TÜRKÇE TWİTTER AKIŞI VE TÜRKÇE GAZETE VERİLERİNDEN SİBER GÜVENLİK OLAYLARININ OTOMATİK TESPİT EDİLMESİ

Jüri Sonrası Tez Düzeltmesi Kapsamında Yapmış Olduğum Çalışmalar:

Jürinin önerdiği üzere geçmiş siber güvenlik olaylarının Twitter ve haber sitesi verileri derlenerek olayın hemen öncesinde (öncül) ve hemen olayı müteakip (artçıl) haber ve tweetlerde frekans analizi yapılarak öncül ve artçıl belirteç kelimelerin bulunması ve bunlara bağlı takip listelerin oluşturulması üzerine bir çalışma yapabilmek için geçmişte gerçekleşen siber güvenlik olaylarını içeren datasetler oluşturmam gerekmekteydi. Geçmiş tarihli istediğim ölçütlerde ve kısıtlamalarda Türkçe hazır datasetler online ortamda bulunmamakta. Bu sebeple dataset oluşturma yazılımını kendim yazmam gerekmekteydi. Tezde Twitter Standart API’sini kullanmaktaydım. Twitterda geçmiş bir olay üzerinden frequency analiz yapmak Twitter'ın Standart Api'si son 7 gün verilerini çekmemize izin verdiği için Apinin ücretsiz versiyonunda yapılabilir bir şey değil. İstenen geçmiş tarihli verilere ancak ücretli versiyon olan Twitter Premium API ile erişim sağlayabilirdim. Twitter’a yapmış olduğum akademik çalışmayı açıklayan, içeriğinde yazmış olduğum tezden bölümler içeren bir mail attım. Twitter çalışmama destek olmayı kabul etti ve aylık 50 request atabileceğim Premium API’yi ücretsiz verdi.

Analizimde kullanmak için etmek için nic.tr DDOS saldırısını seçtim. 14.12.2015 - 29.12.2015 tarihleri arasında içinde "nic.tr" geçen Türkçe Tweetleri çektim. Yaklaşık 1100 Tweet ile eşleşti. Bu tweetlerdeki kelime frekanslarını çıkarttım.

Daha sonra geçmiş 1 yılın verilerini çektim. 10.12.2014 - 13.12.2015 tarihleri arasındaki içinde nic.tr içeren tweetler için sorgu gönderdiğimde sadece 28 tweet içeren bir dataset döndü. Bu tweetlerdeki kelime frekanslarını çıkarttım. Saldırının başladığı gün olan 14 aralık 2015 tarihindeki kelime frekanslarını da ayrıca çıkardım. Bu 3 ayrı kelime frekansı dosyasını analiz edip içlerinden siber güvenlikle ilgili kullanabileceğimiz kelimeleri seçmemiz gerekmekteydi. Bu aşamadkelime frekansı yerine TF-IDF (Term Frequency — Inverse Document Frequency) kullanarak kelimeleri seçmemizin daha doğru olacağına karar verdik ilgili datasetleri kullanarak TF-IDF analizi yaptık ve keyword listemizi bunun doğrultusunda güncelledik. False positive detection’ı minimize etmek için keyword listemizi az sayıda terimden oluşturduk.

Daha sonra oluşturmuş olduğum keyword listesi ile nic.tr saldırısı ile ilgili databaseler oluşturmam gerekmekteydi. Ancak bu databaseleri oluştururken Twitter’ın bana verdiği aylık 50 requestlik premium API hakkım bitti. Twitter Premium API 100-1900 dolar arasında değişen bir aylık ücrete sahip. Bu sebeple satın almak yerine alternative çözümler aramaya başladım. Bu esnada Selenium otomasyon aracının bu tür bir iş için kullanılabileceğini farkettim. Selenium ve chrome driver ile dataları çekmek için bir python kodu yazdım. Çözümüm Api ile çekmek yerine normal bir kullanıcı gibi, chrome browser ı açıyor, Twitterın arama kutusuna belirlediğim query i yazıyor ve daha sonra html sayfasını parse ediyor. Bu çözüm daha önce aklıma gelmişti ama html dinamik olarak yaratıldığı için sayfayı nasıl aşağı kaydırabileceğimi bulamamıştım. O yüzden api çözümüne yönelmiştim. Selenium ve chrome web driver bunu yapabiliyormuş. Selenium kullanarak Twitter dataseti oluşturma daha önce yapılmış bir çözüm değil. İlgili yazılımı diğer kodlarla birlikte githubda paylaştığımda data analizi ile uğraşanlar tarafından sıklıkla kullanılabileceğine inanıyorum.

Selenium çözümünü kullanarak ilgili databaseleri oluşturdum. Saldırı öncesi 1 yıllık süreçteki tweetleri çektiğinde 2310 tweet buldu. Saldırının başlaması sonrası 2 haftalık süreçteki tweetleri çektiğinde 400 tweet buldu. Saldırı günü ise 38 tweet buldu.  
  
Bu databaseleri kullanarak yazılım çözümümü çalıştırdığımda kendi test kriterlerime göre başarılı sonuçlar verdi. Nic.tr saldırısını detect edebildi. Bir olay olduğunda olayın olduğu gün detect edebiliyor olmasını ve ilgili detectionın birkaç hafta boyunca devam edebilmesini duyarlılık olarak düşündüm. Bu açıdan yaptığımız testler başarılı çıktı. False positive sayısının az olmasını belirlilik olarak düşündüm. False positive oranı %20-%25 civarında, bu açıdan da çözümümüzün başarılı olduğuna karar verdik.

Jurinin bir diğer beklediği düzeltme ise tez dokümanında siber güvenlik olaylarının saptanması için kullanılan yöntemin çok daha detaylı olarak anlatılması ve tartışılmasıydı. Bu doğrultuda özellikler tezin 4. bölümünün (“System Architecture and Design”) ve 6. bölümünün (“Results”) daha detaylı ve kapsamlı olacak şekilde geliştirilmesi gerekiyordu. Tezin