pada kasus *non linear* dengan cara menggunakan konsep kernel pada ruang kerja yang berdimensi tinggi [17].

I. Naïve Bayes

Metode *NaiVe Bayes* atau *NaiVe Bayes Classifier* (NBC) merupakan cara yang biasa dimanfaatkan untuk pengklasifikasi *text* pada dokumen. Konsep yang digunakan NBC adalah probabilitas sebagai acuan teorinya. Dalam tulisannya, Han, J. dan Kamber, M. mengatakan: “*Bayesian classifiers* memiliki tingkat kecepatan dan *accuracy* yang unggul ketika diterapkan dalam basedata yang besar” (2001, hlm. 296). Dengan buku tersebut, maka sistem *Naive Bayes* merupakan sistem yang dipergunakan untuk proses tahapan klasifikasi *text* dalam eksperimen ini. Ada 2 tahap pada proses yang dilakukan pada klasifikasi *text*. Yakni proses pertama merupakan *training* terhadap himpunan artikel. Sedangkan untuk proses kedua yang dilakukan merupakan tahapan klasifikasi dokumen yang belum diketahui topiknya [18]. Theorema Bayes:

$$P(C_i | x) = \frac{P(X|C_i) \cdot P(C_i)}{P(X)} \quad (5)$$

$$P(X|C_i) = \prod_{t=1}^n \quad (6)$$

J. Recurrent Neural Network (RNN)

RNN merupakan sistem jaringan saraf yang berulang dan menggunakan *hidden layer* untuk nilai neuron akan diproses untuk data inputan [19]. Proses yang dilakukan untuk *Recurrent Neural Network* (RNN) akan memanggil secara berkali-kali untuk menjalankan inputan yang masuk, yaitu data linear. *Recurrent Neural Network* (RNN) juga mempunyai beragam bentuk, salah satu bentuknya yaitu global digunakan standar *Multi-Layer Perceptron* (MLP) yang ditambah dengan *loop* tambahan. Maka dari itu tahapan ini dapat memanfaatkan kinerja pemetaan *non-linear* yang dari MLP [20]. Terdapat identitas dari *Recurrent Neural Network* (RNN) yaitu mengklasifikasikan data *time series* dan *sekuensial*. Data *time series* sendiri adalah data yang disatukan menurut deretan waktu dengan rentang tertentu, sedangkan untuk data *sekuensial* yaitu suatu contoh data yang dipakai secara terurut dan setiap deretan bersinambungan satu sama lain [21]. *Recurrent Neural Network* (RNN) juga mempunyai fungsi aktivasi deterministik, yaitu fungsi aktivasi *Recurrent Neural*

Network (RNN) ialah han h, dengan perhitungan matematikanya sebagai berikut [22]:

$$h_t = \tan h(S^t_{k=1} W^t -^k W_{in} x_k) \quad (7)$$

K. Confusion Matrix

Confusion Matrix merupakan sebuah proses yang sering dipergunakan untuk menghitung accuracy, recall, precision, dan error rate dapat dilihat pada Tabel 3. Yaitu, precision dapat menilai keahlian dari metode untuk menemukan hasil peringkat yang paling mendekati, dengan cara dideskripsikan sebagai penilaian dokumen yangdi retrieve dan benar-benar mendekati terhadap hasil query. Recall menilai kemampuan metode untuk menemukan semua item yang paling relevan dengan koleksi dokumen dan didefinisikan sebagai penilaian dokumen yang ber- hubungan terhadap query. Accuracy adalah komparasi kasus yang dinyatakan cocok dengan jumlah seluruh case dan error rate adalah kasus yang dipahami salah dengan jumlah seluruh case [23].

Keterangan:

TP = Jumlah data positif dan diprediksi benar.

TN = Jumlah data negative yang diprediksi benar.

FP = Jumlah data negative namun diprediksi data positif.

Tabel 3. Tabel Confusion Matrix

| | | Prediksi | | |
|--------|----------|----------|---------------------|---------------------|
| | | Hoax | TN (True Negative) | FP (False Positive) |
| Aktual | Hoax | | | |
| | Non-Hoax | | FN (False Negative) | TP (True Positive) |

FN = Jumlah data positif namun diprediksi data negatif

Akurasi merupakan seberapa akurat dapat mengklasifikasikan model dengan benar.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (8)$$

Precision merupakan nilai perbandingan antara data yang diminta dengan hasil prediksi model tersebut.

$$Precision = \frac{TP}{TP+FP} \quad (9)$$

Recall merupakan nilai yang menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi.

$$Recall = \frac{TP}{TP+FN} \quad (10)$$

F1 - Score merupakan perbandingan rata-rata dari nilai precision dan nilai recall yang dibobotkan.

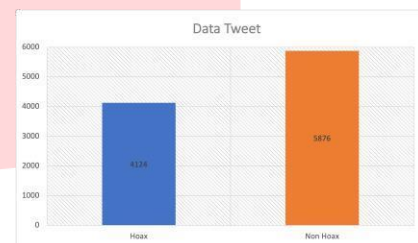
$$F1 - Score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (11)$$

Di gunakan pengukuran performasi Accuracy karena, akurasi merupakan jumlah proporsi yang prediksi benar, mencakup TP+TN lalu di bagi dengan total data. Sedangkan digunakan F-Score karena F-Score merupakan rata-rata yang nantinya akan dibobotkan (harmonic mean) dan gabungan dari recall dan presicion, yang mana cukup untuk membandingkan dari segala kemungkinan matrix.

IV. HASIL DAN PEMBAHASAN

A. Data

Data tweet yang telah didapatkan dari hasil crawling berjumlah 26.948 tweet berbahasa Indonesia yang memuat data yang berhubungan dengan bahasan hoax di Indonesia, namun data tweet yang digunakan berjumlah 10.000. Kemudian data tersebut akan dijadikan sebagai data train dan data test. Data ini terdiri dari beberapa keyword berbeda yang berkaitan dengan hastag, #VaksinUntukKita, #IndonesiaMaju, #NKRI, #Mudik, #Paspampres, #Wakilbupatisangihe, #Gajijakarta. Dengan jumlah penyebarannya data yang telah diberi label hoax dan non-hoax dapat dilihat pada Gambar 2 di bawah ini.



Gambar 3. Persebaran Data

Kemudian, terdapat data pelengkap pembuatan kamus kata menggunakan data yang diambil dari beberapa media berita seperti CNN Indonesia, Sindonews, Kompas, Tempo, Detik.com, Liputan6, dan Republika sebanyak 142.544 data. Digunakan korpus Indonews untuk kombinasi percobaan pada penelitian ini mencari hasil terbaik. Komposisi data IndoNews yang digunakan untuk pembuatan kamus similarity dengan word embedding yang dapat dilihat pada Tabel 4.

Tabel 4. Data Indonews

| Nama Redaksi | Jumlah |
|---------------|---------------|
| CNN Indonesia | 29349 |
| Republika | 53812 |
| Kompas | 15055 |
| Tempo | 15055 |
| SindoNews | 13702 |
| Detik.com | 7974 |
| Total | 142544 |

Terdapat 3 corpus GloVe yang akan dibangun, diantaranya adalah corpus GloVe dengan menggunakan dataset Tweet saja, corpus GloVe dengan menggunakan dataset IndoNews saja dan corpus GloVe yang menggunakan dataset gabungan dari keduanya. Perbandingan jumlah kata yang dihasilkan dalam corpus GloVe yang dibangun dapat dilihat pada Tabel 5.

Tabel 5. Jumlah Data corpus GloVe

| Korpus Glove | Jumlah |
|----------------|---------------|
| Tweet | 10000 |
| Indonews | 142544 |
| Tweet+Indonews | 152544 |
| Total | 338984 |

B. Hasil dan Analisis Eksperimen

Terdapat 3 skenario percobaan yang dilakukan di penelitian ini dengan menggunakan 3 metode klasifikasi yang berbeda yaitu *Naive Bayes* (NB), *Recurrent Neural Network* (RNN) dan *Support Vector Machine* (SVM). Skenario pertama merupakan pembuatan model awal yang menggunakan metode *Naive Bayes*, RNN dan SVM dengan *hyperparameter tuning*. Skenario kedua merupakan percobaan terhadap model yang telah di buat dan dikombinasikan dengan TF-IDF. Skenario ketiga merupakan skenario dimana model dikombinasikan dengan ekspansi fitur GloVe dengan menggunakan beberapa korpus yaitu, korpus *tweet*, korpus *IndoNews* dan korpus *IndoNews + tweet*. Untuk proporsi data yang di gunakan adalah 75:25 data.

4.2.1 Pengujian Pertama

Pengujian ini dilakukan terhadap data *tweet* yang telah *dicrawling* pada gambar 2 dan hasil dari pengujian dapat dilihat pada Tabel 6, untuk menjadi model awal dan sudah dilakukan *hyperparameter tuning* dimana nilai yang terbesar adalah untuk metode *Support Vector Machine* (SVM) dengan nilai akurasi sebesar 76,52% dan nilai *F1-Score* sebesar 69,94%.

Tabel 6. Nilai Akurasi dan *F-1 Score* Model Awal

| Metode | Akurasi(%) | <i>F1-Score</i> |
|-------------|------------|-----------------|
| Naive Bayes | 72,20 | 68,10 |
| RNN | 46,75 | 41,57 |
| SVM | 76,52 | 69,94 |

4.2.2 Pengujian Kedua (Model Awal + TF-IDF)

Pengujian kedua dilakukan dengan menggunakan TF-IDF untuk pembobotan setiap kata pada data *tweet* yang digunakan untuk menjadi perbandingan dengan model awal. Hasil pengujian dengan menggunakan vektor TF-IDF terhadap SVM, *Naive Bayes* dan RNN dapat dilihat pada Gambar 4. Nilai yang naik pada pengujian ini hanya metode *Naive Bayes* dan RNN sedangkan metode SVM mengalami penurunan.

| Metode | Akurasi(%) | | <i>F-1 Score</i> | |
|-------------|------------|---------------------|------------------|---------------------|
| | Model Awal | Model Awal + TF-IDF | Model Awal | Model Awal + TF-IDF |
| Naive Bayes | 72,20 | 73,08(+0,88) | 68,10 | 66,89(+1,21) |
| RNN | 46,75 | 47,09(+0,34) | 41,57 | 43,62(+2,05) |
| SVM | 76,52 | 76,48(-0,04) | 69,94 | 69,90(-0,04) |

Gambar 4. Hasil Pengujian Model Awal dan TF-IDF

4.2.3 Pengujian Ketiga (Model Awal + GloVe)

Dilakukan pengujian ketiga untuk mencari model terbaik pada metode SVM dengan *hyperparameter tuning*, *Naive Bayes* dan RNN dengan menggunakan ekspansi fitur metode *GloVe*. Pengujian menggunakan ekspansi fitur dilakukan terhadap Top 1, 5 dan 10 yaitu dari *feature* yang ada pada

korpus *GloVe*. Korpus *GloVe* yang akan digunakan adalah korpus dari *dataset tweet*, korpus dari *dataset IndoNews*, dan korpus gabungan dari *dataset tweet* dan *IndoNews*. Pemilihan Top 1, Top 5 dan Top 10 adalah jumlah n kata yang memiliki similarity dengan kata yang di tuju. Dengan artian Top 1 merupakan 1 kata paling similar, Top 5 merupakan 5 kata paling similar dan Top 10 merupakan 10 kata paling similar. Pada penelitian ini, melakukan eksperimen dan analisa terhadap pengaruh dari *classification* dengan dan tanpa *Feature Expansion*, dengan harapan memperluas hal tersebut dapat meningkatkan hasil *train* model agar mendapat akurasi yang lebih memuaskan. Lalu mengapa tidak menggunakan Top 1, Top 2 dan Top 3, kami berusaha menghindari *overfit* dan *underfit*, memakai rentang yang besar untuk menghindari *underfit*, dan menggunakan rentang yang tidak terlalu besar untuk menghindari *overfit*.

1. Naive Bayes

Perfomansi untuk untuk nilai akurasi dan *F1-Score* *Feature Expansion* menggunakan algoritma klasifikasi *Naive Bayes* dapat dilihat pada Gambar 5 dan 6. Terjadi peningkatan nilai akurasi pada semua fitur.

| | Akurasi(%) | | | |
|--------|------------|--------------|-----------------|-------------------------|
| | Model Awal | Corpus Tweet | Corpus IndoNews | Corpus Tweet + IndoNews |
| Top 1 | 72,20 | 74,44(+2,24) | 77,52(+5,32) | 82,32(+10,12) |
| Top 5 | 72,20 | 75,04(+2,84) | 82,76(+10,56) | 82,44(+10,24) |
| Top 10 | 72,20 | 77,40(+5,20) | 87,04(+14,84) | 88,32(+16,12) |

Gambar 5. Hasil Akurasi Fitur Ekspansi Pada Metode *Naive Bayes* Feature

| Feature | <i>F1-Score</i> (%) | | | |
|---------|---------------------|--------------|-----------------|-------------------------|
| | Model Awal | Corpus Tweet | Corpus IndoNews | Corpus Tweet + IndoNews |
| Top 1 | 68,80 | 70,23(+2,13) | 72,93(+4,83) | 78,56(+10,46) |
| Top 5 | 68,80 | 71,86(+3,76) | 78,88(+10,78) | 78,72(+10,62) |
| Top 10 | 68,80 | 74,28(+6,18) | 84,10(+16) | 85,60(+17,5) |

Gambar 6. Hasil *F1-Score* Fitur Ekspansi Pada Metode *Naive Bayes*

2. Recurrent Neural Network (RNN)

Perfomansi untuk untuk nilai akurasi dan *F1-Score* *Feature Expansion* menggunakan algoritma klasifikasi *Recurrent Neural Network (RNN)* dapat dilihat pada Tabel 10 dan 11. Terjadi peningkatan nilai akurasi pada fitur Top 1 untuk semua korpus, fitur Top 5 mengalami kenaikan pada korpus *IndoNews* dan *Tweet + IndoNews*, sedangkan fitur Top 10 kenaikan terjadi pada korpus *Tweet* dan *IndoNews*.

| Feature | <i>F1-Score</i> (%) | | | |
|---------|---------------------|--------------|-----------------|-------------------------|
| | Model Awal | Corpus Tweet | Corpus IndoNews | Corpus Tweet + IndoNews |
| Top 1 | 41,57 | 46,02(+4,45) | 47,20(+5,63) | 44,79(+3,22) |
| Top 5 | 41,57 | 44,62(+3,05) | 40,42(-1,15) | 46,52(+4,95) |
| Top 10 | 41,57 | 49,03(+7,46) | 44,61(+3,04) | 42,84(+1,27) |

Gambar 7. Hasil Akurasi Fitur Ekspansi Pada Metode *Recurrent Neural Network (RNN)*

3. Support Vector Machine (SVM)

Perfomansi untuk untuk nilai akurasi dan *F1-Score* *Feature Expansion* menggunakan algoritma klasifikasi *Support Vector Machine (SVM)* dapat dilihat pada Gambar 8 dan 9. Terjadi peningkatan nilai akurasi pada semua fitur.

| Feature | Akurasi(%) | | | |
|---------|------------|--------------|-----------------|-------------------------|
| | Model Awal | Corpus Tweet | Corpus IndoNews | Corpus Tweet + IndoNews |
| Top 1 | 76,52 | 77,44(+0,92) | 82,72(+6,20) | 87,64(+11,12) |
| Top 5 | 76,52 | 85,80(+9,28) | 87,68(+11,16) | 86,48(+9,96) |
| Top 10 | 76,52 | 85,80(+9,28) | 87,68(+11,16) | 91,92(+15,4) |

Gambar 8. Hasil Akurasi Fitur Ekspansi Pada Metode *Support Vector Machine (SVM)*

| Feature | F1-Score(%) | | | |
|---------|-------------|--------------|-----------------|-------------------------|
| | Model Awal | Corpus Tweet | Corpus IndoNews | Corpus Tweet + IndoNews |
| Top 1 | 69,94 | 71,48(+1,54) | 78,50(+8,56) | 84,64(+14,7) |
| Top 5 | 69,94 | 79,12(+9,18) | 84,64(+14,7) | 83,87(+13,93) |
| Top 10 | 69,94 | 79,22(+9,28) | 84,64(+14,7) | 90,66(+20,72) |

Gambar 9. Hasil F1-Score Fitur Ekspansi Pada Metode *Support Vector Machine (SVM)*

C. Analisis Hasil Pengujian

Berdasarkan hasil dari pengujian ini, dapat di lihat melalui diagram Gambar 3. Dengan model Top 1 features terjadi peningkatan nilai akurasi pada metode SVM sebesar 87,64% pada korpus *Tweet+Berita* serta nilai *f1-score* sebesar 84,64% dari model awal yang hasilnya 76,52% dan mengalami kenaikan 11,12%, sedangkan untuk metode *Naive Bayes* mengalami kenaikan yang konsisten tidak mengalami penurunan dari 72,20% sampai 82,32%, dan metode RNN terjadi tidak konsisten mengalami naik dan turun di semua model yang di bangun.

Sedangkan pada model dengan Top 5 features dengan menggunakan metode RNN tidak mengalami perubahan yang signifikan hanya naik sekitar 2%, dan untuk model SVM mengalami kenaikan kecuali pada korpus *Tweet+Berita* menjadi turun sekitar 1%. Pada metode *Naive Bayes* di Top 5 features model *Tweet+Berita* menjadi turun kembali sekitar 0,32%.

Selanjutnya, pada model Top 10 features terjadi penurunan akurasi kembali untuk metode RNN dari nilai model awal 46,75% turun menjadi 44,99% pada korpus *Tweet+Berita* dengan nilai *F1-Score* 42,84%, untuk metode *Naive Bayes* dan SVM selalu mengalami peningkatan yang signifikan.



Gambar 10. Persebaran Data

V. KESIMPULAN

Pada penelitian ini, telah dibangun deteksi hoax menggunakan ekspansi fitur metode *GloVe* dengan metode klasifikasi *Naive Bayes*, *Recurrent Neural Network (RNN)*, *Support Vector Machine (SVM)*. Ekspansi Fitur *GloVe* digunakan pada sistem pendeteksi *hoax* ini bertujuan untuk mengurangi ketidakcocokan kosakata pada kalimat *tweet*. Ekspansi fitur dilakukan dengan menggunakan 3 korpus *GloVe (Tweet, IndoNews, dan Tweet + IndoNews)* dan juga 3 variasi ekspansi fitur (Top 1, Top 5, Top 10) untuk mencari model terbaik. Pada hasil penelitian menunjukkan bahwa metode ekspansi fitur *GloVe* berhasil meningkatkan akurasi nilai pada sistem yang telah di bangun. Pada penelitian ini peningkatan nilai akurasi tertinggi terdapat pada model dengan metode klasifikasi SVM dengan korpus *GloVe*

Tweet+Berita dan menggunakan Top 10, yaitu sebesar 91,92% dengan penambahan nilai akurasi 15,4% dari model awal.

Peneliti juga mengamati peningkatan nilai akurasi yang konsisten terjadi pada Top 1 features dengan metode *Naive Bayes* dan SVM, Top 5 features dengan metode *Naive Bayes* dan SVM mengalami kenaikan tetapi di korpus *Tweet+Berita* mengalami turun kembali, Top 10 features dengan metode *Naive Bayes* dan SVM, mengalami kenaikan di setiap korpus yang di gunakan.

Saran untuk peneliti selanjutnya, peneliti dapat mengembangkan dengan melakukan kombinasi menggunakan *word embedding* dan menggunakan metode klasifikasi yang berbeda.

REFERENSI

- [1] Hans Baars, Jule Hintzbergen, Andre' Smulders, and Kees Hintzbergen. *Foundations of information security based on ISO27001 and ISO27002*. Van Haren, 2015.
- [2] Badan Pengembangan dan Pembinaan Bahasa. Hoaks - kbki daring. 2016.
- [3] S Asur and B Huberman. Predicting the future with social media. in 2010 ieeewic. In *ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pages 492–499.
- [4] Undang-undang republik indonesia nomor 11 tentang informasi dan transaksi elektronik. 2008.
- [5] Errissya Rasywir and Ayu Purwarianti. Eksperimen pada sistem klasifikasi berita hoax berbahasa indonesia berbasis pembelajaran mesin. *Jurnal Cybermatika*, 3(2), 2016.
- [6] Dina Maulina and Rofie Sagara. Klasifikasi artikel hoax menggunakan support vector machine linear dengan pembobotan term frequency-inverse document frequency. *Jurnal Mantik Penusa*, 2(1), 2018.
- [7] Irfan Hanif. *Klasifikasi Perintah Bahasa Natural Menggunakan Global Vectors for Word Representations (GloVe), Convolutional Neural Networks, dan Teknik Transfer Learning pada Aplikasi Chatbots*. PhD thesis, Institut Teknologi Sepuluh Nopember, 2018.
- [8] Erwin B Setiawan, Dwi H Widyantoro, and Kridanto Surendro. Feature expansion for sentiment analysis in twitter. In *2018 5th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, pages 509–513. IEEE, 2018.
- [9] Merinda Lestandy, Abdurrahim Abdurrahim, Lailis Syafa'ah, et al. Analisis sentimen tweet vaksin covid-19 menggunakan recurrent neural network dan naive bayes. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(4):802–808, 2021.
- [10] A Fauzi, EB Setiawan, and ZKA Baizal. Hoax news

- detection on twitter using term frequency inverse document frequency and support vector machine method. In *Journal of Physics: Conference Series*, volume 1192, page 012025. IOP Publishing, 2019.
- [11] Pranav Bharadwaj and Zongru Shao. Fake news detection with semantic features and text mining. *International Journal on Natural Language Computing (IJNLC) Vol, 8*, 2019.
- [12] Putu Manik Prihatini. Implementasi ekstraksi fitur pada pengolahan dokumen berbahasa indonesia. *Matrix: Jurnal Manajemen Teknologi dan Informatika*, 6(3):174, 2017.
- [13] Adi Ryansyah. *Implementasi Algoritma TF-IDF Pada Pengukuran Kesamaan Dokumen*. PhD thesis, Sekolah Tinggi Teknik Musi, 2014.
- [14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
- [15] Erwin B Setiawan, Dwi H Widyantoro, and Kridanto Surendro. Feature expansion using word embedding for tweet topic classification. In *2016 10th International Conference on Telecommunication Systems Services and Applications (TSSA)*, pages 1–5. IEEE, 2016.
- [16] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.
- [17] AS Nugroho, AB Witarto, and D Handoko. Support vector machine—teori dan aplikasi dalam bioinformatika. *Diakses pada tanggal*, 9, 2003.
- [18] Amalia Indranandita, Budi Susanto, and Antonius Rahmat. Sistem klasifikasi dan pencarian jurnal dengan menggunakan metode naive bayes dan vector space model. *Jurnal Informatika*, 4(2), 2011.
- [19] Ahmad Ashril Rizal and Siti Soraya. Multi time steps prediction dengan recurrent neural network long short term memory. *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, pages 115–124, 2018.
- [20] John A Bullinaria. Recurrent neural networks. *Neural Computation: Lecture*, 12, 2013.
- [1] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. *Advances in neural information processing systems*, 28:2980–2988, 2015.
- [2] Annisa Darmawahyuni, Siti Nurmaini, Wahyu Caesarendra, Vicko Bhayyu, M Naufal Rachmatullah, et al. Deep learning with a recurrent network structure in the sequence modeling of imbalanced data for eeg-rhythm classifier. *Algorithms*, 12(6):118, 2019.
- [3] Subari Subari and Ferdinandus Ferdinandus. Sistem informasi retrieval layanan kesehatan untuk berobat dengan metode vector space model (vsm) berbasis webgis. *SNATIKA 2015*, 3(34):1–26, 2015.