Klasifikasi Berita Politik Menggunakan Algoritma *K-nearst Neighbor*

(Classification of Political News Content using K-Nearest Neighbor)

Difari Afreyna Fauziah, Achmad Maududie, Ifrina Nuritha
Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Jember (UNEJ)
Jln. Kalimantan 37, Jember 68121
E-mail: difariafreyna0@gmail.com

Abstrak

Klasifikasi konten berita politik menggunakan algoritma *K-Nearest Neighbor* merupakan suatu proses untuk mengklasifikasikan berita politik ke dalam tiga subkategori yang lebih spesifik yaitu pilkada, UU ORMAS dan *reshuffle* kabinet. Algoritma yang digunakan dalam penelitian ini adalah algoritma *K-Nearest Neighbor.* Algoritma *K-Nearest Neighbor* merupakan suatu pendekatan klasifikasi yang mencari semua data *training* yang paling relatif mirip atau memiliki jarak yang paling dekat dengan data *testing.* Algoritma ini dipilih karena *K-Nearest Neighbor* merupakan algoritma yang sederhana dengan mencari kategori mayoritas sebanyak nilai K yang telah ditentukan sebelumnya. nilai K yang digunakan pada penelitian ini adalah K=3, K=5, K=7 dan K=9. Mekanisme dari sistem klasifikasi konten berita ini dimulai dengan tahap *preprocessing*. Berita politik yang dimasukkan kedalam sistem akan melewati empat tahap *preprocessing* yaitu *case folding, tokenizing, stopword* dan *stemming*. Tahap selanjutnya yaitu tahap pembobotan *term*. Pembobotan atau *term weighting* merupakan proses mendapatkan nilai *term* yang berhasil diekstrak dari proses sebelumnya yaitu proses *preprocessing*. Algoritma yang digunakan untuk tahap pembobotan pada penelitian ini adalah algoritma TFIDF. Setelah didapatkan nilai dari bobot *term*, kemudian dicari nilai jarak antar dokumen menggunakan algoritma *cosine similarity.* Langkah berikutnya adalah melakukan pengurutan data dalam data *training* berdasarkan hasil perhitungan nilai jarak. Selanjutnya, dari hasil pengurutan tersebut diambil sejumlah K data yang memiliki nilai kedekatan. Tujuan dari penelitian ini adalah sistem mampu mengimplementasikan algoritma KNN pada dokumen yang memiliki *similarity* yang tinggi. Pada penelitian ini dilakukan 3 pengujian dengan tiga variasi *dataset* yang berbeda dengan empat nilai K. Hasil akurasi yang terbaik didapatkan ketika sistem menggunakan nilai K=9 yang menunjukkan nilai *precision* sebesar 100%, *recall* sebesar 100% dan nilai *f-measure* sebesar 100%.

Kata Kunci: klasifikasi, algoritma *K-Nearest Neighbor, TFIDF, cosine similarity, confusion matrix* .

Abstract

The classification of political news content using the K-Nearest Neighbor algorithm is a process for classifying political news into three more specific categories: elections, the ORMAS Act and the cabinet reshuffle. The algorithm used in this research is K-Nearest Neighbor algorithm. The K-Nearest Neighbor algorithm is a classification approach that searches all the most relatively similar training data or has the closest distance to the data testing. The algorithm is chosen because K-Nearest Neighbor is a simple algorithm by searching for the majority category as much as the predetermined value of K. K values used in this study were K = 3, K = 5, K = 7 and K = 9. The mechanism of this news content classification system begins with the preprocessing stage. Political news entered into the system will pass through four preprocessing stages: case folding, tokenizing, stopword and stemming. The next stage is the term weighting stage. Weighting or term weighting is the process of obtaining the term value that has been successfully extracted from the previous process of preprocessing process. The algorithm used for weighting phase in this research is TFIDF algorithm. Having obtained the value of the term weight, then searched the distance between documents using cosine similarity algorithm. The next step is to sort the data in training data based on the calculation of distance value. Furthermore, from the sorting results are taken a number of K data that have value proximity. The purpose of this research is the system is able to implement KNN algorithm in documents that have a high similarity. In this study three tests were performed with three different dataset variations with four K values. The best accuracy result was obtained when the system used K = 9 value which showed 100% precision value, 100% recall and 100% f-measure value.

Keywords: *classification, K-Nearest Neighbor algorithm, TFIDF, cosine similarity, confusion matrix* .

PENDAHULUAN

Berita merupakan laporan yang berisi informasi tentang suatu peristiwa, opini, kecenderungan, situasi, kondisi, interpretasi yang penting, menarik, masih baru dan harus secepatnya disampaikan kepada khalayak (Rani, 2013). Berita yang berisi informasi bersifat akurat, relevan dan konsisten yang dapat memberikan pengetahuan bagi penerimanya. Biasanya, berita disajikan dalam bentuk cetak, siaran, internet atau dari mulut ke mulut kepada orang ketiga atau orang banyak.

Di era perkembangan teknologi ini, berita dapat diakses melalui media internet yang disajikan dalam portal-portal berita seperti kompas, detik, *vivanews*, liputan6, *tribunnews* dan portal berita lainnya. Berita yang disajikan biasanya dikelompokkan dalam beberapa kategori berita seperti politik, kesehatan, olahraga dan teknologi. Salah satu

kategori berita yang paling banyak diakses oleh pembaca yaitu berita dengan kategori politik. Hal ini dapat dilihat dari hasil survey yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) pada tahun 2017 yang memberikan gambaran bahwa 36.94% dari 143,26 juta pengguna Internet di Indonesia memanfaatkan Internet untuk mengakses berita politik.

Meskipun beberapa portal berita *online* telah mengelompokkan berdasarkan kategori-kategori berita, namun pengelompokan tersebut masih bersifat umum. Salah satunya portal berita detik.com yang terdiri dari kategori *DetikNews*, *DetikFinance, DetikHot, DetikSport, DetikOto, DetikTravel, DetikFood, DetikHealt,* Wolipop dan Indeks. Hal ini mengharuskan pembaca berita yang ingin mencari berita yang lebih spesifik sebagai contoh berita pilkada harus melakukan pencarian secara manual dengan menelusuri dan membaca satu persatu berita yang diunggah dalam setiap portalnya yang membuat proses pencarian tersebut membutuhkan waktu yang cenderung lama.

Untuk mempermudah dalam pengelompokkan konten berita, sebagai langkah awal peneliti mencoba membangun sistem yang dapat mengenali kategori berita secara otomatis. Sistem ini didasarkan pada teknik klasifikasi sebagai salah satu teknik pengelompokan data. Disamping itu, dalam proses klasifikasinya, sistem ini menerapkan algoritma *K-Nearest Neighbor*. Berdasarkan penelitian yang dilakukan oleh (Ahsanti,2006) (Palinoan, 2014) ketika algoritma ini diterapkan pada objek dengan *similarity* yang rendah mendapatkan nilai akurasi rata – rata diatas 93%. Dengan demikian, disamping membangun sistem klasifikasi konten berita, peneliti juga mencoba meneliti seberapa akurat algoritma *K-Nearest Neighbor* ketika diterapkan untuk melakukan klasifikasi terhadap kategori yang lebih detail (memiliki *similarity* yang tinggi). Pada penelitian ini, objek dengan *similarity* yang tinggi yang digunakan adalah berita dengan kategori politik.

METODE PENELITIAN

1. Pengumpulan *Dataset*

Dokumen yang digunakan dalam klasifikasi ini dibagi menjadi dua bagian, yaitu dokumen yang berfungsi sebagai data *training* dan dokumen yang berfungsi sebagai data *testing* yang akan digunakan sebagai uji coba terhadap data *training*. Data yang digunakan diperoleh dari beberapa portal bgerita *online* yaitu detik, sindonews, liputan6 dan kompas. Kategori yang digunakan dalam penelitian ini adalah pilkada, *reshuffle* kabinet dan UU ORMAS. Sebuah penelitian mengatakan apabila jumlah data *training* berpengaruh terhadap peforma sistem klasifikasi yaitu semakin banyak jumlah data *training* maka semakin besar peluang data *test* terklasifikasi secara benar (Puspitasari & Santoso, 2018). Semakin banyak data *training* yang digunakan maka jumlah variasi *term* unik juga semakin banyak. Penggunaan *dataset* secara lengkap pada penelitian ini dapat dilihat pada lampiran E. Pada penelitian ini untuk pembagian *dataset* dibagi menjadi 3 variasi *dataset*. Pembagian jumlah data *training* dan data *testing* yang digunakan pada penelitian ini dapat dilihat pada Tabel 1.

Tabel 1. Jumlah Dataset

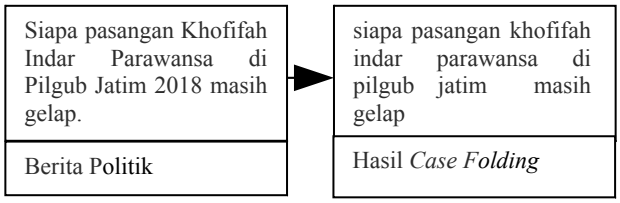
No	Pembagian Dataset	Data *Training*	Data *Testing*
1	Dataset 1	150	150
2	Dataset 2	210	90
3	Dataset 3	270	30

Untuk pembagian *dataset* pada Tabel 3.1, dapat dilihat untuk total keseluruhan *dataset* yang digunakan dalam penelitian ini sebanyak 300 berita. Pembagian data *training* dan data *testing* dibagi secara merata setiap kategorinya. Sebagai contoh pada *dataset* 1, Data *training* yang digunakan sebanyak 150 dokumen berita yang dibagi secara merata sebanyak 50 dokumen yaitu 50 berita pilkada, 50 berita *reshuffle* kabinet dan 50 berita UU ORMAS. Sedangkan jumlah data *testing* yang digunakan sebanyak 150 berita dimana masing-masing kategori 50 yaitu 50 berita pilkada, 50 berita *reshuffle* kabinet dan 50 berita UU ORMAS.

2. *Preprocessing*

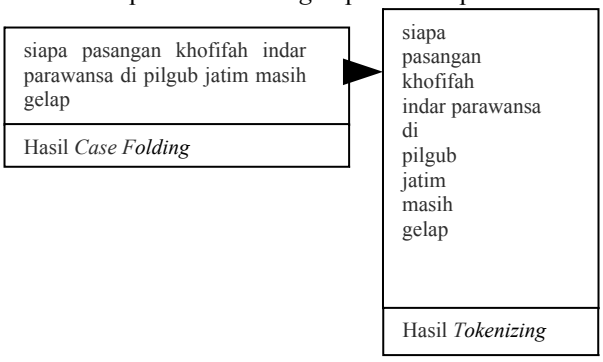
Pada tahapan ini berita politik yang telah didapatkan dari beberapa sumber portal berita akan melalui proses *preprocessing.*Berikut adalah penjelasan masing-masing tahapan *preprocessing*:

a. *Case folding* merupakan sebuah proses awal untuk mengubah semua karakter huruf pada dokumen menjadi huruf kecil. Pada proses ini huruf yang akan diproses hanya huruf *alphabet* yaitu huruf "a" hingga "z" selain huruf tersebut seperti karakter tanda baca akan dihilangkan dan dianggap sebagai *delimiter*. Contoh proses case folding dapat dilihat pada Gambar 1.



Gambar 1. Proses *Case Folding*

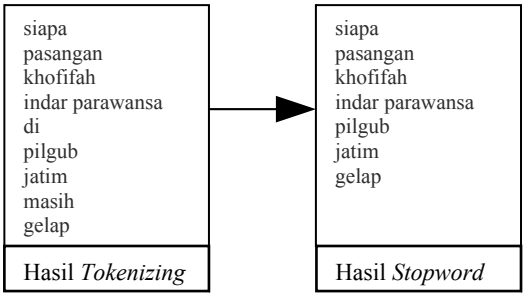
b. *Tokenizing* merupakan Tokenizing merupakan tahap proses pemotongan kumpulan kata menjadi sebuah token. Dalam tahap ini spasi digunakan sebagai pemisah antar kata. Contoh proses *tokenizing* dapat dilihat pada Gambar 2.



Gambar 2. Proses *Tokenizing*

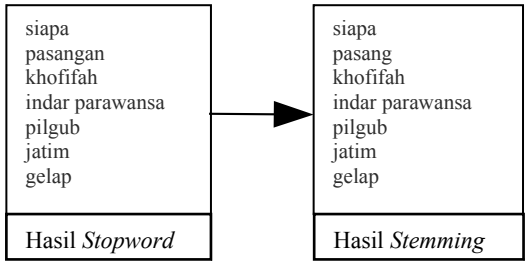
c. *Stopwords*, merupakan sekumpulan kata yang tidak berhubungan *(irrelevant)* dengan subjek utama yang dimaksud, meskipun kata tersebut sering muncul di dalam

data yang digunakan (Setiawan, Kurniawan, & Handiwidjojo, 2013). Kata kata yang biasanya masuk dalam *stopword* yaitu jenis kata sambung, imbuhan dan lain sebagainya. Apabila kata *stopword* dihilangkan maka tidak akan menghilangkan makna dari data atau dokumen teks. Untuk contoh proses *stopwords* dapat dilihat pada Gambar 3.



Gambar 3. Proses *Stopword*

d. Proses *stemming* tahap mengubah kata pada dokumen menjadi kata dasarnya dengan menghilangkan imbuhan atau dengan mengubah kata kerja menjadi kata benda. Pada teks Bahasa Indonesia untuk menghilangkan imbuhan dengan cara menghilangkan selain akhiran (s*uffixes*) adalah awalan (*prefixes*), sisipan (*infixes*), dan kombinasi awalan dan akhiran (*confixes*). Algoritma yang dapat digunakan untuk melakukan tahap stemming yaitu algoritma Porter *Stemmer*, algoritma Nazief-Adriani, algoritma *Confix Stripping* (CS), algoritma *Enhanced Confix Stripping* (ECS). Untuk contoh proses *stemming* dalam penelitian ini dapat dilihat pada Gambar 4.



Gambar 4 Proses *Stemming*

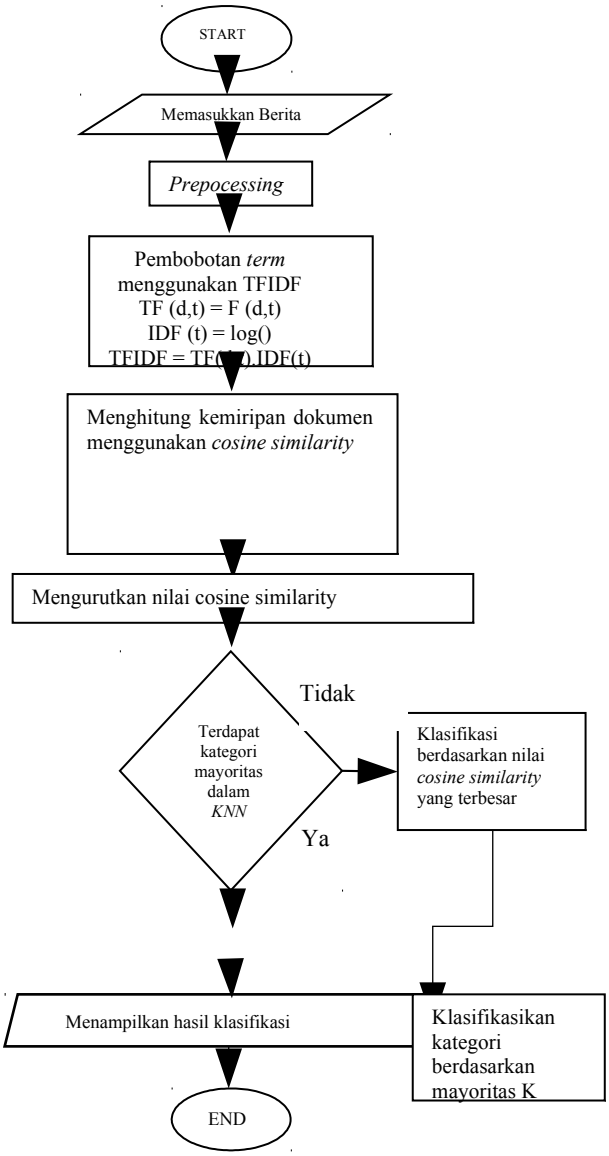
3. Klasifikasi *K-Nearest Neighbor*

Algoritma *K-Nearest Neighbor* adalah sebuah algoritma untuk melakukan klasifikasi terhadap data baru berdasarkan data pembelajaran yang jaraknya paling dekat dengan data tersebut. Untuk menghitung nilai jarak antar dokumen setelah proses *preprocessing* adalah tahap pembobotan. Pada penelitian ini untuk menghitung pembobotan menggunakan algoritma TFIDF. Setelah didapatkan nilai bobot dokumen kemudian dilanjutkan dengan menghitung nilai jarak antar dokumen. Salah satu algoritma perhitungan jarak yang sering digunakan untuk klasifikasi dokumen adalah *cosine similarity*.

Langkah berikutnya adalah melakukan pengurutan data dalam data *training* berdasarkan hasil perhitungan nilai jarak. Selanjutnya, dari hasil pengurutan tersebut diambil kategori mayoritas sejumlah K data yang memiliki nilai kedekatan. Nilai K sudah ditentukan atau diasumsikan sebelumnya sesuai dengan syarat (Rivki & Bachtiar, 2017) yaitu

1. K harus lebih dari satu,
2. Nilai K adalah nilai ganjil (karena jika diambil K adalah bilang genap akan ada kemungkinan hasil klasifikasi sulit ditentukan karena masing-masing kelas bernilai sama.
3. Nilai K lebih dari jumlah kelas.
4. Nilai K tidak melebihi jumlah data *training*.

Untuk mencari kategori mayoritas sebanyak nilai K, pada tahap ini akan dilakukan pengecekan apabila terdapat kategori mayoritas maka kategori akan langsung terklasifikasikan.Namun apabila tidak terdapat kategori mayoritas maka akan dipilih kategori dengan nilai kemiripan (*cosine similarity*) yang paling tinggi. Untuk melihat alur dari klasifikasi konten berita mengggunakan algoritma KNN ini dapat dilihat pada Gambar 5.



Gambar 5 Alur Klasifikasi KNN

5. Pengujian Kualitas Algoritma

Pengujian kualitas dilakukan untuk mengetahui kinerja dari algoritma klasifikasi yang telah diterapkan. Ada beberapa cara untuk mengukur kinerja algoritma klasifikasi tiga diantaranya adalah *precision, recall* dan *f-measure*.

Untuk mengukur kinerja algoritma dapat menggunakan Tabel *confusion matrix multiclass* yang dapat dilihat pada Tabel 2.

Tabel 2 *Confusion Matrix Multiclass*

		Prediksi		
Aktual		*Class 1*	*Class 2*	*Class 3*
	Class 1	TP	E_{12}	E_{13}
	Class 2	E_{21}	TP	E_{23}
	Class 3	E_{31}	E_{32}	TP

1. TP (*True Positive*) menunjukkan jumlah data *testing* yang diklasifikasikan sistem sesuai dengan kategori yang sesungguhnya.
2. FP (*False Positive*) menunjukkan jumlah data *testing* pada kolom yang sesuai kelasnya namun tidak termasuk TP. Contoh untuk FP *Class* 1 = $E_{21} + E_{31}$
3. FN (*False Negative*) menunjukkan jumlah data *testing* pada baris yang sesuai kelasnya namun tidak termasuk TP. Contoh untuk FN *Class* 1 = $E_{12} + E_{13}$ TN (*True Negative*) menunjukkan jumlah data *testing* pada semua kolom dan baris namun tidak termasuk kolom dan baris kelas itu. Contoh untuk TN *Class* 1 = TP *Class* 2 + E_{23} + E_{32} + TP *Class* 3

Precision adalah keakuratan hasil klasifikasi dari seluruh dokumen oleh sistem, sehingga dapat diketahui apakah kategori data yang diklasifikasi sesuai dengan kategori yang sebenarnya. *Precision* dihitung dari jumlah pengenalan data yang bernilai benar oleh sistem dibagi dengan jumlah keseluruhan pengenalan data yang dilakukan pada sistem yang ditunjukkan dengan rumus (Ayu Puspitasari & Santoso, 2018).

$$Precision\ (\mathrm{R}) = \frac{TP + FP}{TP}$$

Keterangan :
TP = *True Positive*
FP = *False Positive*

Recall menunjukkan tingkat keberhasilan sistem dalam mengenali suatu kategori. *Recall* dihitung dari jumlah pengenalan data yang bernilai benar oleh sistem dibagi dengan jumlah data yang seharusnya dapat dikenali sistem yang ditunjukkan dengan rumus (Ayu Puspitasari et al., 2018).

$$Recall\ (\mathrm{R}) = \frac{TP + FN}{TP}$$

Keterangan :
TP = *True Positive*
FN = *False Negative*

F-measure merupakan gambaran pengaruh relatif antara *precision* dan *recall* atau disebut *harmonic mean.* Peforma algoritma yang digunakan dapat disimpulkan dari nilai *F-measure. F-measure* dapat dihitung seperti yang ditunjukkan dengan rumus (Ayu Puspitasari et al., 2018).

$$F_1 = \frac{P + R}{2PR}$$

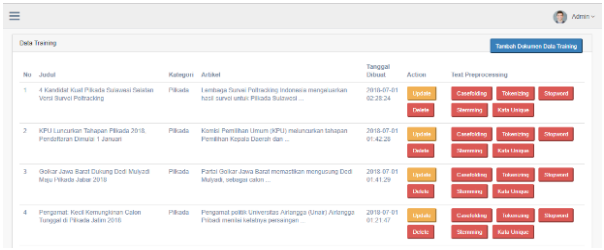
Keterangan :
P = *Precision*
R = *Recall*.

HASIL DAN PEMBAHASAN

Hasil Pembangunan

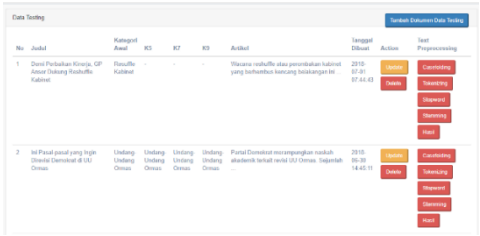
Sistem klasifikasi konten berita ini dibangun dengan 8 fitur yaitu melihat grafik jumlah data *training,* melihat grafik jumlah data *testing,* melihat grafik keakuratan nilai K, mengelola data *training,* mengelola data *testing,* mengelola data *stopword* yang dapat diakses oleh admin dan fitur klasifikasi konten berita yang dapat diakses oleh *guest.*

Halaman fitur data *training* merupakan halaman yang dapat melihat daftar data *training,* menambah, mengubah, menghapus data *training* dan juga dapat melihat hasil *preprocessing*. Halaman fitur data *training* dapat dilihat pada Gambar 6.



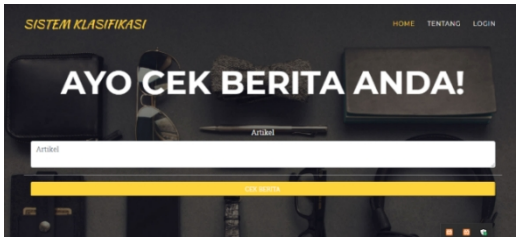
Gambar 6. Halaman Data *Training*

Halaman fitur data *testing* merupakan halaman yang dapat melihat daftar data *testing,* menambah, mengubah, menghapus data *testing* dan juga dapat melihat hasil *preprocessing*. Halaman fitur data *testing* dapat dilihat pada Gambar 7.



Gambar 7. Halaman Mengelola Data *Testing*

Halaman fitur klasifikasi konten berita merupakan halaman untuk *guest* melakukan proses klasifikasi untuk mengecek kategori dari berita politik. Halaman klasifikasi konten berita dapat dilihat pada Gambar 8.



Gambar 8. Halaman Klasifikasi Konten Berita

Hasil Pengujian Kualitas Algoritma *K-Nearest Neighbor*

Pengujian peforma algoritma yang digunakan peneliti di dasarkan pada perhitungan *confusion matrix* dengan menghitung nilai *precision*, *recall* dan *f-measure*. Pengujian kualitas algoritma pada penelitian ini dilakukan sebanyak

tiga kali pengujian dengan menggunakan pembagian *dataset* yang bervariasi seperti yang telah dijelaskan pada bab 3 mengenai pengumpulan *dataset* serta menggunakan sebanyak empat nilai K yaitu k=3, k=5, =7 dan k=9.

a. Pengujian 1 (Pembagian Data *Training* dan Data *Testing* dengan persentase 50% dan 50%)

Pengujian pertama dilakukan menggunakan *dataset* sebanyak 300 berita politik dengan pembagian jumlah data *training* sebanyak 150 berita dan jumlah data *testing* sebanyak 150 berita. Pengujian ini dilakukan dengan menggunakan empat nilai k. Pada Tabel 3-6 merupakan hasil prediksi dari sistem klasifikasi konten berita politik.

Tabel 3. Tabel *Confusion Matrix* Menggunakan K=3

		Prediksi		
	K=3	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	39	0	11
	UU ORMAS	2	46	2
	Pilkada	2	0	48

Tabel 4. Tabel *Confusion Matrix* Menggunakan K=5

		Prediksi		
	K=5	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	49	0	1
	UU ORMAS	2	48	0
	Pilkada	1	0	49

Tabel 5. Tabel *Confusion Matrix* Menggunakan K=7

		Prediksi		
	K=7	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	50	0	0
	UU ORMAS	1	49	0
	Pilkada	0	0	50

Tabel 6. Tabel *Confusion Matrix* Menggunakan K=9

		Prediksi		
	K=9	***Reshuffle* Kabinet**	**UU ORMAS**	**Pilkada**
Aktual	***Reshuffle* Kabinet**	**50**	**0**	**0**
	UU ORMAS	**0**	**50**	**0**
	Pilkada	**0**	**1**	**49**

Dapat dilihat tabel uji K merupakan tabel hasil prediksi sistem klasifikasi konten berita politik menggunakan tabel *confusion matrix*, maka tahap selanjutnya dilakukan perhitungan dari nilai *precision, recall* dan *f-measure* untuk mengetahui peforma dari nilai k terbaik yang digunakan untuk sistem klasifikasi konten berita politik. Untuk perhitungan nilai akurasi dapat dilihat pada tabel 7-10.

Tabel 7. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=3

Perhitungan	**Nilai**
Precision	$\frac{39}{39+4} + \frac{46}{46+0} + \frac{48}{48+13}$ x 100 % = 89,79%
Recall	$\frac{39}{39+11} + \frac{46}{46+4} + \frac{48}{48+2}$ x 100% = 88,66%
F-measure	$\frac{0,8979+0,8866}{2\ x\ 0,8979\ x\ 0,8866}$ x 100 % = 89,22%

Tabel 8. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=5

Perhitungan	**Nilai**
Precision	$\frac{49}{49+3} + \frac{48}{48+0} + \frac{49}{49+1}$ x 100 % = 97,41%
Recall	$\frac{49}{49+1} + \frac{48}{48+2} + \frac{49}{49+1}$ x 100% = 97,33%
F-measure	$\frac{0,9741+0,9733}{2\ x\ 0,9741\ x\ 0,9733}$ x 100 % = 97,37%

Tabel 9. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=7

Perhitungan	**Nilai**
Precision	$\frac{50}{50+1} + \frac{49}{49+0} + \frac{50}{50+0}$ x 100 % = 99,34%
Recall	$\frac{50}{50+0} + \frac{49}{49+1} + \frac{50}{50+0}$ x 100% = 99,33%
F-measure	$\frac{0,9934+0,9933}{2\ x\ 0,9934\ x\ 0,9933}$ x 100 % = 99,33%

Tabel 10. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=9

Perhitungan	**Nilai**
Precision	$\frac{50}{50+1} + \frac{50}{50+0} + \frac{49}{49+0}$ x 100 % = 99,34%
Recall	$\frac{50}{50+0} + \frac{50}{50+0} + \frac{49}{49+1}$ x 100% = 99,33%
F-measure	$\frac{0,9934+0,9933}{2\ x\ 0,9934\ x\ 0,9933}$ x 100 % = 99,33

b. Pengujian 2 (Pembagian Data *Training* dan Data *Testing* dengan persentase 70% dan 30%)

Pengujian pertama dilakukan menggunakan *dataset*

sebanyak 300 berita politik dengan pembagian jumlah data *training* sebanyak 210 berita dan jumlah data *testing* sebanyak 90 berita. Pengujian ini dilakukan dengan menggunakan empat nilai k. Pada Tabel 11-14 merupakan hasil prediksi dari sistem klasifikasi konten berita politik.

Tabel 11. Tabel *Confusion Matrix* Menggunakan K=3

		Prediksi		
	K=3	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	29	0	1
	UU ORMAS	0	29	1
	Pilkada	2	0	28

Tabel 12. Tabel *Confusion Matrix* Menggunakan K=5

		Prediksi		
	K=5	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	30	0	0
	UU ORMAS	0	29	1
	Pilkada	1	0	29

Tabel 13. Tabel *Confusion Matrix* Menggunakan K=7

		Prediksi		
	K=7	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	28	2	0
	UU ORMAS	0	30	0
	Pilkada	1	0	29

Tabel 14. Tabel *Confusion Matrix* Menggunakan K=9

		Prediksi		
	K=9	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	30	0	0
	UU ORMAS	0	30	0
	Pilkada	0	0	30

Dapat dilihat tabel uji K merupakan tabel hasil prediksi sistem klasifikasi konten berita politik menggunakan tabel *confusion matrix*, maka tahap selanjutnya dilakukan perhitungan dari nilai *precision, recall* dan *f-measure* untuk mengetahui peforma dari nilai k terbaik yang digunakan untuk sistem klasifikasi konten berita politik. Untuk perhitungan nilai akurasi dapat dilihat pada tabel 15-18.

Tabel 15. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=3

Perhitungan	**Nilai**
Precision	$\frac{29}{29+2} + \frac{29}{29+0} + \frac{28}{28+2}$ x 100 % = 95,62%
Recall	$\frac{29}{29+1} + \frac{29}{29+1} + \frac{28}{28+2}$ x 100% = 93,33%
F-measure	$\frac{0,9562+0,9333}{2\ x\ 0,9562\ x\ 0,9333}$ x 100 % = 95,59%

Tabel 16. Perhitungan Nilai Precision, Recall dan F-measure k=5

Perhitungan	**Nilai**
Precision	$\frac{30}{30+1} + \frac{29}{29+0} + \frac{29}{29+1}$ x 100 % = 97,81%
Recall	$\frac{30}{30+0} + \frac{29}{29+1} + \frac{29}{29+1}$ x 100% = 97,77%
F-measure	$\frac{0,9781+0,9777}{2\ x\ 0,9781\ x\ 0,9777}$ x 100 % = 97,79%

Tabel 17. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=7

Perhitungan	**Nilai**
Precision	$\frac{28}{28+1} + \frac{30}{30+2} + \frac{29}{29+0}$ x 100 % = 96,76%
Recall	$\frac{28}{28+2} + \frac{30}{30+2} + \frac{29}{29+1}$ x 100% = 96,66%
F-measure	$\frac{0,9676+0,9666}{2\ x\ 0,9676\ x\ 0,9666}$ x 100 % = 96,71%

Tabel 18. Perhitungan Nilai Precision, Recall dan F-measure k=9

Perhitungan	**Nilai**
Precision	$\frac{30}{30+0} + \frac{30}{30+0} + \frac{30}{30+0}$ x 100 % = 100%
Recall	$\frac{30}{30+0} + \frac{30}{30+0} + \frac{30}{30+0}$ x 100% = 100%
F-measure	$\frac{1+1}{2\ x\ 1\ x\ 1}$ x 100 % = 100%

c. Pengujian 3 (Pembagian Data *Training* dan Data *Testing* dengan persentase 90% dan 10%)

Pengujian pertama dilakukan menggunakan *dataset* sebanyak 300 berita politik dengan pembagian jumlah data *training* sebanyak 270 berita dan jumlah data *testing* sebanyak 30 berita. Pengujian ini dilakukan dengan menggunakan empat nilai k. Pada Tabel 19-22 merupakan hasil prediksi dasi sitem klasifikasi konten berita politik.

Tabel 19. Tabel *Confusion Matrix* Menggunakan K=3

		Prediksi		
	K=3	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	9	0	1
	UU ORMAS	0	9	1
	Pilkada	0	0	10

Tabel 20. Tabel *Confusion Matrix* Menggunakan K=5

		Prediksi		
	K=5	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	9	0	1
	UU ORMAS	0	10	0
	Pilkada	0	0	10

Tabel 21. Tabel *Confusion Matrix* Menggunakan K=7

		Prediksi		
	K=7	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	10	0	0
	UU ORMAS	0	10	0
	Pilkada	1	0	9

Tabel 22. Tabel *Confusion Matrix* Menggunakan K=9

		Prediksi		
	K=9	*Reshuffle* Kabinet	UU ORMAS	Pilkada
Aktual	*Reshuffle* Kabinet	10	0	0
	UU ORMAS	0	10	0
	Pilkada	0	0	10

Dapat dilihat tabel uji K merupakan tabel hasil prediksi sistem klasifikasi konten berita politik menggunakan tabel *confusion matrix*, maka tahap selanjutnya dilakukan perhitungan dari nilai *precision, recall* dan *f-measure* untuk mengetahui peforma dari nilai k terbaik yang digunakan untuk sistem klasifikasi konten berita politik. Untuk perhitungan nilai akurasi dapat dilihat pada Tabel 23-26.

Tabel 23. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=3

Perhitungan	**Nilai**
Precision	$\frac{9}{9+0} + \frac{9}{9+0} + \frac{10}{10+0}$ x 100 % = 94,44%
Recall	$\frac{9}{9+1} + \frac{9}{9+1} + \frac{10}{10+0}$ x 100% = 93,33%
F-measure	$\frac{0{,}9444+0{,}9333}{2\ x\ 0{,}9444\ x\ 0{,}9333}$ x 100 % = 93,88%

Tabel 24. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=5

Perhitungan	**Nilai**
Precision	$\frac{9}{9+0} + \frac{10}{10+0} + \frac{10}{10+1}$ x 100 % = 96,96%
Recall	$\frac{9}{9+1} + \frac{10}{10+0} + \frac{10}{10+0}$ x 100% = 96,66%
F-measure	$\frac{0{,}9696+0{,}9666}{2\ x\ 0{,}9696\ x\ 0{,}9666}$ x 100 % = 96,81%

Tabel 25. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=7

Perhitungan	**Nilai**
Precision	$\frac{10}{10+0} + \frac{10}{10+1} + \frac{9}{9+0}$ x 100 % = 96,96%
Recall	$\frac{10}{10+0} + \frac{10}{10+0} + \frac{9}{9+1}$ x 100% = 96,66%
F-measure	$\frac{0{,}9696+0{,}9666}{2\ x\ 0{,}9696\ x\ 0{,}9666}$ x 100 % = 96,81%

Tabel 26. Perhitungan Nilai *Precision, Recall* dan *F-measure* k=9

Perhitungan	**Nilai**
Precision	$\frac{10}{10+0} + \frac{10}{10+0} + \frac{10}{10+0}$ x 100 % = 100%
Recall	$\frac{10}{10+0} + \frac{10}{10+0} + \frac{10}{10+0}$ x 100% = 100%
F-measure	$\frac{1+1}{2\ x\ 1\ x\ 1}$ x 100 % = 100%

Berdasarkan ketiga pengujian tersebut, dapat dilihat hasil rekapitulasi dari nilai *precision, recall* dan *f-measure* yang didapatkan dengan menggunakan empat nilai K serta variasi pembagian dataset antara jumlah data *training* dan data *testing*. Untuk Tabel rekapitulasi nilai *precision, recall* dan *f-measure* dari tiga pengujian dapat dilihat pada Tabel 27.

Tabel 27. Hasil Rekapitulasi Nilai *Precision, Recall* dan *F-measure*

		K3	K5	K7	K9
Pengujian 1	*Precision*	89,79 %	97,41 %	99,34 %	98,41 %
	Recall	88,66 %	97,33 %	99,33 %	98,33 %
	F-measure	89,22 %	97,37 %	99,33 %	98,37 %
Pengujian 2	*Precision*	95,62 %	97,81 %	96,76 %	100%
	Recall	95,55 %	97,77 %	96,66 %	100%
	F-measure	95,59 %	97,79 %	96,71 %	100%
Pengujian 3	*Precision*	94,44 %	96,96 %	96,96 %	100%
	Recall	93,33 %	96,66 %	96,66 %	100%
	F-measure	93,88 %	96,81 %	96,81 %	100%

Pada Tabel 27 ditunjukkan nilai *precision, recall* dan *f-measure* yang didapatkan dari tiga kali pengujian dengan pengujian pertama menggunakan data *training* sebanyak 150 berita politik dan data *testing* sebanyak 150 berita. Pada pengujian ini masih didapatkan banyak kesalahan klasifikasi kategori yang dilakukan oleh sistem. Pada pengujian ini, ditunjukkan apabila menggunakan data *training* tersebut mendapat nilai akurasi yang cukup baik. Untuk nilai k yang terbaik pada pengujian ini yaitu ketika sistem menggunakan nilai k=7 dengan nilai *precision* sebesar 99,43%, nilai *recall* sebesar 99,33% dan nilai *f-measure* sebesar 99,33%.

Pada pengujian kedua dilakukan dengan menggunakan data *training* sebanyak 210 berita dan data *testing* sebanyak 90 berita. Nilai akurasi yang didapat ketika menggunakan data *training* sebanyak 210 berita didapatkan hasil yang baik. Pada pengujian ini nilai k terbaik pada sistem klasifikasi konten berita politik ini ketika menggunakan nilai k=9 dengan menghasilkan nilai *precision* sebesar 100%, *recall* sebesar 100% dan *f-measure* sebesar 100%.

Pada Tabel 27, dapat dilihat pada pengujian ketiga yang menggunakan data *training* sebanyak 270 berita dan data *testing* sebanyak 30 berita menunjukkan nilai akurasi yang baik. Dapat dilihat, nilai akurasi yang terbaik ketika sistem menggunakan nilai k=9. Ketika sistem klasifikasi ini menggunakan nilai k=9 mendapatkan nilai *precision, recall* dan *f-measure* sebesar 100%.

Setelah dilakukan pengujian sebanyak tiga kali dengan menggunakan jumlah *dataset* yang berbeda, menunjukkan nilai k yang konsisten mendapatkan nilai akurasi yang baik yaitu nilai k=9. Sehingga dapat disimpulkan nilai k=9 merupakan nilai k terbaik apabila diimplementasikan pada sistem klasifikasi konten berita politik ini.

KESIMPULAN

Dari hasil pengujian dan analisis nilai precision, recall dan f-measure menggunakan tabel confusion matrix pada algoritma KNN untuk sistem klasifikasi konten berita politik menunjukkan nilai akurasi yang tinggi. Pada penelitian ini dilakukan tiga kali pengujian dengan menggunakan variasi jumlah *dataset* dan menggunakan empat nilai k. Dari hasil pengujian tersebut didapatkan nilai k terbaik yang didapatkan oleh sistem ketika sistem menggunakan nilai k=9 yang memberikan nilai precision sebesar 100%, recall sebesar 100% dan f-measure sebesar 100% pada pengujian kedua yang menggunakan data training sebanyak 210 berita dan pengujian ketiga yang menggunakan data training sebanyak 270 berita. Dengan demikian, algoritma KNN dapat bekerja degan baik ketika menggunakan nilai k=9. Sehingga dapat disimpulkan algoritma KNN cocok untuk diterapkan pada proses klasifikasi dengan dokumen yang memiliki *similarity* yang tinggi.

SARAN

Sistem klasifikasi konten berita yang dibangun masih belum sempurna dan membutuhkan pengembangan yang lebih lanjut. Penulis menyarankan pengembangan penelitian lebih lanjut untuk diimplementasikan ke dalam sistem untuk pengembangan penelitian yang serupa di masa depan yaitu sebagai berikut :

1. Dalam penelitian ini untuk pengumpulan *dataset* berita politik diambil dari beberapa portal berita *online* secara manual. Diharapkan pada penelitian selanjutnya, peneliti dapat melakukan pengumpulan *dataset* berita secara otomatis.
2. Diharapkan pada penelitian selanjutnya, untuk menemukan cara agar waktu pemrosesan lebih cepat ketika mencari kategori berita. Karena semakin banyak dokumen yang digunakan dalam penelitian ini, maka semakin lama waktu pemrosesannya.

DAFTAR PUSTAKA

[1] *Ahsanti, N. (2016). IMPLEMENTASI ALGORITMA K-NEAREST NEIGHBOR DALAM SISTEM CASE BASED REASONING UNTUK PEMBENTUKAN IDENTITAS JAWABAN OTOMATIS DAN PENCARI KEMIRIPAN JAWABAN DARI SOAL-SOAL ALGORITMA*.

[2] Ayu Puspitasari, A., Santoso, E., & Indriati. (2018). Klasifikasi Dokumen Tumbuhan Obat Menggunakan Metode Improved k-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *2*, 486–492.

[3] Palinoan, V. W. (2014). Sistem Klasifikasi Dokumen Bahasa Jawa Dengan Metode K Nearest Neighbor.
[4] *Rani, N. L. R. M. (2013). Persepsi Jurnalis dan Praktisi Humas terhadap Nilai Berita. Jurnal Ilmu Komunikasi, 10(1)*.
[5] *Rivki, M., & Bachtiar, A. M. (2017). IMPLEMENTASI ALGORITMA K-NEAREST NEIGHBOR DALAM PENGKLASIFIKASIAN FOLLOWER TWITTER YANG MENGGUNAKAN BAHASA INDONESIA. Jurnal Sistem Informasi, 13(1), 31. https://doi.org/10.21609/jsi.v13i1.500*.
[6] Setiawan, A., Kurniawan, E., & Handiwidjojo, W. (2013). IMPLEMENTASI STOP WORD REMOVAL UNTUK PEMBANGUNAN APPLIKASI ALKITAB BERBASIS WINDOWS 8, *06*(02), 11.
[7] Windu, G., & Purnomo. (2017). Akurasi Text Mining Menggunakan Algoritma K-Nearest Neighbour pada Data Content Berita SMS. *Format*, *6*(1), 1–13.