Big Data & Data Analytics Psikologi III & IV

Pertemuan 8

Text-Mining

Text Mining adalah salah satu bidang khusus dalam data mining yang memiliki definisi menambang data berupa teks di mana sumber data biasanya didapatkan dari dokumen dan tujuannya adalah mencari kata-kata yang dapat mewakili isi dari dokumen sehingga dapat dianalisa keterhubungan antar dokumen (Mooney,2006)

Text Mining dapat **digunakan** untuk **menganalisa dokumen**, **mengelompokkan dokumen** berdasarkan kata-kata yang terkandung di dalamnya serta **menentukan kesamaan** di antara dokumen untuk mengetahui bagaimana mereka berhubungan dengan variable lainnya (Statsoft, 2015)

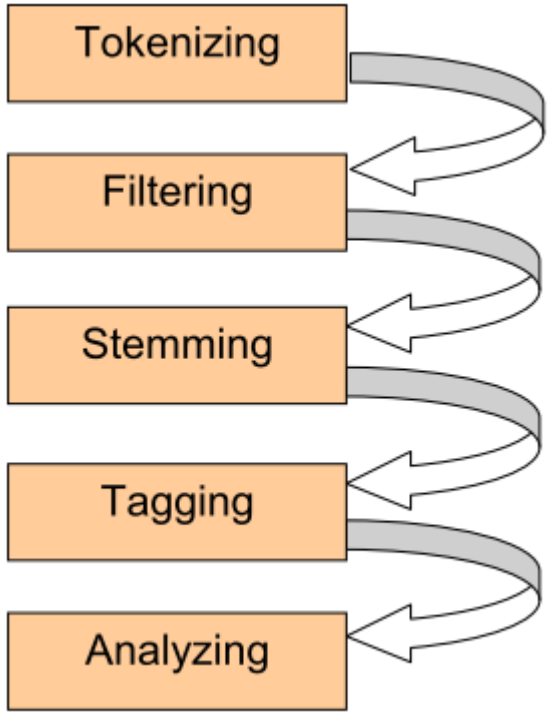
Tipe Text Mining

- **Search & Information Retrieval** :Mencari & menemukan Kembali dokumen teks, termasuk mesin pencari dan keyword
- **Document Clustering** : Pengelompokan & Kategorisasi istilah, potongan, paragraph atau dokumen menggunakan metode mining
- **Document Classification** : Pengelompokan & Kategorisasi istilah, dokumen atau paragraph dengan metode klasifikasi
- **Web Mining & Text Mining** Pada Internet Yang Fokus Pada Skala & Antar hubungan website
- **Information Extraction** : Identifikasi & Ekstraksi Fakta Yang Relevan
- **Natural Language Processing** : Pemrosesan bahasa tingkat rendah biasanya untuk bahasa komputasi
- **Concept Extraction** : Pengelompokan kata atau frasa dalam grup yang sama dalam tahap Preporcessing text.

Preprocessing Text Mining

Text Preprocessing merupakan tahapan awal terhadap pengolahan teks untuk mempersiapkan teks menjadi data yang akan diolah lebih lanjut.

Tahapan Preprocessing Text Mining



Tokenizing

Tahap Tokenizing adalah tahap pemotongan string atau kalimat input berdasarkan tiap kata yang menyusunnya.

Contoh :

Fenomena paparan kekerasan sangat represif masuk ke kehidupan anak dari berbagai media.

Hasil Tokenizing :

Fenomena, paparan, kekerasan, sangat, represif, masuk, ke, kehidupan, anak, dari, berbagai, media

Case folding

Tahap Case Folding adalah tahap mengubah setiap kata hasil tokenizing menjadi huruf kecil semua.

Contoh :

Fenomena, paparan, kekerasan, sangat, represif, masuk, ke, kehidupan, anak, dari, berbagai, media

Hasil Case Folding :

fenomena, paparan, kekerasan, sangat, represif, masuk, ke, kehidupan, anak, dari, berbagai, media

Filtering

- **Filtering** adalah tahap mengambil kata – kata penting dari hasil tokenizing.
- Bisa menggunakan algoritma stop list (membuang kata yang kurang penting) atau word list (menyimpan kata penting).

Contoh : dari hasil case folding sebelumnya

fenomena, paparan, kekerasan, sangat, represif, masuk, ke, kehidupan, anak, dari, berbagai, media

Hasil Filtering :

fenomena, kekerasan, masuk, kehidupan, anak, media

Stemming

Tahap Stemming adalah tahap mencari kata dasar (root) dari setiap kata hasil filtering.

Contoh : dari hasil filtering

Fenomena, kekerasan, masuk, kehidupan, anak, media

Hasil Stemming :

Fenomena, keras, masuk, hidup, anak, media

Tagging & Analyzing

- **Tagging** merupakan tahap untuk mencari bentuk awal dari tiap kata lampau atau hasil dari stemming yang masih memuat beberapa kata lampau yang dikembalikan ke bentuk awalnya.
- **Analyzing** merupakan tahap penentuan seberapa jauh keterhubungan antar kata atau term terhadap suatu dokumen atau kalimat dengan menghitung nilai/ bobot keterhubungan.

Hasil Stemming	Hasil Tagging
Was	Be
Used	Use
Went	Go

Algoritma Term Frequency-Inverse Document Frequency (TF-IDF)

- Digunakan untuk menghitung bobot terminology kata
- Metode ini paling umum digunakan dalam retrieval informasi karena relative lebih akurat, mudah dan efisien.

Persamaan Penghitungan Bobot masing-masing Dokumen terhadap kata kunci :

$$\boldsymbol{W_{d,t} = TF_{d,t} * IDF}$$

Dengan $\boldsymbol{IDF = \log \frac{D}{df}}$

dan $$\boldsymbol{TF_{d,t} = \frac{jumlah\ kemunculan\ kata\ ke\ t\ dalam\ dokumen}{total\ jumlah\ seluruh\ kata\ dalam\ dokumen}}$$

Keterangan

$W_{d,t}$: bobot dokumen ke- n

df : jumlah dokumen yang mengandung kata kunci

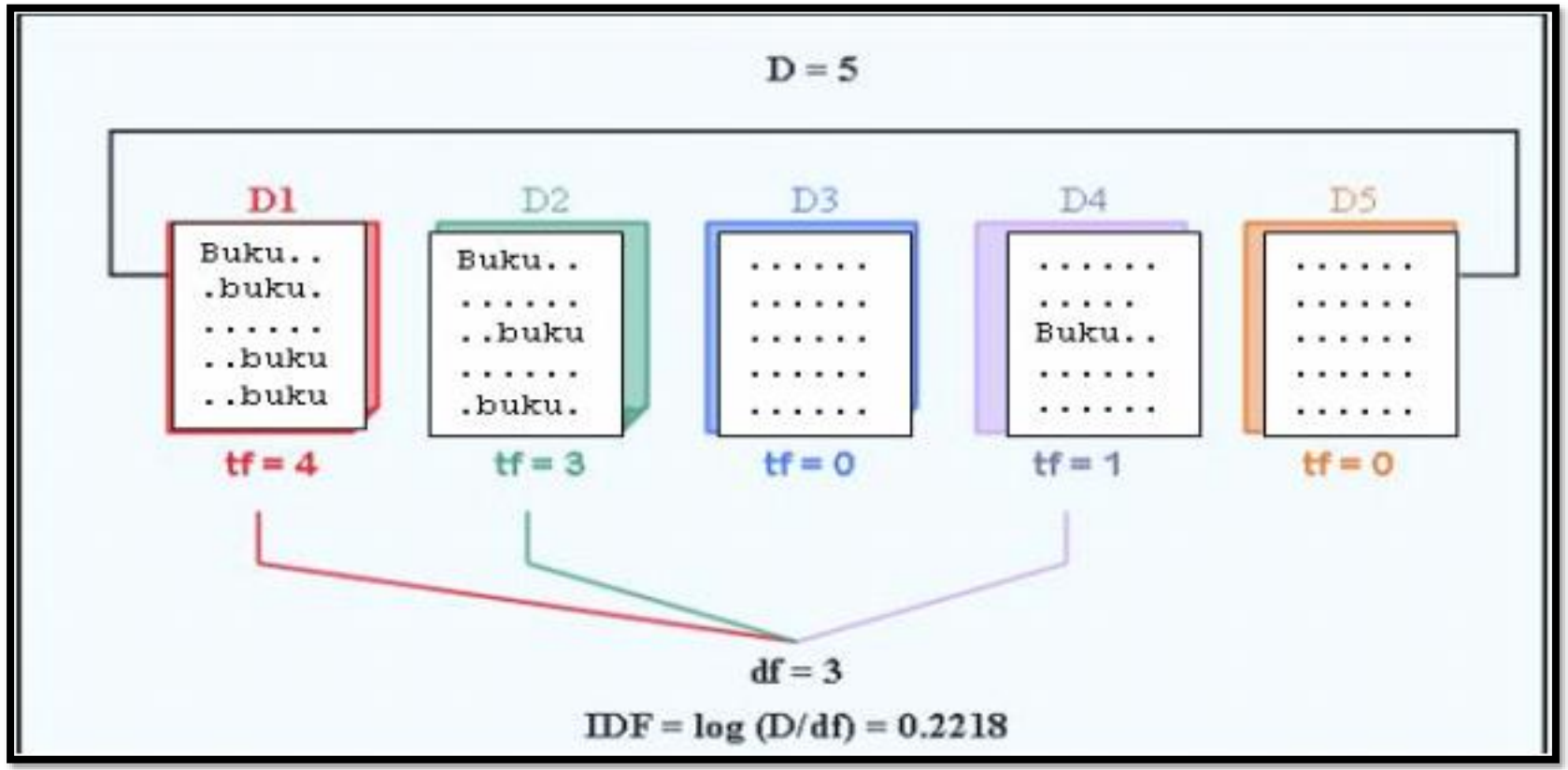
$TF_{d,t}$: jumlah kemunculan kata pada dokumen D

d : dokumen

t : kata kunci

D: total kalimat dalam dokumen

Ilustrasi Algoritma Text Mining



Contoh

Misalkan diberikan sebuah paragraf sbb :

Saya sedang belajar menghitung tf.idf. Tf.idf merupakan frekuensi kemunculan term pada dokumen. Langkah awal perhitungan tersebut adalah menghitung tf, kemudian menghitung df dan idf. Langkah terakhir menghitung nilai tf.idf. Mari kita belajar!

Tentukan pembobotan TF-IDF !

Penyelesaian

Paragraf tersebut dipecah menjadi 4 dokumen (kalimat) sbb :

1. Saya sedang belajar menghitung tf.idf.
2. Tf.idf merupakan frekuensi kemunculan term pada dokumen.
3. Langkah awal perhitungan tersebut adalah menghitung tf, kemudian menghitung df dan idf.
4. Langkah terakhir menghitung nilai tf.idf.
5. Mari kita belajar!

Sehingga diperoleh $D = 5$

- **Cleansing** :

1. Saya sedang belajar menghitung tf idf
2. Tf idf merupakan frekuensi kemunculan term pada dokumen
3. Langkah awal perhitungan tersebut adalah menghitung tf kemudian menghitung df dan idf
4. Langkah terakhir menghitung nilai tf idf
5. Mari kita belajar

- **Tokenizing** :

1. Saya, sedang, belajar, menghitung, tf, idf
2. Tf, idf, merupakan, frekuensi, kemunculan, term, pada, dokumen
3. Langkah, awal, perhitungan, tersebut, adalah, menghitung, tf, kemudian, menghitung, df, dan, idf,
4. Langkah, terakhir, menghitung, nilai, tf, idf
5. Mari, kita, belajar

- **Case folding** :

1. saya, sedang, belajar, menghitung, tf, idf
2. tf, idf, merupakan, frekuensi, kemunculan, term, pada, dokumen
3. langkah, awal, perhitungan, tersebut, adalah, menghitung, tf, kemudian, menghitung, df, dan, idf,
4. langkah, terakhir, menghitung, nilai, tf, idf
5. mari, kita, belajar

Filtering :

1. saya, belajar, menghitung, tf, idf
2. tf, idf, frekuensi, kemunculan, term, dokumen
3. awal, perhitungan, menghitung, tf, menghitung, df, idf,
4. terakhir, menghitung, tf, idf
5. kita, belajar

Stemming :

1. saya, ajar, hitung, tf, idf
2. tf, idf, frekuensi, muncul, term, dokumen
3. awal, hitung, hitung, tf, hitung, df, idf,
4. akhir, hitung, tf, idf
5. kita, ajar

Penghitungan TF-IDF

Term (t)	D1 (Dokumen 1)	D2	D3	D4	D5
Saya	1	0	0	0	0
Ajar	1	0	0	0	1
Hitung	1	0	3	1	0
Tf	1	1	1	1	0
Idf	1	1	1	1	0
Frekuensi	0	1	0	0	0
Muncul	0	1	0	0	0
Dokumen	0	1	0	0	0
awal	0	0	1	0	0
Df	0	0	1	0	0
akhir	0	0	0	1	0
kita	0	0	0	0	1

Menghitung document frequency (df) dan idf

$$idf = \log \frac{D}{df}$$

Term (t)	df	idf
Saya	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$
Ajar	2	$\log\left(\frac{5}{2}\right) = 0{,}3979$
Hitung	5	$\log\left(\frac{5}{5}\right) = 0$
Tf	4	$\log\left(\frac{5}{4}\right) = 0{,}0969$
Idf	4	$\log\left(\frac{5}{4}\right) = 0{,}0969$
Frekuensi	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$
Muncul	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$
Dokumen	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$
awal	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$
Df	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$
akhir	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$
kita	1	$\log\left(\frac{5}{1}\right) = 0{,}6989$

Penghitungan TF-IDF

Term (t)	D1	D2	D3	D4	D5	idf	TF*IDF				
							D1	D2	D3	D4	D5
Saya	1	0	0	0	0	0,6989	0,6989	0	0	0	0
Ajar	1	0	0	0	1	0,3979	0,3979	0	0	0	0,3979
Hitung	1	0	3	1	0	0	0	0	0	0	0
Tf	1	1	1	1	0	0,0969	0,0969	0.0969	0.0969	0.0969	0
Idf	1	1	1	1	0	0,0969	0.0969	0.0969	0.0969	0.0969	0
Frekuensi	0	1	0	0	0	0,6989	0	0,6989	0	0	0
Muncul	0	1	0	0	0	0,6989	0	0,6989	0	0	0
Dokumen	0	1	0	0	0	0,6989	0	0,6989	0	0	0
awal	0	0	1	0	0	0,6989	0	0	0,6989	0	0
Df	0	0	1	0	0	0,6989	0	0	0,6989	0	0
akhir	0	0	0	1	0	0,6989	0	0	0	0,6989	0
kita	0	0	0	0	1	0,6989	0	0	0	0	0,6989

Latihan

Hitunglah Pembobotan TF-IDF dari dokumen berikut :

Droplet bisa menempel di pakaian atau benda di sekitar penderita pada saat batuk atau bersin. Namun, partikel droplet cukup besar sehingga tidak akan bertahan atau mengendap di udara dalam waktu yang lama. Oleh karena itu, orang yang sedang sakit, diwajibkan untuk menggunakan masker untuk mencegah penyebaran droplet.