SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-xxxx-xxxx

AUTONÓMNY ALGORITMUS NA ZAPISOVANIE A VYHODNOCOVANIE PRODUKTOVEJ SUPERPOZÍCIE

BAKALÁRSKA PRÁCA

2022 **Martina Mahelyová**

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-xxxx-xxxx

AUTONÓMNY ALGORITMUS NA ZAPISOVANIE A VYHODNOCOVANIE PRODUKTOVEJ SUPERPOZÍCIE

BAKALÁRSKA PRÁCA

Študijný program:	Aplikovaná informatika
Číslo študijného odboru:	2511
Názov študijného odboru:	9.2.9 Aplikovaná informatika
Školiace pracovisko:	Ústav informatiky a matematiky
Vedúci záverečnej práce:	doc. Ing. Milan Vojvoda, PhD.
Konzultant:	Ing. Martin Rástocký

Bratislava 2022 **Martina Mahelyová**

SÚHRN

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Študijný program:	Aplikovaná informatika
Autor:	Martina Mahelyová
Bakalárska práca:	Autonómny algoritmus na zapisovanie a vyhodnocovanie produktovej superpozície
Vedúci záverečnej práce:	doc. Ing. Milan Vojvoda, PhD.
Konzultant:	Ing. Martin Rástocký
Miesto a rok predloženia práce:	Bratislava 2022

Jeden odsek 100-500 slov.

Kľúčové slová: kľúčové slovo1, kľúčové slovo2, kľúčové slovo3

ABSTRACT

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION TECHNOLOGY

Study Programme:	Applied Informatics
Author:	Martina Mahelyová
Bachelor's thesis:	Autonomous algorithm for recording and evaluating product superposition
Supervisor:	doc. Ing. Milan Vojvoda, PhD.
Consultant:	Ing. Martin Rástocký
Place and year of submission:	Bratislava 2022

One paragraph. 100-500 words.

Keywords: keyword1, keyword2, keyword3

Vyhlásenie autora

Čestne vyhlasujem, že som bakalársku prácu s názvom: Autonómny algoritmus na zapisovanie a vyhodnocovanie produktovej superpozície na základe poznatkov získaných počas štúdia a informácií z dostupnej literatúry uvedenej v práci. Uvedenú prácu som vypracovala pod vedením Ing. Martina Rástockého.

V Bratislave dňa dd.MM.yyyy

.....
podpis autora

Poďakovanie

I would like to express a gratitude to my thesis supervisor.

Obsah

Zoznam obrázkov a tabuliek

Zoznam algoritmov

Zoznam výpisov

Úvod

TODO: dopísať úvod do problematiky

Disponentské modely sú v službách a špeciálne v bankovníctve a poisťovníctve veľmi dôležité. Určujú, ktorý klient má na čo oprávnenie, kto môže s jeho produktami nakladať a aký je jeho pohľad na majetok. Banky a poisťovne ponúkajú mnoho rôznych produktov a v súčasnosti je ich už toľko, že spoločnosti predávajú aj cudzie produkty, nie iba svoje. Preto sa tieto modely počítajú veľmi komplikovane. A tak vznikla potreba vytvoriť produktový zoznam a disponentský model pre skupinu používateľov, ktorý bude mať na klientských produktoch príznaky. Takéto príznaky hovoria o tom, z akej spoločnosti produkt pochádza a ktorý klient ho využíva. O tom, kde a aký príznak treba pridať, by mal rozhodovať jednoduchý rozhodovací systém na základe rôznych parametrov a zdrojov. Návrh a implementácia takéhoto systému je predmetom tejto bakalárskej práce.

1 Teoretická časť

2 Analytická časť - Analýza možných riešení problematiky

V časti Analýza problému autor uvádza súčasný stav riešenej problematiky doma i v zahraničí, dostupné informácie a poznatky týkajúce sa danej témy. Zdrojom pre spracovanie sú aktuálne publikované práce domácich a zahraničných autorov. Základné definície a formalizmy potrebné na riešenie problematiky.
Tu by mal študent analyzovať relevantné existujúce riešenia, sumarizovať ich klady a nedostatky (vhodné sú napr. sumárne tabuľky) a identifikovať z toho vyplývajúce špecifické požiadavky na vlastnú prácu (čo má riešenie robiť).

Keďže sú dáta najčastejšie uchovávané v databázach, najjednoduchšou cestou k identifikovaní ich potenciálnych duplicít a ich následné odstránenie je práve skr databázu.

1. T-SQL párovanie reťazcov "natvrdo"

Zdanlivo najjednoduchším prístupom by bol bolo napísať jednoduchý program, ktorý by v cykle čítal všetky záznamy a na základne jednoznačného identifikátora by ich navzájom porovnával a následne vkladal záznamy do novej tabuľky s alebo bez príznaku. Muselo by sa stanoviť, ako porovnávať záznamy v prípade chýbajúceho jednoznačného identifikátoru. Potom by pravdepodobne do porovnávacích podmienok muselo vstupovať viacero rôznych parametrov v závislosti od zdrojových dát a existencie jednoznačného identifikátoru. Takéto riešenie by však nebolo najvhodnejšie, veľmi ľakho sa totiž môže stať, že sa v takejto analýze vynechá jeden krajný prípad, ktorý by mohol spôsobiť stovky a tisícky zle vyhodnotených príznakov v záznamoch. Takto vyhodnotené príznaky by sa hladali ťažko, keďže krajný prípad nebol definovaný v analýze, a ak by sa v testovaje fáze našla chyba, musela by sa znova prepisovať analytický časť a upravovať už existujúce podmienky.

2. T-SQL fuzzy logika

Upraviť dáta. Levenshtein alebo Damerau-Levenshtein vzdialenosť.
https://nanonets.com/blog/fuzzy-matching-fuzzy-logic/

3. Python knižnica Linkage Toolkit

https://recordlinkage.readthedocs.io/en/latest/about.html

4. Genetický algoritmus

http://www0.cs.ucl.ac.uk/staff/C.Ragkhitwetsagul/files/cmu/Genetic_algorithm_final_report_cha

Ďalším spôsobom vyhotovenia takéhoto programu by bol genetický algoritmus. Ten vychádza z Darwinovej evolučnej teórie. Algoritmus by mohol vyzerať nasledovne: Najskôr by bolo potrebné inicializovať populáciu o veľkosti x chromozónov. Chromozón je jedinec, ktorý predstavuje riešenie problému. Číselná hodnota, ktorá bude predstavovať vhodnosť/správnosť jedinca je priradená každému chromozónu. Na základe tejto hodnoty sa vyberú najvhodnejšie jedince na kríženie a mutáciu. V závislosti od vhodných kandidátov na správne riešenie sa vynegeruje nová populácia a celý proces sa opakuje niekoľkokrát. Výsledkom by mal byť najvhodnejší chromozón, ktorý predstavuje korektné riešenie.

5. Neurónová sieť

Pri vhodne upravených dátach by bolo možné záznamy z rôznych zdrojov medzi sebou porovnávať. Najskôj by museli byť zostrojené trénovacie dáta s manuálne pridaným správnym výsledkom, na ktorých by sa sieť natrénovala. Na ostrých dátach, bez správneho výsledku, by potom sieť vyhodnotila, či sa jedný o duplicitný záznam alebo nie.

3 Návrhová časť

Tu sa rieši otázka ako postupovať pri riešení práce. Študent by mal spracovať návrh riešenia: vychádzajúc z analytickej časti by mal identifikovať hlavné komponenty a interakcie riešenia (architektúra, use-cases, data-flow), a pripraviť softvérový návrh jednotlivých komponentov do hĺbky. Identifikuje/navrhne vhodné algoritmy. Pri experimentálnej/výskumnej práci identifikuje a zdôvodní metodiku experimentov. Táto časť je charakteristická diagramami a schémami, využívajú sa v nej postupy softvérového inžinierstva.

4 Implementačná časť - Opis riešenia

Časť Opis riešenia jasne, výstižne a presne charakterizuje predmet riešenia. Súčasťou sú aj rozpracované čiastkové ciele, ktoré podmieňujú dosiahnutie hlavného cieľa. Ak je práca implementačná, tak jej súčasťou musí byť aj softvérová špecifikácia požiadaviek, návrh, implementácia, overenie riešenia. Treba podľa možností vychádzať zo známych prístupov. Táto časť práce závisí od konkrétneho zadania. Je dôležité prezentovať návrhové rozhodnutia, alternatívy, ktoré sa zvažovali pri riešení a samotný návrh riešenia zadaného problému. Štruktúra textu by mala vychádzať zo zadanej úlohy, ktorá sa rieši. Najmä v tejto časti študent preukazuje svoj originálny prístup k riešeniu problémov a kritické myslenie.
Súčasťou môže byť metodika práce a metódy skúmania, ktoré spravidla obsahujú: a) charakteristiku objektu skúmania b) pracovné postupy c) spôsob získavania údajov a ich zdroje d) použité metódy ich vyhodnotenia a interpretácie výsledkov Implementácia musí byť otestovaná. Výsledok musí byť porovnaný s inými riešeniami.

todo: tu treba dopísať riešenie problému

rišenie prostredníctvom rekurentnej neúronovej siete lebo ..
popísať ako treba upraviť vstupné dáta - normalizácia
+ do každej tabuľky označiť prioritu zdroja ? možno nebude treba keď budem podľa priority joinovať ostatné tabulky
asi na začiatku bude treba vytvoriť dáta na trénovanie a manuálne im označiť match na hodnotu 0/1
nahranie tabulky sporky do tabulky suporpozicia - čo bude výstup programu
k sporke najoinovať každý zdroj, to by mal byť koniec pre upravu dat
popísať ako má vyzerať výsledná najoinovaná tabuľka, jeden záznam, druhý záznam a match
natrénovať model, testovanie, validácia - rozdeliť dáta
rnn na nelablovaných dátach, výstupom bude tabulka s match hodnotou
v závislosti od percenta potom budem pridávať príznak k hlavnému záznamu do superpozicie

5 Výsledky a testovanie

Študent by mal sumarizovať výsledky testov a vyhodnotiť riešenie. Je tiež vhodné porovnať vlastné riešenie s existujúcimi alternatívami. Pri experimentálnych/vedeckých prácach je táto časť najbohatšia, obsahuje relevantné, spracované výsledky (tabuľky, grafy) a z nich odvodené závery.

Záver

Výsledky (vlastné postoje alebo vlastné riešenie vecných problémov), ku ktorým autor dospel, sa musia logicky usporiadať a pri popisovaní sa musia dostatočne zhodnotiť. Zároveň sa komentujú všetky skutočnosti a poznatky v konfrontácii s výsledkami iných autorov. Ak je to vhodné, výsledky práce a diskusia môžu tvoriť samostatné časti ZP. Je potrebné, aby čitateľ dostal sumár výsledkov, zhodnotenie riešenia a jeho prínosov.

Prílohy