Büyük Veri Sistemleri

Yapay Zekâ ve Veri Dersleri Rehberi · Kısım IV: Veri

Üç V | Dağıtık depolama | MapReduce ve Spark | Akış işleme | NoSQL ve CAP | Komşu dersler

Sercan KÜLCÜ · Giresun Üniversitesi · Bilgisayar Mühendisliği

© Sercan Külcü. Tüm hakları saklıdır. · Sunumun etkileşimli HTML sürümü: https://sercankulcu.github.io/files/ai_courses/sunumlar/Bolum_27_Buyuk_Veri_Sistemleri.html

GİRİŞ

Ders Hangi Soruyu Sorar?

> **Tek bir makineye sığmayan veri nasıl saklanır ve işlenir?**

Büyük veri sistemleri, **tek bir makineye sığmayan** ya da tek makinenin yetişemeyeceği hızda gelen veriyi çok sayıda makineye dağıtarak saklamayı ve işlemeyi inceler. Ders bölümleme, kopyalama, hata toleransı ve tutarlılık arasındaki ödünleşimlerle ilgilenir.

Bir bakışta

YAKLAŞIM dağıtık depolama ve hesaplama	**GİRDİ** çok büyük ve hızlı akan veri	**ÇIKTI** işlenmiş veri, sorgu sonucu
BAŞARI ÖLÇÜTÜ ölçeklenme, gecikme, maliyet	**MATEMATİK YÜKÜ** düşük	**VERİ İHTİYACI** çok fazla

Kısa Tarihçe

2001
Laney veri yönetimindeki zorluğu hacim, hız ve çeşitlilik (3V) olarak tanımlar.

2003
Google File System makalesi: ucuz makinelerden kurulan, hataya dayanıklı dağıtık dosya sistemi.

2004
Dean ve Ghemawat MapReduce'u yayımlar: binlerce makinede basit bir programlama modeli.

2006
Hadoop, GFS ve MapReduce'un açık kaynak karşılığı olarak ayrı bir Apache projesi olur.

2010
Spark açık kaynak olarak yayımlanır: ara sonuçları bellekte tutarak yinelemeli işleri hızlandırır.

2011
LinkedIn Kafka'yı açık kaynak yapar: olay akışları için dağıtık kayıt (log) sistemi.

DERS

Temel Kavramlar

Hacim, hız, çeşitlilik
Verinin boyutu, geliş hızı ve biçim çokluğu.

Dağıtık dosya sistemi
Dosyayı bloklara bölüp makinelere kopyalamak.

MapReduce
Eşle, karıştır, indirge: veriye hesabı götürmek.

Bellek içi işleme
Spark: ara sonuçları diske yazmadan zincirlemek.

Akış işleme
Veri gelirken, sınırsız olay akışını işlemek.

NoSQL
Anahtar-değer, belge, sütun ailesi ve çizge depoları.

CAP teoremi
Ağ bölünmesinde tutarlılık ile erişilebilirlik arası seçim.

Bölümleme, kopyalama
Yükü dağıtmak ve makine bozulunca veri kaybetmemek.

DERS

Tipik İş Akışı



Örnek: Bir video sitesi günde milyarlarca izleme olayını Kafka ile toplar, nesne deposunda tarih bazında bölümler, Spark ile gecelik olarak kullanıcı başına izleme özetleri çıkarır ve sonuçları analistlerin SQL ile sorgulayabileceği tablolara yazar.

DERS

Derste Neler İşlenir?

1. Ölçekleme sorunu: dikey ve yatay büyüme
2. Dağıtık dosya sistemleri: GFS ve HDFS
3. MapReduce programlama modeli
4. Spark: RDD, DataFrame ve Spark SQL
5. Veri modelleri: ilişkisel, belge, sütun ailesi
6. Depolama motorları: LSM ağacı, B ağacı, sütunlu dosya
7. Bölümleme ve yük dengeleme
8. Kopyalama ve tutarlılık modelleri
9. CAP, dağıtık işlemler ve uzlaşma: Paxos, Raft
10. Olay akışları: Kafka ve yayımla-abone ol
11. Akış işleme: pencereler, olay zamanı, Flink
12. Veri gölü, veri ambarı ve iş akışı düzenleme

Kazandırdığı beceriler

PySpark ile dağıtık iş yazmak | Kafka üreticisi ve tüketicisi yazmak | Parquet ile sütunlu depolama | Cassandra ya da MongoDB ile veri modeli | yavaş bir Spark işini çözümlemek

Tipik bir dönemin akışıdır; sıra ve kapsam üniversiteden üniversiteye değişir.

DENE

Arama Kayıtlarında Sözcük Sayımı

Bir sitenin arama kayıtları tek makineye sığmıyor; hangi sözcüğün kaç kez arandığını işçi makineler birlikte saymalı. Adım adım ilerleyin: Map → Karıştır → Reduce. İşçi sayısını ve birleştiriciyi değiştirip süreye bakın.

Map işçileri

İ1
veri bulut veri
disk veri ağ
kod veri sorgu
bulut ağ veri disk

İ2
sorgu veri kod
veri disk disk
ağ bulut veri
kod sorgu veri veri

İ3
bulut disk ağ
veri kod bulut
sorgu ağ veri
disk veri bulut kod

Reduce işçileri

Sonraki adım ▶ | Baştan

İşçi sayısı = **3**

Birleştirici (combiner)

Adım 1/4: Bölme

Bölme: 29 ms · Map: 160 ms · Karıştırma: 123 ms · Reduce: 116 ms

Geçen süre: **29 ms**

Metin 3 parçaya bölündü; her işçi kendi parçasını okur.

Büyük veri dersi işi veriye götürmeyi öğretir: Map ve Reduce birbirinden bağımsız çalıştığı için işçi eklemek hızlandırır, ama ağ trafiği ve en yavaş işçi bir sınır koyar. **Birleştirici** ağdaki çift sayısını azaltır.

KONUM

Ön Koşullar

Bu dersten önce bilinmesi yararlı olan dersler ve neden gerektikleri.

Ders	Neden gerekir
İşletim Sistemleri	süreç, iş parçacığı, dosya sistemi ve G/Ç maliyeti
Java ile Nesneye Yönelik Programlama	Hadoop, Spark ve Kafka JVM üzerinde çalışır
Veritabanı ve SQL	ilişkisel model, dizin, işlem (transaction)

KONUM

Bu Ders Sana Göre mi?

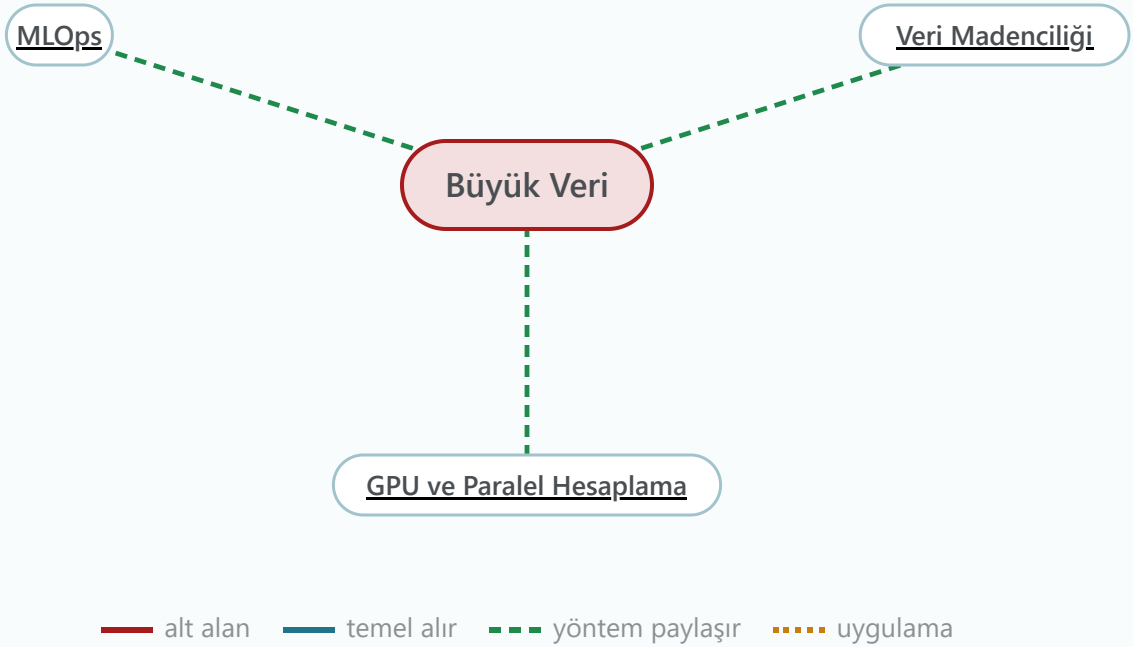
Kendine sor; "hayır" dediklerin için sağdaki sütun nereden başlayacağını gösterir.

Kendine sor	Hayırsa önce
Süreç ile iş parçacığı arasındaki farkı açıklayabiliyor musun?	İşletim Sistemleri dersi: süreçler ve eş zamanlılık
JOIN ve GROUP BY içeren bir SQL sorgusu yazabiliyor musun?	Bir veritabanı dersindeki SQL ve ilişkisel model
Java ya da Python'da sınıf ve koleksiyonlarla program yazabiliyor musun?	Java ile Nesneye Yönelik Programlama dersi
Linux komut satırında dosya taşıyıp bir betik çalıştırabiliyor musun?	Linux komut satırına kısa bir giriş

Çoğuna "evet" diyebiliyorsan derse hazırsın; bir ya da iki "hayır" dönem içinde kapanabilir.

KONUM

Dersin Komşuları



ÖRTÜŞME

Örtüştüğü Dersler

İlişki	Ders	Ortak konu
yöntem paylaşır	Veri Madenciliği ile yöntem paylaşır	ölçeklenebilir madencilik
yöntem paylaşır	Makine Öğrenmesi Sistemleri ve MLOps ile yöntem paylaşır	veri boru hatları
yöntem paylaşır	GPU Programlama ve Paralel Hesaplama ile yöntem paylaşır	paralel ve dağıtık hesaplama

Ayrıştığı Noktalar (1/2)

Ölçüt	Büyük Veri	Veri Madenciliği	Veri Bilimi
Temel soru	Tek bir makineye sığmayan veri nasıl saklanır ve işlenir?	Büyük veri yığınlarındaki gizli ve işe yarar örüntüler nasıl bulunur?	Veriden doğru soruları sorup karara dönüşen sonuçlar nasıl çıkarılır?
Yaklaşım	dağıtık depolama ve hesaplama	keşfe dayalı çözümleme	soru, veri, analiz, anlatı
Girdi	çok büyük ve hızlı akan veri	büyük tablo ya da işlem verisi	ham, dağınık veri
Çıktı	işlenmiş veri, sorgu sonucu	kural, küme, aykırı değer	rapor, görselleştirme, model
Başarı ölçütü	ölçeklenme, gecikme, maliyet	ilginçlik, destek, güven	kararlara katkı, doğruluk
Matematik yükü	düşük	orta	orta: istatistik
Veri ihtiyacı	çok fazla	çok	orta ve çok

- **Veri Madenciliği:** Veri madenciliği hangi örüntünün aranacağını sorar; büyük veri sistemleri o aramanın çok makinede nasıl çalışacağını.
- **Veri Bilimi:** Veri bilimi veriden soru ve sonuç çıkarır; büyük veri sistemleri bunu veri tek makineye sığmadığında mümkün kılar.

AYRIŞMA

Ayrıştığı Noktalar (2/2)

Ölçüt	Büyük Veri	GPU ve Paralel Hesaplama	MLOps
Temel soru	Tek bir makineye sığmayan veri nasıl saklanır ve işlenir?	Binlerce çekirdek aynı hesabı nasıl paylaşır?	Bir model laboratuvardan üretime nasıl taşınır ve orada nasıl ayakta tutulur?
Yaklaşım	dağıtık depolama ve hesaplama	veri paralel hesaplama	yazılım mühendisliği ve MÖ
Girdi	çok büyük ve hızlı akan veri	paralelleştirilebilir algoritma	eğitilmiş model, veri boru hattı
Çıktı	işlenmiş veri, sorgu sonucu	hızlandırılmış program	çalışan ve izlenen hizmet
Başarı ölçütü	ölçeklenme, gecikme, maliyet	hızlanma, bellek bant genişliği	gecikme, kesintisizlik, veri kayması
Matematik yükü	düşük	düşük-orta	düşük
Veri ihtiyacı	çok fazla	gerekmez	akan üretim verisi

- **GPU ve Paralel Hesaplama:** GPU tek makinede binlerce çekirdekle hesabı hızlandırır; büyük veri sistemleri veriyi makinelere dağıtır, darboğaz ağ ve disktir.
- **MLOps:** MLOps modelin yaşam döngüsünü yönetir; büyük veri sistemleri onu besleyen veri boru hatlarını ve depolamayı kurar.

UYGULAMA

Uygulama Alanları

Günlük analizi
Sunucu günlüklerinden hata ve trafik.

Tıklama akışı
Kullanıcı davranışını anlık izlemek.

Öneri altyapısı
Milyonlarca kullanıcının geçmişi.

Nesnelerin interneti
Algılayıcılardan sürekli gelen ölçüm.

Finansal işlemler
Anlık sahtekârlık uyarısı.

Arama dizini
Web sayfalarını taramak ve dizinlemek.

UYGULAMA

Riskler ve Etik

Sessiz veri kaybı

Yanlış ayarlanmış kopyalama ya da saklama süresi dolup silinen Kafka kayıtları, kimse fark etmeden veriyi yok edebilir.

Her şeyi saklama eğilimi

Depolama ucuz diye toplanan kişisel veri sızıntı yüzeyini büyütür; KVKK verinin amaçla sınırlı ve ölçülü tutulmasını ister.

Silme hakkını uygulamak

Bir kişinin verisi yedeklere, kopyalara ve türetilmiş tablolara dağılır; silme talebini eksiksiz karşılamak zordur.

Bulut faturası ve enerji

Açık unutulan bir küme ya da gereksiz tam tablo taraması büyük bir fatura ve boşa harcanan enerji demektir.

Ayrıntılı tartışma: Açıklanabilir Yapay Zekâ ve Yapay Zekâ Etiği.

UYGULAMA

Araçlar ve Örnek Veriler

Araçlar ve kütüphaneler

Apache Hadoop | Apache Spark | Apache Kafka | Apache Flink | Apache Cassandra | MongoDB | Apache Hive

Tanınmış veri setleri

Common Crawl | Wikipedia dökümleri | NYC TLC taksi kayıtları | GDELT | Google Books Ngram

UYGULAMA

Sık Karıştırılanlar

Yaygın düşünce	Doğrusu
Büyük veri çok veri demektir.	Ölçüt boyut kadar hız ve çeşitliliktir; asıl soru verinin tek makineye sığıp sığmadığıdır.
Dağıtık sistem her zaman daha hızlıdır.	Küçük veride tek makine (pandas, SQL) çoğu zaman daha hızlıdır; dağıtmanın ek yükü vardır.
Makineyi iki katına çıkarmak süreyi yarılar.	Karıştırma (shuffle), ağ trafiği ve en yavaş görev hızlanmayı sınırlar.
NoSQL, SQL'in yerini aldı.	NoSQL farklı tutarlılık ödünleşimleri sunar; Spark SQL ve Hive gibi SQL motorları yaygındır.

SONRASI

Bu Dersten Sonra

Bu dersi temel alan ya da genişleten dersler

Bu ders çoğunlukla uç noktadır; öteki dersleri temel alır.

İlgili meslekler

Veri mühendisi | Büyük veri mimarı | Veri platformu mühendisi | Bulut mühendisi

Büyük veri sistemleri, veri kısmının altyapısıdır. Buradan modelleri üretime taşıyan MLOps'a, hesabı hızlandıran GPU ve paralel hesaplamaya ya da bu altyapının üzerinde çalışan veri madenciliği ve veri bilimine geçilebilir.

SONRASI

Kaynaklar

- Kleppmann, *Designing Data-Intensive Applications*, O'Reilly, 2017.
- White, *Hadoop: The Definitive Guide*, 4. baskı, O'Reilly, 2015.
- Chambers ve Zaharia, *Spark: The Definitive Guide*, O'Reilly, 2018.
- Dean ve Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", *OSDI*, 2004.

Son

Bölüm 27: Büyük Veri Sistemleri

← Önceki bölüm · Alanın haritası · Sonraki bölüm →