PEMPROSESAN TEKS

Saucha Diwandari, S.Kom., M.Eng

Deskripsi Singkat Mata Kuliah

Mata kuliah ini bertujuan untuk membekali mahasiswa untuk mengetahui konsep dan teknik data mining untuk mencari pola dalam data teks yang tercipta pada media online setiap harinya. Penekanan dalam mata kuliah ini ialah mahasiswa mampu untuk mengidentifikasi dan mengembangkan keterampilan dalam melakukan pengolahan data berupa teks dengan teknik data mining sehingga menghasilkan informasi baru yang dapat dimanfaatkan untuk tujuan tertentu, Teknik ini biasa disebut dengan Text Mining. Sumber data yang digunakan pada text mining adalah kumpulan teks yang memiliki format yang tidak terstruktur atau minimal semi terstruktur. Adapun tugas khusus dari text mining ada dua, yaitu pengkategorisasian teks (text classification) dan pengelompokan teks (text clustering). Mata kuliah ini menjelaskan tentang konsep umum Text Mining meliputi Crawling Data (melalui media sosial atau dataset online), Prepoccessing data, Summarization atau Data Transformasi, Pattern Discovery yang meliputi Document Clustering dan Document Classification sehingga dengan demikian mahasiswa bisa menghasilkan kumpulan data teks yang lebih terstruktur dan bisa digunakan untuk keperluan tertentu.

Penilaian dalam matakuliah ini mencakup seluruh proses kegiatan mandiri yang terdiri dalam beberapa tahap, yaitu :

1. Tahap pertama (10%) melakukan crawling data teks dari media online untuk melakukan analisis sentimen terkait pembelajaran daring di masa Pandemi Covid-19
2. Tahap kedua (25%) melakukan pre-processing data teks analisis sentimen terkait pembelajaran daring di masa Pandemi Covid-19
3. Tahap ketiga (25%) melakukan proses untuk mengenali pola-pola (pattern discovery) mengguakan teknik supervised & unsupervised learning
4. Tahap keempat (20%) melakukan proses analisis & visualisasi hasil dari penemuan pola untuk mengetahui analisis sentimen terkait pembelajaran daring di masa pandemi Covid-19

Ujian Tengah Semester (10%) dan Ujian Akhir Semester (10%) digunakan untuk mengevaluasi kemampuan mahasiswa secara komprehensif yang mencakup cara berpikir yang logis dan sistematis, ketepatan metode yang dipilih dalam tahap pattern discovery, ketepatan analisis yang dilakukan dan visualisasi hasil dari analisis, serta kematangan penyajian studi kasus baik secara visual maupun verbal.

Capaian Mata Kuliah

M1	**Mahasiswa memahami pengertian, konsep text mining**
M2	Mahasiswa memahami langkah-langkah crawling data text melalui social media atau dataset online
M3	Mahasiswa mampu melakukan proses Prepocessing data text
M4	Mahasiswa mampu melakukan Pattern Discovery

Outline

Before UTS

(Week 1 – 2) Pengantar Text Mining dan Information Retrieval

(Week 2) Teknik Crawling Data Analis Sentimen Kebijakan Pembelajaran Daring dimasa Pandemic Covid-19

(Week 3) Teknik Preprocessing Data Text Analis Sentimen Kebijakan Pembelajaran Daring dimasa Pandemic Covid-19

(Week 4) Konsep Pembobotan Kata pada Analis Sentimen Kebijakan Pembelajaran Daring dimasa Pandemic Covid-19

(Week 5) Teknik Summarization Analis Sentimen Kebijakan Pembelajaran Daring dimasa Pandemic Covid-19

(Week 6 - 7) Pattern Discovery: Supervised & Unsupervised

After UTS

(Week 8 – 10) Klastering Data Analisis Sentimen Kebijakan Pembelajaran Daring dimasa Pandemic Covid-19

(Week 11 – 12) Klasifikasi Data Analis Sentimen Kebijakan Pembelajaran Daring dimasa Pandemic Covid-19

(Week 13 – 14) Review Jurnal dan Proyek Mandiri Text Mining Analis Sentimen Kebijakan Pembelajaran Daring dimasa Pandemic Covid-19

Komponen Penilaian

No.	Elemen	Bobot (%)
1	CP-MK M1	**10**
2	CP-MK M2	**20**
3	CP-MK M3	**25**
4	CP MK M4	**25**
5	Ujian Tengah Semester	**10**
6	Ujian Akhir Semester	**10**
Total		**100**

Referensi

Utama:

Aggarwal, Charu C., and ChengXiang Zhai, editors. Mining Text Data. Springer US, 2012.

Pendukung:

1. Marmanis, H., Babenko, D., “Algorithms of the intelligent web”, Manning Publication Co, 2009.
2. Weiss, S. M., Indurkhya, N., Zhang, T., Damerau, F. J., “Text mining: Predictive methods for analyzing unstructured information”, Springer, 2005.
3. Grossman, D.A., Frieder, O., “Information retrieval: Algorithms and Heuristics”, 2nd edition, Springer, 2004.
4. Konchady, M., “Text mining application programming”, Charles River Media, 2006.
5. Liu, B., “Web data mining: Exploring hyperlinks, contents, and usage data”, Springer, 2007.

Wittern, I.H., Frank, E., “Data mining: Practical machine learning tools and techniquues”, Elsevier Inc, 2005.

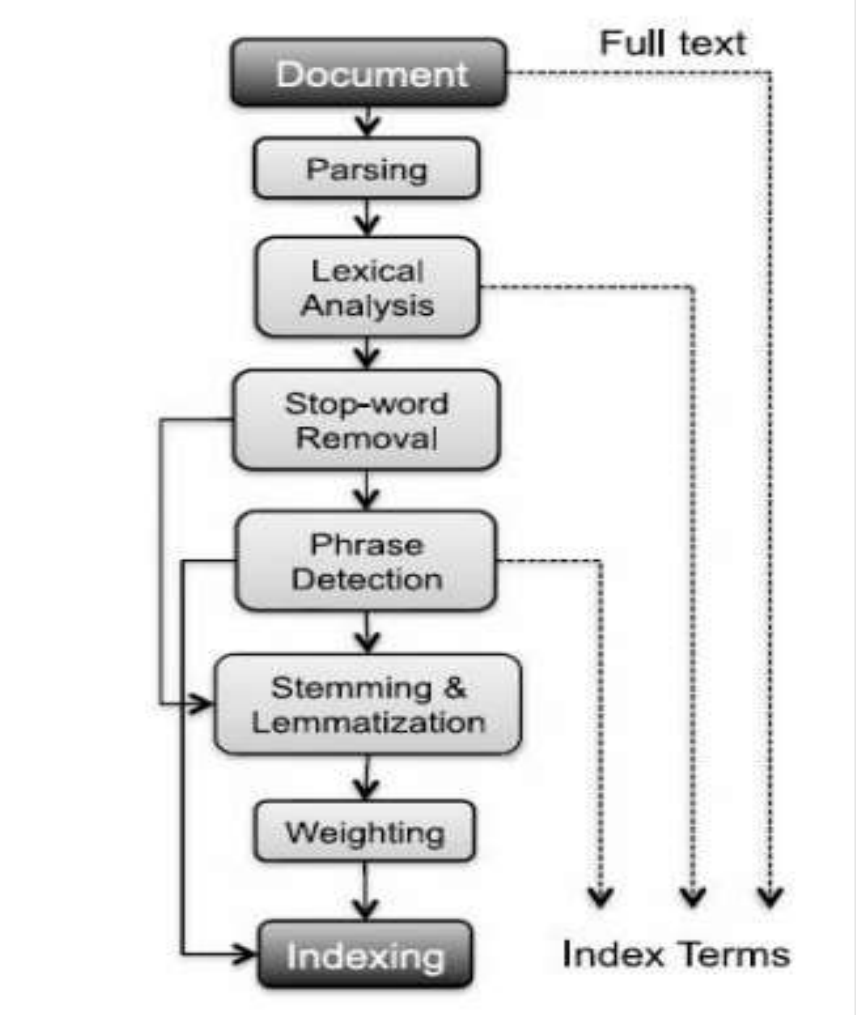
Pengantar Pemprosesan Text

- Dokumen-dokumen yang terdapat di media online/media penyimpanan kebanyakan **tidak memiliki struktur yang pasti** sehingga informasi di dalamnya tidak bisa diekstrak secara langsung
- Tidak semua kata mencerminkan makna/isi yang terkandung dalam sebuah dokumen
- Preprocessing diperlukan untuk memilih kata yang akan digunakan sebagai **indeks**
- **Indeks** ini adalah kata-kata yang **mewakili dokumen** yang nantinya digunakan untuk membuat pemodelan untuk Information Retrieval maupun aplikasi teks mining lain.

Definisi Text Processing

- Definisi Pemrosesan Teks (Text Preprocessing) adalah suatu proses pengubahan bentuk data yang belum terstruktur menjadi data yang terstruktur sesuai dengan kebutuhan, untuk proses mining yang lebih lanjut (sentiment analysis, peringkasan, clustering dokumen, etc.)
- Preprocessing adalah merubah teks menjadi term index
- Tujuan: menghasilkan sebuah set term index yang bisa mewakili dokumen

Langkah-langkah Pemprosesan Teks



Langkah 1 : Parsing

- Tulisan dalam sebuah dokumen bisa jadi terdiri dari berbagai macam bahasa, character sets, dan format
- Sering juga, dalam satu dokumen yang sama berisi tulisan dari beberapa bahasa. Misal, sebuah email berbahasa Indonesia dengan lampiran PDF berbahasa Inggris
- Parsing Dokumen berurusan dengan pengenalan dan "pemecahan" struktur dokumen menjadi komponen-komponen terpisah. Pada langkah preprocessing ini, kita menentukan mana yang dijadikan satu unit dokumen

Langkah 1 : Parsing

- Contoh, email dengan 4 lampiran bisa dipisah menjadi 5 dokumen : 1 dokumen yang merepresentasikan isi (body) dari email dan 4 dokumen dari masing-masing lampiran
- Contoh lain, buku dengan 100 halaman bisa dipisah menjadi 100 dokumen; masing-masing halaman menjadi 1 dokumen
- Satu tweet bisa dijadikan sebagai 1 dokumen. Begitu juga dengan sebuah komentar pada forum atau review produk

Langkah 2 : Lexical Analysis

- Lebih populer disebut Lexing atau Tokenization / Tokenisasi
- Tokenisasi adalah proses pemotongan string input berdasarkan tiap kata penyusunnya
- Pada prinsipnya proses ini adalah memisahkan setiap kata yang menyusun suatu dokumen
- Pada proses ini dilakukan penghilangan angka, tanda baca dan karakter selain huruf alfabet, karena karakter-karakter tersebut dianggap sebagai pemisah kata (delimiter) dan tidak memiliki pengaruh terhadap pemrosesan teks
- Pada tahapan ini juga dilakukan proses case folding, dimana semua huruf diubah menjadi huruf kecil

Langkah 2 : Lexical Analysis

- Pada tahapan ini juga Cleaning
- Cleaning adalah proses membersihkan dokumen dari komponen-komponen yang tidak memiliki hubungan dengan informasi yang ada pada dokumen, seperti tag html, link, dan script, dsb

Tokens, Types, and Terms

- Text: “apakah culo dan boyo bermain bola di depan rumah boyo?”
- Token adalah kata-kata yang dipisahpisah dari teks aslinya tanpa mempertimbangkan adanya duplikasi
- Token: “culo”, “dan”, “boyo”, “bermain”, “bola”, “di”, “depan”, “rumah”, “boyo”

Tokens, Types, and Terms

- Text: “apakah culo dan boyo bermain bola di depan rumah boyo?”
- Term adalah type yang sudah dinormalisasi (dilakukan stemming, filtering, dsb)
- Term : “culo”, “boyo”, “main”, “bola”, “depan”, “rumah”
- Term: “culo”, “boyo”, “main”, “bola”, “depan”, “rumah

Contoh Tokenisasi

Langkah 3 : Stopword Removal

- Disebut juga Filtering
- Filtering adalah tahap pemilihan katakata penting dari hasil token, yaitu katakata apa saja yang akan digunakan untuk mewakili dokumen

Stopword Removal : Metode

- Algoritma stoplist
- Wordlist adalah kata-kata yang deskriptif (penting) yang harus disimpan dan tidak dibuang dengan pendekatan bag-of-words
- Kita memiliki database kumpulan katakata yang tidak deskriptif (tidak penting), kemudian kalau hasil tokenisasi itu ada yang merupakan kata tidak penting dalam database tersebut, maka hasil tokenisasi itu dibuang

Langkah 4 : Stemming

- Dengan dilakukanya proses stemming setiap kata berimbuhan akan berubah menjadi kata dasar, dengan demikian dapat lebih mengoptimalkan proses teks mining

Langkah 4 : Stemming

Hasil Token	Hasil Filtering	Hasil Stemming
they	-	-
are	-	-
applied	applied	apply
to	-	-
the	-	-
words	words	word
in	-	-
the	-	-
texts	texts	text

Langkah 4 : Stemming

Hasil Token	Hasil Filtering	Hasil Stemming
namanya	namanya	nama
adalah	-	-
santiago	santiago	santiago
santiago	santiago	santiago
sudah	-	-
memutuskan	memutuskan	putus
untuk	-	-
mencari	mencari	cari
sang	-	-
alkemis	alkemis	alkemis

Langkah 4 : Stemming

Implementasi proses stemming sangat beragam , tergantung dengan bahasa dari dokumen

Beberapa metode untuk Stemming :

Porter Stemmer (English & Indonesia)

Stemming Arifin-Setiono (Indonesia)

Stemming Nazief-Adriani (Indonesia)

Khoja (Arabic)

Stemming : Metode

- Metode Lemmatization
- Lemmatization : Stemming berdasarkan **kamus**
- Menggunakan *vocabulary* dan *morphological analysis* dari kata untuk menghilangkan imbuhan dan di kembalikan ke bentuk dasar dari kata.

Stemming : Metode

- Metode Lemmatization
- Stemming ini bagus untuk kata-kata yang mengalami **perubahan tidak beraturan** (terutama dal am english)
- Contoh : “see” -> “see”, “saw”, atau “seen”
- Jika ada kata “see”, “saw”, atau “seen”, bisa dikembalikan ke bentuk aslinya yaitu “se e”

PEMBOBOTAN KATA

Saucha Diwandari, S.Kom., M.Eng

Outline

- **Document indexing**
  - Bag-of-words model
- **Pembobotan Kata (*Term weighting*)**
  - Binary model
  - Raw term-frequency model
  - Log-frequency model
  - Document frequency/Inverse document frequency
  - Tf-idf model

Document Indexing

- Tahapan ***preprocessing*** menghasilkan sekumpulan ***term*** yang akan dijadikan sebagai **indeks**
- **Indeks** merupakan perwakilan dari dokumen dan merupakan **fitur** dari dokumen tersebut
- **Indeks** menjadi dasar untuk pemrosesan selanjutnya dalam *text mining* maupun *information retrieval*

Bag of words model

- Indeks dari suatu dokumen dibuat hanya berdasarkan kemunculan kata, tanpa memperhatikan urutan kata
- Sebagai contoh, terdapat dua dokumen sebagai berikut:
- **d1** : Kucing makan ikan
- **d2** : Ikan makan kucing
- Kedua dokumen tersebut memiliki indeks yang sama, yaitu : kucing, makan, ikan
- Metode pembuatan indeks seperti ini disebut dengan **bag of words model**

Metode pembobotan kata

Beberapa metode pembobotan kata :

1. Binary term weighting
2. Raw-term frequency
3. Log-frequency weighting
4. Term-frequency inverse document frequency

Binary Term-Weighting

Kelebihan :

- Mudah diimplementasikan

Kekurangan :

- Tidak dapat membedakan term yang sering muncul ataupun term yang hanya sekali muncul

Contoh Dokumen

Contoh term dari dokumen setelah preprocessing

d1	d2	d3
suka, masak, nasi, goreng	ukur, nasi, makan	nasi, beras, tanam, sawah

d4	d5
mobil, bus, angkut, tumpang, ukur	bus, ukur, sawah, tumpang

Document Frequency

	D1	D2	D3	D4	D5	df
suka	1.301	0	0	0	0	1
masak	1.477	0	0	0	0	1
nasi	1.477	1.301	1.000	0	0	3
goreng	1.477	0	0	0	0	1
ukur	0	1.000	0	1.000	1.000	3
makan	0	1.000	0	0	0	1
beras	0	0	1.301	0	0	1
tanam	0	0	1.301	0	0	1
sawah	0	0	1.301	0	1.000	2
mobil	0	0	0	1.602	0	1
bus	0	0	0	1.301	1.301	2
angkut	0	0	0	1.000	0	1
tumpang	0	0	0	1.000	1.000	2

Document frequency

- *Document frequency (dft)* merupakan jumlah dokumen yang mengandung term *t*
- *Rare terms* merupakan term yang memiliki nilai *df* yang kecil
- *Frequent terms* merupakan term yang memiliki nilai *df* besar
- *Rare* terms seharusnya memiliki bobot yang lebih besar dari *Frequent terms* karena *rare terms* lebih informatif

Inverse document frequency weight

- df_t = Document frequency of t (jumlah dokumen yang mengandung term t)
  - df_t merupakan ukuran kebalikan dari keinformatifan term t
  - $df_t \leq N$ (Nilai df_t lebih kecil atau sama dengan jumlah dokumen)
- idf (Inverse document frequency) dari t adalah :

$$idf_t = \log_{10}\left(\frac{N}{df_t}\right)$$

  - Perhitungan idf_t dapat menggunakan logaritma basis berapapun

TF-IDF weighting

- Nilai *tf-idf* dari sebuah *term t* merupakan perkalian antara nilai *tf* dan nilai *idf* nya.

$$w_{t,d} = \log(1 + tf_{t,d}) \log_{10}(\frac{N}{df_t})$$

- tf-idf merupakan *term weighting* yang paling populer
  - Catatan : tanda "-" pada notasi tf-idf adalah tanda hubung, bukan pengurangan!
- Term yang sering muncul di satu dokumen dan jarang muncul pada dokumen lain akan mendapatkan nilai tinggi

Title Lorem Ipsum

LOREM IPSUM DOLOR SIT AMET, CONSECTETUER ADIPISCING ELIT.

NUNC VIVERRA IMPERDIET ENIM. FUSCE EST. VIVAMUS A TELLUS.

PELLENTESQUE HABITANT MORBI TRISTIQUE SENECTUS ET NETUS.