**BAB II**

**KAJIAN PUSTAKA**

**A. Dokumen Berita untuk Pengklasifikasian Data**

Berita sendiri memiliki arti yakni laporan tercepat dari suatu peristiwa atau kejadian yang faktual. Pada penelitian ini terdapat kumpulan dokumen berita tervalidasi berita *hoax* yangnantinya digunakan untuk pembuatan *library* kata. Berita *hoax* ini didapat dari website *turnbackhoax.id* yang memilki beberapa berita yang terbukti bukan berita benar atau tervalidasi berita *hoax.* Penggunaan klasifikasi dinilai tepat untuk identifikasi berita hoax dikarenan dalam metode klasifikasi terdapat teknik untuk mengidentifikasi pola dari kata pada berita yang terindikasi *hoax.* Data dokumen yang sudah dikumpulkan dibagi menjadi 2 buah bagian yakni data uji dan data latih. Data latih yakni berita tervalidasi hoax yang nantinya akan digunakan untuk pembuatan *library* kata. Data uji yakni data baik berita *hoax* maupun non *hoax* yang digunakan untuk mengukur performa sistem yang digunkan untuk klasifikasi berita *hoax*.

**B. Metode *Waterfall* untuk Pembuatan Aplikasi Klasifikasi Berita *Hoax***

Metode *waterfall* yakni metode pengembangan yang digunakan untuk penelitian dalam bidang sistem informasi. Model ini melakukan pendekatan secara urut dan sistematis yang dimulai dari tahap awal kebutuhan sistem kemudian tahap analisis, selanjutnya tahap desain, tahap *coding,* tahap *testing*, tahap *verifikasi,* dan tahap *maintenance* (Pascapraharastyan et al, 2014). Model metode ini dijuluki dengan *waterfall* dikarenakan tahapan tahapan yang dilakukan berjalan secara runtut dari tahap awal dilanjut dengan tahap selanjutnya yang dapat dikerjakan jika tahap sebelumnya telah diselesaikan.

***C. Natural Language* *Processing***

*Natural Languge Processing*  adalah sebuah pemrosesan bahasa alami. Digunakan untuk mengkaji bahasa manusia ke dalam bahasa computer yang mencakup pemrosesan simbolik dan static (Loper, 2004). NLP ini berguna untuk memproses dan memahami bahasa manusia ke dalam komputer sehingga maksud dari target dapat tersampaikan dengan benar ke dalam komputer. Dalam hal ini salah satu kesulitan yang dihadapi yakni soal ambiguitas dari kata yang diberikan oleh manusia sehingga komputer salah dalam memproses maksud yang diinginkan. (Khurana dkk, 2014)

Penerapan teknik ini dalam dunia teknologi yakni steaming / pemotongan kata menjadi bentuk dasar, pembuatan ringkasan sebuah cerpen,pembuatan botchat untuk took-toko online dan lain sebagainya. Dalam hal ini *Natural Language processing*  sangat berperan penting guna menghubungkan antara Bahasa Manusia ke dalam komputer sehingga dapat diolah sedemikian rupa sesuai kebutuhan.

Berikut adalah tahapan-tahapan pemrosesan dari NLP :

1. *Tokenizing*

Yakni sebuah proses pemisahan text menjadi per kata sehingga tiap kata dapat diolah dengan mudah.

*2. Stopwords Removal*

Stopword merupakan kata umum yang digunakan / kata yang tidak penting untuk dianalisi / diproses. Seperti contoh kata : dan, atau, akan tetapi, jika. Tujuan dari penghapusan kata ini yakni mengurangi pembengkakan jumlah index yang digunakan.

3. *Stemming*

Proses *stemming* yakni sebuah penghapusan kata yang memilki awalan / akhiran sehingga didapat kata dasar. Guna dari *stemming* ini yakni meminimalisir persamaan kata yang memiliki perbedaan awalan / akhiran sehingga dapat memperkecil jumlah indeks yang akan diproses. (Perkin, 2014).

**D. Teknik *TF-IDF***

Metode *TF-IDF* (*Term Frequency* *and* *Inverse Document* *Frequency)* yakni sebuah metode pembobotan sebuah kata yang digunakan untuk memberikan nilai seberapa penting kata tersebut dalam pembentukan text yang digunakan. Dalam metode ini memiliki algoritma yang digunakan untuk mengukur bobot tiap-tiap kata dalam sebuah dokumen. Semakin besar bobot sebuah kata, maka semakin penting pula kata tersebut dalam sebuah dokumen. (Gifti, 2018)

Proses dari TF-IDF yakni *Term Frequent* akan menghitung frekuensi kata yang muncul dan dibandingkan jumlah kata yang terdapat dalam sebuah dokumen. Berikut rumus persamaan dari metode *Term Frequent*:

*Tf(i)* = ………… ……..*(2.1)*

Keterangan :

*Tf (i)* = Nilai *Term Frequent* kata (i)

*freq ti* = Jumlah kemunculan kata(i) dalam sebuah dokumen

*t* = Jumlah keselurahan kata dalam sebuah dokumen

Untuk proses IDF (*Inverse Document Frequent*) yakni menghitung jumlah seluruh dokumen yang dibandingkan dengan dokumen yang memiliki kata i. Berikut rumus persamaan yang digunakan :

*Idf(i) = log ……………(2.2)*

Keterangan :

*Idf* = Nilai *Inverse Documen Frequent* yang terdapat dalam dokumen

*D* = Jumlah seluruh dokumen

= Jumlah dokumen yang mengandung kata (i)

Setelah diketahui Tf dan Idf dari 2 buah rumus persamaan yang telah dipaparkan maka langkah selanjutnya yakni mencari weight / bobot dari sebuah kata dengan mengalikan hasil dari TF dan IDF sebuah kata (Gifti, 2018).

*W = tf(i) x idf(i) ……………… (2.3)*

Keterangan :

*W* = *Weight* / Bobot dari sebuah kata

*tf(i)* = *TermFrequent* dari sebuah kata

*idf(i)* = *Inverse Document Frequent* dari sebuah kata

**E. *Apriori* *Method***

Metode *apriori* yakni sebuah metode yang digunakan untuk mencari pola frequensi penjualan sehingga dapat memaksimalkan laba penjualan dari sebuah toko.Secara umum, dalam pembentukan pola asosiasi algoritma apriori memiliki 2 buah tahap yakni pencarian *frequent itemset* dan membentuk pola asosiasi (Listriani dkk, 2016). Metode ini dapat di aplikasikan kedalam *stemming* kata dengan memanfaatkan hasil *support* dari metode *apriori.* Cara kerja algoritma ini yakni :

1. *Itemset Frequent* yakni pencarian item yang sering muncul bersamaan dalam sebuah data. Contoh : kata berita dan hoax sering muncul dalam sebuah paragraph.

2. *Knowledge* Pencarian data / informasi yang penting dalam sebuah paragraph

3. *Support* (nilai penunjang) yakni presentasi dari *record* yang mengandung kombinasi item dibanding dengan jumlah *record* contoh : jika ada kata a dan b maka *support* dari {a,b} yakni peluang sebuah kata a dan b yang muncul dalam sebuah dokumen (Saputro, 2017).

Rumus untuk menghitung nilai *support* pada suatu itemset yakni:

*Support (A) = …… (2.4)*

Keterangan :

*Support(A) =* Nilai Penunjang dari sebuah kata

**F. Bahasa Pemorgraman *Python***

Bahasa pemrograman *python* merupakan bahasa pemrograman yang termasuk kedalam *highlevel language. Hightlevel language* merupakan kategori bahasa pemrograman yang mendekati bahasa manusia. Penulisan dari bahasa python ini sangat *simple* sehingga tidak memerlukan banyak *space* untuk digunakan. Selain itu, keuntungan dari bahasa pemrograman *python* yakni tidak memakan banyak waktu dalam pembuatanya dikarenakan bahasa yang tidak terlalu rumit. Hal ini membuat bahasa *python* sangat mudah untuk dipelajari baik untuk pemula maupun bagi yang sudah menguasai (Hokya, 2017).

Bahasa pemrograman python termasuk kedalam bahasa pemrograman *open source* yakni dapat digunakan secara bebas sehingga banyak perusahaan yang memanfaatkan bahasa pemrograman *python* untuk memberikan pelayanan.Selain itu, *python* merupakan Bahasa pemrograman tingkat tinggi sehingga dapat memaksimalkan kinerja *system* yang ingindibuat. Salah satu aplikasi yang dapat digunakan untuk mengembangkan sebuah software dengan mendukungbahasa pemrograman python yakni pycharm. Python juga mendukung pengembangan pada berbagai sistem operasi seperti : Linux, Android, Mac Os, Windows, Palm.

(Wahono, 2003)

**G. *Database* MySQL**

Yakni sebuah manajemen basis data SQL yang digunakan untuk penyimpanan, pengaturan dan pengolahan data yang nantinya akan dapat diakses dengan mudah. MySQL termasuk kedalam *system database* gratis yang didistribusikan dibawah lisensi GPL (*General Public License*). Pada awalnya, MySQL ditemukan oleh Michael Monty Widenius pada tahun 1979 dari Swedia (Saputro dkk, 2012). Kelebihan dari database MySQL ini yakni :

1. Memiliki keamanan yang cukup baik sehingga pengguna dapat mengguanakan database ini untuk keperluan pribadi maupun komersial.

2. Gratis sehingga dapat digunakan oleh siapapun

3. Stabil dalam pengoperasianya sehingga dapat digunakan dengan optimal

4. Fleksibel dalam berbagai macam program sehingga dapat di akses dalam berbagai platform dan berbagai bahasa program

5. Merupakan *Database Management System* (DBMS)

6. MySQL merupakan *database relational*

7. Menggunakan enkripsi *password* sehingga aman untuk digunakan

**H. Pengukuran Performa**

Setelah sebuah *system* telah selesai dibuat dan menampilkan hasil yang telah diinginkan, maka langkah selanjutnya yakni akan dilakukan pengukuran performa dari *system* yang dibuat. Pengukuran performa dilakukan bertujuan menguji kinerja dan akurasi dari *system* yang sudah dibuat.Pengukuran performa ini dilakukan dengan menggunakan metode *Precision , Recall* dan *Accuracy.* (Gifti , 2018)

*Precision dan Recall* yakni berguna untuk mengukur keefektifan pengambilan informasi (Purnama, 2012). Rumus formula dari *Precision* dan *Recall*  yakni :

*Precision = ………….. (2.5)*

*Recall = ………………... (2.6)*

**Tabel 2.1 *Precision and Recall***

|  |  |  |
| --- | --- | --- |
| Berita | Relevan | Tidak Relevan |
| Hoax  Tidak Hoax | *True positve (tp)*  *False negative (fn)* | *Flase postive (fp)*  *True negative (tn)* |

Berdasarkan Tabel 2.1 dapat dirumuskan formula perhitungan akurasi sebuah *system* yakni

*P = tp / (tp+fp) …....…………………………... (2.7)*

*R = tp / (tp+fn) ……………………………….. (2.8)*

Keterangan :

P = *Precision*

R = *Recall*

tp = Berita hoax terklasifikasi hoax (*True Positive)*

*fp =* Beritareal terklasifikasi hoax *(False Positive)*

*tn =* Beritareal terklasifikasi real *(True Negative)*

*fn =* Berita hoax terklasifikasi real *(False Negative)*

Contoh terdapat 10 buah dokumen berita yang terdiri dari 5 buah konten berisi *fake news* dan 5 buah konten berisi berita *real.* Jika dalam *system* ini mendeteksi terdapat 6 buah berita yang terindikasi *fake* yakni 4 buah berisi konten *fake* *news* dan 2 buah termasuk berita *real*  maka 4 dari berita yang telah dipilih termasuk tn (*true positive*) 2 yang dipilih termasuk tn (*false psotive*). 1 berita *fake* yang tidak terdeteksi termasuk *fn* (*false negative*). Dan 4 berita sisa yang tidak terdeteksi termasuk tn (*True Negative*)

Selain *Precision and Recall,* Perhitungan akurasi *system* juga diperlukan untuk memastikan seberapa akurat *system* dapat digunakan.maka langkah selanjutnya yakni perhitungan akurasi *system* dengan menggunakan rumus persamaan :

*ac = x 100% ………………………….. (2.9)*

Keterangan :

*ac* = Tingkat akurasi

= Berita Hoax yang terindikasi hoax

= Berita Real yang yang tidak terindikasi berita hoax

= Berita real yang terindikasi hoax

= Berita Hoax yang terdeteksi real

Jumlah deteksi benar yakni hasil dari penjumlahan *true positive* dan *True negative.*Lalu hasil tersebut akan dibagi dengan total dari seluruh data yang digunakan untuk pengujian. Setelah itu dikalikan 100% agar dapat mencari persentase tingkat akurasi dari peneltian ini (Gifti, 2018).

**I. Penelitian yang Relevan**

Dalam penelitian ini dilakukan penelusuran studi literatur pada beberapa penelitian terkait dengan klasifikasi berita hoax . Beberapa penelitian tersebut antara lain :

1. Penilitian yang dilakukan oleh Frista Gifti (2018) menjelaskan pendeteksian konten hoax berbahasa Indonesia menggunakan metode *Levenshtein Distance.* Dalam penilitan tersebut penulis menggunakan metode *Tf-Idf* untuk memberikan bobot kata yang nantinya akan dicari jarak kata asal dengan kata sumber menggunakan metode *Levenshtein Distance* pada sebuah berita. Hasil batas yang didapat yakni 0,0014 pada data 100 berita yang terindikasi berita *hoax.* Metode yang diambil dari penelitian ini yakni proses pengklasifikasian berita *hoax.*

2. Penelitian yang dilakukan oleh Marin Vukovic (2009) yang berjudul “*An Intelegent Automatic Hoax Detection Sistem*” menjelaskan tentang pengklasifikasian email hoax dengan membandingkan pola tersimpan yang sama. Kelemahan dari peneleitian ini yakni jika ada email yang memiliki pola baru, maka *system* belum dapat mengidentifikasi email tersebut. Metode yang diambil dari penelitian ini yakni proses pengklasifikasian berita *hoax*.

3. Penelitian yang dilakukan oleh Munjiah Nur Saadah, Widar Atmagi, Dyah S, Agus Zainal (2013) yang berjudul “Sistem Temu Kembali Dokumen Texs dengan pembobotan TF-Idf dan LCS” menjelaskan tentang bagaimana membangun system pengembalian sejumlah dokumen dengan metode tertentu yang memiliki relevansi tinggi sesuai dengan permintaan pengguna. Metode yang digunakan penulis yakni metode pembobotan dengan menggunakan Tf-Idf yang disesuaikan dengan menggunakan *LCS* guna mempertimbangkan kemunculan urutan kata yang sama antara *query* dengan teks dalam document. Metode yang diambil dari penelitian ini yakni metode pembobotan kata.

**J. Kerangka Pemikiran**

Penelitian ini menggunakan parameter berupa beriita hoax yang nantinya akan di proses dengan menggunakan metode *NLP ,Apriori dan TF-Idf untuk dijadikan dataset.* Hasil dari inputan tersebut yakni berupa kata yang memiliki nilai bobot tersendiri yang nantinya digunakan untuk menilai berita baru yang di inputkan sebagai data uji. Setelah di proses, hasil dari meotde akan diuji akurasi dengan menggunakan perhitungan akurasi *system.*

1. Proses input data latih :

Dalam Prose *input* data latih, data uang di masukkan berupa data berita yang terindikasi berita hoax.Berita yang sudah dimasukkan nantinya akan diproses sehingga didapat *library* data kata-kata hoax.

Input Pre-Prosesing Proses (1)

*Input* Berita Hoax

Prosses NLP

Pencarian Bobot TF-IDF

Pencarian Nilai *Support Apriori*

Pencarian Hasil Rataan Support dan Bobot

Library Kata Didapat

Output Proses (3) Proses (2)

**Gambar 2.1 Proses Input Data Latih**

* 1. *Input* Berita Hoax

Proses *input* berita terindikasi *hoax* guna membuat sebuah data latih.

* 1. Proses NLP (Neural Language Prosesing)

Proses untuk memisah kata dari kalimat, memilah tanda baca dan pengahapusan kata yang tidak penting.

* 1. Proses (1)

Proses pencarian nilai bobot sebuah kata dengan menggunakan rumus *Tf-Idf* yang nantinya akan disimpan menjadi sebuah data latih.

* 1. Proses (2)

Proses pencarian Nilai sebuah kata dengan menggunakan rumus dari metode *apriori* yang nantinya akan disimpan menjadi sebuah data latih.

1.5 Proses (3)

Proses untuk mencari Nilai sebuah kata dengan mencari rataan dari hasil support dan nilai bobot yang nantinya akan dijadikan acuan dari data latih.

* 1. Output

Proses penyimpanan library kata (data latih) yang nantinya digunakan sebagai pembanding data uji.

1. Proses *input* data uji :

Data *input* data uji yakni terdiri dari data berita real dan berita yang terindikasi berita *hoax.* Data ini digunakan untuk menguji tingkat keberhasilan system yang dibuat.

*Input* Pre-Prossesing Proses(1)

*Input* Berita

Prosses NLP

Pencarian kata pada data latih

Pencarian nilai hasil

Rataan Nilai Hasil

Hasil

Output Proses(3) Proses (2)

**Gambar 2.2 Proses Input Data Uji**

* 1. *Input* Berita Hoax

Proses *input* berita terindikasi *hoax* guna membuat sebuah data latih.

* 1. Proses NLP (Neural Language Prosesing)

Proses untuk memisah kata dari kalimat, memilah tanda baca dan pengahapusan kata yang tidak penting.

* 1. Proses (1)

Proses *Searching* kata pada inputan pada data latih

*2.4* Proses (2)

Pencarian Nilai Hasil yang terdapat pada kata yang telah ditemukan

*2.5* Proses (3)

Proses mencari rata-rata dari nilai yang telah didapat dari penacrian Nilai Hasil

* 1. Hasil

Hasil yang didapat yakni apakah data yang di inputkan terindikasi data hoax atau tidak.