Çankaya Üniversitesi

3. Mühendislik ve Teknoloji Sempozyumu

Geniş Veri Kümeleri Üzerinde Paralel Veri Madenciliği Yaklaşımları: Wavecluster Yöntemi ile Öbekleme Uygulaması

Ahmet Artu YILDIRIM
Efe ÇİFTCİ
Cem ÖZDOĞAN

Nisan 2010

Genel İçerik

- Veri Madenciliği
- Öbekleme Analizi
- Wavecluster Algoritması
- Paralel Wavecluster Algoritması
- Sonuçlar

Veri Madenciliği

- Büyük miktardaki veri kümesinin içinde bulunan gizli örüntüleri keşfetme süreci
- Kullanım amacı:
  - Geleceğe dönük tahminleme yapma
  - Veri elemanlarını tanımlayabilme
- Günümüzde astronomi, pazarlama, kuantum kimyası, kredi kartı dolandırıcılığı tespiti vs. konularında sıkça kullanılmaktadır

Öbekleme Analizi

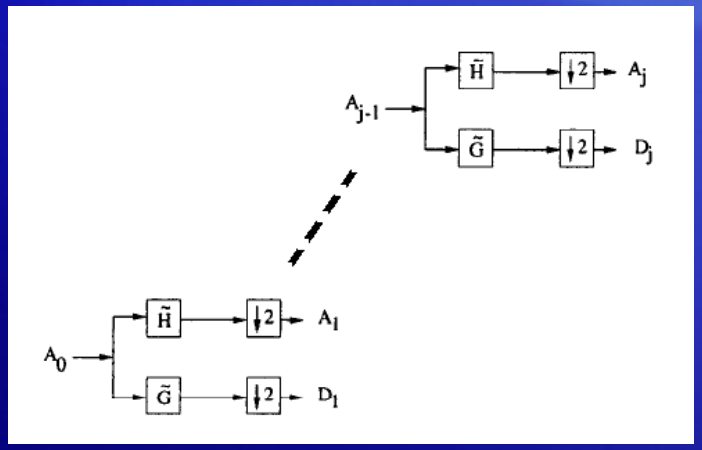
- Veri kümesindeki nesneleri belirli bir benzerlik kıstasına göre önceden tanımlanmamış öbeklere ayırma süreci
- Örüntü tanımada, coğrafi bilgi sistemlerinde, görüntü işlemede, makine öğrenmesinde, web dokümanlarının sınıflandırılması vs. kullanılmaktadır
- İyi bir öbekleme algoritması
  - Öbek şeklinden bağımsız olmalı (konkav şekil)
  - Gürültü veya aykırı değerlerden etkilenmemeli

Wavecluster Algoritması

- Çok boyutlu uzamsal veri kümeleri için kullanılan ızgara tabanlı öbekleme analizi algoritması
- Uzamsal veri kümesinden frekans alanına dönüştürmek için wavelet dönüşümü uygulanır
- Dönüştürülmüş frekans alanı üzerindeki yoğun bölgeler öbek olarak tanımlanır
- İyi bir öbekleme algoritmasında olan tüm koşulları sağlamaktadır

Wavelet Dönüşümü

- Filtreleme yaparak, işaretin zaman-frekans gösterimini sağlayan dönüşüm türü
- Kullanım amacı; veri kümesini yoğunlaştırmak, aykırı değerlerden arındırmak ve farklı seviyelerde öbek tespit etmek



- Haar wavelet, daubechies wavelet, mexican hat wavelet, meyer wavelet, coiflets wavelet …

Şekil 1- Çok-çözünürlüklü Wavelet Dönüşümü

Wavecluster Algoritması Adımları

Girdi: Çok boyutlu veri nesneleri örnek vektörleri
Çıktı: Öbeklere ayrılmış nesneler
1. Öznitelik uzayındaki nesneleri birimlere atayarak sayısallaştır
2. Sayısallaştırılmış veri kümesi üzerinde wavelet dönüşümü uygula
3. Yüksek frekanslı bileşen üzerinde farklı seviyelerde bağlı parçaları bul
4. Bağlı ünitelere öbek numarası ile etiket ata
5. Arama tablosunu oluştur
6. Arama tablosunu kullanarak nesneleri öbeklerle ilişkilendir.

Waveletcluster Örnek Çalışma

a) Orijinal Veri Kümesi

b) ρ = 2, öbek sayısı = 6

c) ρ = 3, öbek sayısı = 3

Şekil 2- Farklı ρ (wavelet dönüşüm uygulama sayısı) değerleri için oluşturulan düşük frekanslı bileşenler ve tespit edilen öbek sayısı

Paralel Wavecluster Algoritması

Wavecluster algoritmasının uygulanmasında karşılaşılan sorunlar:

1. Hafıza yetersizliğinden dolayı çok büyük veri kümeleri ile çalışılamıyor
2. Büyük veri kümeleri için çalışma zamanı fazla

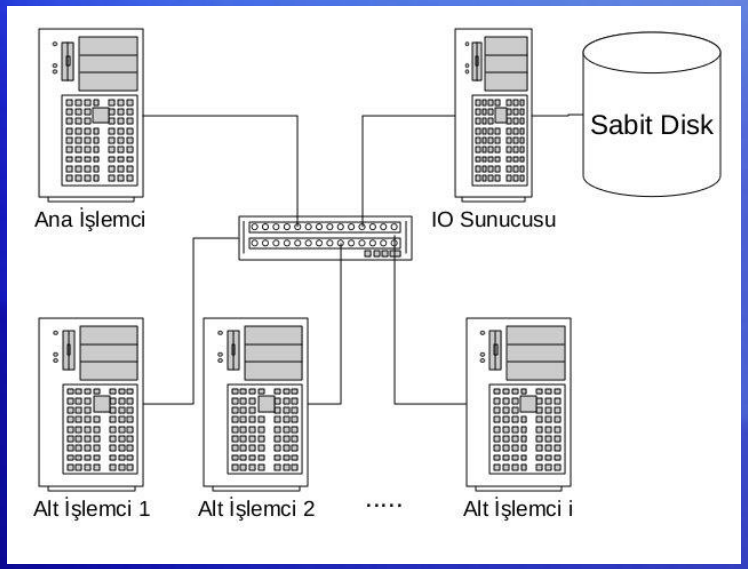
Çözümümüz:

- Geniş veri kümesini işlemciler arasında dağıtarak wavecluster algoritmasını koşut zamanlı hale getirmek.

Paralel Hesaplama

- Sıralı çalışan sürecin yaptığı işi, işlemciler arasında dağıtarak eş zamanlı hesaplama yaptırmak
- Bu şekilde;
  - Algoritmanın çalışma zamanın kısaltmak
  - Kaynakları etkin bir şekilde kullanmak

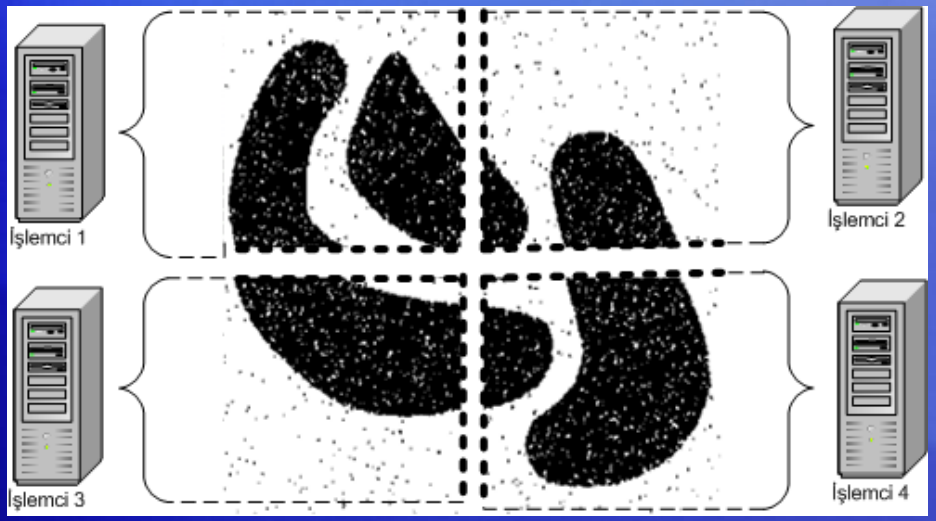
Çankaya Üniversitesi Boron Bilgisayar Öbek Sistemi



Şekil 3 – Boron Bilgisayar Öbek Sistemi

- Dağıtık hafıza sistemi tabanlı
- Sistem yıldız ağ mimarisine sahip
- İşlemciler birbirlerine hızlı eternet anahtarlama cihazı ile bağlı
- Tüm işlemciler veri kümesinin bulunduğu sabit diske ağ üzerinden erişebiliyorlar

Paralel Wavecluster Veri Kümesi Dağıtılması



- Tüm işlemciler geniş veri kümesini bölümlemeli olarak eşit miktarda paylaşmaktalar

Şekil 4 – Dört işlemcili paralel sistem için veri kümesinin dağıtılması örnek gösterimi

Paralel Wavecluster Algoritması

- İşlemciler birbirleri ile mesaj geçirme arayüzü MPI (Message Passing Interface) kullanarak haberleşmekteler
- Metodoloji olarak ana işlemci – alt işlemci modeli uygulandı
- Ana işlemcinin görevi:
  - Koordinat bilgisi yollayarak veri paylaşımı yapmak
  - Alt işlemcilerden aldığı düşük frekanslı bileşene ait sınır veri kümelerini karşılaştırmak, nesne komşuluğuna göre her bir işlemci için birleştirme tablosunu oluşturmak ve sonucu işlemcilere bildirmek

Paralel Wavecluster Algoritması

- Alt işlemcilerin görevi:
  - Yerel veri kümesi üzerinde wavelet dönüşümü uygulamak
  - Düşük frekanslı bileşen üzerinde bağlı parçaları işaretleme algoritmasını çalıştırmak ve yerel öbekleri tespit etmek
  - Ana işlemciden gelen birleştirme tablosuna göre öbek numaraların güncellemek
  - Arama tablosu vasıtası ile öbekleri orijinal veri kümesindeki nesnelerle ilişkilendirmek

Sonuçlar

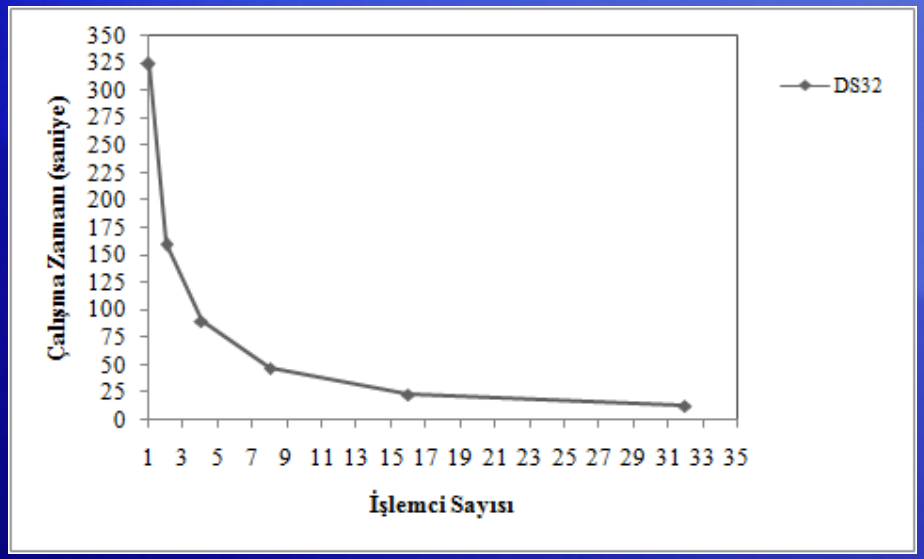
- Çalışmamız Çankaya Üniversitesinde bulunan boron bilgisayar öbeğinde çalıştırıldı, 2.34 GHz gücünde 32 işlemciden faydalanıldı
- Algoritma C programlama dili kullanılarak geliştirildi
- İşlemciler arasındaki iletişimi sağlamak için MPI kütüphanesi olan OpenMPI kütüphanesi kullanıldı, yardımcı kütüphane olarak GLib kütüphanesi kullanıldı
- Çalışmamızda sentetik olarak oluşturulmuş iki boyutlu DS32 (1.073.741.824 nesne) geniş veri kümesi kullanıldı

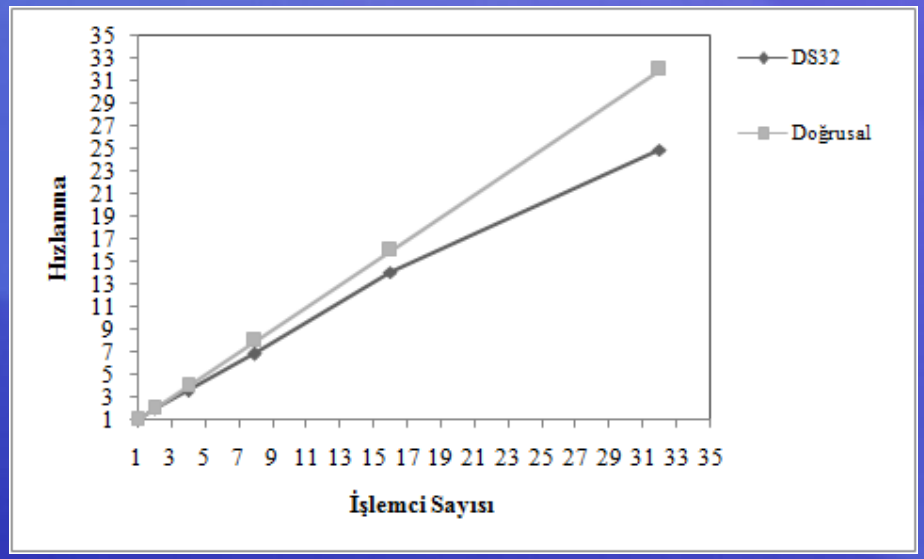
Sonuçlar

Hızlanma Katsayısı = t_s / t_p

t_s : tek işlemcili sıralı sistemin çalışma süresi

t_p : birden fazla işlemcili paralel sistemin çalışma süresi





Şekil 5 – ρ = 3 ve işlemci sayısı = 1, 2, 4, 8, 16, 32 için, Çalışma Süresi ve Hızlanma Grafikleri

Sonuçlar

- Paralel wavecluster algoritmamızın doğrusallık karakteristiği gösteriyor
- Bellek miktarının yetmediği veri kümeleri ile çalışılabilir
- Wavelet dönüşümü sayısı arttıkça paralel algoritmanın çalışma süresi düşüyor
- Algoritmamız geniş veri kümeleri için oldukça uygun

Teşekkür ederiz…