VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ BAKALAURO STUDIJŲ PROGRAMA

Neuroninio tinklu pritaikymas automatizuotam programinės įrangos sistemų problemų sprendimui

Automated software system issue solving using deep neural networks

Bakalauro baigiamojo darbo planas

Atliko: Mantas Petrikas (parašas)

Darbo vadovas: dr. Vytautas Valaitis (parašas)

Vilnius – 2023

Darbo planas

Bakalauriniame darbe tiriamos modernių natūralios kalbos apdorojimo modelių pritaikymo galimybės programinės įrangos sistemų problemų sprendimui. Dauguma modernių giliaisiais neoriniais tinklais paremtų natūralios kalbos apdorijų sistemų tokiu kaip Salesforce sukurtas CodeT5 [WWJ+21], Microsoft CodeBERT [FGT+20] ar OpenAI GPT-3 [BMR+20] sugeba labai gerai generuoti kodo dalis, tačiau tokių modelių pritaikymo galimybės spręsti sistemos lygio problemas nėra iki galo ištirtos.

Darbo tikslas ir uždaviniai

Darbo tikslas - sukurti giliaisiais neuroniniais tinklais paremtą sistemą kuris pagal programinės sistemos problemos aprašymą sugebėtų identifikuoti kodo dalis kurias reikėtų pakeisti norint išspręsti problemą ir pateikti galimus kodo pakeitimus.

Uždaviniai:

1. išanalizuoti ir įvertinti dabartinių kodo generavimo įrankių galimybės spęsti sistemos lygio problemas
2. surinkti sistemos funkcionalumo pakeitimų aprašymų ir realizuotų kodo pakeitimų skirta modelio mokymui
3. suprojektuoti sistemą, kuri priima sistemos kodą ir funkcionalumo aprašymą ir gražiną sąrašą failų kuriuos reikėtų pakeisti norint įgyvendinti funkcionalumą
4. apmokyti suprojektuotos sistemos giliuosius neuroninius modelius naudojant surinktą duomenų rinkinį ir įvertinti modelių veikimą
5. praplėsti sukurtos sistemos funkcionalumą pridedant kodo generavimo galimybę
6. įvertinti sukurtą sistemą palyginant su kitomis koda generuojančiomis sistemos

Laukiami rezultatai

1. sistemos funkcionalumo pakeitimų aprašymų ir realizuotų kodo pakeitimų duomenų rinkinys
2. kuriamos sistemos architektūros aprašymas
3. realizuota sistema sugebanti identifikuoti failus, kurios reikia pakeisti norint įgyvendinti sistemos funkcionalumo pakeitimą
4. realizuota sistema generuojanti kodo pakeitimus reikalingus sistemos funkcionalumo pakeisti
5. sukurtos sistemos giliųjų neuroninių tinklų modelių įvertinimo ataskaita

Darbo metodai

- Susijusios literatūros, duomenų rinkinių ir giliųjų neorinių tinklų modelių analizė
- Skirtingu kodo generavimo modelių palyginamoji analizė
- Atvirai pasiekiamų funkcionalumo pakeitimų aprašymų ir realizuotų kodo pakeitimų šaltinių analizė
- Neuroninių tinklų pritaikymas problemai spęsti naudojant perkeliamąjį mokymasi

Darbo eiga

Darbe bus įvertinami esami atviro ir uždaro kodo produktų galimybės generuoti sistemos pakeitimus spręsti, pateikiant naujo funkcionalumo aprašymą. Darbe bus palyginamos skirtingos giliųjų neuroninių tinklų architektūros naudojamos šiai užduočiai spęsti. Apžvelgiami skirtingi duomenų rinkiniai naudoti apmokyti giliųjų neuroniniams tinklams ir jų aktualumas tiriamai problemai. Darbe bus įvertinos skirtingi duomenų apdorojimo ir tokenizavimo technikos norint paruošti duomenys modelio mokymui. Taip pat apžvelgiami modelio mokymui ir validavimui naudojami kriterijai.

Darbo praktinėje dalyje naudojantis laisvais prieinamais šaltiniais ir duomenų rinkiniais bus atrinktas duomenų rinkinys tinkamas šio darbo nagrinėjamos problemos sprendimui naudojamo giliojo neutroninio tinklo mokymui. Duomenų rinkinį turėtų sudaryti sistemos kodas, sistemoje esančios problemos ar naujo funkcionalumo aprašas, ir kodo pakeitimai sistemos funkcionalumo pakeitimui įgyvendinti. Norint pagreitinti sistemos sukūrimo ir apmokymo laiką bus bandoma kuo labiau pritaikyti jau laisvai prieinamus apmokytus modelius, tokius kaip CodeBERT [FGT+20] ar GPT-2 [RWC+19]. Turint duomenų rinkinį bus atrinka užduočiai tinkama teksto tokenizavimui skirta sistema. Tada bus sukurta sistemos architetūra ir apmokytas gilusis neoroninis tinklas sugebentis identifikuoti kurios sistemos failus reikia pakeisti norint pakeisti sistemos funkcionalumą. Realizavus šią sistemą, atsižvelgiant į darbui pateikti likusį laiką, bus bandomą ją praplėsti galimybe generuoti kodo pakeitimus reikalingus problemos sprendimui.

Literatūra

[BMR+20] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah ir k.t. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[FGT+20] Zhangyin Feng, Daya Guo, Duyu Tang, Nan Duan ir k.t. Codebert: A pre-trained model for programming and natural languages. *arXiv preprint arXiv:2002.08155*, 2020.

[RWC+19] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever ir k.t. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[WWJ+21] Yue Wang, Weishi Wang, Shafiq Joty ir Steven CH Hoi. Codet5: Identifier-aware unified pre-trained encoder-decoder models for code understanding and generation. *arXiv preprint arXiv:2109.00859*, 2021.