

Büyük Veri Analitiği (Big Data Analytics)

M. Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Bu dersin sunumları, "Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data, Bing Liu, Springer, 2011." kitabı kullanılarak hazırlanmıştır.

Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

Giriş

- **Klasik information retrieval (IR) sistemleri, temel bilgi birimini doküman olarak alır, Web üzerinde ise temel bilgi birimi Web sayfalarıdır.**
- Bilgiye erişim, kullanıcı sorgusuyla ilgili bir grup dokümanın bulunmasını ifade eder.
- **En yaygın kullanılan sorgu formatı keyword (term) listesi şeklindedir.**
- IR ile bilgiye erişim, veritabanlarında SQL ile bilgiye erişimden çok farklıdır.
- Ancak, klasik IR sistemlerinde doküman sayısı daha az olduğundan etkinlik en önemli gereksinim değildir.
- **Veritabanları yapılandırılmıştır, ancak metin içerisindeki bilgi yapılandırılmamıştır.**

3

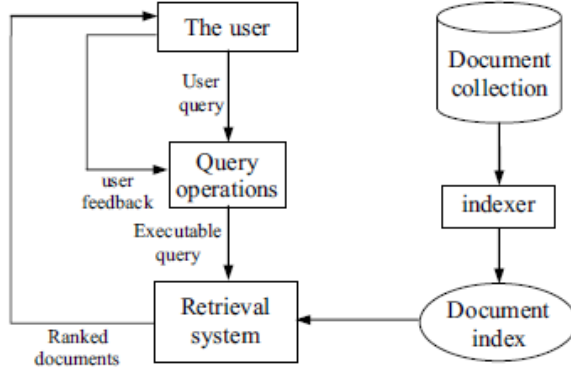
Konular

- Giriş
- **Bilgi Erişiminde Temel Yaklaşımlar**
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

4

Bilgi Erişiminde Temel Yaklaşımlar

- IR ile kullanıcının ihtiyaç duyduğu bilgilerle uyumlu bilgilerin bulunması amaçlanır.
- IR, **bilginin toplanması, organize edilmesi, depolanması, geri kazanımı ve dağıtılması** konularıyla ilgilenir.



Bilgi Erişiminde Temel Yaklaşımlar

- Kullanıcı sorguları **keyword sorguları, Boolean sorguları, phrase sorguları, proximity sorguları, full doküman sorguları ve doğal dil sorguları** şeklinde olabilir.

Keyword sorguları

- Kullanıcı ihtiyaç duyduğu bilgiyi keyword listesi olarak verir.
- Keyword listesindeki **tüm terimlerin birbirine AND ile bağlandığı varsayılır.**
- Elde edilen listedeki tüm dokümanların keyword listesindeki tüm terimleri bulundurması zorunlu olmayabilir.

Bilgi Erişiminde Temel Yaklaşımlar

Boolean sorguları

- Kullanıcı Boolean ifadeleri ile karmaşık sorgular oluşturabilir.
- Sorgu içerisinde **AND, OR** ve **NOT** gibi Boolean operatörler yer alabilir.
- Arama motorları Boolean sorguların kısıtlı versiyonunu kullanır.

Phrase sorgular

- Bu sorgular sıralı kelimelerden oluşur ve bir ifadeyi gösterir.
- Elde edilen dokümanlar en az bir kez sorgunun tamamını içerisinde bulundurmak zorundadır.
- Arama motorlarında **phrase sorguları çift tırnak içerisinde yer alır.**

7

Bilgi Erişiminde Temel Yaklaşımlar

Proximity sorgular

- Phrase sorgular ve keyword'lerden oluşabilir.
- **Proximity sorgulardaki terimlerin doküman içerisinde birbirine uzaklıkları ranking algoritmalarında kullanılır.**
- Tüm terimleri ve phrase'leri bulunduran dokümanlardan bu **terimler birbirine daha yakın olanlar daha üst sırada yer alır.**
- Bazı sistemler terimler arasındaki **maksimum mesafeyi de kısıtlayabilmektedir.**
- Popüler arama motorlarının büyük çoğunluğu terimlerin yakınlığını ve sırasını değerlendirir.

8

Bilgi Erişiminde Temel Yaklaşımlar

Full doküman sorguları

- Bu tür sorgular verilen bir dokümanın benzeri olan dokümanları bulmayı amaçlar.
- **Sorgu sayfasında dokümana ait URL girilir.**

Doğal dil sorguları

- **En karmaşık ancak ideal sorgu türüdür.**
- Kullanıcı isteğini doğal dil ifadesiyle verir.
- Ancak, doğal dili anlamak halen oldukça zordur tam olarak başarı sağlanamamıştır.

9

Bilgi Erişiminde Temel Yaklaşımlar

Sorgu ön işlemleri

- Sorgu içerisindeki anlamı **önemli olmayan kısımlar** (the, a, in, ki, de/da, için, ...) **çıkartılır.**
- Daha önce yapılan sorgularda kullanıcıdan alınan **ilgililik geribildirim**i (relevance feedback) gözönüne alınarak **orijinal sorgular yeniden düzenlenir.**
- Indexer modülü ile orijinal dokümanlar hızlı erişimi sağlamak için indekslenir.
- **Informasyon Retrieval Sisteminin en önemli görevi** kullanıcı sorgusuyla ilgili olduğu belirlenen **dokümanların sıralanmasıdır.**

10



Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- **Bilgi Erişim Modelleri**
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

11



Bilgi Erişim Modelleri

- IR modeli, **dokümanların ve sorguların nasıl gösterileceği ve dokümanlar ile kullanıcı sorguları arasındaki ilginin nasıl tanımlanacağını** belirler.
- Temel olarak 4 tane IR modeli bulunmaktadır:
 - Boolean modeli
 - Vector space modeli
 - İstatistiksel dil modeli
 - Probabilistic model
- En yaygın kullanılan IR modelleri, Boolean modeli ve vector space modelidir.
- IR modelleri, dokümanları ve sorguları farklı gösterse de, **dokümanlar ve sorgular kelimelerden oluşan küme olarak alınır.**

12

Bilgi Erişim Modelleri

- IR modellerinde kelimelerin sırası ve cümle içerisindeki konumu önemli değildir.
- Doküman veya sorgu içerisindeki **tüm kelimeler hesaplanan bir ağırlık değeri ile ilişkilidir.**
- Bir doküman topluluğu D olsun.
- Farklı terimlerden oluşan $V = \{t_1, t_2, \dots, t_{|V|}\}$ kümesi bu doküman topluluğunun **sözlüğü (vocabulary)** olarak ifade edilir.
- Burada, $|V|$ sözlüğün boyutudur.
- Bir $t_i \in V$ kelimesi ile $d_j \in D$ dokümanı arasındaki ilişki $w_{ij} \geq 0$ ağırlığı ile gösterilir.

13

Bilgi Erişim Modelleri

- Bir t_i kelimesi d_j dokümanında bulunmuyorsa $w_{ij} = 0$ olur.
- Her doküman d_j aşağıdaki gibi gösterilir.

$$d_j = (w_{1j}, w_{2j}, \dots, w_{|V|j}),$$

burada, her w_{ij} ağırlığı, $t_i \in V$ ile ilişkilidir ve t_i kelimesinin d_j dokümanı için önem seviyesini gösterir (burada kelimelerin sırası önemli değildir).

- Doküman topluluğu ilişkisel tablo veya matris şeklinde ifade edilebilir.
- Bu tabloda her kelime (terim) bir niteliği (attribute) ve her ağırlık nitelik değerini (value) gösterir.
- Farklı IR modelleri w_{ij} ağırlık değerini farklı şekillerde hesaplar.

14

Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

15

Boolean Model

- En eski IR modellerinden birisidir.
- Doküman ve sorgu için **Boolean cebri kullanılır.**
- Her doküman $\mathbf{d}_j$ aşağıdaki gibi gösterilir.

Doküman gösterimi

- **Her kelime için** dokümanda ve sorguda bulunup bulunmadığına bakılır ve $w_{ij} \in \{0, 1\}$ olur.
- Eğer, t_i kelimesi $\mathbf{d}_j$ dokümanı içinde varsa $w_{ij} = 1$, yoksa $w_{ij} = 0$ olur.

$$w_{ij} = \begin{cases} 1 & \text{if } t_i \text{ appears in } \mathbf{d}_j \\ 0 & \text{otherwise.} \end{cases}$$

16

Boolean Model

Boolean sorgular

- Sorgularda **AND**, **OR** ve **NOT** Boolean operatörleri kullanılır.
- Boolean sorgular, $((x \text{ AND } y) \text{ AND } (\text{NOT } z))$ şeklinde kesin anlama sahiptirler (x, y, z terimlerdir).

Doküman erişimi

- Verilen Boolean sorgu için elde edilen dokümanlar sorguyu mantıksal olarak doğru yapan dokümanlardır.
- **Bir doküman ya tam ilgilidir ya da tam ilgisizdir (exact match).**
- Çoğu arama motoru sınırlı Boolean operatör kullanır (+ inclusion, – exclusion).
- **Örnek:** mining –data + “equipment price”

17

Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - **Vector space model**
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

18

Vector Space Model

Doküman gösterimi

- Vector space modelinde **dokümanlar ağırlık vektörü olarak gösterilir.**
- Çoğu veri madenciliği uygulamasında, dokümanların konularına göre gruplandırılması gerekir.
- **Dokümanların konuları belirli anahtar kelimelere göre belirlenebilir.**
- Ağırlık değerleri **TF (Term Frequency)** ve **TF-IDF (Term Frequency–Inverse Document Frequency)** yöntemleriyle elde edilir.
- d_j dokümanı içindeki t_i kelimesinin w_{ij} ağırlık değeri hesaplanma yöntemine göre herhangi bir değer olabilir.

19

Vector Space Model

Doküman gösterimi

- Bir dokümanda **sık geçen kelimelerin o doküman için önemli olduğu tahmin edilebilir.**
- Bazen sık kullanılan kelimeler konu belirlemek için uygun olmayabilir.
- ‘the’, ‘and’ gibi kelimeler (stop words) İngilizce dokümanlarda çok sık kullanılır.
- Bir dokümanda bir kelimenin az kullanılması da konu belirlemek için tek başına yeterli değildir.

20

Vector Space Model

Term Frequency–Inverse Document Frequency (TF–IDF)

- Bir doküman topluluğundaki toplam doküman sayısı N olsun.
- f_{ij} ise t_i kelimesinin d_j dokümanında bulunma sayısı olsun.
- d_j dokümanında t_i kelimesinin normalize edilmiş TF değeri,

$$tf_{ij} = \frac{f_{ij}}{\max\{f_{1j}, f_{2j}, \dots, f_{|V|j}\}}$$

- Maksimum değer, tüm terimlerin d_j dokümanında en çok bulunan kelimenin adedine eşittir. $|V|$ sözlük boyutudur.
- Eğer bir t_i kelimesi d_j dokümanında yoksa $tf_{ij} = 0$ olur.
- TF dokümanların büyük çoğunluğunda bulunan kelimelerin ayırt edici olmadığını göz önüne almaz.

21

Vector Space Model

Term Frequency–Inverse Document Frequency (TF–IDF)

- df_i ise t_i kelimesinin en az bir kez bulunduğu doküman sayısı olsun.
- t_i kelimesinin inverse document frequency (IDF) değeri,

$$idf_i = \log \frac{N}{df_i}$$

- Bir t_i kelimesi çok sayıda dokümanda varsa ayırt edicilik özelliği ve önemi olmaz.
- TF-IDF ağırlık değeri aşağıdaki gibi hesaplanır.

$$w_{ij} = tf_{ij} \times idf_i$$

22

Vector Space Model

Örnek

- Veritabanında 2^{20} doküman olsun.
- Bir w kelimesi 2^{10} dokümanda geçiyorsa $IDF_w = \log_2 (2^{20} / 2^{10}) = 10$ olur.
- Bir j dokümanında w kelimesi 20 kez geçiyorsa ve bu en sık geçen kelime ise $TF_{wj} = 1$ olur.
- $TF.IDF_{wj} = 10$ olur.
- Bir k dokümanında w kelimesi 1 kez geçiyorsa ve en sık geçen başka bir kelime ise 20 kez geçiyorsa $TF_{wk} = 1/20$ olur.
- $TF.IDF_{wk} = 10 \times (1 / 20) = 1/2$ olur.

23

Vector Space Model

Sorgular

- Bir q sorgusu doküman topluluğundaki bir doküman ile aynı şekilde gösterilir.
- q sorgusu içindeki her t_i kelimesinin w_{iq} ağırlık değeri dokümanlardaki gibi hesaplanır.
- Salton ve Buckley tarafından w_{iq} ağırlık değeri hesabı için aşağıdaki eşitlik önerilmiştir.

$$w_{iq} = \left(0.5 + \frac{0.5 f_{iq}}{\max \{f_{1q}, f_{2q}, \dots, f_{|V|q}\}} \right) \times \log \frac{N}{df_i}$$

24

Vector Space Model

Doküman erişimi ve ilgililik sıralaması

- Bir dokümanın bir sorguyla ilgili olup olmadığına karar vermek çok zordur.
- Dokümanlar **sorguya ilgililik derecelerine göre sıralanır.**
- **İlgililik derecesi, q sorgusu ile d_j dokümanının benzerliğini hesaplayarak elde edilir.**
- **Benzerliğin hesaplanması için temel yaklaşımda sorgu ile dokümanda bulunan ortak kelimelerin ağırlık değerleridir.**
- Bazı ranking algoritmaları kelimelerin doküman içerisindeki **yakınlıklarını da gözönüne alarak benzerlik** hesaplayabilmektedir.

25

Vector Space Model

Doküman erişimi ve ilgililik sıralaması

- Metin ve doküman kümelemede en yaygın kullanılan benzerlik **cosine similarity**'dir.

$$\text{cosine}(\mathbf{d}_j, \mathbf{q}) = \frac{\langle \mathbf{d}_j \bullet \mathbf{q} \rangle}{\|\mathbf{d}_j\| \times \|\mathbf{q}\|} = \frac{\sum_{i=1}^{|V|} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{|V|} w_{ij}^2} \times \sqrt{\sum_{i=1}^{|V|} w_{iq}^2}}$$

- w_{ij} , t_i kelimesinin $\mathbf{d}_j$ dokümanındaki ağırlık değeridir.
- w_{iq} , t_i kelimesinin $\mathbf{q}$ sorgusundaki ağırlık değeridir.
- Daha basit benzerlik ölçütü iki vektör ile hesaplanabilir.

$$\text{sim}(\mathbf{d}_j, \mathbf{q}) = \langle \mathbf{d}_j \bullet \mathbf{q} \rangle$$

26

Vector Space Model

Doküman erişimi ve ilgililik sıralaması

- **Okapi** metin ve doküman kümelemede popüler yöntemlerdendir. Genellikle kısa sorgularda daha etkindir.

$$okapi(d_j, q) = \sum_{t_i \in q, d_j} \ln \frac{N - df_i + 0.5}{df_i + 0.5} \times \frac{(k_1 + 1)f_{ij}}{k_1(1 - b + b \frac{dl_j}{avdl}) + f_{ij}} \times \frac{(k_2 + 1)f_{iq}}{k_2 + f_{iq}}$$

Bulunduğu doküman sayısı arttıkça önemi azalır.

Dokümanda ve sorguda bulunma sıklığı arttıkça önemi artar.

Doküman boyutu arttıkça sık geçmesinin önemi azalır.

f_{ij} , t_i kelimesinin d_j dokümanındaki bulunma sayısıdır.

f_{iq} , t_i kelimesinin q sorgusundaki bulunma sayısıdır.

df_i , t_i kelimesini bulunduran doküman sayısıdır.

dl_j , d_j dokümanının uzunluğudur.

$avdl$, dokümanların ortalama uzunluğudur.

$k_1 = [1, 0 - 2, 0]$
 $b = 0,75$ (genellikle)
 $k_2 = [1 - 1000]$

27

Vector Space Model

Doküman erişimi ve ilgililik sıralaması

- **Pivoted normalization weighting** yöntemi ile metin ve dokümanlara skor değeri hesaplanabilir.

$$pnw(d_j, q) = \sum_{t_i \in q, d_j} \frac{1 + \ln(1 + \ln(f_{ij}))}{(1 - s) + s \frac{dl_j}{avdl}} \times f_{iq} \times \ln \frac{N + 1}{df_i}$$

Bulunma sıklığı arttıkça önemi artar.

Bulunduğu doküman sayısı arttıkça önemi azalır.

Doküman boyutu arttıkça sık geçmesinin önemi azalır.

f_{ij} , t_i kelimesinin d_j dokümanındaki bulunma sayısıdır.

f_{iq} , t_i kelimesinin q sorgusundaki bulunma sayısıdır.

df_i , t_i kelimesini bulunduran doküman sayısıdır.

dl_j , d_j dokümanının uzunluğudur.

$avdl$, dokümanların ortalama uzunluğudur.

$s = 0,2$ (genellikle)

28

Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- **Değerlendirme Ölçütleri**
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

29

Değerlendirme Ölçütleri

- **Web aramada dokümanların kullanıcı sorgusuyla ilgili olup olmadığına yönelik bir karar verilmez.**
- Bunun yerine, **kullanıcı için dokümanların rank değerleri hesaplanır** ve sıralama yapılır.
- Veritabanındaki dokümanların kümesi D , doküman sayısı N olsun.
- Verilen bir q sorgusu için retrieval algoritmaları D içerisindeki her doküman için ilgililik skoru hesaplar.
- Ardından, **ilgililik skorlarına göre tüm dokümanlar için R_q rank değerleri oluşturulur.**

$$R_q : < \mathbf{d}_1^q, \mathbf{d}_2^q, \dots, \mathbf{d}_N^q >$$

- Burada, $\mathbf{d}_I^q \in D$ kullanıcı sorgusuna en ilgili eleman, $\mathbf{d}_N^q \in D$ ise en ilgisiz elemandır.

30

Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

31

Precision ve recall

- $D_q \subseteq D$, kullanıcı sorgusu q ile gerçekten ilgili dokümanlar kümesi ise **precision ve recall değerleri hesaplanabilir.**

Recall

- Recall değeri, listede i .sıradaki bir doküman $\mathbf{d}_i^q$ için, $\mathbf{d}_1^q$ ve $\mathbf{d}_i^q$ arasındaki **ilgili dokümanların tüm ilgili dokümanlara oranıdır.**
- D_q içerisinde $\mathbf{d}_1^q$ ve $\mathbf{d}_i^q$ arasındaki **ilgili dokümanların sayısı s_i ise,** recall değeri aşağıdaki gibi hesaplanır:

$$r(i) = \frac{s_i}{|D_q|}$$

32

Precision ve recall

Precision

- Precision değeri, listede i .sıradaki bir doküman $\mathbf{d}_i^q$ için, $\mathbf{d}_1^q$ ve $\mathbf{d}_i^q$ arasındaki ilgili dokümanların i sayısına oranıdır.

$$p(i) = \frac{s_i}{i}$$

33

Precision ve recall

Örnek

- D doküman kümesinde 20 doküman olsun.
- Kullanıcı sorgusu q ile gerçekten ilgili doküman sayısı 8 olsun.
- Retrieval algoritması tablodaki rank değerlerini oluştursun.
- Precision ve recall değerleri tablodaki gibi hesaplanabilir.
- Tabloda, “+” ilgili, “-” ilgili olmayan dokümanları göstermektedir.

Rank i	+/-	$p(i)$	$r(i)$
1	+	1/1 = 100%	1/8 = 13%
2	+	2/2 = 100%	2/8 = 25%
3	+	3/3 = 100%	3/8 = 38%
4	-	3/4 = 75%	3/8 = 38%
5	+	4/5 = 80%	4/8 = 50%
6	-	4/6 = 67%	4/8 = 50%
7	+	5/7 = 71%	5/8 = 63%
8	-	5/8 = 63%	5/8 = 63%
9	+	6/9 = 67%	6/8 = 75%
10	+	7/10 = 70%	7/8 = 88%
11	-	7/11 = 63%	7/8 = 88%
12	-	7/12 = 58%	7/8 = 88%
13	+	8/13 = 62%	8/8 = 100%
14	-	8/14 = 57%	8/8 = 100%
15	-	8/15 = 53%	8/8 = 100%
16	-	8/16 = 50%	8/8 = 100%
17	-	8/17 = 53%	8/8 = 100%
18	-	8/18 = 44%	8/8 = 100%
19	-	8/19 = 42%	8/8 = 100%
20	-	8/20 = 40%	8/8 = 100%

Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

35

Ortalama precision

- Kullanıcı sorgusu q için farklı retrieval algoritmalarını karşılaştırmak amacıyla ortalama precision değeri kullanılabilir.
- Ortalama precision değeri tüm ilgili dokümanların precision değerlerinin aritmetik ortalaması hesaplanarak elde edilir.

$$p_{avg} = \frac{\sum_{d_i^q \in D_q} p(i)}{|D_q|}$$

$$p_{avg} = \frac{100\% + 100\% + 100\% + 80\% + 71\% + 67\% + 70\% + 62\%}{8} = 81\%$$

36

Konular

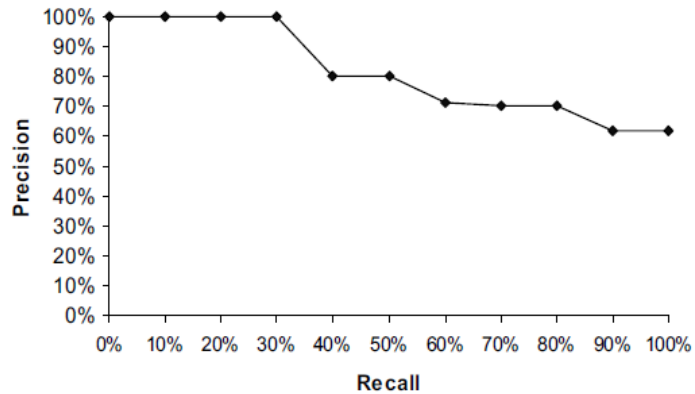
- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

37

Precision recall eğrisi

- Her bir rank pozisyonu için precision ve recall değerlerine göre precision-recall grafiği çizilebilir (x eksenini recall, y eksenini precision).
- Genellikle 11 standart aralık için çizilir (%0, %10, %20, ..., %100).

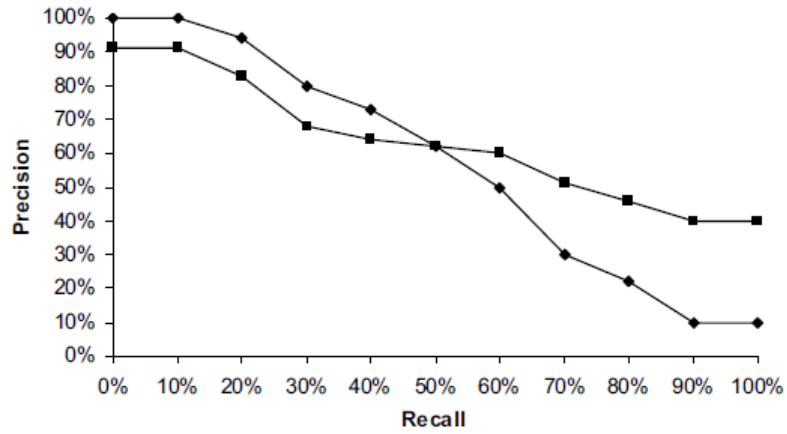
i	$p(r_i)$	r_i
0	100%	0%
1	100%	10%
2	100%	20%
3	100%	30%
4	80%	40%
5	80%	50%
6	71%	60%
7	70%	70%
8	70%	80%
9	62%	90%
10	62%	100%



38

Precision recall eğrisi

- Farklı algoritmalar precision-recall grafiğine göre karşılaştırılabilir.
- **Birinci algoritmada düşük recall değerleri için precision daha iyi yüksek recall değerlerinde precision daha düşüktür.**



39

Konular

- Giriş
- Bilgi Erişiminde Temel Yaklaşımlar
- Bilgi Erişim Modelleri
 - Boolean model
 - Vector space model
- Değerlendirme Ölçütleri
 - Precision ve recall
 - Ortalama precision
 - Precision recall eğrisi
 - F-score

40

F-score

- F-score değeri farklı retrieval algoritmalarının karşılaştırılması için yaygın bir şekilde kullanılmaktadır.
- F-score değeri, **hem precision hem de recall değerlerinin yüksek olduğu değerlerde yüksek değere sahiptir.**

$$F(i) = \frac{2}{\frac{1}{r(i)} + \frac{1}{p(i)}} = \frac{2p(i)r(i)}{p(i) + r(i)}$$