Avrupa Bilim ve Teknoloji Dergisi
Özel Sayı, S. 292-301, Ekim 2019
© Telif hakkı EJOSAT'a aittir
Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
Special Issue, pp. 292-301, October 2019
Copyright © 2019 EJOSAT
Research Article

ImpSlidingWindow: A New Model to Improve the Performance of the Sliding Window Based Streaming Data Summarization Method

Ali Şenol[1], Hacer Karacan[2]

[1] Bilgisayar Mühendisliği Bölümü / Mühendislik Fakültesi, Ardahan Üniversitesi, Türkiye (ORCID: 0000-0003-0364-2837)
[2] Bilgisayar Mühendisliği Bölümü / Mühendislik Fakültesi, Gazi Üniversitesi, Türkiye (ORCID: 0000-0001-6788-008X)

(This publication has been presented orally at HORA congress.)

(First received 1 August 2019 and in final form 25 October 2019)

(**DOI:** 10.31590/ejosat.638096)

ATIF/REFERENCE: Şenol, A., & Karacan, H. (2019). ImpSlidingWindow: A New Model to Improve the Performance of the Sliding Window Based Streaming Data Summarization Method. *European Journal of Science and Technology*, (Special Issue), 292-301.

Abstract

Sliding window based data summarization which is a quantity based summarization is commonly used in data stream clustering area in which the recent data is more important. In this data summarization method, w which is a predefined variable, of the most recent data is taken as the summary each time a new data arrives and the window slides one by one. This means that the model processes all the data in the data window each time a new data arrives. This approach causes the performance to reduce. Therefore, there is a need of new studies to be proposed in this area. In this study, a new sliding window model named ImpSlidingWindow (ISW) is proposed as a solution to the mentioned problem. In the proposed model, we propose that clustering model to work whenever a certain number of data accumulates instead of each data entry. With this new model, the sliding window width is divided into four equal parts and the clustering model works at the end of each part. As a result, a significant increase in the performance is achieved by enabling the clustering model to run four times instead of working as much as the number of data in the window width. When the proposed model applied to KD-AR Stream algorithm which is a proposed algorithm in the data stream clustering area, it has been found that up to 80% improvement obtained in run-time complexity.

Keywords: Data stream clustering, data summarization, sliding window.

ImpSlidingWindow: Kayan Pencere Tabanlı Akan Veri Özetleme Yönteminin Performansını Arttırmaya Yönelik Yeni Bir Model

Öz

Kayan pencere tabanlı veri özetleme, akan veri kümeleme alanında son gelen verilerin daha önemli olduğu uygulamalarda sıkça kullanılan miktar tabanlı bir veri özetleme yaklaşımıdır. Bu veri özetleme yaklaşımında, her yeni veri gelişinde ön tanımlı bir değişken olan en son gelen w tane veri özet olarak alınır ve pencere birer birer kaymaktadır. Yani model her yeni veri girişinde veri penceresinde bulunan tüm verileri işler. Bu da performansı olumsuz etkilemektedir. Bu nedenle bu probleme çözüm üretecek çalışmalara ihtiyaç duyulmaktadır. Bu çalışmada sözü edilen probleme çözüm olarak ImpSlidingWindow (ISW) isimli yeni bir kayan pencere modeli önerilmektedir. Önerilen modelde her veri girişinde kümeleme modelinin çalışması yerine belirli sayıda veri biriktikçe kümeleme modelinin çalışması önerilmektedir. Bu yeni model ile kayan pencere genişliği dört eşit parçaya bölünmekte ve her parçanın sonunda kümeleme modelinin çalışması sağlanmaktadır. Sonuç olarak pencere genişliğinde bulunan veri sayısı kadar kümeleme modelinin çalışması yerine dört defa çalışması sağlanarak performansta çok önemli bir artış sağlanmaktadır. Önerilen model akan veri kümeleme

[1] Ali Şenol: Ardahan Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Ardahan, Türkiye, ORCID: 0000-0003-0364-2837, alisenol@ardahan.edu.tr

alanında önerilmiş bir algoritma olan KD-AR Stream algoritmasına uygulandığında çalışma zamanında %80'lara varan iyileştirmeler elde edilmiştir.

Anahtar Kelimeler: Akan veri kümeleme, veri özetleme, kayan pencereler.

1. Giriş

Teknolojinin gelişmesinin ve internet hızının artmasının yanında; akıllı telefon, tablet ve bilgisayarlar gibi elektronik cihazlar ile Facebook, Twitter ve Instagram gibi sosyal medya ortamlarının kullanımının yagınlaşmasıyla berbaber bilgisayar ortamına aktarılan veri miktarında çok ciddi bir artış söz konusudur (AlNuaimi, Masud, Serhani ve Zaki, 2019; Görmüş, Aydın ve Ulutaş, 2018; Kanmaz ve Aydin, 2018; Martín, Julián ve Cos-Gayón, 2019; Oussous, Benjelloun, Ait Lahcen ve Belfkih, 2018). Bu nedenle klasik veri madenciliği yöntemleri artık yetersiz kalmaktadır. Dolayısıyla gerçek zamanlı veri işleme yöntemlerine ihtiyaç duyulmaktadır. Bu ihtiyacı karşılamya yöntelik önerilen akan veri kümeleme yaklaşımları son yılların popüler konularından biridir. Bu tür çalışmalar veri akarken gerçek zamanlı sonuç üretmeyi amaçlayan çalışmalardır.

Akan veri kümeleme tıklama verisi (Antonellis, Makris ve Tsirakis, 2009), saldırı tespit sistemleri (Li, 2014; Yin, Xia ve Wang, 2017, 2018), finansal uygulamalar (Hendricks, 2017), bilimsel araştırmalar (Aggarwal, 2010), sağlık araştırmaları (Gravina, Alinia, Ghasemzadeh ve Fortino, 2017; King, Villeneuve, White, Sherratt, Holderbaum ve Harwin, 2017; Manzi, Dario ve Cavallo, 2017), nesnelerin interneti (IoT) (Diaz-Rozo, Bielza ve Larrañaga, 2018) ve mobil uygulamalar (Tasnim, Caldas, Pissinou, Iyengar ve Ding, 2018) gibi pek çok alanda kullanılmaktadır (Ankleshwaria ve Dhobi, 2014; Ikonomovska, Loskovska ve Gjorgjevik, 2007; Şenol ve Karacan, 2018).

Akan veri kümeleme alanında pek çok çalışma önerilmiştir. STREAM (O'Callaghan, Mishra, Meyerson, Guha ve Motwani, 2002), CluStream (Aggarwal, Han, Wang ve Yu, 2003), STING (Wang, Yang ve Muntz, 1997), D-Stream (Tu ve Chen, 2009), MR-Stream (Wan, Ng, Dang, Yu ve Zhang, 2009), ClusTree (Kranen, Assent, Baldauf ve Seidl, 2011), HPStream (Charu, Jiawei, Jianyong ve Philip, 2004), DUCstream (Gao, Li, Zhang ve Tan, 2005), DenStream (Cao, Estert, Qian ve Zhou), E-Stream (Udommanetanakit, Rakthanmanon ve Waiyamai, 2007), SE-Stream (Chairukwattana, Kangkachit, Rakthanmanon ve Waiyamai, 2013), DD-Stream (Jia, Tan ve Yong, 2008), STREAMKM++ (Ackermann, Martens, Raupach, Swierkot, Lammersen ve Sohler, 2012), HDDStream (Ntoutsi, Zimek, Palpanas, Kröger ve Kriegel, 2012), LeaDen-Stream (Amini ve Wah, 2013), DBSTREAM (Hahsler ve Bolaños, 2016), FEAC-Stream (Silva, Hruschka ve Gama, 2017), DPStream (Xu, Wang, Li, Deng ve Gou, 2017), CEDAS (Hyde, Angelov ve MacKenzie, 2017), LLDStream (Laohakiat, Phimoltares ve Lursinsap, 2017), BOCEDS (Ahmed, 2019), StreamSW (Reddy ve Bindu, 2018) ve KD-AR Stream (Şenol ve Karacan, 2019) bunlardan bazıları olarak sayılabilir.

Akan veri kümeleme yaklaşımlarının odaklandığı temel noktalardan biri veriyi hızlı bir şekilde işlemektir. Çünkü teknolojinin gelişmesi ile beraber çok hızlı bir veri akışı söz konusudur. Bu yüzden veriyi akarken hızlı bir şekilde kümelemek gerekir. Ancak hem veri akış hızı hem de verinin sahip olduğu nitelik sayısı nedeniyle performans düşebilmektedir. Bu nedenle akan veri kümeleme alanında veri özetleme yaklaşımları kullanılmaktadır. Veri özetleme yaklaşımları belirli kurallara göre verinin bir alt kümesini almaktadır. Random Sampling (Guha, Rastogi ve Shim, 2001), Histogram tabanlı özetleme (Chairukwattana ve diğerleri, 2013; Charu ve diğerleri, 2004; Udommanetanakit ve diğerleri, 2007), Sliding Window (Kayan Pencere) (Badiozamany, Orsborn ve Risch, 2016; Datar, Gionis, Indyk ve Motwani, 2002; Reddy ve Bindu, 2018; Ren ve Ma, 2009), Micro-Cluster tabanlı özetleme (Hahsler ve Bolaños, 2016; Hyde ve diğerleri, 2017; Kranen ve diğerleri, 2011) ve Wavelet (Keim ve Heczko, 2001) gibi pek çok veri özetleme yaklaşımı kullanılmaktadır.

Kayan pencere tabanlı akan veri özetleme yaklaşımında kullanıcının belirlediği kadarlık veri kısmı özet olarak alınmaktadır. Yani her veri girişinde en son gelen *w* (ön tanımlı değişken) kadarlık veri miktarı özet olarak alınmakta ve kümelemeye tabi tutulmaktadır. Dolayısıyla pencere genişliğinde bulunan her veri *w* defa işlenmektedir. Bu da performansın düşmesine neden olmaktadır. Bu çalışmada bu probleme çözüm olarak kayan pencere tabanlı özetlemenin gelişmiş bir versiyonu olan ImpSlidingWindow yaklaşımı önerilmektedir. Önerdiğimiz bu yani yaklaşımda her veri girişinde veri özetini kümelemeye tabi tutmak yerine, her *w/4* veri biriktikçe veri özetinin kümelemeye tabi tutulmasını öneriyoruz. Bu yaklaşım ile hem makul bir veri kümeleme yaklaşımının elde edilmesi, hem de performsın arttırılması amaçlanmaktadır. Önerdiğimiz bu yeni model akan veri kümeleme alanında geliştirilmiş olan tamamen çevrim içi çalışan, evrimsel kümeleme yeteneğine sahip, uyarlanabilir yarıçap özelliği olan, kayan pencere tabanlı bir özetleme yaklaşımı kullanan, kümelerin geçmiş bilgilerini tutan ve yüksek kümeleme yeteneğine sahip bir algoritma olan KD-AR Stream uygulamasına uyarlanmış ve elde edilen sonuçlar karşılaştırılmıştır.

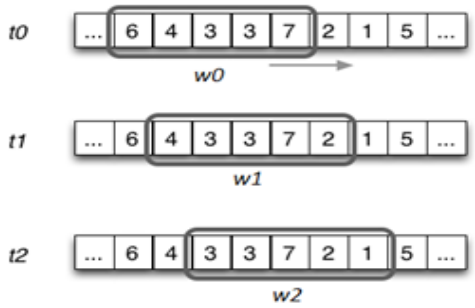
Çalışmanın geri kalanı şu şekilde sıralanmaktadır. 2. Bölümde önerdiğimiz yaklaşım detaylı olarak açıklanmaktadır. 3. Bölümde deneysel çalışma ve sonuçları değerendirilmekte iken 4. ve son bölümde sonuç ve öneriler yer almaktadır.

2. Materyal and Metot

2.1. Kayan Pencere (Sliding Window) Tabanlı Akan Veri Özetleme

Kayan pencere tabanlı veri özetlemede belirlenmiş bir pencere genişliğinde bulunan tüm veriler özet olarak alınır. Bu yaklaşımın en önemli özelliği son gelen veriler üzerinde işlem yapmasıdır. Bankacılık, güvenlik ve sağlık gibi son gelen verilerin daha önemli olduğu uygulamalarda sıkça kullanılmaktadır (Badiozamany ve diğerleri, 2016; Datar ve diğerleri, 2002; Reddy ve Bindu, 2018; Ren ve Ma, 2009). Bu özelliğinin yanında miktar veya zaman tabanlı çeşitlerinin bulunması ve kullanımının kolay olması nedeniyle de sıkça kullanılmaktadır. Ayrıca ölçeklenebilir bir yapısının olması da önemli bir diğer avantajı olarak öne çıkmaktadır.

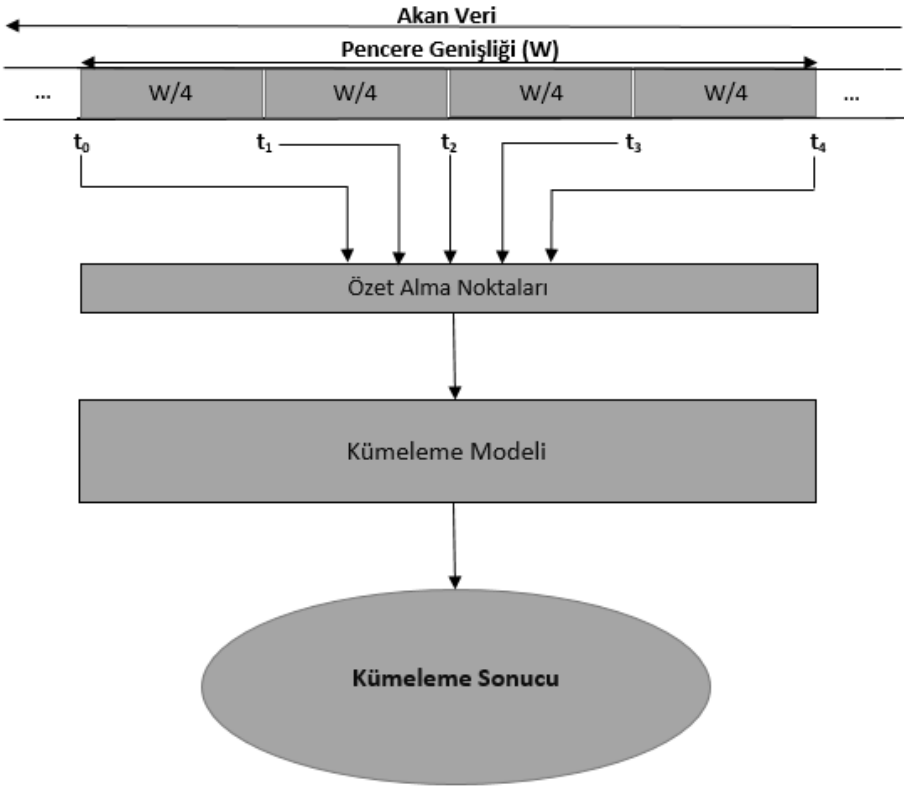
Kayan pencere tabanlı özetleme yaklaşımında her yeni veri girişinde son gelen *w* tane veri özet olarak alınır ve kümeleme modeline gönderilir. Şekil 1'de de görüldüğü gibi pencere her yeni veri girişinde bir kademe kaymaktadır. Dolayısıyla her veri için w kere işlenme söz konusudur. Bu da performansını belirli bir oranda düşürmektedir.



Şekil 1. Kayan pencere tabanlı veri özetleme örneği

2.2. ImpSlidingWindow: Kayan Pencere Tabanlı Akan Veri Özetleme Yönteminin Performansını Arttırmaya Yönelik Yeni Bir Model

ImpSlidingWindow (ISW) klasik kayan pencere tabanlı akan veri özetleme yaklaşımının her veri girişinde kümeleme modeline verileri göndermesi nedeniyle performansı düşürmesine çözüm olarak önerilen belirli sayıda veri biriktikçe özet veriyi kümeleme modeline gönderen bir yaklaşımdır. Şekil 2'de de görüldüğü gibi önerdiğimiz yaklaşımda her *w/4* veri biriktikçe pencere içerisinde (*w*) bulunan tüm veriler özet olarak kümeleme modeline gönderilmektetir. Bu yaklaşım ile verilerin gereksiz yere tekrar tekrar işlenmesinin önüne geçilmektedir. Bu da performansı arttırmaktadır.



Şekil 2. Önerilen ImpSlidingWindow (ISW)

Önerdiğimiz yaklaşım zaman karmaşıklığını düşürürken kümeleme başarısnı da yüksek tutmayı amaçlamakatadır. Çünkü yüksek performanslı kümeleme tek başına yeterli değildir. Aynı zamanda yüksek kümeleme başarısının da yakalanması gerekir. Bu amaca ulaşmak adına önerdiğimiz yaklaşım kümeleme modeline özet alma noktaklarında *w/4* tane veriyi göndermek yerine pencere genişliğinde bulunan tüm verileri göndermektedir.

Pencere genişliğinda çok sayıda özet alma noktasının olması peformansta aşağı yönlü bir eğilime neden olurken, kümeleme başarısında yukarı yönlü bir eğlime neden olacaktır. Benzer şekilde özet alma noktalarının sayısının azalması performansı arttırırken,

kümeleme başarısında aşağı yönlü bir eğlime neden olacaktır. Bu nedenle pencere genişliğinin 4 eşit parçaya bölünmesinin hem performans hem de kümeleme başarısında optimum değerlerin yakalanması açısından en iyi değer olacağı düşünülmektedir.

3. Sonuçların Değerlendirilmesi ve Tartışma

3.1. Deneysel Tasarım

Deneysel çalışmada UCI'nin KDD, Fisher Iris ve Breast Cancer ile DPStream (Xu ve diğerleri, 2017) algoritmasında kullanılan ExclaStar veri setleri kullanılmıştır. Kullanılan veri setlerinin özellikleri Tablo 1'de verilmiştir. Bilgisayar olarak Intel® Core™ i5-4460S CPU 2.90 GHz işlemcili ve 8 GB RAM kapasiteli ve Windows 10 yüklü bir bilgisayar ve Matlab 2017b kullanılmıştır.

Table 1. Kullanılan Veri Setlerinin Özellikleri

Veri Seti	Türü	Veri Miktarı	Nitelik Sayısı	Sınıf Sayısı	Özellik
KDD	*Gerçek*	*494020*	*38*	*23*	*Akan veri kümelemede sıkça kullanılır*
Fisher Iris	*Gerçek*	*150*	*4*	*3*	*Sıkça kullanılır*
Breast Cancer	*Gerçek*	*699*	*9*	*2*	*Veri Kümelemede sıkça kullanılır*
ExclaStar	*Sentetik*	*755*	*2*	*3*	*DPStream de kullanılan sentetik*

Önerdiğimiz yaklaşım KD-AR Stream algoritmasına uyarlanmış ve elde edilen sonuçlar yalın hali ile elde edilen sonuçlarla karşılaştırılmıştır. Önerdiğimiz yaklaşım için en yüksek Accuracy değerini üreten parametreler tespit edilerek kullanılmıştır. KD-AR Stream algoritması için ise çalışmada belirtilen parametreler kullanılmıştır. Her iki algoritmada kullanılan parametreler Tablo 2 ve Tablo 3'te verilmiştir.

Kümeleme başarısını değerlendirirken Purity, Accuracy, F-Score ve Silhouette indeks parametreleri üzerinden karşılaştırma yapılmıştır. Yaklaşımların performansını değerlendirirken de algoritmaların çalışma zamanları üzerinden karşılaştırma yapılmıştır.

Table 2. KD-AR Stream Algoritmasında Kullanılan Parametreler

	Veri setleri			
Parametreler	**KDD**	**Fisher Iris**	**Breast Cancer**	**ExclaStar**
N	*90*	*5*	*3*	*14*
Window_Size (w)	*160*	*90*	*200*	*50*
r	*1.4*	*1*	*7.5*	*4.5*
r_threshold	*4*	*0.55*	*2.5*	*1.5*
r_max	*6.55*	*1.55*	*10.35*	*6*

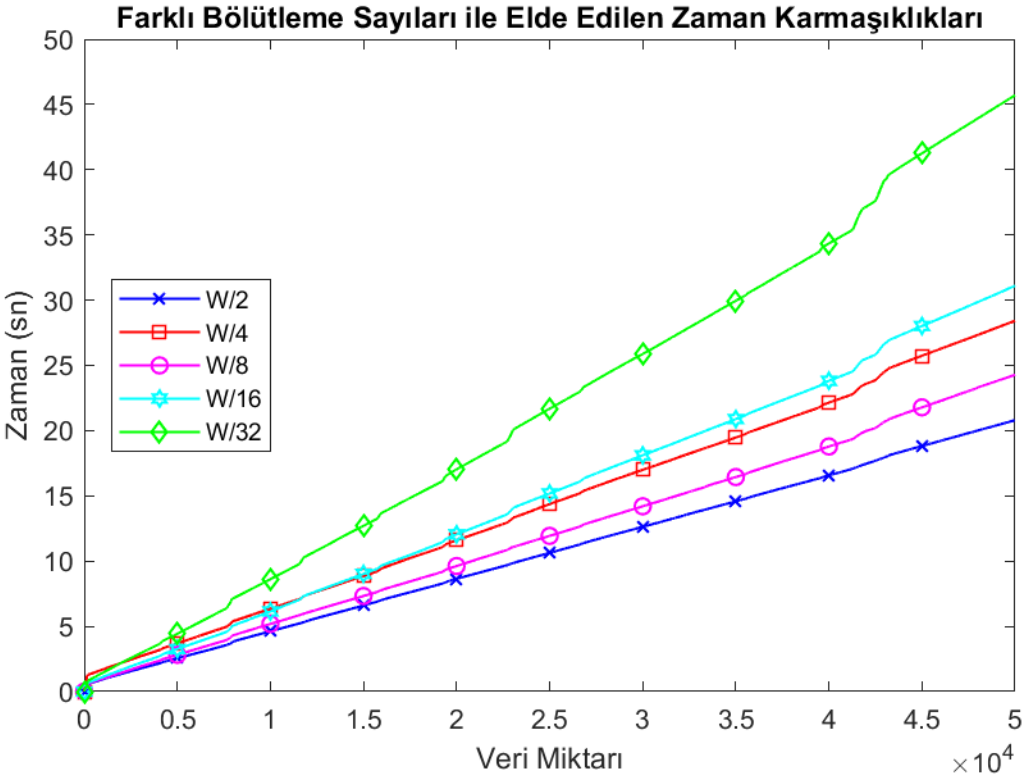
Table 3. ImpSlidingWindow Tabanlı KD-AR Stream Algoritmasında Kullanılan Parametreler

	Veri setleri			
Parametreler	**KDD**	**Fisher Iris**	**Breast Cancer**	**ExclaStar**
N	*90*	*5*	*3*	*14*
Window_Size (w)	*160*	*90*	*200*	*50*
r	*1.4*	*1*	*7.5*	*4.5*
r_threshold	*4*	*0.55*	*2.5*	*1.5*
r_max	*6,55*	*1.55*	*10.35*	*6*

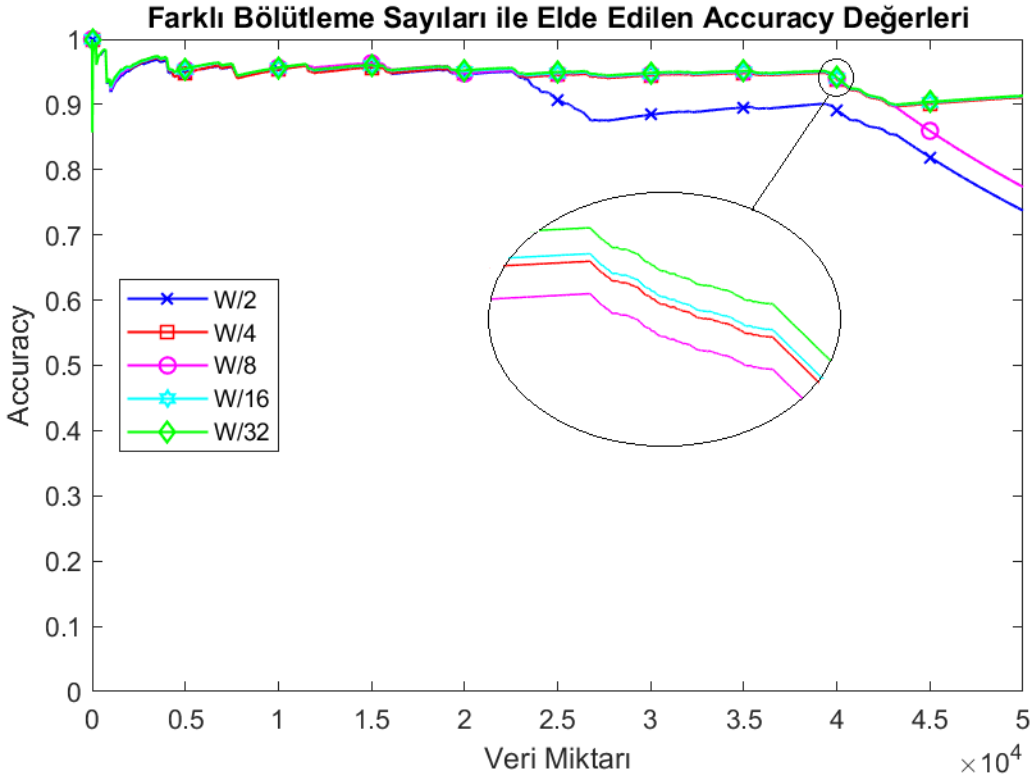
3.2. Sonuçların Değerlendirilmesi

Önerdiğimiz yaklaşımın en önemli avantajı veriyi çok hızlı bir şekilde işlemesidir. Bunun da nedeni her veri girişinde penceredeki tüm verileri işlemek yerine belirli sayıda veri biriktikçe penceredeki tüm verileri işlemesidir. Veri biriktirme veya pencere bölütleme yaklaşımı gereksiz işlem yükünü düşürmektedir. Bölütleme sayısı ve zaman karmaşıklığı arasında ters; bölütleme sayısı ile kümeleme başarısı arasında doğru orantı vardır. Şekil 3 ve Şekil 4'te de görüldüğü gibi bölütleme sayısı arttıkça performans düşmekte ama kümeleme başarısında artış söz konusudur. Bu nedenle önerdiğimiz yaklaşımda KD-AR Stream algoritmasının yalın hali ile

karşılaştırma yapılırken hem kümeleme balşarısı açısından makul olan, hem de hızlı bir şekilde veriyi işlemeye imkan tanımasından dolayı bölütleme saysı 4 olarak seçilmiştir.



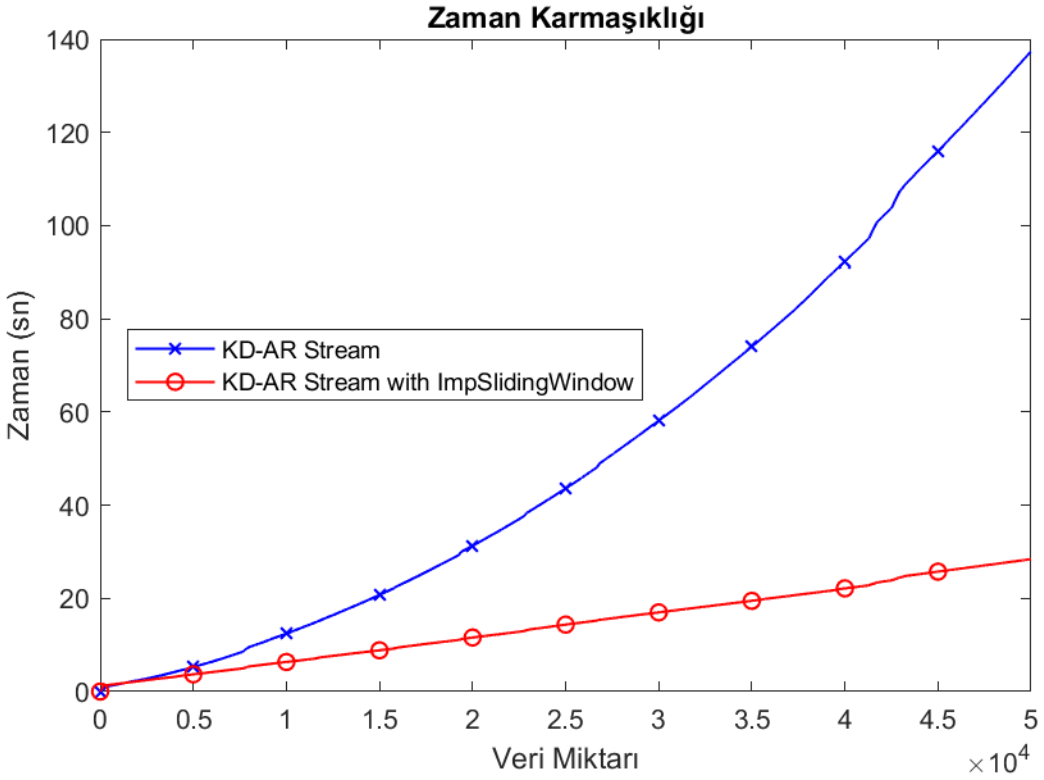
Şekil 3. Farklı bölütleme sayısının çalışma zamanına etkisi



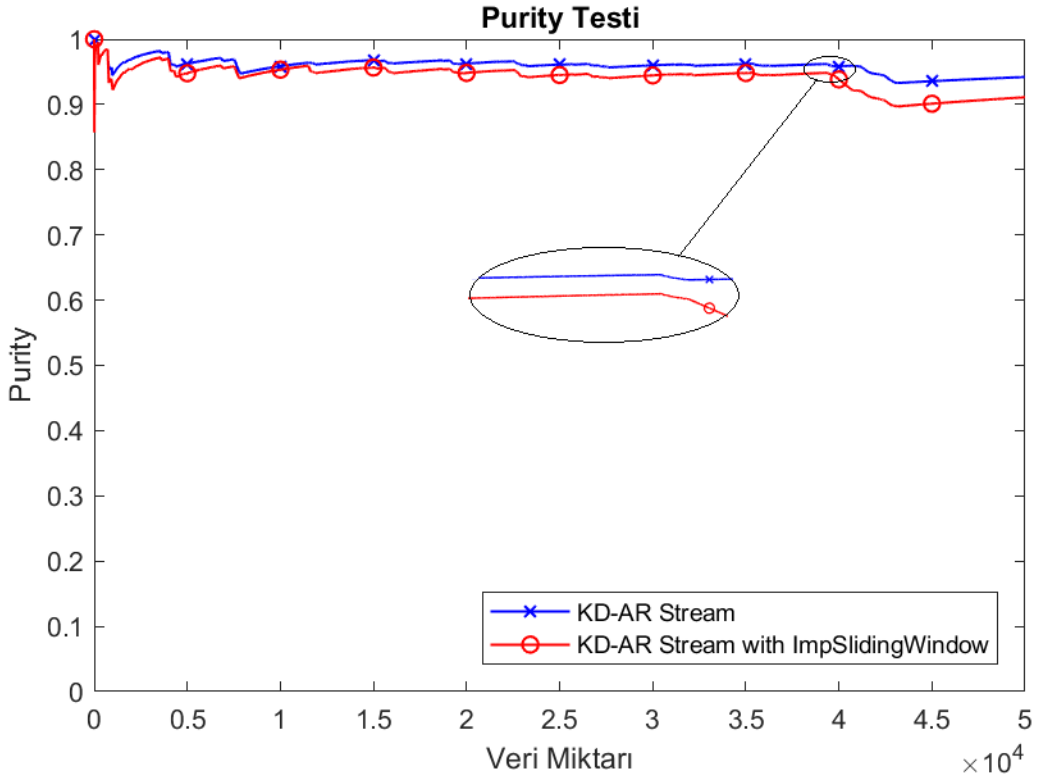
Şekil 4. Farklı bölütleme sayısının kümeleme başarısına etkisi

Şekil 5'te de görüldüğü gibi KD-AR Stream'in yalın hali KDD veri setini yaklaşık 130 sn.'de işlerken önerdiğimiz yaklaşım söz konusu veriyi 28 sn. gibi çok kısa bir sürede işlemektedir. Kuşkusuz kümeleme modeli tek başına yeterli değildir. Önerilen modelin kümeleme başarısının da makul bir seviyede olması gerekir. Şekil 6'da da görüldüğü gibi önerdiğimiz yaklaşımın elde ettiği Purtiy

değeri KD-AR Stream'in yalın halinin ürettiği Purity değerine çok yakın dır. Bu da önerdiğimiz yaklaşımın da verileri doğru bir şekilde kümelere ayrıdığını göstermektedir.



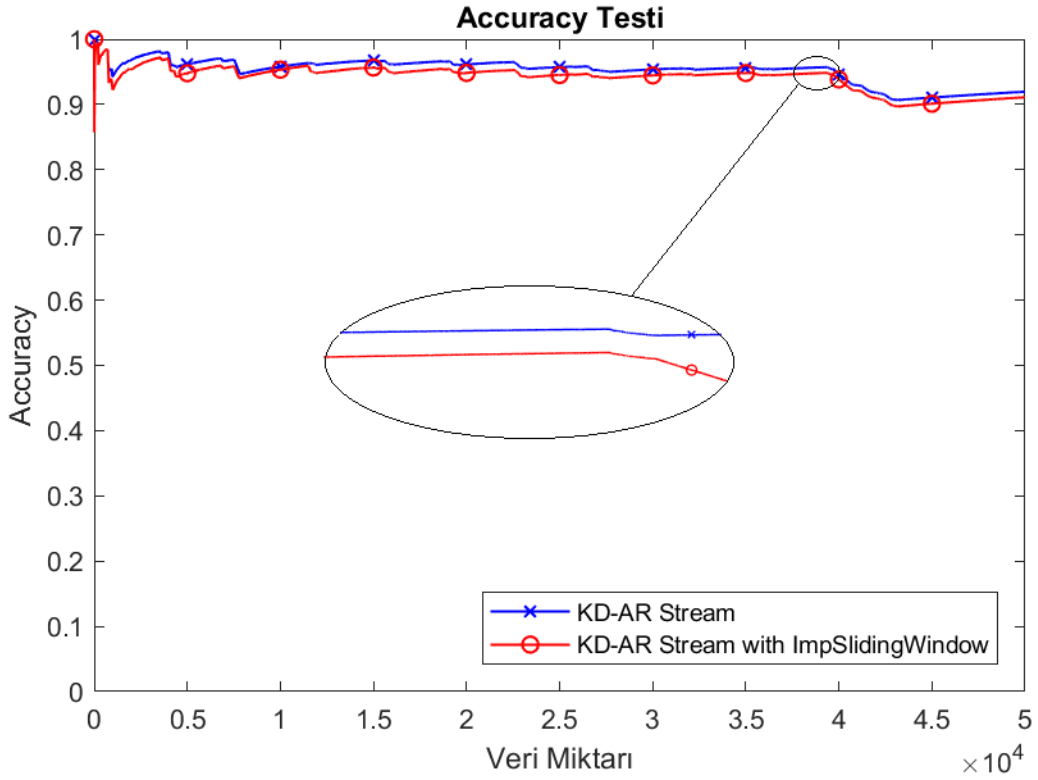
Şekil 5. Yaklaşımların toplam çalışma zamanlarının karşılaştırması



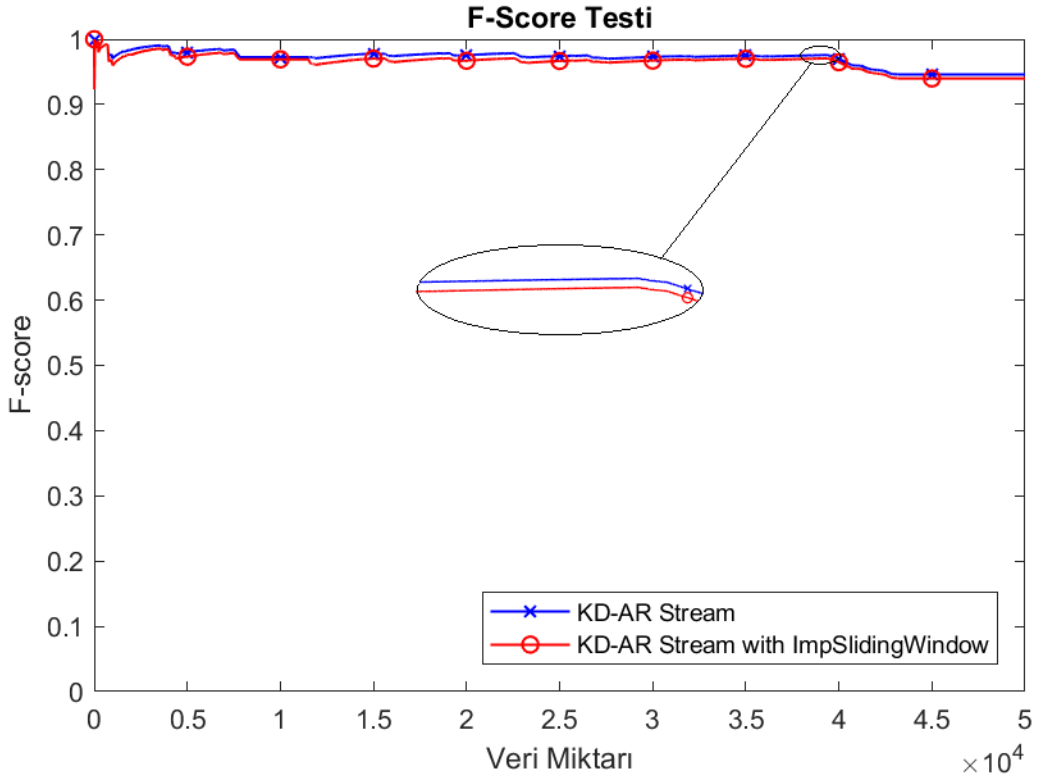
Şekil 6. Algoritmaların KDD veri seti üzerinde test edilmesi ile elde edilen Purity değerlerinin karşılaştırması

Şekil 7 ve 8'de de görüldüğü gibi Accuracy ve F-Score değerleri de Accuracy değerlerine paralellik göstermekte ve KD-AR Stream'in yalın halinin ürettiği değerlere oldukça yakındır. Bunun yanında Şekil 9'da görülen Silhouette indeks sonuçları karşılaştırıldığında ise önerilen modelin ürettiği değerin KD-AR Stream'in yalın halinin ürettiği Silhouette indeks değerine eşit olduğu

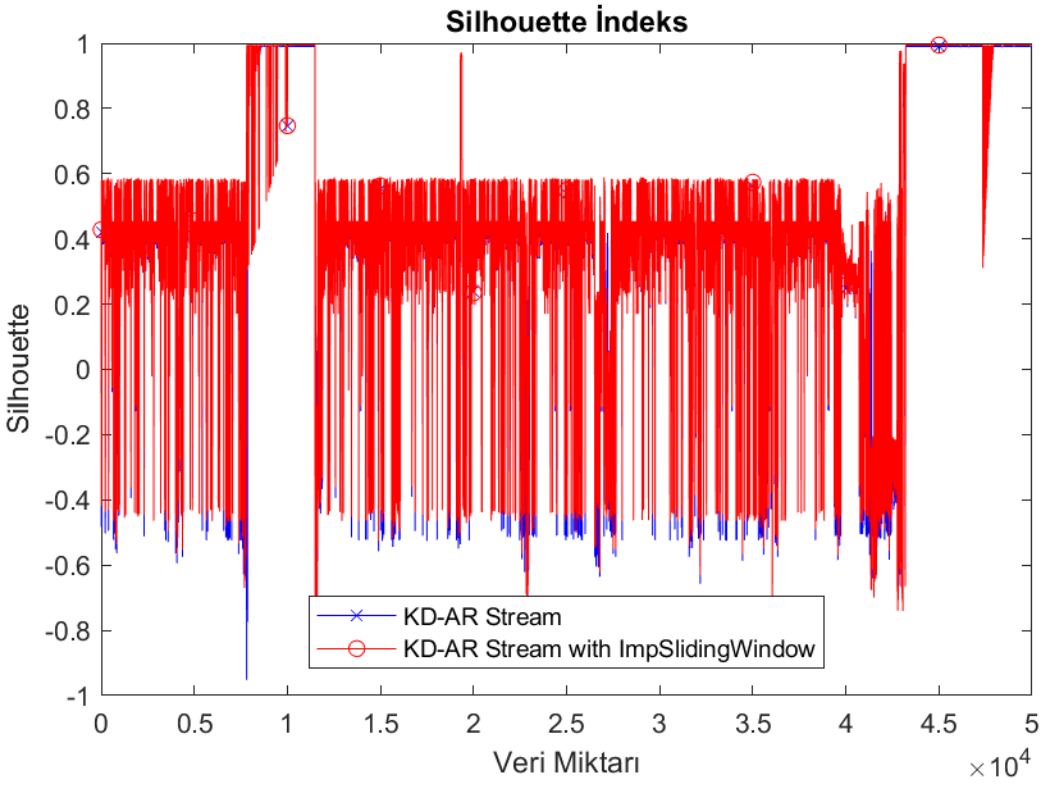
görülmektedir. Bu da önerdiğimiz yaklaşımın verileri küme etiketinden bağımsız olarak kümelere bölme yaklaşımının çok tutarlı olduğunu göstermektedir.



Şekil 7. Algoritmaların KDD veri seti üzerinde test edilmesi ile elde edilen Accuracy değerlerinin karşılaştırması



Şekil 8. Algoritmaların KDD veri seti üzerinde test edilmesi ile elde edilen F-Score değerlerinin karşılaştırması



Şekil 9. Algoritmaların KDD veri seti üzerinde test edilmesi ile elde edilen Silhouette indeks değerlerinin karşılaştırması

Kuşkusuz bir tek veri seti üzerinden karşılaştırma yaparak algoritmaların başarısı hakkında bir sonuca varmak yeterli değildir. Bu nedenle önerilen yaklaşımın birkaç veri seti üzerinde test edilerek sonuçların karşılaştırılması gerekir. Önerdiğimiz yaklaşım Tablo 1'de sözü edilen veri setleri üzerinde test edilmiş ve elde edilen sonuçlar KD-AR Stream'in yalın hali ile elde edilen sonuçlarla karşılaştırılmış ve Tablo 4'te verilmiştir. Tabloda da görüldüğü gibi kümeleme başarısı açısından önerdiğimiz yaklaşım yüksek oranlarda kümeleme başarısı yakalamaktadır. Önerdiğimiz yaklaşım, yapılan 16 testin 10'unda en yüksek başarı oranını yaklamaktadır. Bu da önerdiğimiz yaklaşımın kümeleme başarısı açısından oldukça tatmin edici olduğunu göstermektedir.

Table 4. Elde Edilen Kümeleme Başarılarının Tek Tabloda Karşılaştırması

		Purity	**Accuracy**	**F-Score**	**Silhouette İndeks**
KD-AR Stream	*KDD*	***95,86***	***95,14***	***97,07***	***0,52***
	Fisher Iris	*96,88*	*96,26*	*95,84*	***0,39***
	Breast Cancer	*90,05*	*90,05*	*91,90*	***0,56***
	ExclaStar	***99,87***	***99,87***	***99,88***	*0,56*
ImpSlidingWindow Tabanlı KD-AR Stream	*KDD*	*94.14*	*94.14*	*96.45*	***0.52***
	Fisher Iris	***97.62***	***97.62***	***97.19***	*0,37*
	Breast Cancer	***91.55***	***91.16***	***92.75***	*0.35*
	ExclaStar	***99.87***	***99.87***	***99.88***	*0,45*

4. Sonuç ve Öneriler

Bu çalışmada akan veri kümeleme alanında sıkça kullanılan kayan pencere tabanlı veri özetleme yaklaşımının performansını geliştirmeye yönelik gelişmiş bir versiyonu olan ImpSlidingWindow yaklaşımı önerilmiştir. Önerdiğimiz model hem yüksek oranlarda kümeleme başarısı yakalamakta hem de çok hızlı sonuç üretmeye imkân tanımaktadır. Performans açısından bakıldığı zaman veri setinin sahip olduğu nitelik sayısına bağlı olarak %80'lere varan performans iyileştirmeleri yakalanabilmektedir. Özellikle veri akışının çok hızlı olduğu uygulamalarda önerdiğimiz yaklaşım çok faydalı olacak bir yaklaşım olarak dikkat çekmektedir.

Önerdiğimiz pencere genişliğini 4 eşit parçaya bölme yaklaşımı uygulama ve ihtiyaca göre değiştirilebilir bir yaklaşımdır. Bölünen parça sayısının fazla olması performansta aşağı yönlü bir eğilime neden olurken, kümeleme başarısı açısından yukarı yönlü bir eğilime neden olmaktadır. Benzer şekilde parça sayısının az olması performası arttırırken, kümeleme başarısı açsından aşağı yönlü bir eğilime neden olmaktadır. En doğru parça sayısı veri setine ve ihtiyaca göre belirlenecek bir değerdir.

Kaynaklar

Ackermann, M. R., Martens, M., Raupach, C., Swierkot, K., Lammersen, C. ve Sohler, C. (2012). StreamKM++: A clustering algorithm for data streams. *J. Exp. Algorithmics, 17*, 2.1-2.30. doi:10.1145/2133803.2184450

Aggarwal, C. C. (2010). Data Streams: An Overview and Scientific Applications. In M. M. Gaber (Ed.), *Scientific Data Mining and Knowledge Discovery: Principles and Foundations* (pp. 377-397). Berlin, Heidelberg: Springer Berlin Heidelberg.

Aggarwal, C. C., Han, J., Wang, J. ve Yu, P. S. (2003). *A framework for clustering evolving data streams*. Paper presented at the Proceedings of the 29th international conference on Very large data bases - Volume 29, Berlin, Germany.

Ahmed, M. (2019). Buffer-based Online Clustering for Evolving Data Stream. *Information Sciences*. doi:https://doi.org/10.1016/j.ins.2019.03.022

AlNuaimi, N., Masud, M. M., Serhani, M. A. ve Zaki, N. (2019). Streaming feature selection algorithms for big data: A survey. *Applied Computing and Informatics*. doi:https://doi.org/10.1016/j.aci.2019.01.001

Amini, A. ve Wah, T. Y. (2013). LeaDen-Stream: A Leader Density-Based Clustering Algorithm over Evolving Data Stream. *Journal of Computer and Communications, 1*, 26-31. doi:10.4236/jcc.2013.15005

Ankleshwaria, T. B. ve Dhobi, J. S. (2014). Mining Data Streams: A Survey. *International Journal of Advance Research in Computer Science and Management Studies, 2*(2), 379-386.

Antonellis, P., Makris, C. ve Tsirakis, N. (2009). Algorithms for clustering clickstream data. *Information Processing Letters, 109*(8), 381-385. doi:https://doi.org/10.1016/j.ipl.2008.12.011

Badiozamany, S., Orsborn, K. ve Risch, T. (2016). *Framework for real-time clustering over sliding windows*. Paper presented at the Proceedings of the 28th International Conference on Scientific and Statistical Database Management, Budapest, Hungary.

Cao, F., Estert, M., Qian, W. ve Zhou, A. Density-Based Clustering over an Evolving Data Stream with Noise *Proceedings of the 2006 SIAM International Conference on Data Mining* (pp. 328-339).

Chairukwattana, R., Kangkachit, T., Rakthanmanon, T. ve Waiyamai, K. (2013, 4-6 Sept. 2013). *Efficient evolution-based clustering of high dimensional data streams with dimension projection.* Paper presented at the 2013 International Computer Science and Engineering Conference (ICSEC).

Charu, C. A., Jiawei, H., Jianyong, W. ve Philip, S. Y. (2004). A framework for projected clustering of high dimensional data streams *Proceedings of the Thirtieth international conference on Very large data bases - Volume 30 %@ 0-12-088469-0* (pp. 852-863). Toronto, Canada: VLDB Endowment.

Datar, M., Gionis, A., Indyk, P. ve Motwani, R. (2002). *Maintaining stream statistics over sliding windows: (extended abstract)*. Paper presented at the Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms, San Francisco, California.

Diaz-Rozo, J., Bielza, C. ve Larrañaga, P. (2018). Clustering of Data Streams with Dynamic Gaussian Mixture Models. An IoT Application in Industrial Processes. *IEEE Internet of Things Journal*, 1-1. doi:10.1109/JIOT.2018.2840129

Gao, J., Li, J., Zhang, Z. ve Tan, P.-N. (2005). *An Incremental Data Stream Clustering Algorithm Based on Dense Units Detection*, Berlin, Heidelberg.

Görmüş, S., Aydın, H. ve Ulutaş, G. (2018). Nesnelerin interneti teknolojisi için güvenlik: Var olan mekanizmalar, protokoller ve yaşanılan zorlukların araştırılması. *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 33*, 1247-1272.

Gravina, R., Alinia, P., Ghasemzadeh, H. ve Fortino, G. (2017). Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges. *Information Fusion, 35*, 68-80. doi:https://doi.org/10.1016/j.inffus.2016.09.005

Guha, S., Rastogi, R. ve Shim, K. (2001). Cure: an efficient clustering algorithm for large databases. *Information Systems, 26*(1), 35-58. doi:https://doi.org/10.1016/S0306-4379(01)00008-4

Hahsler, M. ve Bolaños, M. (2016). Clustering Data Streams Based on Shared Density between Micro-Clusters. *IEEE Transactions on Knowledge and Data Engineering, 28*(6), 1449-1461. doi:10.1109/TKDE.2016.2522412

Hendricks, D. (2017). Using real-time cluster configurations of streaming asynchronous features as online state descriptors in financial markets. *Pattern Recognition Letters, 97*, 21-28. doi:https://doi.org/10.1016/j.patrec.2017.06.026

Hyde, R., Angelov, P. ve MacKenzie, A. R. (2017). Fully online clustering of evolving data streams into arbitrarily shaped clusters. *Information Sciences, 382-383*, 96-114. doi:https://doi.org/10.1016/j.ins.2016.12.004

Ikonomovska, E., Loskovska, S. ve Gjorgjevik, D. (2007). *A survey of stream data mining*. Paper presented at the Eighth International Conference with International Participation – ETAI 2007, Ohrid, Republic ofMacedonia.

Jia, C., Tan, C. ve Yong, A. (2008, 25-26 Sept. 2008). *A Grid and Density-Based Clustering Algorithm for Processing Data Stream.* Paper presented at the 2008 Second International Conference on Genetic and Evolutionary Computing.

Kanmaz, M. ve Aydin, M. A. (2018). Kablosuz Sensör Ağlarda Konumlandırma Yöntemleri ve K-means++ Kümeleme Yöntemi ile Yeni Yaklaşım. *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 2018*, 0-0.

Keim, D. A. ve Heczko, M. (2001). *Wavelets and their Applications in Databases.* Paper presented at the 17th International Conference on Data Engineering (ICDE'01), Heidelberg, Germany, 2001.

King, R. C., Villeneuve, E., White, R. J., Sherratt, R. S., Holderbaum, W. ve Harwin, W. S. (2017). Application of data fusion techniques and technologies for wearable health monitoring. *Medical Engineering & Physics, 42*, 1-12. doi:https://doi.org/10.1016/j.medengphy.2016.12.011

Kranen, P., Assent, I., Baldauf, C. ve Seidl, T. (2011). The ClusTree: indexing micro-clusters for anytime stream mining. *Knowledge and Information Systems, 29*(2), 249-272. doi:10.1007/s10115-010-0342-8

Laohakiat, S., Phimoltares, S. ve Lursinsap, C. (2017). A clustering algorithm for stream data with LDA-based unsupervised localized dimension reduction. *Information Sciences, 381*, 104-123. doi:https://doi.org/10.1016/j.ins.2016.11.018

Li, Z. Q. (2014). A New Data Stream Clustering Approach about Intrusion Detection. *Advanced Materials Research, 926-930*, 2898-2901. doi:10.4028/www.scientific.net/AMR.926-930.2898

Manzi, A., Dario, P. ve Cavallo, F. (2017). A Human Activity Recognition System Based on Dynamic Clustering of Skeleton Data. *Sensors (Basel, Switzerland), 17*(5), 1100. doi:10.3390/s17051100

Martín, A., Julián, A. B. A. ve Cos-Gayón, F. (2019). Analysis of Twitter messages using big data tools to evaluate and locate the activity in the city of Valencia (Spain). *Cities, 86*, 37-50. doi:https://doi.org/10.1016/j.cities.2018.12.014

Ntoutsi, I., Zimek, A., Palpanas, T., Kröger, P. ve Kriegel, H.-P. (2012). *Density-based Projected Clustering over High Dimensional Data Streams.* Paper presented at the SIAM International Conference on Data Mining.

O'Callaghan, L., Mishra, N., Meyerson, A., Guha, S. ve Motwani, R. (2002, 26 Fe.-1 March 2002). *Streaming-data algorithms for high-quality clustering.* Paper presented at the Proceedings 1st International Conference on Data Engineering, San Jose, CA, USA, USA.

Oussous, A., Benjelloun, F.-Z., Ait Lahcen, A. ve Belfkih, S. (2018). Big Data technologies: A survey. *Journal of King Saud University - Computer and Information Sciences, 30*(4), 431-448. doi:https://doi.org/10.1016/j.jksuci.2017.06.001

Reddy, K. S. S. ve Bindu, C. S. (2018). StreamSW: A Density-based Approach for Clustering Data Streams over Sliding Windows. *Measurement*. doi:https://doi.org/10.1016/j.measurement.2018.11.041

Ren, J. ve Ma, R. (2009, 14-16 Aug. 2009). *Density-Based Data Streams Clustering over Sliding Windows.* Paper presented at the 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery.

Silva, J. d. A., Hruschka, E. R. ve Gama, J. (2017). An evolutionary algorithm for clustering data streams with a variable number of clusters. *Expert Syst. Appl., 67*(C), 228-238. doi:10.1016/j.eswa.2016.09.020

Şenol, A. ve Karacan, H. (2018). A Survey on Data Stream Clustering Techniques. *European Journal of Science and Technology*(13), 17-30.

Şenol, A. ve Karacan, H. (2019). K-boyutlu ağaç ve uyarlanabilir yarıçap (KD-AR Stream) tabanlı gerçek zamanlı akan veri kümeleme. *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, (Basımda)*.

Tasnim, S., Caldas, J., Pissinou, N., Iyengar, S. S. ve Ding, Z. (2018, 5-8 March 2018). *Semantic-Aware Clustering-based Approach of Trajectory Data Stream Mining.* Paper presented at the 2018 International Conference on Computing, Networking and Communications (ICNC).

Tu, L. ve Chen, Y. (2009). Stream data clustering based on grid density and attraction. *ACM Trans. Knowl. Discov. Data, 3*(3), 1-27. doi:10.1145/1552303.1552305

Udommanetanakit, K., Rakthanmanon, T. ve Waiyamai, K. (2007). *E-Stream: Evolution-Based Technique for Stream Clustering*, Berlin, Heidelberg.

Wan, L., Ng, W. K., Dang, X. H., Yu, P. S. ve Zhang, K. (2009). Density-based clustering of data streams at multiple resolutions. *ACM Trans. Knowl. Discov. Data, 3*(3), 1-28. doi:10.1145/1552303.1552307

Wang, W., Yang, J. ve Muntz, R. R. (1997). *STING: A Statistical Information Grid Approach to Spatial Data Mining*. Paper presented at the Proceedings of the 23rd International Conference on Very Large Data Bases.

Xu, J., Wang, G., Li, T., Deng, W. ve Gou, G. (2017). Fat node leading tree for data stream clustering with density peaks. *Knowledge-Based Systems, 120*, 99-117. doi:https://doi.org/10.1016/j.knosys.2016.12.025

Yin, C., Xia, L. ve Wang, J. (2017, 2017). *Application of an Improved Data Stream Clustering Algorithm in Intrusion Detection System.* Paper presented at the Advanced Multimedia and Ubiquitous Engineering, Singapore.

Yin, C., Xia, L. ve Wang, J. (2018, 2018). *Data Stream Clustering Algorithm Based on Bucket Density for Intrusion Detection.* Paper presented at the Advances in Computer Science and Ubiquitous Computing, Singapore.