Flowing Data Cluster for KD-AR Stream Approach

Akan Veri Kümeleme İçin KD-AR Stream Yaklaşımı

Melike AKALAN

Kırıkkale Üniversitesi

ABSTRACT

Flowing data clustering is a highly sought-after topic in which the increasing amount of data has reached a serious dimension. Some deficiencies in the data clustering methods flowing in the increase in the amount of data brought with it, therefore the KD-AR algorithm was developed. It is tried to briefly explain what is an approach that aggregates K-dimensional tree and adaptive radius-based (KD-AR Stream) real-time flowing data, its application style, performance tests and its contribution to flowing data studies.

Keywords: KD-AR stream, addaptive radius, time-based data summarization, evolutionary clustering

ÖZET

Akan veri kümeleme, artan veri miktarının ciddi bir boyuta geldiği günümüzün oldukça rağbet gören konusudur. Veri miktarındaki artışta akan veri kümeleme yöntemlerindeki bazı eksiklikleri beraberinde getirmiş bu sebeple KD-AR algoritması geliştirilmiştir. K-boyutlu ağaç ve uyarlanabilir yarıçap tabanlı (KD-AR Stream) gerçek zamanlı akan verileri kümeleyen bir yaklaşımdır. Makalede bu yaklaşımın ne olduğu, uygulanış biçimi, performans testleri ve akan veri çalışmalarına olan katkısı kısaca anlatılmaya çalışılmıştır.

Anahtar Kelimeler: KD-AR stream, uyarlanabilir yarıçap, zaman tabanlı veri özetleme, evrimsel kümeleme

1. GİRİŞ

Günümüzde kullanılan akıllı telefonlar, tabletler vb. cihazlar üzerinden Twitter, Instagram, Facebook gibi sosyal ağlar bilgisayar ortamına aktarılan veri miktarında ciddi artışa neden olmaktadır. Bu veriler de sürekli güncellendiğinden, değiştiğinden bir nevi akan veriyi oluşturur aslında. Akan veri kümeleme; saldırı tespit sistemleri, finansal uygulamalar, bilimsel araştırmalar, sağlık araştırmaları, nesnelerin interneti (IoT) ve mobil uygulamalar gibi pek çok alanda kullanılmaktadır.

Bildiğiniz üzere veriler; tek başına anlamı olmayan varlıklardı, üzerinde işlem yapılmadıkları takdirde arşiv özelliğinde saklanabiliyorlardı; fakat bu veriler arasından aradığımız anlamlı, doğru ve geçerli verilere ulaşabilmek için birtakım aşamalardan geçerek bilgiye dönüştürülmesi gerekir. İşte bu noktada karşımıza çıkan klasik veri işleme yöntemleri; giderek hızla artan veri karşısında yetersiz kalarak, yerlerini **"gerçek zamanlı akan veri kümeleme"** yöntemine bırakmışlardır.

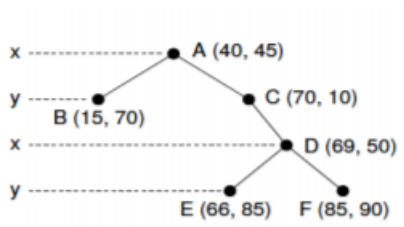
Akan veri, miktar olarak çok büyük, sonsuz ve süreklidir. Veri sürekli değiştiğinden veri hakkında tahminde bulunmak zordur. Buna bağlı olarak yarıçap ve küme sayısı gibi parametreleri doğru bir şekilde belirlenmesi de zorlaşır. Dolayısıyla geliştirilecek yöntemin ya bu değerleri otomatik belirlemesi ya da esnek bir yapıda olması gerekir. Bize de bunu **"KD-AR Stream algoritması"** sağlar. Çünkü KD-AR Stream'de küme sayısını belirlemeye, ona bir sınır koymaya gerek yoktur; küme yarıçapı ise uyarlanabilir bir yapıdadır.

2. GELİŞTİRİLEN YÖNTEM

2.1 K-Boyutlu Ağaç ve Alan Araştırması

K-boyutlu ağaçlarda belirli bir aralıktaki verilere ulaşma çabası bir alan araştırması işlemidir.

K-boyutlu ağaç çok boyutluluğu destekleyen bir dengeli ikili arama ağacı türüdür. Veriler ağaca yerleştirilirken veya ağaçta arama yapılırken ağacın her seviyesi farklı bir niteliğe göre sıralanmaktadır. Aşağıdaki gibi iki boyutlu veriler ağaca yerleştirilecekse, ilk önce x niteliğine sonrasında y niteliğine göre yerleştirilmektedir ve tekrar x'e geri giderek işlem tekrarlanır.

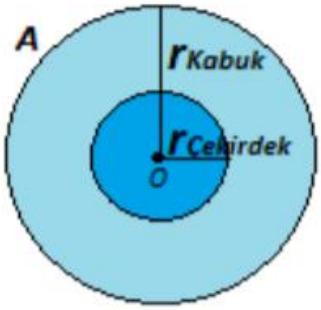


2.2 Zaman Tabanlı Veri Özetleme

Zaman tabanlı olarak verileri özetleyen KD-AR Stream'de; veriler kullanıcıların belirlediği son 1 saniyede, son 1 saatte gelen veriler şeklinde özetlenebilmektedir. Ancak bu süre zarfında gelen veri miktarı 100 yahut 1000 olabileceği gibi on binlerle ifade edilen veri de gelebilir. Bu durumda bu verileri işleme imkânı da azalır. Bu problemi aşmak için, söz konusu bu verilerden en son gelen TN (ön tanımlı değişken) tanesi alınarak işlem yapılmaktadır.

2.3 Uyarlanabilir Yarıçap

Önerdiğimiz bu yaklaşımın temel özelliklerinden birisi de küme yarıçapının uyarlanabilir olmasıdır. Gelen verinin en yakın küme yarıçapına olan uzaklığına göre küme yarıçapı güncellenir. Bir veri kümesinde iki çeşit yarıçap vardır:



- Kabuk yarıçap: Kümeye ait verilerden küme merkezine en uzak olan verinin uzaklığıdır.
- Çekirdek yarıçap: Kümede verilerinin yoğunlaştığı alan yarıçapıdır.

Kümenin çekirdek yarıçapını bulurken aşağıdaki formül kullanılır.

$$\sigma = \frac{1}{d}\sum_{j}^{d}\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(X_i^j - \mu_i^j\right)^2}$$

Formulde verilen d verilerin sahip olduğu nitelik sayısı, N kümenin sahip olduğu veri sayısı, μ j niteliğine ait ağırlık merkezi değeri ve Xi üzerinde işlem yapılan veridir.

2.4 Temel İşlemler

2.4.1 Yeni bir küme oluşturma

Belirli yarıçap içinde ve diğer kümelerden yeterince uzakta belirli bir sayıdaki veri yeni bir küme olarak tanımlanır.

Yeni küme oluşturmak için hiçbir kümeye ait olmayan veriler K-boyutlu ağaca yerleştirilir ve bu veriler üzerinde alan araştırması işlemi gerçekleştirilir. Amaç hiçbir kümeye ait olmayan N (kullanıcının belirlediği ön tanımlı değişken) tane veri belirlenmiş yarıçapta (r – ön tanımlı değişken) bulunuyorsa ve bu verilerin oluşturduğu aday kümenin merkezinin diğer kümelerin merkezlerine olan uzaklığı yeterli ise bu aday küme yeni bir küme olarak tanımlanır.

2.4.2 Küme yarıçapının artırılması ve azaltılması

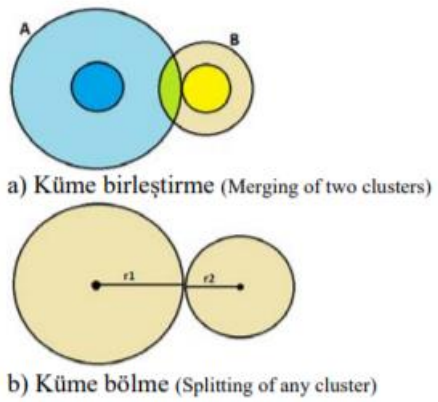
Yeni bir veri geldiği zaman bu verinin en yakın olduğu küme tespit edilir. Verinin küme merkezine olan uzaklığı, eğer kümenin kabuk yarıçapından az ise yani veri kümenin içinde ise veri ilgili kümeye atanır ve küme merkezi güncellenir. Verinin küme merkezine olan uzaklığı, kümenin kabuk yarıçapından fazla, maksimum yarıçaptan az ve kümenin kabuk yarıçapı ile yarıçap arttırma eşik değerinin toplamından az ise bu veri bu kümeye atanır.

Bir kümenin kabuk yarıçapı belirlenmiş olan maksimum yarıçap değerine kadar arttırılabilir; ancak söz konusu artış aşama aşama gerçekleştirilmektedir. Bu artış miktarı her seferinde belirlenmiş yarıçap arttırma eşik değeri kadar yapılabilmektedir. Böyle bir yapının kullanılmasının nedeni kümelerin daha kararlı bir yapıda olmasını sağlamak ve sapan verilere karşı kümeleri daha dirençli hale getirmektir.

2.4.3 Veri yaşlandırma ve ömrünü tamamlayan verilerin silinmesi

Anlatıldığı üzere KD-AR Stream zaman tabanlı bir özetleme yaklaşımı kullanır. Bu nedenle her bir verinin ömrü belirlenmiş olan süre kadardır. Ömrünü tamamlayan her veri silinir. Silinen veri hiçbir kümeye ait olmayan bir veri olabileceği gibi bir kümeye de ait olabilir. Eğer silinen veri bir kümeye aitse bu kümenin kabuk yarıçapı, çekirdek yarıçapı, merkez koordinatları ve sahip olduğu veri sayısı güncellenmektedir.

2.4.4 Küme Birleştirme/Bölme



a) Küme birleştirme (Merging of two clusters)

b) Küme bölme (Splitting of any cluster)

İki kümenin birleştirilip birleştirilemeyeceğine karar verirken merkezleri arasındaki Öklid uzaklığına bakılır. Eğer bir kümenin kabuk yarıçapı başka bir kümenin çekirdek yarıçapı ile kesişiyorsa bu iki küme birleştirilir.

Aktif bir kümede bulunan verilerden belirli sayıdaki bir kısmının oluşturduğu aday kümenin kabuk yarıçapı ile kümenin geri kalanının oluşturduğu kümenin kabuk yarıçapı kesişmiyorsa bu iki küme bölünür.

2.4.5 Aktif bir kümenin pasif yapılması ve pasif bir kümenin yeniden aktive edilmesi

Aktif bir kümenin sahip olduğu veri sayısı verilerin zamana bağlı olarak silinmesi nedeniyle küme oluşturmak için belirlenmiş olan eşik değerin altına düşerse bu küme pasif yapılır. Pasif durumdaki bir kümenin sahip olduğu veri sayısı yeni veri alması sonucunda eşik değerin üzerine tekrar çıkarsa bu küme yeniden aktif hale getirilir.

2.5 Evrimsel Bir Akan Veri Kümeleme

KD-AR Stream yukarıda saydığımız tüm bu işlemleri(yeni küme oluşturma, var olan bir kümeyi pasif yapma, pasif bir kümeyi yeniden aktive etme, iki kümeyi birleştirme veya bir kümeyi ikiye bölme ve küme yarıçapını güncelleme) evrimsel olarak yani gerçek zamanlı olarak yapmaktadır.

2.6 Algoritma

KD-AR Stream algoritmasında kullanıcının tanımlaması gereken 6 tane değişken vardır. Bunlardan N, küme oluşturmak için gereken minimum veri sayısına karşılık gelirken, t her bir verinin ömrünü tanımlamaktadır. Belirlenmiş t sürede gelen veri miktarı çok fazla olursa en son gelen TN tane veri özet olarak alınır. r değişkeni, yeni küme tanımlarken minimum N tane verinin bulunması gereken alanın yarıçapını ifade etmektedir. r_threshold değişkeni küme yarıçapını arttırma/azaltma eşik değerini ifade ederken, r_max ise küme yarıçapının ulaşabileceği maksimum değeri ifade eder.

KD-AR Stream'in temel algoritması aşağıda verilmiştir.

While yeni bir veri geldiginde **do**
  Yaslandirma;
  YeniKumeOlustur;
  CakisanKumeleriBul;
  KumeBol;
  EnYakınKumeyiBul;
  KumeMerkezleriniGuncelle;
  YarıcaplariGuncelle;
  KumeleriAktiflestir;
End

"Yaşlandırma" algoritmasında, işlenen tüm veriler yaşlandırılmakta ve ömrünü tamamlayan veriler silinerek silinenVeri değişkenine eklenmektedir. Veri ömrü eşik değeri, uygulamanın türüne ve ihtiyaca göre belirlenmelidir. Örneğin bankacılık gibi bir uygulamada her verinin ömrü için 10 saniye veya 10 dakika gibi bir süre belirlenirken, Corona Virüs gibi salgın izleme uygulaması için 1 gün veya 1 hafta olarak belirlenebilir.

"YeniKumeOlustur", "CakisanKumeleriBul" ve "KumeBol" algoritmalarından yukarıda bahsedilmiştir.

"EnYakınKümeyiBul" algoritması yeni gelen veya hiçbir kümeye ait olmayan veriler arasında kümelerin durumunun değişmesi nedeniyle ilgili kümelere dâhil edilmek için yeterli yakınlığa ulaşan veri var ise bu verileri ilgili kümelere ekler.

"KümeMerkezleriniGüncelle" kümelerin merkez koordinatlarını günceller. Ancak bu güncelleme işlemini kümenin aktiflik durumuna bağlı olarak iki şekilde yapmaktadır. Eğer küme aktif bir küme ise

bu kümenin merkez koordinatlarını kümenin aktif verileri üzerinden hesaplar. Ama eğer küme pasif bir küme ise bu durumda kümenin merkez koordinatlarını varsa kümenin aktif verileri ve kümenin silinmiş verileri üzerinden hesaplar.

"YarıçaplarıGüncelle" tüm kümeler için çekirdek yarıçap ve kabuk yarıçap değerlerini hesaplar. Eğer küme aktif bir küme değilse kümenin kabuk yarıçapını r/2 olarak atar. Bu değerin atanmasının nedeni kümeyi sapan verilerden korumak içindir.

"KümeleriAktifleştir" her kümenin sahip olduğu veri sayısına bakarak kümelerin aktif mi yoksa pasif mi olacağına karar vermektedir.

Burada verilmeyen "KBAğacıOluştur" algoritması aldığı verileri bir K-boyutlu ağaca yerleştirir. Özyinelemeli bir yapıda çalışarak verileri sağ ve sol alt ağaçlara yerleştirir. Bu şekilde oluşturduğu ağacı döndürür.

3. PERFORMANS TESTLERİ VE TARTIŞMA

KD-AR Stream'in başarısını ölçmek adına akan veri kümeleme alanında kullanılan bazı algoritmalar(CEDAS, DPStream, SE-Stream) ile test sonuçları karşılaştırılmıştır.

Karşılaştırma işlemi hem kümeleme başarısı, hem de zaman karmaşıklığı açısından gerçekleştirilmiştir. Tüm karşılaştırma işlemleri Intel® Core™ i5-4460S CPU 2.90 GHz işlemcili ve 8 GB RAM kapasiteli ve Windows 10 yüklü bir bilgisayarda, Matlab 2017 versiyonu kullanılarak gerçekleştirilmiştir.

3.1 Veri Seti

Algoritmaların başarılarını karşılaştırmak için inceleyeceğimiz performans testlerinde "KDD" veri setinin 50.000 verilik bir kısmı kullanılmıştır. Bu veri seti ağ saldırısı verilerinden oluşmaktadır. Ayrıca veri setinde gerçek veriler kullanılmıştır.

3.2 Kümeleme Başarısını Ölçme

Kümeleme başarısını ölçmek için Saflık (Purity), Doğruluk (Accuracy), F-Skor (F-Score) ve Silhouette İndeks parametreleri kullanılmıştır.

- **Saflık** küme etiketini küme sayısına bölerek yüzdesini alır. Kümelerin kendi içerisinde saflık yüzdesini bulmaya çalışan bir testtir.
- **Doğruluk** testi modelin ürettiği küme etiketlerini gerçek küme etiketleri ile karşılaştırarak modelin başarısını ölçen bir testtir.
- **Silhouette İndeks**, kümeleme başarısını farklı kümelere ait verilerin birbirinden ne kadar uzak ve aynı kümeye ait verilerin birbirine ne kadar yakın olduğuna bakarak ölçen testtir.

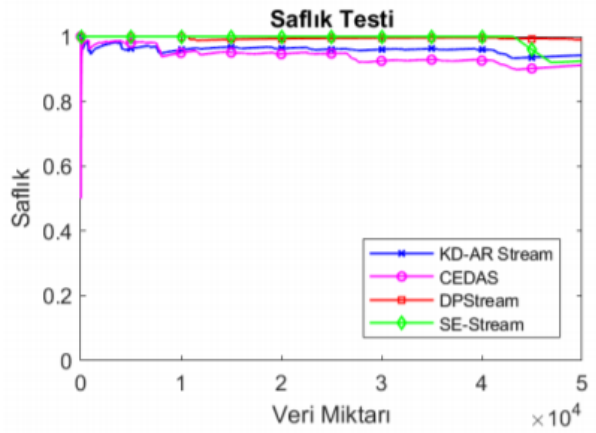
- **F-Skor** testi Kesinlik (Precision) ve Duyarlılık (Recall) değerlerinin harmonik ortalamasını hesaplar.

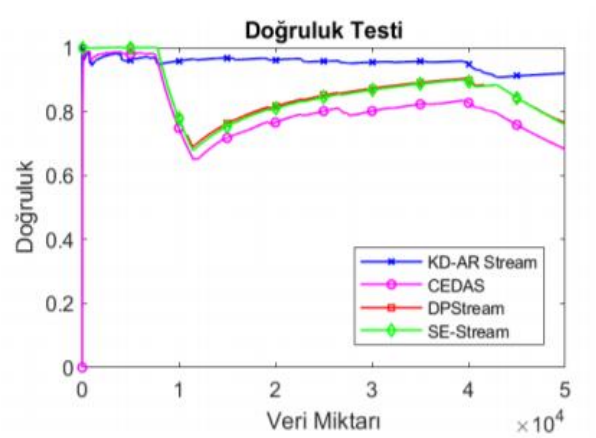
3.3 Test Sonuçları

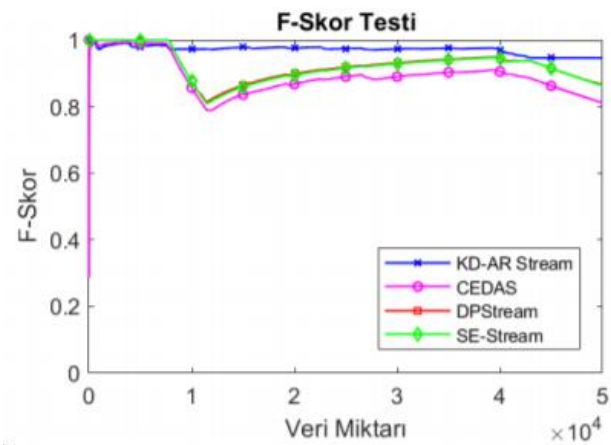
Grafiklerden de görülebileceği gibi KD-AR Stream'in, Saflık testinde DPStream ve SE-Stream algoritmaları dışında bütün testlerde en iyi sonucu ürettiği görülmektedir. DPStream ve SE-Stream algoritmaları neredeyse tüm verileri tek bir kümeye atadığından Saflık değerleri yüksek çıkmaktadır. KD-AR Stream' de ise veriler yarıçaplarına göre kümelenmektedir; dolayısıyla başarısını bir tek bu test ile anlayamayız.

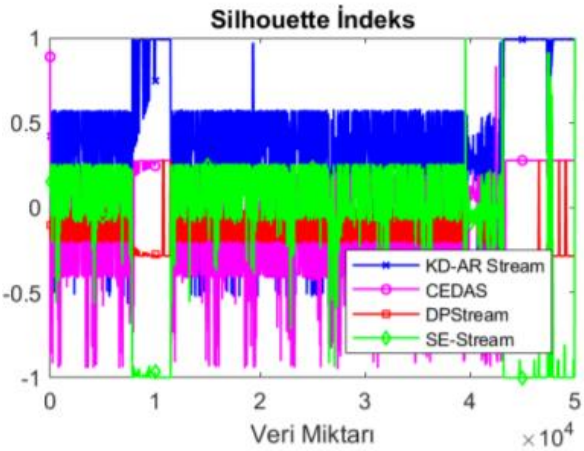
Doğruluk ve F-Skor grafiklerine baktığımız zaman KD-AR Stream algoritması dışındaki diğer algoritmalarda bazı bölgelerde oldukça önemli düşüşler görülmektedir. Bu düşüşlerin nedeni ise diğer algoritmaların bu bölümde oluşması gereken kümeleri tespit edememesinden kaynaklanmaktadır.

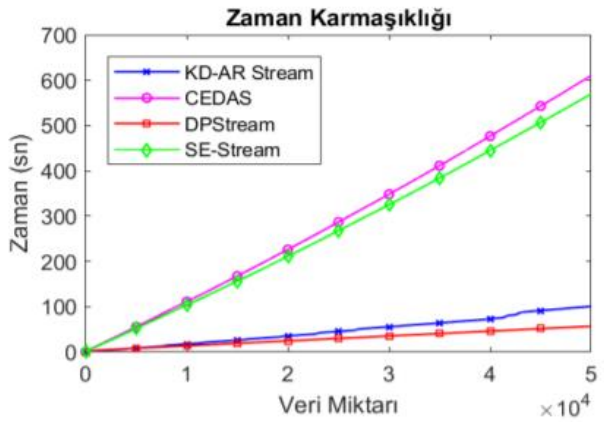
Silhouette İndeks sonuçları karşılaştırıldığında KD-AR Stream algoritmasının çok daha iyi sonuç verdiği görülmektedir. Bu da KD-AR Stream algoritmasının küme etiketinden bağımsız olarak kümeleri daha iyi ayrıştırdığını gösterir.



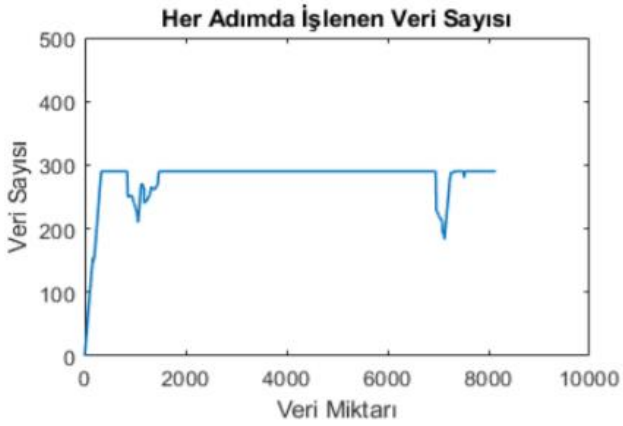








ŞEKİL 1



ŞEKİL 2

KD-AR Stream algoritmasında 6 adet değişken tanımlanması gerekmekteydi. SE-Stream'in 8, CEDAS algoritmasının 3 ve DPStream algoritmasının 8 adet değişken kullandığı dikkat edildiğinde KD-AR Stream algoritmasının daha uygun sayıda değişken kullandığı söylenebilir. Yine KD-AR Stream evrimsel değişimi destekleyen bir algoritmaydı. Bu nedenle ŞEKİL-1 den de anlaşılacağı üzere KD-AR Stream zamana bağlı olarak küme yapılarında oluşan değişimleri başarıyla gerçekleştirebilmektedir. Bu özellik kümeleme başarısını arttırmaktadır.

KD-AR Stream algoritması bilindiği üzere zaman tabanlı özetleme yapmaktaydı. Bu sayede belirlenmiş süre zarfında gelen veri sayısı beklenenden çok daha fazla olduğu zaman da kullanıcının tanımladığı kadarlık kısmını(TN) özet olarak almaktaydı. ŞEKİL-2 den de anlaşılacağı üzere t zamanda gelen veri bazen makul bazense eşik değerini aşmıştır. Bu durumda verileri işlemeye kolaylık sağlaması açısından en son gelen verilerin TN tanesi alınmıştır. Burada KD-AR Stream'in sapan verilere karşı dirençli bir algoritma olduğu gözlemlenebilir.

4. SONUÇ

Bu makalede veriyi tamamen çevrimiçi işleyen, uyarlanabilir yarıçap özelliğine sahip ve zaman tabanlı bir özetleme yapan KD-AR Stream algoritması önerilmiştir. Gerçekleştirilen performans testlerinde, bu algoritmanın diğer algoritmalarla kıyaslandığında, KD-AR Stream'in daya uygun bir sürede daha yüksek kümeleme başarısı gösterdiği görülmektedir.

Akan veri çalışmalarına Olan Katkıları:

- ✓ Tamamen çevrimiçi çalışması,
- ✓ Zaman tabanlı özetleme yapması,
- ✓ Evrimsel bir algoritma olması
- ✓ Yarıçap gibi parametrelerin uyarlanabilir olmasından ötürü akan verilerin daha sistematik bir şekilde daha gerçekçi olarak kümelemektedir.

Ayrıca zamanla sahip olduğu verileri kaybeden kümeleri silmek yerine pasif duruma getirmesi ve daha sonra bu tür kümelerin yeni veri alarak tekrar yeterli veriye sahip olması durumunda yeniden aktive etmesi KD-AR Stream'in, akan verilerin değişken yapısını destekleyen oldukça önemli bir diğer özelliğidir. İlerleyen zamanlarda akan veri kümeleme için KD-AR Stream algoritması üzerine daha çok gidilecek, performansını arttıracak ve parametrelerin otomatik seçilebileceği çalışmalar olacağı beklenilmektedir.

5. KAYNAKÇA

http://static.dergipark.org.tr/article-download/d2c6/d70f/ae62/5db2a66002e96.pdf?

https://acikerisim.sakarya.edu.tr/bitstream/handle/20.500.12619/81937/T02781.pdf?sequence=1&isAllowed=y

https://dspace.gazi.edu.tr/bitstream/handle/20.500.12602/151135/39e5c72b88d9204cf84cc80fa2bffe37.pdf;jsessionid=7FBE2F93D1F3A691F2961F71097A4992?sequence=1

http://nek.istanbul.edu.tr:4444/ekos/TEZ/45671.pdf

https://dergipark.org.tr/tr/download/article-file/677647