BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768

Identifikasi Kata Benda Dan Bukan Kata Benda Menggunakan *Single Layer Perceptron Network*

Yusuf Unggul Budiman[1]

[1]Fakultas Teknik dan Informatika, Program Studi Teknologi Informasi, Universitas Bina Sarana Informatika Jakarta, Indonesia
Email: [1]yusuf.yub@bsi.ac.id

Abstrak - Pada saat ini kebutuhan akan data yang sangat signifikan dengan adanya komputasi yang dapat membantu kinerja manusia. Dalam suatu data yang akan menghasilkan suatu informasi dibutukan proses identifikasi data yang berbentuk kata dengan menggunakan metode Single Layer Perceptron Network atau yang lebih dikenal sebagai jaringan saraf tiruan sehingga dalam penelitian ini memberikan akurasi yang akurat untuk suatu data yang di hasilkan pada penelitian. Penelitian ini menghasilkan eksperimen data akurasi untuk mencari kata benda dan bukan kata benda dengan proses taning dan testing yang di dapat akurasi 49 % pada traning dan 44 % pada testing. Sehingga kedepanya identifikasi untuk suatu data menjadi lebih mudah

Kata Kunci: Identifikasi kata benda, *Mechine Learning, Single Layer Perceptron Network, word2vec*

Abstract−At this time the need for data will be very significant with the presence of computing that can help human performance. In a data that will produce information, a data search process is needed in the form of words using the Single Layer Perceptron Network method or better known as a neural network so that in this study it provides accurate causation for the data generated in the study. This study resulted in experimental data accuracy to find nouns and non-nouns with a taning and testing process that got 49% accuracy on training and 44% on testing. So that in the future for a data becomes easier. .

Keywords: *Noun identification, Mechine Learning, Single Layer Perceptron Network, word2vec*

1. PENDAHULUAN

Kebutuhan akan data saat ini begitu banyak tentu dalam hal ini diperlukan seiring dengan banyaknya informasi yang dibutuhkan. Dalam mengolah data perlu mengetahui proses untuk membuat data itu sesuai dengan yang dibutuhkan. Pertumbuhan data saat ini sangat signifikan dengan adanya komputasi saat ini data adalah hal terpenting untuk kemncari informasi.

Data merupakan hal yang sangat penting pada era saat ini. Seiring dengan perkembangan dunia Teknologi Informasi yang begitu pesat maka dalam hal data yang dihasilkan akan semakin banyak. Di sisi lain pemanfaatan komputasi untuk mengolah data dengan mesin pembelajaran atau yang sering disebut sebagai *Mechine Learning* adalah hal yang sangat di perlukan. (Wissenschaftliche Dienste des deutschen Bundestags & Data, 2013)

Hal ini sangat dibutuhkan untuk mengklasifikasikan data untuk membantu tugas manusia. Salah satu metode untuk mengklasifikasikan data dengnan Single Layer Perceptron dalam hal ini dilakukan untuk mengidentifikasi kata benda dan bukan kata benda. Klasifikasi yang dihasilkan dengan menggunakan metode *Single Layer Perceptron***.**

Dalam penelitian peneliti membuat klasifikasi dari sekumpulan data yang di dalamnya terdapat kata – kata yang akan di lakukan proses pemilahan data. Pemilahan data atau klasifikasi data berdasarkan mana yang kata benda dan bukan kata benda.

Dengan menggunakan metode *algoritma Single Layer Perceptron*. Peneliti akan menguji ketepatan dari akurasi untuk mengklasifikasikan kata benda dan bukan kata benda.

Metode *single layer perceptron* adalah metode paling dasar untuk mechine learning dan paling sederhana. *Single layer perceptron* merupakan feedforward type merupakan type NN dimana neuron pada suatu layer hanya bisa berkoneksi dengan *neuron* yang berada pada layer yang berbeda. maka dalam hal ini metode *Single Layer Perceptron* Network termasuk *Supervised Learning* karena metode pembelajarannya dilakukan dengan mempelajari contoh-contoh yang diketahui *input* dan *output*-nya. Jaringan akan di-*training* dengan sekumpulan contoh-contoh yang diketahui *input* dan outputnya. Selama proses belajar tersebut jaringan akan menyesuaikan nilai bobotnya agar menghasilkan *output* yang diinginkan. Jadi semakin banyak mesin belajar maka, maka akan semakin

BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768

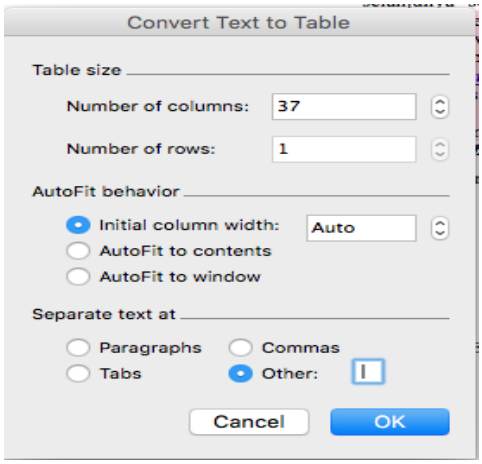
besar pula kebenaran mesin dalan menentukan mana kata benda dan bukan kata benda *Supervised learning*, dimana ada target outputnya, sehingga error dihitung dari *output* hasil perhitungan dikurangi dengan target *output*.(Ng, 2012)

2. METODE

2.1 Pengumpulan Data

Data yang di kumpulkan untuk melakukan penelitian ini adalah data yang terdapat pada jurnal-jurnal. Data ini merupakan ada awal yang di dapat dari kumpulan jurnal. Pada proses setiap jurnal di jadikan dalam satu folder untuk mempermudah sehingga nanti data tersebut dapat valid sesuai dengan data yang telah dikumpulkan. File ini merupakan file beformat pdf. Data yang telah dikumpulkan inilah yang akan menjadi acuan awal untuk penelitian untuk dijadikan dataset pengolahan data.

File yang masih berformat .pdf yang telah di kumpulkan Selanjutnya setelah file di konversi menjadi file berformat .doc dengan aplikasi berbasis web yaitu *http://convertonlinefree.com*. Maka mengkonversi file yang berformat .doc di jadikan satu kata menjadi satu tabel dengan menggunakan fitur pada Microsoft word yaitu *convert text to table*. Untuk menggunakan fitur ini dalam Microsoft Word terdapat pada Menu Table kemudian blok paragraf yang akan di konversi ke dalam tabel menjadi satu kata satu table kemudian merubah pada separete text at menjadi other dengan di isi nilai spasi. Untuk melakukan koversi setiap kata menjadi 1 tabel.



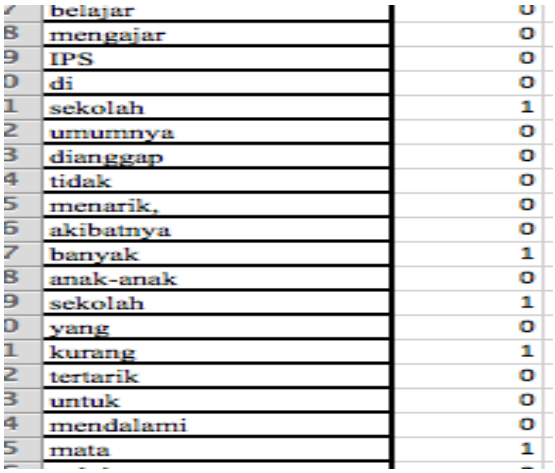
Gambar 1 *Convert to table* dengan menggunakan fitur yang terdapat di *Microsoft Word*

2.2 Pelabelan Data

Untuk membuat identifikasi kata benda dan bukan kata benda dalam penelitian ini. Di bututuhkan pelabelan dengan menggunakan klasifikasi binner yaitu dengan memberikan nilat terhadap kata benda yaitu 1 dan bukan kata benda bernilai 0. Dengan memindahkan data setiap 1 kata menjadi satu tabel ke Microsoft Excel sehingga nantinya dataset tersebut menjadi variabel Kamus untuk menentukan klasifikasi kata benda dan bukan kata benda. Pelabelan ini menjadi entitas yang unik sehinga nanti pada saat melakukan penelitian menjadi mudah

BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768

.



7	belajar	0
8	mengajar	0
9	IPS	0
0	di	0
1	sekolah	1
2	umumnya	0
3	dianggap	0
4	tidak	0
5	menarik,	0
6	akibatnya	0
7	banyak	1
8	anak-anak	0
9	sekolah	1
0	yang	0
1	kurang	1
2	tertarik	0
3	untuk	0
4	mendalami	0
5	mata	1

Gambar 2 *Labeling Dataset* dengan menggunakan *Microsoft Excel*

2.2 Konversi Word ke *Vector*

Pada proses word to vec akan menghasilkan deretan kata yang telah menjadi angka real. Setelah melakukan proses word2vec dapat memberikan gambaran dari suatu nilai dari kata dalam proses yang telah di jalankan dalam menjalakan word2vec ini dapat dijalan dengan program dengan bahasa c++ dan cli pada sistem operasi linux. File yang dihasilkan dalam proses ini berektensi dengan format .bin kemudian dari file tersebut di masukan ke dalam program matlab sehingga manjadi database di matlab. Hasil dari 2 file matlab index kata dan index nilai dari kata yang bernilai binner.

2.3 *Traning*

Pada penelitian ini proses traning di lakukan dengan program Matlab. traning memberikan gambaran data yang dilatih dengan mengacu pada label dari *dataset* yang telah dikerjakan proses ini memiliki 80 % data yang digunakan dalam proses traning.

2.4 *Testing*

Dari dataset menjadi acuan pada proses traning di proses testing ini akan dilakukan perhitungan akurasi data yang menggambarkan seberapa besar ketetpatan pada proses testing. dalam proses ini data yang digunakan adalah 20% dari dataset .

2.5 Evaluasi

Penelitian ini akan melakukan akurasi rata – rata terhadap akurasi yang dari script yang telah di jalankan pada matlab sehingga hasil yang ada di rata – ratakan untuk mengatahui akurasi dalam identifikasi kata benda dan bukan kata benda pada penelitian.

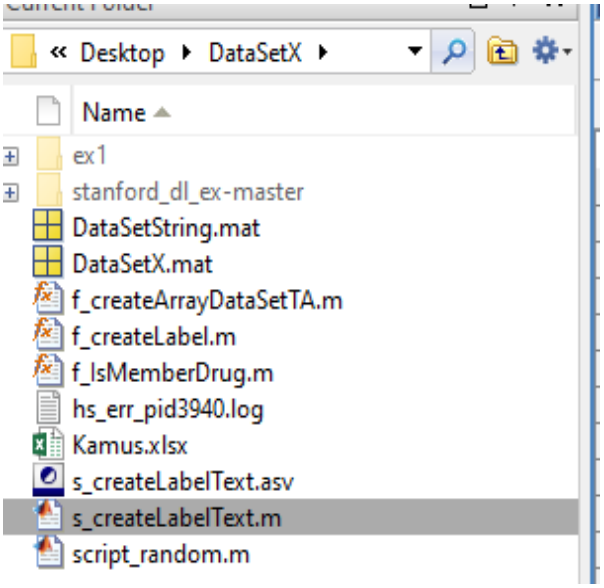
3. HASIL DAN PEMBAHASAN

Penelitian ini dibagi 3 tahapan dimana secara umum dapat di klasifikasikan yaitu *pre processing*, *processing* dan *pra processing*

3.1 Tahap *Pre Processing*

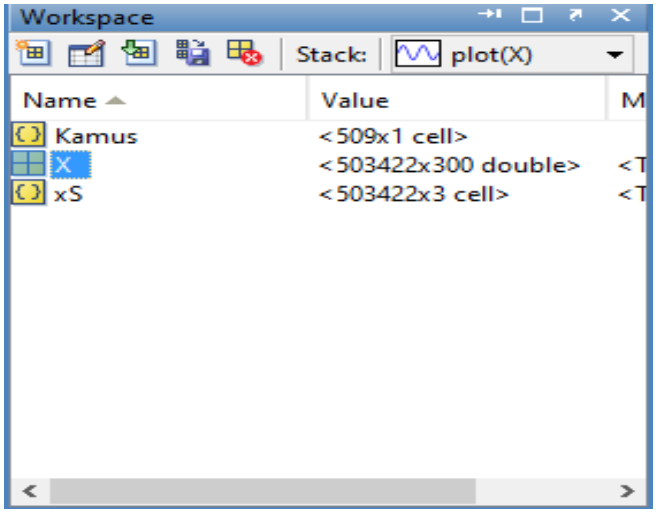
Sebelum melakukan *training* dan *testing* hal yang selanjutnya perlu dilakukan adalah pembuatan dataset yang nantinya akan digunakan pada saat training dan testing pada matlap. Berikut adalah tahapanya. File data matlab dan *script*, atau fungsi dimasukkan dalam satu folder.

BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768



Gambar 3 *File* yang di *import* di *matlab* untuk melakukan pelabelan

Untuk memulai penelitian dangan menggunakan *metode single layer perceptron* membuat *dataset* awal yang terdiri dari data variabel X yaitu kata – kata yang di kumpulkan dari jurnal, variabel Kamus yaitu data hasil pelabelan kata benda dan variabel Xs yaitu data dari variabel X yang telah di lakukan proses *word2vec*.

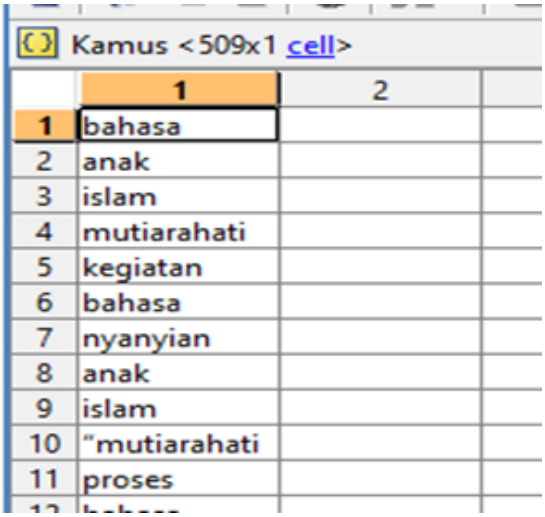


Gambar 4 *Workspace* pada matlab awal dengan variabel awal yang di gunakan

Pada gambar di atas untuk variabel Kamus berisi 509 Kata benda yang terdiri dari satu baris, untuk variabel X terdiri dari 503422 baris dengan kata yang digunakan sebagai *dataset* awal yang telah di laukan word2vec menghasilkan nilai *real* dari setiap kata , untuk variabel Xs yaitu 503422 kata yang terdiri menjadi 3 kolom kata dari proses *word2vec*.

Dalam variabel Kamus berisi kata – kata yang menjadi acuan untuk menidentifikasi kata benda untuk menajadi parameter di penelitian ini

BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768



Kamus <509x1 cell>

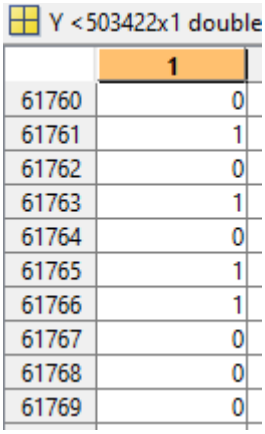
	1	2
1	bahasa	
2	anak	
3	islam	
4	mutiarahati	
5	kegiatan	
6	bahasa	
7	nyanyian	
8	anak	
9	islam	
10	"mutiarahati	
11	proses	

Gambar 5 Variabel Isi Kamus yang di gunakan sebagai acuan *labeling*

Setelah menyiapkan dataset menjalankan syntax dengan aplikasi matlab untuk membuat label identifikasi biner 0 atrau 1 dengan menngunakan *Script* dan fungsi yang digunakan untuk melakukan pelebelan ini bersumber dari *GitHub*, yang berjudul *stanford_dl_ex* yang dibagikan oleh Andrew Maas dan Sameep Tandon. *Script* yang dijalankan pada tahap ini adalah *s_createLabelText.m*.(A. Maas and S. Tandon, 2016) Berikut adalah *script* yang digunakan.

```
datasize=size(X,1);
Y=zeros(datasize,1);
for ik=1:datasize
[find_]=f_IsMemberDrug(Kamus,xS(ik,
2));
  if (find_)Y(ik)=1;
    else
    Y(ik)=0;
  end
end
```

Setelah menjalankan script akan didapat dataset Y yang memiliki satu kolom dan 503422 baris seperti pada gambar berikut.



Y <503422x1 double

	1
61760	0
61761	1
61762	0
61763	1
61764	0
61765	1
61766	1
61767	0
61768	0
61769	0

Gambar 6 *Dataset Y yang berisi nilai dari kata yang di labelkan.*

BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768

Dataset Y memiliki jumlah baris yang sama dengan *dataset xS* karena *dataset Y* merupakan *dataset xS* dengan *Kamus* yang menjadi parameter dalam penelitian. Pada baris variabel Y akan di isi nilai 0 jika tidak terdapat kata pada variabel Kamus dan sebaliknya jika terdapat kata dalam kamus maka hasilnya bernilai 1. Menjalankan script_random.m yang telah di *import* pada proses ini dengan *scrip*t :

```
n = size(Y,1);
ix=zeros(n,1);
indx=zeros(n,1);
a = randperm(n);
b = bsxfun(@plus, a,
transpose(1:20));
R = mod(b, n) + 1;
```

Akan didapatkan *matrik R* sebanyak *20* baris, dan *503422(sesuai total data)* kolom. Tabel *R* yang didapatkan memiliki 20 baris, setiap baris merupakan hasil *random index X*, maka pada setiap baris *R* memiliki urutan yang berbeda-beda berdasarkan hasil *random index X*.

3.2 Tahap *Processing*

Sebelum proses *traning* dan *testing* dilakukan masukan *train.X*, *train.Y*, *test.X***,** dan *test.Y* dari *X* dan *Y* dengan perbandingan 80% untuk *training*, dan 20% untuk testing.

Pengujian akan dilakukan sebanyak 10 kali dengan *train.X*, *train.Y*, *test.X*, dan *test.Y* yang berbeda dari *dataset X* dan *Y* berdasarkan setiap baris pada tabel *R (hasil random)*. Dari setiap pengujian yang dilakukan akan diperoleh tingkat akurasi *testing* dan *training* yang kemudian akan dirata-ratakan.

Untuk mendapatkan *train.X*, *train.Y*, *test.X*, dan *test.Y* yang berbeda dari *dataset X* dan *Y* berdasarkan setiap baris pada tabel *R*, maka dilakukan *looping* terhadap *source code* yang dipakai untuk membuat *train.X*, *train.Y*, *test.X*, dan *test.Y*. Berikut adalah *source code* untuk membuat *train.X*, *train.Y*, *test.X*, dan *test.Y* dari *dataset X* dan *Y* berdasarkan baris tabel *R*:

```
iTrain=int32(n*0.8);
train.X=X(R(1,1:iTrain),:);
train.Y=Y(R(1,1:iTrain),:);
test.X=X(R(1,iTrain+1:n),:);
test.Y=Y(R(1,iTrain+1:n),:);
```

R(1,.... merupakan baris pertama pada tabel *R*, maka untuk membuat *train.X*, *train.Y*, *test.X*, dan *test.Y* dari *dataset X* dan *Y* berdasarkan setiap baris tabel *R*, dilakukan *looping* untuk baris selanjutnya pada tabel *R*. Berikut adalah *soure code*-nya:

```
acc={};
for i=1:10;
  iTrain=int32(n*0.8);
  train.X=X(R(i,1:iTrain),:)';
  train.Y=Y(R(i,1:iTrain),:)';
  test.X=X(R(i,iTrain+1:n),:)';
  test.Y=Y(R(i,iTrain+1:n),:)';
  train.X
  =[ones(1,size(train.X,2));train.X];
  test.X = [ones(1,size(test.X,2));
  test.X];
end
```

acc adalah *variable* baru yang akan dibuat, yang merupakan tempat akurasi *training* dan *testing* dari setiap pengujian yang dilakukan.

BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768

Sedangkan *R(i,....* merupakan baris pada tabel *R*, jadi setiap kali *looping i* akan ditambah *1 (i+1)*.

Script dan fungsi yang digunakan untuk proses *training* dan *testing* bersumber dari *GitHub*, yang berjudul *stanford_dl_ex* yang dibagikan oleh Andrew Maas dan Sameep Tandon(A. Maas and S. Tandon, 2016), dengan nama *script ex1b_logreg.m*.

3.2.1. *Training*

Seperti yang telah dijelaskan sebelumnya, *single layer perceptron network* termasuk *supervised learning*, artinya metode pembelajarannya dilakukan dengan mempelajari contoh-contoh yang diketahui *input* dan *output*-nya. Jaringan akan di-*training* dengan sekumpulan contoh-contoh yang diketahui *input* dan outputnya. Selama proses belajar tersebut jaringan akan menyesuaikan nilai bobotnya agar menghasilkan *output* yang diinginkan. Jadi semakin banyak mesin belajar maka, maka akan semakin besar pula kebenaran mesin dalan menentukan mana kata benda dan bukan kata bendac. Berikut *source code* yang digunakan pada saat *training*:

```
theta = rand(n,1)*0.001;
theta=minFunc(@logistic_regression_vec,
theta, options, train.X, train.Y)
```

*theta = rand(n,1)*0.001; adalah* Inisialisasi *Theta* dengan *random.* Sedangkan baris selanjutnya proses *learning* yang menghasilkan nilai θ.

Masukan dari data training berupa variable *X* dan *Y* dan train.*X, train.Y* yang sebelumnya dimasukan. Setelah dilakukan proses *training* didapatkan nilai dari θ yang akan digunakan untuk memprediksi variable *Y* yang merupakan target pada proses *testing*.

ArrAkurasi, ArraAkurasi, Kamus, R, X, Y, a, acc, accuracy, b, binary_digits, datasize, find_, i, iTrain, ik, indx, ix, m, n, options, test, theta, traccuracy, train, tstaccuracy, xS

3.2.2. *Testing*

Pada tahap *testing source code* yang digunakan adalah sebagai berikut:

```
sigmoid(theta(:)'*X(:,1))
theta(:)'*X(:,1)
ytest=sigmoid(theta'*X) > 0.5;
```

Proses *testing* ini membuat nilai θ menjadi kisaran [0,1].

3.3 Tahap *Pra Processing*

Pada tahapan ini mendapatkan akurasi dari masing masing proses dengan script berikut :

```
accuracy=(ytest==Y)
```

Potongan *script* diatas digunakan untuk membandingkan hasil y dengan testy dengan acuan dari keluaran variabel Y yang masing – masing bernilai benar.

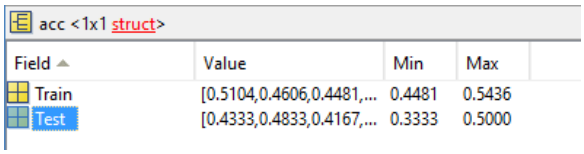
```
sum(accuracy)
```

SUM pada *script* ini merupakan proses menghitung akurasi yang didapat pada saat proses *testing*.

```
sum(accuracy)/size(y,2)
```

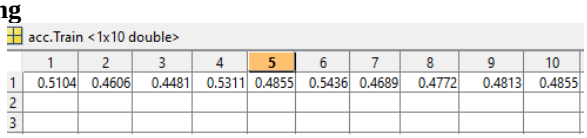
BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768

Pada *script* diatas melakukan pengungurangan akurasi menjadi persen (%).
Pengujian dilakukan sebanyak 10 kali dari *train.X*, *train.Y*, *test.X*, *test.Y* yang berbeda dari *dataset X* dan *Y* berdasarkan setiap baris *R* hasil *random* sebelumnya. Berikut adalah hasilnya:



Gambar 7. test

3.3.1 Hasil Traning

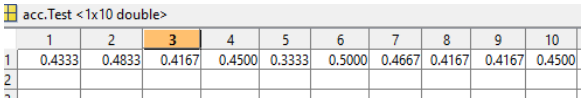


Gambar 8. Hasil Akurasi Train dengan 10 kali percobaan

Table 1 Hasil Traning dan Akurasi

Traning	Akurasi (%)
1	51%
2	46%
3	45%
4	53%
5	49%
6	54%
7	47%
8	48%
9	48%
10	49%

3.3.2 Hasil Testing



Gambar 8. Hasil Akurasi Test dengan percobaan 10 kali.

Table 2 Hasil Testing Dan Akurasi

Testing	Akurasi (%)
1	43%
2	48%
3	42%
4	45%
5	33%
6	50%

BULLET : Jurnal Multidisiplin Ilmu
Volume 01, No. 5 (Oktober – November) 2022
ISSN 2829-2049 (media online)
Hal 759-768

7	47%
8	42%
9	42%
10	45%

Untuk mengetahui rata-rata seberapa akurat *algoritma single layer peceptron* pada saat *training* dan *testing* adalah dengan menjumlahkan semua hasil *training* dan *testing* lalu dibagi 10.
Setelah mengetahui hasil masing – masing akurasi dari training dan testing. Selanjutnya adalah mencari nilai rata rata dari masing masing dengan menggunakan fungsi Average pada tabel berikut :

Table 3 Rata Rata Traning dan Testing

Exp	Training	Testing
1	51%	43%
2	46%	48%
3	45%	42%
4	53%	45%
5	49%	33%
6	54%	50%
7	47%	47%
8	48%	42%
9	48%	42%
10	49%	45%
Rata - Rata	**49%**	**44%**

4. KESIMPULAN

Maka dapat di simpulkan pada penelitian dengan menggunakan menggunakan *Single Layer Perceptron* untuk identifikasi kata benda dan bukan kata benda Dengan menggunakan metode ini untuk mengidentifikasi berjalan dengan baik sesuai dengan yang diharapkan. Dalam penelitian ini yang manjadi acuan akhir dari penelitian ini adalah hasil *traning* sebanyak 80 % dan testing 20 % sehingga mendapatkan hasil rata – rata *traning* 49 % dan testing 44 %. Sehingga dalam penelitian ini memberikan hasil yang cukup baik.

Untuk memberikan akurasi yang lebih baik harus di butuhkan dataset yang lebih banyak agar proses traning dan testing mendapatkan hasil yang lebih baik. Dalam penelitian selanjutnya agar bisa lebih baik lagi dengan tidak hanya memberikan akurasinyanya saja namun juga dapat meng identifikasi kata benda dan bukan kata benda yang mudah dengan menggbungkan beberapa fitur yang ada dalam dunia IT.

REFERENCES

A. Maas and S. Tandon. (2016). *stanford_dl_ex," https://github.com.* Online.

Anwar, A., & Jatmiko, D. D. (2014). *Analisis dan implementasi metode single layer perceptron pada data mining penerimaan mahasiswa baru jalur jppan*. *1*(single layer perceptron).

Awodele, O., & Jegede, O. (2009). Neural Networks and its Application in Engineering. *Proceedings of Informing Science & IT Education Conference (InSITE) 2009*, *April 2016*.

Farsiah, L., Fuadi Abidin, T., & Munadi, K. (2010). *Klasifikasi Gambar berwarna menggunkan K-Nearest Neighbor dan Support Vector Machine*. 1–5.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 142–150. https://doi.org/978-1-932432-87-9

Martiana, E. (2010). Introduction to Machine Learning Entin Martiana Learning from Data. *Article*, 2.

Ng, A. (2012). Supervised learning. *Machine Learning*, 1–30.

Paper, C. (2016). *Improving Behavior Prediction Accuracy by Using Machine Learning for Agent Based Simulation*. *March*.

Riadi, J., & Nurmahaludin. (2009). *Aplikasi Jaringan Syaraf Tiruan Multi Layer Perceptron Pada Aplikasi Prakiraan Cuaca*. *1*.

Wissenschaftliche Dienste des deutschen Bundestags, & Data, B. (2013). Big Data. *Aktueller Begriff*, *37*(13), 23–26.