

Modelovanie bezpečnosti softvéru pre automatizovanú výrobu na neurónovej sieti*

Anastasiia Alekperova

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
xalekperova@stuba.sk

6. november 2021

Abstrakt

Článok je venovaný problematike matematického modelovania a umelých neurónových sietí pre programovanie bezpečnosti komplexných softvérových systémov. Ukáže sa, že na modelovanie spoľahlivosti je možné použiť vertikálne vrstvené siete.

1 Úvod

V medzinárodnej norme ISO 9126: 1991 [1] je spoľahlivosť zvýraznená ako jedna z hlavných charakteristík kvality softvéru. Štandardný slovník [2] termíny softvérového inžinierstva definuje softvérovú bezpečnosť ako schopnosť systému alebo komponentu vykonávať požadované funkcie v špecifikovanom podmienok počas stanoveného časového obdobia.

V súčasnej dobe výroba stále viac automatizuje monotónne úlohy s pomocou automatizovaných riadiacich systémov založených na priemyselnej alebo osobnej počítačovej technológii, programovateľných logických automatoch alebo špecializovaných počítačových systémoch [4]. Celá táto počítačová technológia vyžaduje na svoju činnosť softvér. Nesprávna funkcia najmenej jednej z uvedených komponentov bude mať určite za následok zlyhanie celého systému, čo môže mať za následok nielen výrobné straty, ale aj poškodenie zdravia. Podľa štatistík je 50% bezpečnostných problémov v softvérových produktoch dnes spôsobené nesprávnou a nebezpečnou štruktúrou softvéru [5].

Niekedy je ľudským faktorom príčinou chyby výpočtov v analytických údajoch, takže je lepšie dôverovať prevádzke neurónovej siete.

2 Typy matematických modelov

Modely bezpečnosti softvéru sú rozdelené na analytické a empirické modely [3]. Analytické modely umožňujú vypočítať kvantitatívne ukazovatele spoľahlivosti na základe

*Semestrálny projekt v predmete Metódy inžinierskej práce, ak. rok 2021/22, vedenie: Vladimír Mlynárovíč

údajov o správaní programu počas testovacieho procesu. Empirické modely sú založené na analýze štrukturálnych vlastností programov.

Analytické modely sú prezentované v dvoch skupinách: dynamické a statické. V dynamických modeloch sa správanie softvéru zohľadňuje v čas. Zaujímajú nás kvantitatívne bezpečnostné indikátory, preto uvažujme o analytických modeloch

2.1 Analytické dynamické modely

Napríklad, **model La Padula** vyzerá tak [6]:

$$R(t) = R(\infty) - A/i, i = 1, 2, \dots \quad (1)$$

kde A - rastový parameter, $R(\infty) = \lim_{x \rightarrow \infty} R(i)$ - maximálna spoľahlivosť softvéru. Neznáme množstvá možno nájsť z nasledujúcich rovníc:

$$\sum_{i=1}^m (S_i - m_i)/S_i - R(\infty) + A_i/i = 0 \quad (2)$$

kde S_i - počet testov v i -tom štádiu, m_i - počet porúch počas i -tej etapy ($i = 1, 2, \dots, m$).

Podľa tohto modelu sa vykonávanie postupnosti testov vykonáva v m etapách. Každá etapa končí zmenami softvéru.

Model je dynamický, ale vo vzorci sa nepoužíva čas t . Dynamika sa prejavuje v opakovanom výpočte výsledku pri rôznych stavoch softvéru.

V **modeli Djelinsky - Moranda** základným postojom, na ktorom je model založený, je, že v procese testovania softvéru sa hodnota testovacích časových intervalov medzi detekciou dve chyby majú exponenciálnu distribúciu s mierou zlyhania úmernou počtu ešte nezistených chýb. Každá zistená chyba je odstránená, počet zostávajúcich chýb sa zníži na jednu.

Funkcia hustoty času:

$$P(t_i) = \lambda_i * \exp(-\lambda_i * t_i) \quad (3)$$

kde λ_i - intenzita poruchy.

Hodnota λ_i sa vypočíta z výsledkov testov, použitím segmentov času od spustenia programu do poruchy.

Toto je jeden z prvých a najjednoduchších modelov klasického typu, ktorý slúžil ako základ pre ďalší vývoj v tomto smere. Model bol použitý vo vývoji takýchto významných softvérových projektov ako program APOLLO (niektoré z jeho modulov).

2.2 Analytické statické modely

(Poznámka:clanok je vo vyvoji)

Tu bude povedané o ďalších typoch modelov

1. prvy
2. druhy

3 Testovania pomocou neurónovej siete

3.1 Transformácia výrazov

V článku [4] je prediktívny model bezpečnosti softvéru. Je založený na modeli Djelinsky - Moranda, ale vyzerá to trochu iné.

$$H(t_1, t_2, \dots, t_n) = \sum_{i=1}^N v_i * \beta_i * \exp(-\gamma_i * t_i) \quad (4)$$

To znamená, že pre vysokú presnosť predikciu spoľahlivosti, riešenie veľkého počtu nelineárnych rovníc a každý obsahuje exponent.

Zjednodušíme model a transformujeme ho na vzorec vhodný pre modelovanie. Nahradíme premenné a získame množstvo exponentov [4].

$$H(t) = \sum_{i=1}^N \exp(-b_i * t + a_i) \quad (5)$$

Získaný výsledok má teoretické riešenie, ale v praxi je ťažké použiť:

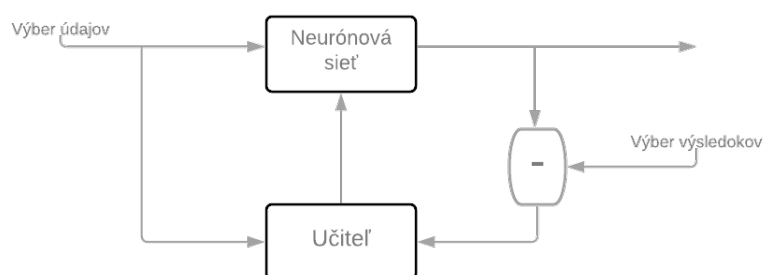
1. Riešenie jednej rovnice určite vedie k inému rozhodnutiu a tak ďalej
2. Je ťažké určiť hodnotu N, ktorá je potrebná pre presný výsledok.

3.2 Použitie umelej nervovej siete

Použitie neurónových sietí v bezpečnosti nie je nové. Existuje dostatočne veľký počet systémov založených na nervových sieťach a podporných funkciách, detekcii, odhaľovaní nesprávnej práce [5].

Naša neurónová sieť sa musí naučiť predpovedať výsledok založený na časových údajoch.

Jednoduchá schéma prevádzky neurónovej siete znázornená na obr.1.



Obr. 1: schéma prevádzky neurónovej siete

4 Zhodnotenie

Tým padom...

Literatúra

- [1] Thamer A Alrawashdeh, Mohammad Muhairat, and Ahmad Althunibat. Evaluating the quality of software in erp systems using the iso 9126 model. *International Journal of Ambient Systems and Applications (IJASA)*, 1(1):1–9, 2013.
- [2] Martha Branstad and Patricia B Powell. Software engineering project standards. *IEEE transactions on software engineering*, (1):73–78, 1984.
- [3] Viktor Alekseevich Grateful, Vladimir Aleksandrovich Volnin, and Kirill Felixovich Poskakalov. Standardization of software development. 2005.
- [4] Kabak I.S. and Rapoport G.N. Estimation of the reliability of software on its mathematical model. *Problems of creating flexible automated production*, 1987.
- [5] S.V. Polupudnikov and A.A. stakhov. Tools for analyzing software based on neural networks. In *ADVANCED SCIENCE*, pages 46–48, 2018.
- [6] N.V. Vasilenko and V.A. Makarov. Software reliability evaluation models. *Vestnik Novgorod State University named Yaroslav Mudry*, (28), 2004.