Text Mining and Information Retrieval

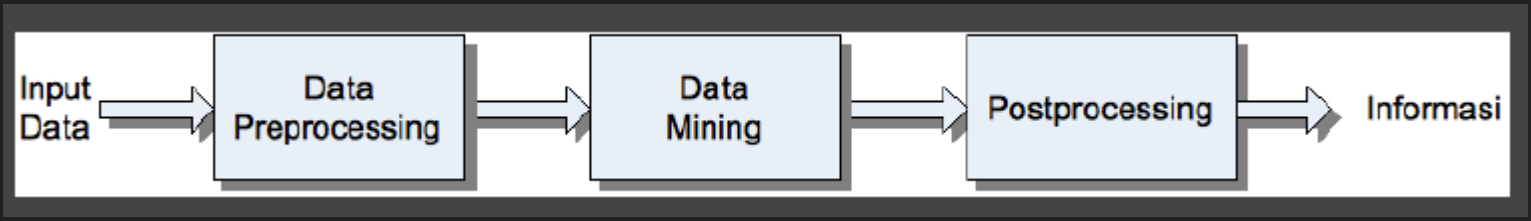
Text Mining

Data Mining

Data mining adalah suatu proses yang secara otomatis mencari atau **menemukan informasi** yang bermanfaat dan suatu kumpulan data yang besar.

Tahapan Data Mining

Data Mining lebih dekat pada bidang **pencarian pengetahuan** dalam basis data (knowledge discovery in database / KDD), yang merupakan proses **konversi** dari data mentah menjadi informasi yang bermanfaat.



Kelompok Data Mining

- Data mining dibagi dalam dua kelompok jenis tugas analisis data:
  - Predictive task : bertugas untuk **memprediksi nilai** sebuah atribut tertentu (target) didasarkan pada nilai atribut lain *(explanatory)*
  - Descriptive task : bertugas **mendapatkan pola** analisis asosiasi *(association analysis), pengelompokan (clustering),* penyimpangan *(anomaly detection) yang meringkas* hubungan-hubungan dalam data

Text Mining

- Text mining merupakan penerapan konsep dan teknik data mining untuk **mencari pola dalam teks**
- Teks Mining : Proses penganalisisan teks guna **menyarikan informasi** yang bermanfaat untuk tujuan tertentu.

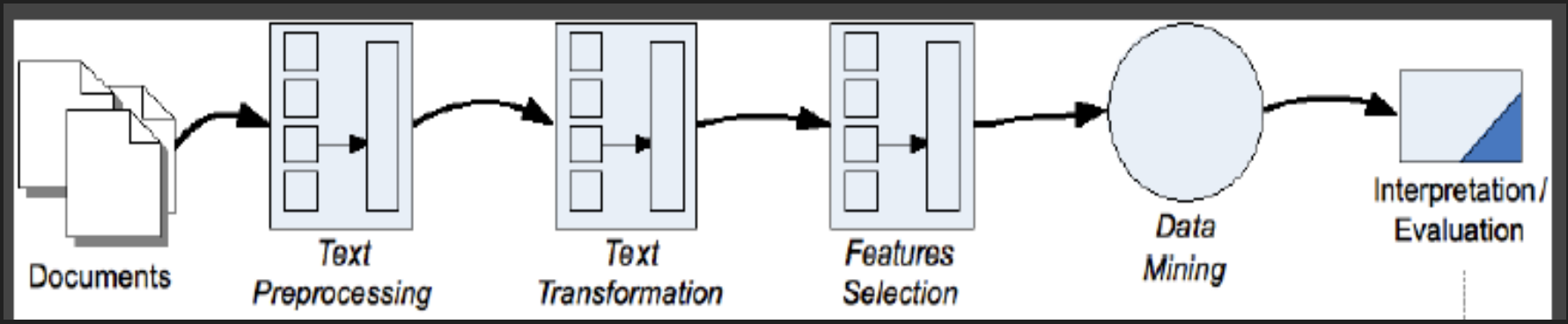
Text Mining

Perbedaan mendasar dengan Data Mining pada umumnya, **Text Mining** mengolah data **teks yang tidak terstruktur**, maka proses text mining memerlukan beberapa tahap awal ***(preprocessing)*** yang pada intinya adalah mempersiapkan agar teks dapat diubah menjadi lebih terstruktur.

Text Mining

Perbedaan	Data Mining	Text Mining
Data Object	Numerical & categorical data	Textual data
Data structure	Structured	Unstructured & semi-structured
Data representation	Straightforward	Complex
Space dimension	< tens of thousands	> tens of thousands
Methods	Data analysis, machine learning, Neural Network, etc.	Data mining, information Retrieval, NLP, etc.
Maturity	Broad implementation since 1994	Broad implementation starting 2000

Tahapan Text Mining



Masalah Umum yang ditangani

- Pengorganisasian dan Clustering Dokumen
- Klasifikasi Dokumen
- Information Extraction
- Web Mining
- Natural Language Processing (NLP)
- Information Retrieval (IR)

Clustering Dokumen

- Clustering adalah pengorganisasian kumpulan pola ke dalam cluster (kelompok-kelompok) berdasar atas kesamaannya.
- Pola-pola dalam suatu cluster akan memiliki kesamaan ciri/sifat daripada pola-pola dalam cluster yang lainnya.

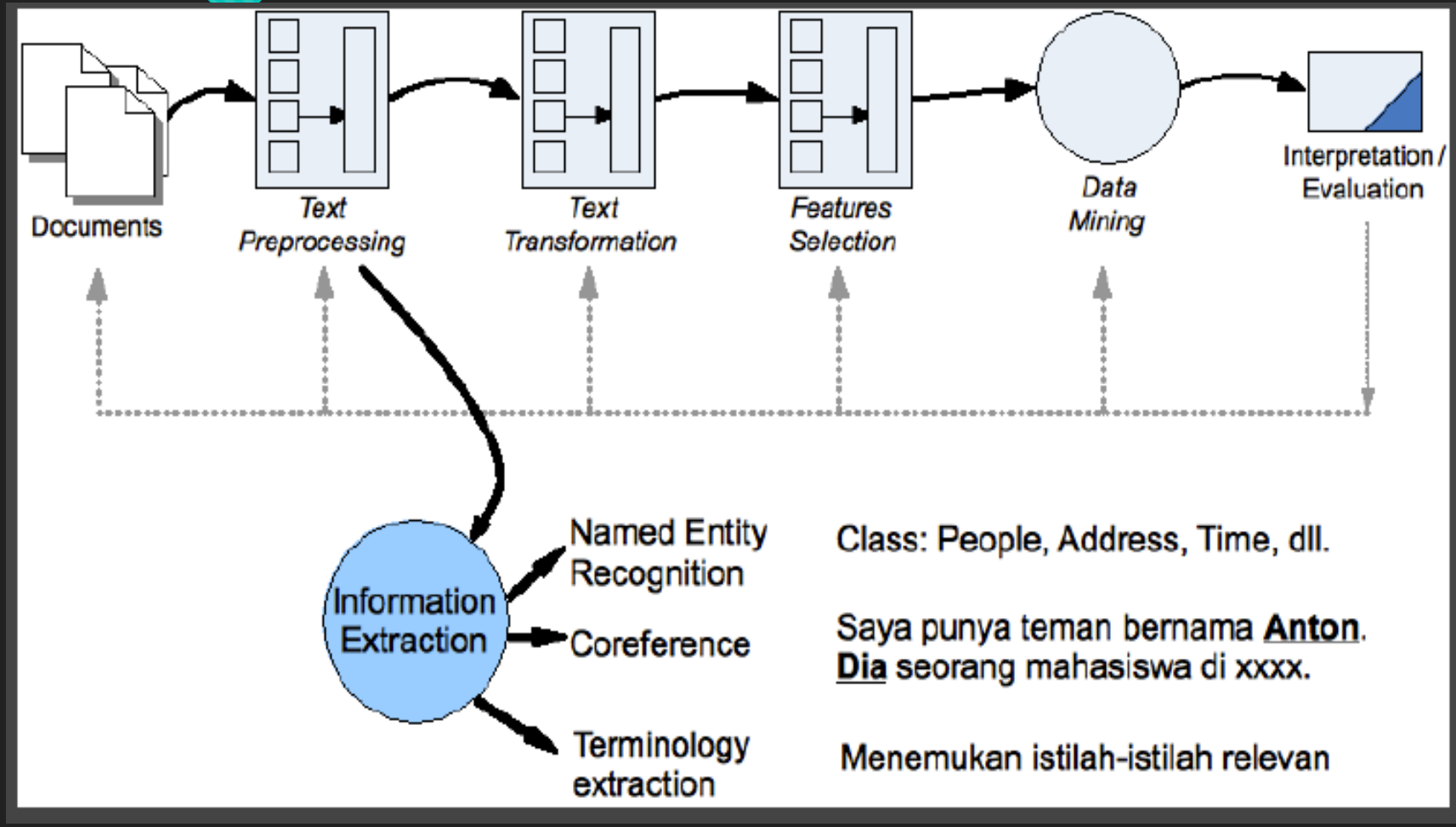
Clustering Dokumen

- Clustering bermanfaat untuk melakukan analisis pola-pola yang ada, mengelompokkan, dan membuat keputusan.
- Metodologi clustering lebih cocok digunakan untuk eksplorasi hubungan antar data untuk membuat suatu penilaian terhadap strukturnya.

Klasifikasi Dokumen

- Klasifikasi adalah mengelompokkan dokumen berdasarkan data training yang sudah dilabeli.
- Perbedaanya dengan clustering adalah pada klasifikasi, kelas/kategorinya sudah ditentukan di awal, sedangkan pada clustering tidak.

Information Extraction



Information Extraction

- Information Extraction bermanfaat untuk menggali struktur informasi dari sekumpulan dokumen.
- Dalam menerapkan IE, perlu sekali dilakukan pembatasan domain problem.
- IE sangat memerlukan NLP untuk mengetahui gramatikal dari setiap kalimat yang ada.
- Sebagai contoh:
  - "Indonesia dan Singapore menandatangani MoU kerjasama dalam bidang informasi dan komunikasi."
  - KerjaSama(Indonesia, Singapore, TIK)

Information Extraction

Dengan IE, kita dapat menemukan:

- concepts (CLASS)
- concept inheritance (SUBCLASS-OF)
- concept instantiation (INSTANCE-OF)
- properties/relations (RELATION)
- domain and range restrictions (DOMAIN/RANGE)
- equivalence

Web Mining: Karakteristik Web

- Jumlah data/informasi di web sangat besar dan terus bertambah.
  - tipe data beragam
  - informasi pada web sangat beragam.
  - informasi-informasi di web saling terhubung.
  - informasi di web sangat "kotor".
  - web juga merupakan service.
  - web dinamis
  - web merupakan sarana komunitas sosial virtual.

Web Mining

- Web Mining bertujuan untuk menemukan informasi atau pengetahuan dari
  - **Web hyperlink structure,** contoh :
    - menemukan halaman web terpenting
    - menemukan komunitas pemakai yang berbagi ketertarikan topik yang sama
  - **Page content**
  - **usage data, contoh :**
    - menemukan pola akses pemakai terhadap web, melalaui click stream.

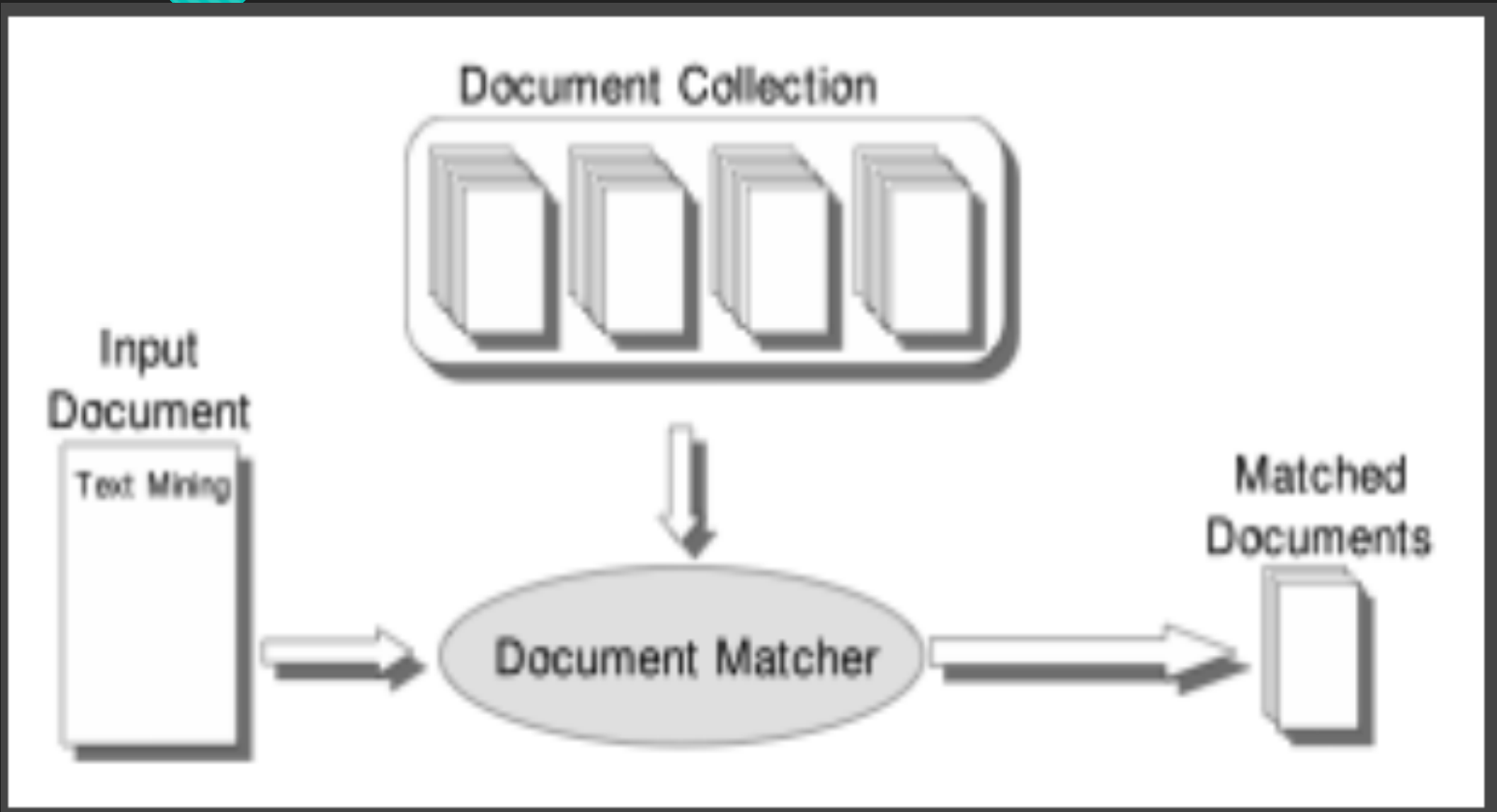
NLP

Natural Language Processing (NLP) adalah melakukan pengolahan untuk **memahami** **Bahasa** **alami** yang diucapkan manusia

NLP

- Bahasa alami adalah bahasa yang secara umum digunakan oleh manusia dalam berkomunikasi satu sama lain.
- Bahasa yang diterima oleh komputer butuh untuk diproses dan dipahami terlebih dahulu supaya maksud dari user bisa dipahami dengan baik oleh komputer.

Information Retrieval



Information Retrieval

Konsep dasar dari IR adalah pengukuran kesamaan

- sebuah perbandingan antara dua dokumen, mengukur sebearapa mirip keduanya.
- Setiap input query yang diberikan, dapat dianggap sebagai sebuah dokumen yang akan dicocokan dengan dokumendokumen lain.
- Pengukuran kemiripan menggunakan cosine similarity

Information Retrieval

Perbedaan mendasar antara Text Mining dan IR :

- Text Mining : Discovery of novel information

  Extracting Ore from otherwise worthless rock : menemukan informasi yang relevan dan bermanfaat dari sekumpulan data besar yang kelihatanya tidak berguna.

- IR : Retrieval of Non-novel Information

  Finding needles in a needle-stack : mencari informasi yang relevan di antara informasi-informasi lain yang berguna namun tidak relevan

Search Engine

Search Engine merupakan aplikasi nyata dari **Information Retrieval** pada bidang web.

Search Engine

Search Engine

Information Retrieval

Text Mining - Text Mining & Information Retrieval

Boolean Retrieval Model

- Pada Boolean Information Retrieval query yang digunakan berupa Boolean.
- Dokumen-dokumen yang ada dibedakan menjadi sesuai atau tidak sesuai dengan query tersebut

Boolean Retrieval Model

- Contoh, Query : Antony **AND** mercy

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

- Maka yang dianggap **relevan** adalah dokumen nomor 1 (Antony and Cleopatra) dan dokumen nomor 6 (Macbeth), karena kedua dokumen tersebut **mengandung** kata Antony dan kata mercy

Boolean Retrieval Model

- Contoh, Query : Calpurnia **OR** Brutus

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

- Maka yang dianggap **relevan** adalah dok nomor 1 (Antony and Cleopatra), dok nomor 2 (Julius Caser), dan dok nomor 4(Hamlet) karena ketiga dokumen tersebut **mengandung** salah satu atau dua2nya **(OR)** dari kata Calpurnia dan kata Brutus

Kelebihan Boolean IR

- Mengembalikan dokumen pencarian yang *match* saja atau tidak sama sekali.
- Sangat cocok untuk *expert user* yang sudah pengalaman tentang kebutuhan pencarian mereka (misalnya di library search, bookstore search, etc).
- Aplikasi dengan konsep Boolean Retrieval Model dapat menghemat konsumsi waktu pencarian dokumen dalam search engine.

Kekurangan Boolean IR

- Tidak bagus untuk sebagian besar pengguna.
- Sebagian besar pengguna tidak mampu menulis Query Boolean dengan baik (mereka berpikir itu akan menambah pekerjaan dalam pencarian).
- Sebagian besar pengguna tidak ingin mengarungi hasil pencarian yang banyak dan kurang spesifik (misalnya, web search).

Kekurangan Boolean IR

- Query Boolean sering menghasilkan pencarian dengan jumlah yang kadang-kadang terlalu sedikit (=0) atau terlalu banyak (=1000000). Contohnya :
  - Query 1: "standard AND user AND dlink AND 650" → 200,000 hits
  - Query 2: "standard AND user AND dlink AND 650 AND NO found" → 0 hits
- Butuh skill bagus dalam memilih query agar menghasilkan hasil pencarian yang tepat.
  - AND memberikan hasil terlalu sedikit; OR memberikan hasil terlalu banyak

Ranked retrieval models

- Pada **ranked retrieval**, system mengurutkan dokumen-dokumen berdasarkan relevansinya, bukan hanya relavan atau tidak relevan.
- Dokumen yang paling relevan memiliki ranking tertinggi dan dokumen yang kurang relevan memiliki ranking lebih rendah

Ranked retrieval models

- Setiap dokumen diberikan skor sesuai tingkat relevansinya.
- Misal diberikan nilai dalam rentang [0, 1] pada setiap dokumen
- Skor tersebut mengukur tingkat kococokan antara dokumen dan query

Ranked retrieval models

Metode yang paling sering digunakan adalah **Vector Space Model** untuk representasi fiturnya dan **Cosine Similarity** untuk menghitung kemiripan antara dokumen dan query

Ranked retrieval models

Vector Space Model adalah Model proses pencarian informasi dari query yang menggunakan ekspresi kemiripan berdasarkan frekuensi terms/token/kata yang terdapat pada dokumen.

Ranked retrieval models

Vector Space Model adalah Model proses pencarian informasi dari query yang menggunakan ekspresi kemiripan berdasarkan frekuensi terms/token/kata yang terdapat pada dokumen.

Ranked retrieval models

- Contoh Vector Space Model

$W_{t,d}$	Antony & Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1.524831652	1.366152196	0	0	0	0
Brutus	0.482268112	0.962061659	0	0.301029996	0	0
Caesar	0.266483532	0.265734309	0	0.103017176	0.079181246	0.07918
Calpurnia	0	1.556302501	0	0	0	0
Cleopatra	2.144487465	0	0	0	0	0
Mercy	0.103017176	0	0.116960302	0.134526562	0.134526562	0.07918
Worser	0.22910001	0	0.176091259	0.176091259	0.176091259	0

Ranked retrieval models

- **Cosine similarity** adalah fungsi yang digunakan untuk menghitung besarnya derajat kemiripan diantara dua vektor.
- Ukuran nilai Cosine similarity dihitung berdasarkan besarnya nilai fungsi cosine terhadap sudut yang dibentuk oleh dua vektor.

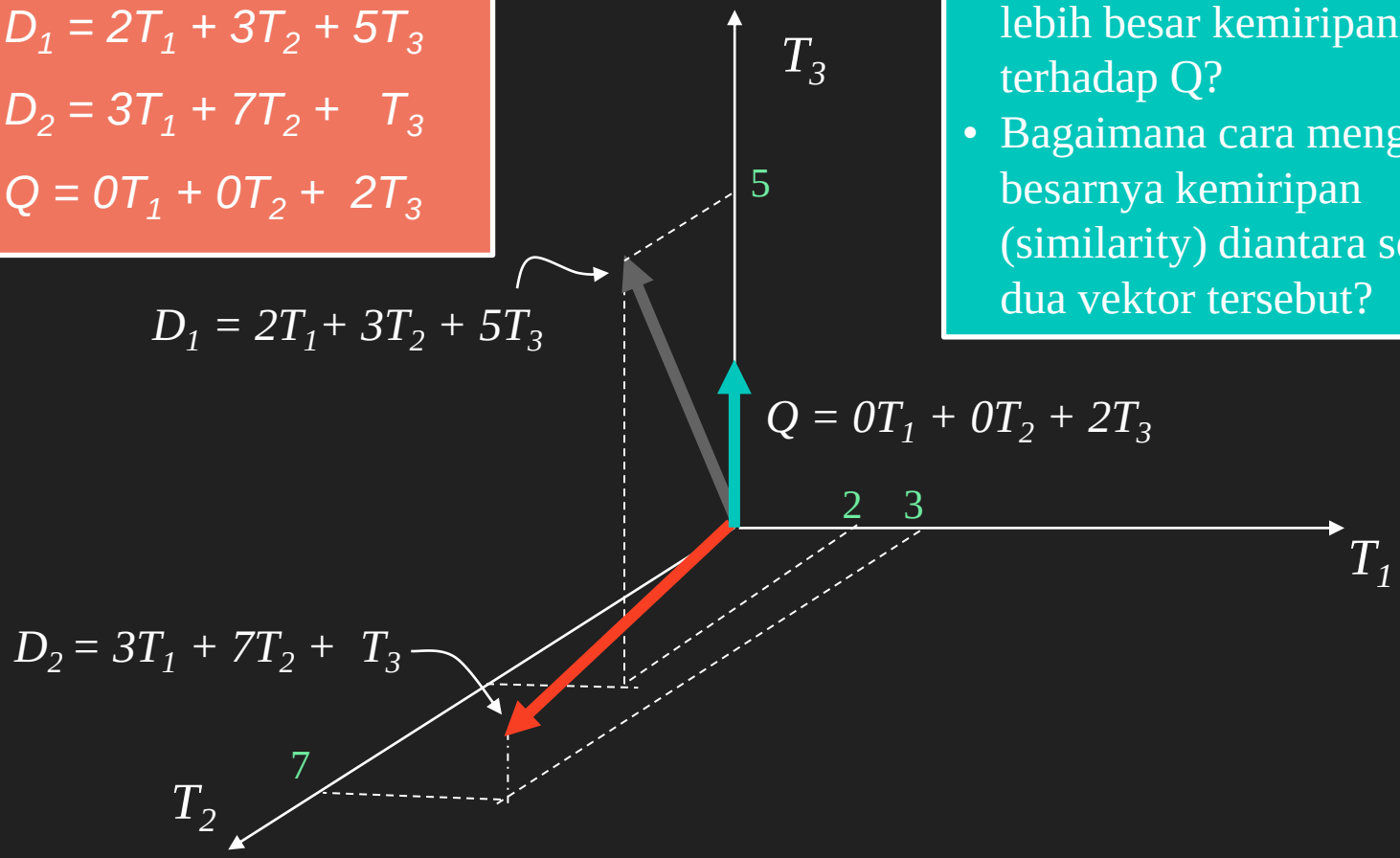
Ranked retrieval models

Misalnya :

$D_1 = 2T_1 + 3T_2 + 5T_3$

$D_2 = 3T_1 + 7T_2 + T_3$

$Q = 0T_1 + 0T_2 + 2T_3$



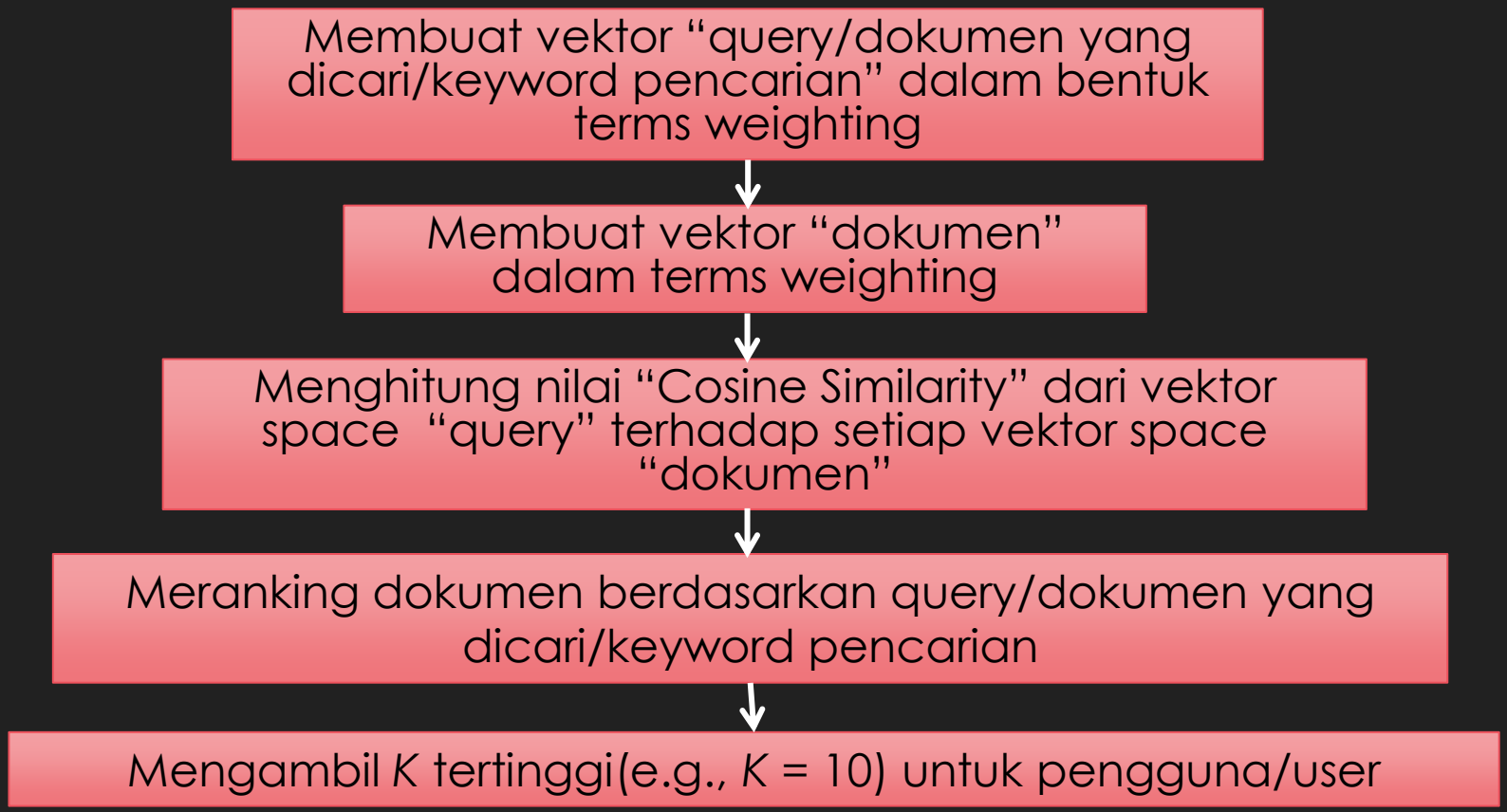
- Apakah D_1 atau D_2 yang lebih besar kemiripannya terhadap Q?
- Bagaimana cara menghitung besarnya kemiripan (similarity) diantara setiap dua vektor tersebut?

Ranked retrieval models

- Rumus untuk menghitung nilai similarity diantara dua vektor adalah sebagai berikut :

$$\text{CosSim}(\mathbf{d}_j, \mathbf{q}) = \frac{\vec{d}_j \cdot \vec{q}}{\left|\vec{d}_j\right| \cdot \left|\vec{q}\right|} = \frac{\sum_{i=1}^{t} (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^{t} w_{ij}^{2}} \cdot \sqrt{\sum_{i=1}^{t} w_{iq}^{2}}}$$

Ranked retrieval models



Case Study B (1 of 5)

- Perhatikan tabel dokumen-dokumen novel berikut :
  - AAC : *Ayat-Ayat Cinta*
  - KCB : *Ketika Cinta Bertasbih*
  - ADH : *Asmara Di Atas Haram*

tf	AAC	KCB	ADH
Cinta	227	99	0
Benci	15	10	0
Cemburu	7	0	10
Wanita	0	12	19

Setiap dokumen direpresentasikan dengan nilai real dari frekuensi setiap token/term/kata dalam bentuk vektor space. Tentukan nilai kemiripan dari setiap novel tersebut menggunakan konsep **Cosine Similarity** !

Case Study B (2 of 5)

- Hitung tf weight

$$w_{\mathrm{tf}_{t,d}} = \begin{cases} 1 + \log_{10} \mathrm{tf}_{t,d}, & \text{if } \mathrm{tf}_{t,d} > 0 \\ 0, & \text{otherwise} \end{cases}$$

W_{tf}	AAC	KCB	ADH
Cinta	1 + ^{10}log(227)	1 + ^{10}log(99)	0
Benci	1 + ^{10}log(15)	1 + ^{10}log(10)	0
Cemburu	1 + ^{10}log(7)	0	1 + ^{10}log(10)
Wanita	0	1 + ^{10}log(12)	1 + ^{10}log(19)

W_{tf}	AAC	KCB	ADH
Cinta	3.356025857	2.995635195	0
Benci	2.176091259	2	0
Cemburu	1.84509804	0	2
Wanita	0	2.079181246	2.278753601

Case Study B (3 of 5)

- Hitung df_t :

tf	AAC	KCB	ADH	df_t
Cinta	227	99	0	2
Benci	15	10	0	2
Cemburu	7	0	10	2
Wanita	0	12	19	2

- Hitung nilai idf_t (Inverse Document Frequency) :

tf	AAC	KCB	ADH	df_t	idf_t	idf_t
Cinta	227	99	0	2	$^{10}\log(3/2)$	0.17609126
Benci	15	10	0	2	$^{10}\log(3/2)$	0.17609126
Cemburu	7	0	10	2	$^{10}\log(3/2)$	0.17609126
Wanita	0	12	19	2	$^{10}\log(3/2)$	0.17609126

$$\mathrm{idf}_t = \log_{10} N/\mathrm{df}_t$$

Case Study B (4 of 5)

- Hitung $W_{t,d}$:

W_{tf}	AAC	KCB	ADH	idf_t
Cinta	3.356025857	2.995635195	0	0.17609126
Benci	2.176091259	2	0	0.17609126
Cemburu	1.84509804	0	2	0.17609126
Wanita	0	2.079181246	2.278753601	0.17609126

$$w_{t,d} = w_{tf\ t,d} \times idf_t$$

$W_{t,d}$	AAC	KCB	ADH
Cinta	0.590966819	0.527505173	0
Benci	0.38319065	0.352182518	0
Cemburu	0.324905637	0	0.352182518
Wanita	0	0.366125643	0.401268591

Case Study B (5 of 5)

- Hitung Normalization $W_{t,d}$:

$$w_{t,d} = \frac{w_{t,d}}{\sqrt{\sum_{t=1}^{n} w_{t,d}^{2}}}$$

$W_{t,d}$	AAC	KCB	ADH
Cinta	0.761894003	0.720287255	0
Benci	0.494022082	0.480891169	0
Cemburu	0.418879112	0	0.659641586
Wanita	0	0.49992995	0.75158032

- Hitung Cosine Similarity :

CosSim($\mathbf{d}_j$, $\mathbf{q}$) = $\vec{d}_j \cdot \vec{q} = \sum_{i=1}^{t} (w_{ij} \cdot w_{iq})$

cosSim	AAC	KCB	ADH
AAC	1	0.786353396	0.276310082
KCB	0.786353396	1	0.375737512
ADH	0.276310082	0.375737512	1

Kesimpulan :

Jadi dokumen Novel AAC dan KCB lebih besar similarity-nya daripada Novel AAC dengan ADH.

Note :

Semakin kecil sudut yang dibentuk oleh dua vektor, maka akan semakin besar nilai similarity-nya.
Semakin besar sudut yang dibentuk oleh dua vektor, maka similarity-nya akan semakin kecil.

Latihan Individu 2

- Perhatikan tabel dokumen-dokumen novel berikut :
  - AAC : *Ayat-Ayat Cinta*
  - KCB : *Ketika Cinta Bertasbih*
  - ADH : *Asmara Di Atas Haram*

tf	AAC	KCB	ADH
Cinta	115	58	20
Benci	10	7	11
Cemburu	2	0	6
Wanita	0	0	38

Jika $W_{t,d} = W_{tf}$.Tentukan nilai kemiripan dari setiap novel tersebut menggunakan konsep **Cosine Similarity** !