Резултат (Result) 2.2

Отчет със сравнителен анализ за прилагане на машинно обучение в различните етапи от жизнения цикъл на софтуера

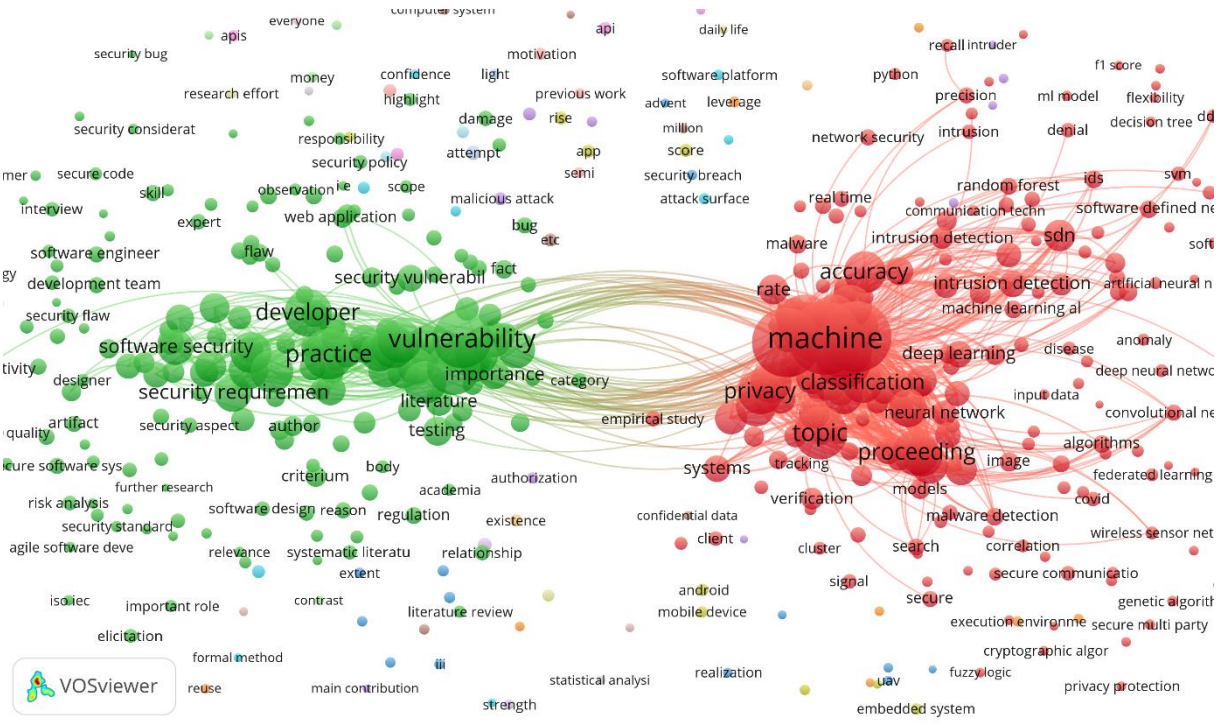
1. Методология на изследването.

Методологията на изследването за откриване и анализ на аномалии в софтуерни продукти вклщчва няколко стъпки:

1.) Търсене на научни публикации в различни източника с търсене по тема, анотация и ключови думи чрез използване на дигитални библиотеки. Заявката за търсене беше:

```
TITLE-ABS-KEY((''Secure Software'') AND (''machine learning'') AND (''Secure Software Development Lifecycle'' OR ''Software Development Lifecycle''))
```

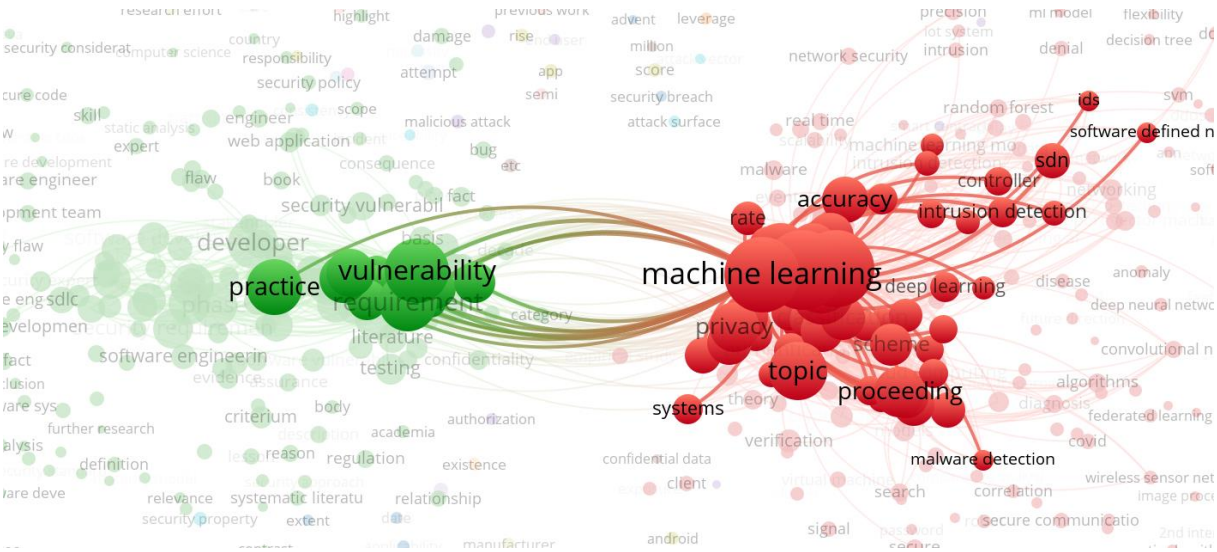
2.) След това създаваме подобна карта въз основа на индексираните ключови думи (фиг. 1). Те се различават от ключовите думи, предоставени от авторите, и предоставят повече възможности за наблюдение и заключения по време на анализа.



Фиг. 1 Индексна карта на ключовите думи.

Открити са 33037 понятия, от които 819 се повтарят повече от 10 пъти. За всеки от 186-те понятия се изчислява оценка на релевантността. Въз основа на тази оценка са избрани най-подходящите термини. По подразбиране се избират 60% от най-подходящите термини (491 понятия).

На фиг. 2 са показани основните връзки с другите понятия на ключовата дума “machine learning” към “Secure Software”.



Фиг. 2 Индексна карта на ключовата дума “machine learning” .

2. Жизнен цикъл на разработване на сигурен софтуер

През последните години се обръща специално внимание на осигуряването на сигурност на всички етапи от жизнения цикъл на разработване на софтуер (SDLS), като той се превръща в жизнен цикъл на разработване на сигурен софтуер (Secure Software Development Life Cycle - SSDLC)[1]. Изследванията на рисковете за сигурността на софтуера[2], подчертават основните източници на рискове, които се дължат на липсата на подходящо внимание към въпросите на сигурността по време на всички етапи на SDLC: разработване на изисквания, проектиране, разработване, тестване и поддръжка. По този начин широко се възприема организационен модел, известен като DevSecOps, който има за цел да установи непрекъснат цикъл на интеграция и доставка и съчетава разработването на приложения със съображения за сигурност и операции[3]. Факторите, влияещи върху сигурността на софтуера, могат да бъдат класифицирани в 4 основни категории[4]: институционален контекст, хора и действия, процес на разработване на системата и съдържание на проекта.

Фокус на изследването е в различните етапи от жизнения цикъл на разработване на софтуер, кои подходи от машинно обучение е подходящо да се използват от гледна точка на осигуряване на висока сигурност на софтуера. За целта се идентифицират за разглеждане следните етапи: разработване на изисквания, проектиране, разработване,

---

[1] J. Fonseca, M. Vieira, A Survey on secure software development lifecycles, In K. Buragga, N. Zaman (eds.), Software Development Techniques for Constructive Information Systems Design, IGI Global, pp. 57-73, 2013.

[2] R. Khan, S. Khan, H. Khan, M. Ilyas, Systematic literature review on security risks and its practices in secure software development, IEEE Access, Vol. 10, pp. 5456-5481, 2022.

[3] H. Myrbakken, R. Colomo-Palacios, DevSecOps: A Multivocal Literature Review, in A. Mas, A. Mesquida, R. O'Connor, T. Rout, A. Dorling (eds), Software Process Improvement and Capability Determination, Communications in Computer and Information Science, Vol. 770. Springer, Cham, 2017.

[4] S. Kanniah, M. Mahri, A review on factors influencing implementation of secure software development practices, International Journal of Computer and Systems Engineering, Vol. 10, No. 8, pp. 3032-2039, 2016.

тестване и поддръжка. Етапът на проектирането се разделя на проектиране на модел на заплахите и проектиране на архитектурата на приложението.

Софтуерната уязвимост е дефинирана като недостатък на сигурността, грешка или слабост, открита в софтуерен код, която може да бъде използвана от нападател (източник на заплаха)[5] и е тясно свързана с откриването на софтуерни аномалии. Изчерпателен списък на защитените уязвимости се поддържа от MITRE Corporation като част от програмата CVE[6], както и от Националната база данни за уязвимости на NIST[7].

Също така на етапите на разгръщане, тестване и поддръжка на системата в рамките на SDLC откриването на аномалии се разглежда като процес по време на работа, основан на данни от мониторинга на поведението на системата. Ръчното откриване на аномалии не е практически възможно дори за малки и не много сложни софтуерни системи. Ето защо се правят много опити да се предложи и приложи автоматично откриване на аномалии в софтуерни системи, като се поставя специален акцент върху прилагането на подходи, базирани на машинно обучение и дълбоко обучение.

3. Алгоритми за машинно обучение за откриване на уезвимости в софтуерни системи в различните етапи от жизнения цикъл на софтуера

Машинното обучение за откриване на уезвимости в софтуерни продукти включва прилагане на алгоритми за идентифициране на необичайни модели или поведение в данни, свързани със софтуер. Това може да бъде от решаващо значение за откриването на уезвимости, които могат да показват заплахи за сигурността, софтуерни грешки, проблеми с производителността или друго неочаквано поведение.

При контролираното откриване на отклонения моделът за машинно обучение се обучава върху маркиран набор от данни, в който изрично се идентифицират както нормалните, така и аномалните случаи. По време на обучението моделът се научава да прави разлика между нормалните и отклоняващите се модели. При неконтролираното откриване на отклонения моделът се обучава върху набор от данни без маркирани отклонения. Алгоритъмът трябва да идентифицира аномалиите въз основа на присъщите модели или структури, присъстващи в данните.

На фиг. 3 е представена таксономия на алгоритмите за откриване на отклонения. Подходите се разглеждат като методи за количествено обучение и методи за семантично обучение[8], като методите за количествено обучение са независими от контекста на набора от данни, докато методите за семантично обучение идентифицират отклоненията в рамките на специфичен за приложението контекст.

[5] K. Dempsey, P. Eavy, G. Moore, E. Takamura Automation Support for Security Control Assessments: Software Vulnerability Management, National Institute of Standards and Technology, NISTIR 8011, Vol. 4, 2020.
[6] Common Vulnerabilities and Exposures, MITRE, https://cve.mitre.org
[7] National Vulnerability Database, NIST, https://nvd.nist.gov
[8] R. Kashef, M. Gencarelli, A. Ibrahim, Classification of outlier's detection methods based on quantitative or semantic learning, in Z. Fadlullah, A. Pathan (eds) Combating security challenges in the age of big data. advanced sciences and technologies for security applications, Springer, Cham, 2020.

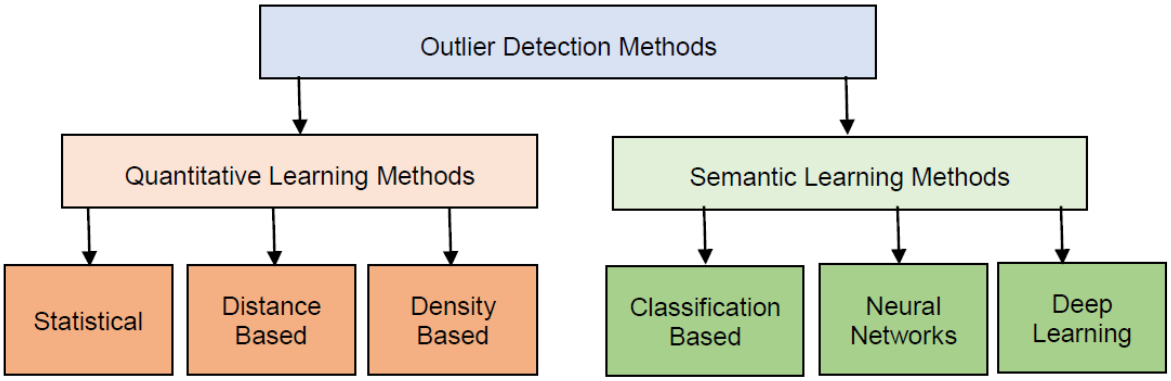


Fig. 3. *Таксономия на методите за откриване на отклонения*

Алгоритмите за машинно обучение, които се използват за откриване на отклонения, могат да се разглеждат още като статистически базирани, базирани на разстояния, базирани на плътност или базирани на ансамбъл алгоритми. Специфичен клас алгоритми за машинно обучение са невронните мрежи и дълбокото обучение.

3.1. Статистически подходи за откриване на аномалии

Статистическите подходи за откриване на отклонения разчитат на анализ на статистическите свойства на данните, за да идентифицират случаи, които се отклоняват значително от очакваното или нормалното поведение. Тези техники обикновено са лесни за прилагане и са полезни в случаите, когато входният набор от данни е ограничен по размер или когато се очаква данните да отговарят на определено статистическо разпределение. Статистическите подходи се основават на разработване на модел на вероятностно разпределение и разглеждане на това колко вероятни са обектите при този модел[9]. Сред статистическите подходи за откриване на отклонения са:

- Анализ на Z-точките: Измерва колко стандартни отклонения има дадена точка от средната стойност на разпределението. Точки с Z-резултат над определен праг могат да се считат за аномалии. Подходящ за едномерни данни, при които аномалиите се идентифицират въз основа на отклонението им от средната стойност;

- IQR (междуквартален диапазон): Междуквартилният диапазон е диапазонът между първия квартил (Q1) и третия квартил (Q3) на набор от данни. Точките извън диапазона се отбелязват като потенциални отклонения. Устойчив на отклонения и подходящ за едномерни данни.

- Анализ на хистограмата: хистограмата разделя данните на интервали и отчита броя на наблюденията във всеки интервал. Отклоненията от очакваните честоти на биновете могат да показват отклонения. Подходящ за изследване на разпределението на непрекъснати данни. Отклоненията могат да бъдат идентифицирани въз основа на необичайните бройки на биновете.

[9] G. Madhuri, M. Rani, Statistical approaches to detect anomalies, in P. Krishna, M. Obaidat (eds) Emerging research in data engineering systems and computer communications, Advances in Intelligent Systems and Computing, Vol. 1054, Springer, Singapore, 2020

- Ядрена оценка на плътността (KDE): Оценява функцията на плътността на вероятността на случайна променлива и осигурява плавна оценка на разпределението на данните. Отклоненията могат да бъдат идентифицирани като точки с ниска вероятност. Ефективно за оценка на основното разпределение на непрекъснати данни.

- Анализ на времеви редове: За анализ на времеви редове могат да се използват статистически методи, като например пълзящи средни, експоненциално изглаждане и авторегресивни интегрирани модели на пълзящи средни (ARIMA). Отклоненията от очакваните тенденции могат да показват аномалии. Полезни са за наблюдение на системни показатели и идентифициране на необичайни модели или тенденции във времето.

Статистическите подходи са универсални и могат да се прилагат за различни видове данни в софтуерните системи. Те са особено полезни, когато могат да се направят предположения за разпределението на данните.

3.2. Подходи, базирани на разстояния

Подходите за откриване на отклонения, базирани на разстоянието, се основават на измерване на различието между точките с данни чрез използване на някаква мярка за разстояние или метрика, получена от разстоянието между точките. Отклоненията се идентифицират като случаи, които се отклоняват значително от по-голямата част от данните в съответствие с приетите метрики за разстояние.

Най-широко използваният подход за откриване на отклонения, базиран на разстоянието, е k-Nearest Neighbors (kNN). KNN класифицира точките от данни въз основа на мнозинството от класовете на техните k най-близки съседи, като вариантите на алгоритъма се основават на разстоянието до всички точки, разстоянието до най-близкия съсед, средното разстояние до k съседи, медианното разстояние до k съседи. При откриване на аномалии случаите с по-малко съседи от същия клас могат да се считат за отклонения. Метриката за разстояние определя близостта. Сред метриките за разстояние, използвани в kNN, са Евклидовото разстояние (най-често срещаното), разстоянието на Манхатън, разстоянието на Махаланобис. Класификаторът KNN е универсален и ефективен както за едномерни, така и за многомерни набори от данни. Изборът на k влияе върху чувствителността към локални модели.

Подходите, основани на разстоянието, са универсални и ефективни, особено когато не се приема, че разпределението на основните данни следва определена форма. Тези методи могат да бъдат адаптирани към различни видове данни в софтуерни системи и често са изчислително ефективни. Предимствата на подходите, базирани на разстояния, са способността да се обработват големи масиви от данни, въпреки че изключително високата размерност драстично намалява производителността.

3.3. Подходи, базирани на плътността

Подходите за откриване на аномалии, базирани на плътността, се фокусират върху идентифицирането на области с по-ниска плътност на данните, като считат аномалиите за случаи, които се отклоняват от по-голямата част от данните. Подходите, основани на плътността, обикновено се основават на к-средни клъстери, които групират

точките с данни в k клъстера въз основа на тяхното сходство, което може да бъде изчислено чрез Евклидовото разстояние в пространството на признаците или други показатели за сходство. Случаите, които не се вписват добре в нито един клъстер, се считат за потенциални отклонения. Сред подходите за откриване на отклонения, базирани на плътността, са:

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise - пространствено клъстериране на приложения с шум): клъстеризира точките с данни въз основа на тяхната плътност. Подходът дефинира основни точки, които имат определен брой съседи в рамките на определен радиус, и гранични точки, които имат по-малко съседи, но са в рамките на радиуса на основната точка. Точките, които не са основни или гранични точки и имат по-малко съседи, се третират като шум. Използва два параметъра: радиуса около точката с данни и минималния брой точки с данни в рамките на радиуса, за да се счита точката за основна точка. DBSCAN е ефективен за набори от данни с различна гъстота и клъстери с неправилна форма;

- LOF (Local Outlier Factor): измерва отклонението на локалната плътност на точка от данни по отношение на нейните съседи. Подходът съчетава аспектите на подходите, базирани на плътността и разстоянието. Използва параметър k за броя на използваните съседи. Аномалиите имат по-ниска локална плътност, което показва, че се намират в по-слабо населени райони. LOF е ефективен за набори от данни с различна плътност и идентифицира точките в по-малко плътни региони като аномалии.

Подходите, базирани на плътността, са особено ефективни, когато аномалиите се характеризират с отклонение от преобладаващата плътност на данните и са подходящи за набори от данни с неправилна форма на клъстерите и различна плътност. Може да се наложи параметрите на конкретния алгоритъм да бъдат внимателно настроени въз основа на характеристиките на данните. Методите, базирани на плътността, могат да се сблъскат с предизвикателства, свързани с мащабируемостта в пространства с голяма размерност.

3.4. Подходи за откриване на аномалии, базирани на класификация

Подходите за откриване на отклонения, базирани на класификация, включват обучение на модел за машинно обучение върху маркирани данни, при което нормалните и аномалните случаи се идентифицират изрично по време на етапа на обучение. След това моделът класифицира новите случаи като нормални или отклоняващи се въз основа на научените модели. Сред подходите за откриване на отклонения, базирани на класификация, са:

- Логистична регресия: линеен класификатор, който моделира вероятността дадена точка от данни да принадлежи към определен клас. При откриването на отклонения логистичната регресия може да бъде обучена върху нормални случаи и аномалиите се идентифицират въз основа на ниски вероятностни резултати. Логистичната регресия е проста и разбираема, подходяща за линейно разделящи се данни;

- Дървета на решенията: могат да се използват за контролирано откриване на аномалии чрез обучение на модела върху нормални случаи. Аномалиите се

идентифицират въз основа на пътищата за вземане на решения, които се отклоняват значително от научените нормални модели. Дърветата на решенията са прост и интерпретируем подход за откриване на отклонения, ефективен за улавяне на нелинейни връзки;

- Поддържащи векторни машини (SVM): в условията на класификация SVM могат да бъдат обучени както за нормални, така и за аномални случаи. Хиперплоскостта, научена от SVM, се използва за отделяне на нормалните случаи от аномалиите. SVM от един клас (Support Vector Machine) е алгоритъм за машинно обучение, който обикновено се използва за откриване на отклонения, особено в сценарии, при които за обучение са налични само нормални случаи. Той принадлежи към семейството на методите за еднокласно обучение, които имат за цел да създадат модел на нормално поведение и да идентифицират отклоненията от това поведение като аномалии. Еднокласният SVM поддържа различни типове ядра, като линейно, функция с радиална основа (RBF) и полиномно. Изборът на ядрото зависи от характеристиките на данните.

Подходите за откриване на отклонения, базирани на класификация, могат да се прилагат в различни области, включително киберсигурност, откриване на измами и контрол на качеството на софтуерни продукти, както на базата на метрики, така и на лог данни. Изборът и проектирането на характеристики, които са от значение за откриването на аномалии, е от решаващо значение за ефективността на модела. Прецизната настройка на хиперпараметрите на избрания класификационен алгоритъм за оптимална производителност е важен въпрос, както и справянето с дисбалансите в разпределението на нормалните и аномалните случаи в обучаващата съвкупност.

3.5. Подходи за откриване на аномалии, базирани на ансамбъл

Базираните на ансамбъл подходи за откриване на аномалии включват комбиниране на множество модели за подобряване на цялостната ефективност на откриването, като по този начин се използва разнообразието на отделните модели за подобряване на общата устойчивост и точност. Сред ансамбловите подходи за откриване на отклонения са:

- Random Forest: метод за ансамблово обучение, който изгражда множество дървета на решенията и комбинира техните прогнози. В контекста на откриването на аномалии моделът се обучава върху нормални случаи и аномалиите се идентифицират въз основа на несъгласието между дърветата. Всяко дърво в гората гласува за класа на дадена точка от данни. Аномалиите се идентифицират въз основа на нивото на несъгласие между дърветата, което осигурява устойчивост на шума в данните.

- Isolation Forest: ансамблов метод, специално разработен за откриване на аномалии. Той конструира гора от дървета за изолиране, където всяко дърво изолира аномалии, като изисква по-малко разклонения в структурата на дървото. Комбинира резултатите от няколко дървета в гората. Аномалиите се идентифицират въз основа на средната стойност или гласуването на прогнозите на отделните дървета. Ефективна е както за едномерни, така и за многомерни набори от данни и е приложима за данни с висока размерност. Също така е ефективен за откриване на редки аномалии, което е

особено важно за откриване на отклонения на софтуерни продукти, базирани на метрики или лог данни.

- Алгоритми за усилване на ансамбъл: итеративно комбинират слаби обучаеми (модели, които се представят малко по-добре от случайността), за да формират силен класификатор. За откриване на аномалии може да се приложи усилване, за да се комбинират модели, обучени върху нормални случаи. На всеки слаб обучаем се присвояват тегла въз основа на неговото представяне. Аномалиите се идентифицират въз основа на колективното решение на усилените модели. Ефективно е, когато отделните модели улавят различни аспекти на нормалното поведение, като по този начин могат да подобрят цялостното представяне на модела.

- Системи за гласуване: простите системи за гласуване включват комбиниране на прогнозите на множество модели въз основа на схема за гласуване с мнозинство или претеглени гласове. Аномалиите се идентифицират въз основа на решението на мнозинството или претеглената комбинация от прогнозите на отделните модели. Системите за гласуване са лесни за прилагане и тълкуване и са ефективни, когато отделните модели показват различни граници на решенията.

Подходите, базирани на ансамбъл, са мощни инструменти за откриване на аномалии, като използват силните страни на множество модели за постигане на подобрена точност и устойчивост. Изборът на конкретен ансамблов метод зависи от характеристиките на данните и естеството на аномалиите, налични в софтуерния продукт. Ефективността на ансамбловите методи зависи от разнообразието на отделните модели. Разнообразните модели трябва да обхващат различни аспекти на нормалното поведение. Обучението на ансамбъла обикновено изисква маркирани данни, особено когато се използват техники с наблюдение, но може да се прилага и при обучение без наблюдение и без анотирани набори от данни. Въпреки това осигуряването на представителна съвкупност от данни с нормални и аномални случаи е от решаващо значение както при неанотираните, така и при анотираните съвкупности от данни. Необходим е внимателен подбор на базовите модели, за да се осигури разнообразие и допълняемост при улавянето на аномалии.

3.6. Невронни мрежи и дълбоко обучение за откриване на аномалии

Невронните мрежи предлагат мощен подход за откриване на аномалии, особено в сценарии, в които аномалиите са сложни и не отговарят на лесно дефинирани модели. Сред многото различни архитектури на невронните мрежи най-широко използвани за откриване на отклонения са[10] [11]:

- Невронна мрежа с автоенкодер: автоенкодерът се състои от енкодер и декодер. Кодерът компресира входните данни в по-нискоизмерно представяне (кодиране), а декодерът реконструира входните данни от това кодиране. Автоенкодерът се обучава да минимизира разликата между входа и реконструирания изход, като ефективно се научава

[10] E. Filho, L. Brandão, B. Fernandes, A. Maciel, A review of neural networks for anomaly detection, IEEE Access, Vol. 10, pp. 112342-112367, 2022

[11] G. Pang, C. Shen, L. Cao, A. Hengel, Deep learning for anomaly detection: A review, ACM Computing Surveys, Vol. 54, No. 2, pp. 1-38, 2021

на компресирано представяне на нормалните данни. По време на обучението автоенкодерът е изложен само на нормални случаи и се научава да кодира нормалните данни, като улавя присъщите им модели и структури. Моделът се обучава да реконструира нормалните случаи точно, като насърчава мрежата да улавя основните характеристики на данните. Аномалиите се идентифицират чрез измерване на разликата между входните данни и реконструирания изход. По-голямата грешка при реконструкцията показва, че входните данни се отклоняват от научените нормални модели. Грешката при реконструкцията може да се изчисли с помощта на показатели като средна квадратна грешка (MSE) или други мерки за сходство. За грешката при реконструкция се определя праг. Случаите, при които грешките на реконструкция надвишават прага, се класифицират като аномалии. Прагът може да се определи въз основа на статистически методи, познания в областта или чрез използване на данни за валидиране. Определянето на подходящ праг за грешката при реконструкция е от решаващо значение. То може да изисква експериментиране или познания в областта. Автоенкодерите имат хиперпараметри, като например броя на слоевете, размера на кодиращия слой и функциите за активиране. Тези параметри трябва да бъдат прецизно настроени за постигане на оптимална производителност. Вариационните автоенкодери (VAE) са вариант на автоенкодерите, които въвеждат вероятностни компоненти в процеса на кодиране. Те моделират несигурността в данните и могат да осигурят по-стабилно откриване на аномалии. Ефективността на автоенкодерите зависи от избора на характеристики и от това колко добре кодирането улавя основните модели на нормалните данни. Ако наборът от данни е силно дисбалансиран с малък дял аномалии, може да се наложи моделът да бъде коригиран, за да се справи с дисбалансираните класове.

- Конволюционна невронна мрежа (CNN): CNN са клас дълбоки невронни мрежи, предназначени за обработка на пространствени йерархии от характеристики и са особено подходящи за обработка на данни, подобни на мрежи, като например изображения, данни за времеви редове или всякакви данни, които могат да бъдат представени като мрежа. CNN обикновено се използват за класификация на изображения и откриване на обекти, но могат да бъдат адаптирани и за откриване на отклонения в сценарии, при които данните имат мрежовидна структура. CNN могат да бъдат адаптирани за откриване на отклонения, като се обучават върху нормални случаи и се идентифицират аномалии въз основа на отклонения от научените модели. Мрежата може да бъде проектирана така, че да научи йерархични представяния, които улавят сложни модели в данните, което ѝ позволява да открива аномалии, които се отклоняват от тези модели. Архитектурата на CNN може да бъде персонализирана въз основа на характеристиките на данните. Обикновено тя включва конволюционни слоеве за извличане на признаци и обединяващи слоеве за намаляване на извадката. По време на етапа на обучение CNN е изложен само на нормални случаи. Мрежата се научава да улавя йерархичните характеристики и модели, присъстващи в нормалните данни. Конволюционните слоеве използват филтри за откриване на модели в различни области на входните данни. Активациите в картите на признаците, създадени от тези филтри, могат да бъдат показателни за специфични модели или структури в данните. Слоевете за обединяване се използват за намаляване на пространствените измерения на данните, като същевременно се запазват важни характеристики. Това помага за създаването на по-абстрактно и компактно представяне. Към мрежата често се добавят сплескани и напълно свързани слоеве, за да се свържат

абстрахираните характеристики и да се направят прогнози. Изборът на функцията за загуба за CNN е от решаващо значение и тя трябва да се избере въз основа на естеството на данните и целта за откриване на отклонения, като широко се използва средната квадратична грешка (MSE). Подобно на автоенкодерите, грешката при реконструкцията между входа и реконструирания изход може да се използва за идентифициране на аномалии, като случаите с по-високи грешки при реконструкцията е по-вероятно да са отклонения. Регулирането на хиперпараметри, като например броя на слоевете, размера на филтъра и скоростта на обучение, е от съществено значение за оптимизиране на работата на CNN. Предварително обучените CNN, обучени върху големи набори от данни за изображения, могат да бъдат прецизно настроени за задачи за откриване на отклонения, като се използва трансферно обучение, което използва научените характеристики от голям набор от данни и ги адаптира към конкретна задача[12]. CNN могат да бъдат прилагани към поточни данни или данни в реално време, което ги прави подходящи за сценарии, при които аномалиите трябва да се откриват в момента на възникването им.

- Рекурентна невронна мрежа (RNN): RNN са вид архитектура на невронни мрежи, предназначени за обработка на последователни данни чрез улавяне на зависимости и модели във времето. RNN имат способността да поддържат памет за минали входове, което ги прави подходящи за задачи, включващи последователности, като например данни за времеви редове или последователности от събития. RNN обработват входните последователности по един елемент в даден момент, като поддържат вътрешно състояние, което съдържа информация за предишните елементи в последователността. RNN могат да бъдат адаптирани за откриване на отклонения, като се обучават на последователности от нормални случаи и се идентифицират аномалии въз основа на отклонения от научените времеви модели. Архитектурата на RNN включва повтарящи се слоеве, които имат връзки, образуващи цикъл, позволяващ предаването на информация от една стъпка на последователността към следващата. Дългосрочната краткотрайна памет (LSTM) и затворената рекурентна единица (GRU) са популярни видове рекурентни слоеве, които решават проблема с изчезващия градиент и са способни да улавят дългосрочни зависимости[13]. По време на обучението RNN е изложена само на последователности от нормални случаи и по този начин мрежата се научава да улавя времеви модели и зависимости в нормалните последователности. Аномалиите могат да бъдат идентифицирани чрез измерване на разликата между входната последователност и реконструираната изходна последователност. По-големите грешки при реконструкцията показват случаи, които се отклоняват от научените времеви модели. Определя се праг на грешката на реконструкция, за да се класифицират случаите като аномалии, като случаите с грешки на реконструкция, надвишаващи прага, се считат за отклонения. Регулирането на хиперпараметри, като например броя на рекурентните слоеве, скритите единици и скоростта на обучение, е от съществено значение за оптимизиране на работата на RNN. Предварително обучените RNN, особено тези, които

[12] R. Chen et al., LogTransfer: cross-system log anomaly detection for software systems with transfer learning," IEEE 31st International Symposium on Software Reliability Engineering (ISSRE), Coimbra, Portugal, pp. 37-47, 2020.

[13] R. Vinayakumar, K. Soman, P. Poornachandran, Long short-term memory based operation log anomaly detection, International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, pp. 236-242, 2017

са обучени на големи последователни данни, могат да бъдат прецизно настроени за задачите за откриване на отклонения. RNN могат да бъдат прилагани към поточни или последователни данни в реално време, което ги прави подходящи за сценарии, при които аномалиите трябва да бъдат откривани в момента на възникването им.

- Генерираща адверсантска мрежа (GAN): GAN са клас невронни мрежи, въведени за генериране на нови данни, които са подобни на даден набор от данни. GAN се състоят от две мрежи - генераторна и дискриминаторна - които се обучават едновременно чрез обучение по неприятелски принцип. Въпреки че GAN се използват предимно за генериране на нови данни, те могат да бъдат използвани за откриване на отклонения, като се използва способността на дискриминатора да прави разлика между реални и генерирани данни. Мрежата на генератора създава синтетични екземпляри, които трябва да приличат на нормалните екземпляри в набора от данни. Тя приема случаен шум като вход и генерира образци, които в идеалния случай са неразличими от реалните случаи. Дискриминиращата мрежа се обучава да прави разлика между реалните случаи от набора от данни и синтетичните случаи, генерирани от генератора. Това е двоичен класификатор, който извежда вероятността даден пример да е реален. В контекста на откриването на отклонения GAN се обучава върху нормални случаи от набора от данни. Дискриминантът се използва като детектор на аномалии, тъй като става умел в разграничаването на реални и синтетични случаи. По време на обучението GAN е изложен само на поредици от нормални случаи. Генераторът се опитва да генерира синтетични случаи, които приличат на нормалните данни, а дискриминаторът се научава да прави разлика между реални и синтетични случаи. След като GAN е обучен, дискриминаторът може да се използва самостоятелно като детектор на аномалии: при даден случай дискриминаторът присвоява вероятностна оценка, показваща вероятността той да е реален случай. Случаите с ниска вероятност се считат за аномалии. За да се класифицират случаите като нормални или аномалии, се задава праг на вероятностните оценки: случаите с вероятностни оценки под прага се считат за аномалии. Прецизната настройка на хиперпараметри, като например скоростта на обучение, архитектурата на генератора и дискриминатора и продължителността на обучението, е от съществено значение.

Основните предимства на невронните мрежи и дълбокото обучение, приложени за откриване на отклонения, са следните:

- разпознаване на сложни модели: невронните мрежи са отлични в улавянето на сложни модели и връзки в данните, което ги прави подходящи за откриване на аномалии в сложни структури, които могат да бъдат предизвикателство за традиционните методи;
- изучаване на характеристики: моделите за дълбоко обучение могат автоматично да научат съответните характеристики от данните, като намаляват необходимостта от ръчно разработване на характеристики и се адаптират добре към набори от данни с голяма размерност;
- адаптивност: невронните мрежи са адаптивни към различни типове данни, включително структурирани и неструктурирани данни, и могат да бъдат адаптирани към специфични типове аномалии в различни области;
- обработка в реално време: моделите за дълбоко обучение, особено когато са оптимизирани за внедряване, могат да се справят с обработка в реално

време, което ги прави подходящи за приложения, при които навременното откриване на аномалии е от решаващо значение, какъвто е случаят с мониторинга на софтуерни продукти, основан на откриване на аномалии.

Основното предизвикателство при използването на моделите за дълбоко обучение е, че те често се разглеждат като "черни кутии" и тълкуването на техните решения може да бъде трудно. Освен това за ефективното им обучение са необходими големи количества маркирани или немаркирани данни, а в сценарии с ограничени данни моделите може да се затруднят да обобщават добре. Освен това моделите за дълбоко обучение са склонни към свръхнастройка, особено когато наборът от данни е малък или когато моделът е твърде сложен, и често се налага използването на техники за регуларизация, за да се смекчи свръхнастройката. Обработката на небалансирани набори от данни, при които броят на нормалните случаи значително надвишава аномалиите, изисква внимателно обмисляне, за да се предотврати тенденциозното обучение на модела. Въпреки че моделите за дълбоко обучение могат да се справят добре с конкретни задачи, способността им да обобщават за нови, невиждани аномалии или да се адаптират към променящите се условия може да бъде ограничена.