UNIVERZITA PARDUBICE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

# POSUDEK VEDOUCÍHO BAKALÁŘSKÉ PRÁCE

**Jméno studenta:** **Martinek Daniel**

**Název práce:** **Implementace evoluční strategie v prostředí Apache Spark**

**Autor posudku:** **doc. Dr. Ing. Tomáš Brandejský**

**Cíl práce:** **Sepište rešerši informačních zdrojů o prostředí Apache Spark, jeho programování a evolučních algoritmech. V některém z podporovaných jazyků v prostředí Apache Spark (Python, Java, Scala) vytvořte efektivní paralelní implementaci optimalizačního algoritmu evoluční strategie. Demonstrujte jeho činnost na řešení optimalizačních úloh, které jsou popsány daty uloženými v databázi dotazované prostřednictvím SparkSQL.**

| **Povinná kritéria hodnocení práce** | **Stupeň hodnocení** (známka) | | | |
|---|---|---|---|---|
| | **A** | **C** | **E** | **F** |
| Práce svým zaměřením odpovídá studovanému oboru | ☒ | ☐ | ☐ | ☐ |
| Vymezení cíle a jeho naplnění | ☐ | ☐ | ☐ | ☒ |
| Zpracování teoretických aspektů tématu | ☐ | ☐ | ☒ | ☐ |
| Zpracování praktických aspektů tématu | ☐ | ☐ | ☐ | ☒ |
| Adekvátnost použitých metod, způsob jejich použití | ☐ | ☒ | ☐ | ☐ |
| Hloubka a správnost provedené analýzy | ☐ | ☐ | ☒ | ☐ |
| Práce s literaturou | ☒ | ☒ | ☐ | ☐ |
| Logická stavba a členění práce | ☒ | ☐ | ☐ | ☐ |
| Jazyková a terminologická úroveň | ☐ | ☐ | ☒ | ☐ |
| Formální úprava a náležitosti práce | ☒ | ☐ | ☐ | ☐ |
| Vlastní přínos studenta | ☐ | ☐ | ☒ | ☐ |
| Využitelnost výsledků práce v teorii (v praxi) | ☐ | ☐ | ☒ | ☐ |

**Dílčí připomínky a náměty:**

Problematický je už seznam značek a zkratek. Některé nejsou v práci použity (proč jsou tedy vysvětlovány), jako Machine Learning Library, GraphX, další jsou nepřesné, jako Evoluční strategie, Genetické programování (nemusí používat stromovou reprezentaci), Yarn, batch processsing (je o 50 let staří než Hadooop, se kterým ho student spojuje), u mnoha pojmů chybí odkaz na literaturu, i když je v seznamu – vedle zmiňovaných by to prospělo např. pojmům BigData, Mesos, Yarn… Generace evolučního algoritmu jsou vysvětleny matoucím způsobem (generace je totéž jako populace, ne

množina všech populací do určité doby). DataFrame je použit ve Sparku jako jednotný pohled na data, ne pro paralelizaci. Spark je popisován slovz „Jeho hlavní předností je unifikovaný framework, který integruje batch processing, streamování, SQL dotazy, strojové učení a analýzu grafů.“, při tom pro analýzu grafů je použita samostatná knihovna GraphX, nejedná se o vlastnost Sparku. Na str. 19 jsou sice vyjmenovány podporované jazyky (viz zadání), ale ve skutečnosti tato čtveřice je celá množina, slovo „jako“ není na místě. Mnohem užitečnější by byla informace, že API pro každý z těchto jazyků podporuje jinou funkcionalitu a žádný z nich nepodporuje všechny funkce Sparku.

RDD slouží k předávání informací mezi jádrem Sparku a vlákny, rozhodně se pomocí nich nekomunikuje se Sparkem (str. 20). „Originální data, informace o transformacích a akcích a rozdělení na nodech je důležité uchovat, dokud se neprovedou všechny zamýšlené akce, aby šlo dopočítat chybějící data, pokud node selže.“ – správně tyto informace uchovává Spark, uvedený text navozuje dojem, že je to starost programátora.

Na str. 21 jsou poněkud nepřesné informace o DataFrame – „K RDD přidává schema“. Ve skutečnosti DataFrame slouží k reprezentaci dat např. z externích DB. Vzájemný převod mezi DataFrame a RDD je možný oběma směry, ale rozhodně by takto DataFrame neměl být definován. Na této straně se čtenář dozví i že „lambda funkce jsou inspirací pro lambda funkce“.

Str. 23. Popis lambda funkcí je také matoucí. Spark především využívá toho, že labda funkce neodkazují na globální proměnné a jsou tedy snadno paralelně vykonávatelné...

Str. 26. Bakalář je pravděpodobně stoupencem Hadoopu, protože na str. 26 uvádí „Hadoop MapReduce byl překonán v některých případech Sparkem.“ Uvedl bych, že Spark je vlastně nadstavbou a z některých případů bych uvedl, že např. v rychlosti překonává Hadoop podle literatury cca 100x. Výhodou in-memory řešení Sparku není jen odstranění latence úložišť, které uvádí student, ale také odstranění jejich nadměrné degradace neustálými zápisy informací. Ne všechna úložiště jsou v praxi čistě flashová.

Str. 27, student měl na mysli, že při zabezpečené komunikaci se používá předsdílený klíč, ne předplacený.

Str. 28. Všechna jádra jsou ale využita pouze pokud toto povolí programátor, z mnoha důvodů tomu tak ve skutečnosti být nemusí.

Str. 32: všechny EA nepoužívají sekvenci křížení-mutace, ale např. náhodně volí jednu z těchto dvou operací.

Str. 34: Genetické programování už od raných prací J. Kozy může využívat i jiné struktury, např. obecné grafy při návrhu el. obvodů, nemusí tedy jen navrhovat programy a nemusí jen vyměňovat podstromy mezi rodiči, ale např. kopírovat, náhodně generovat atd.

Str. 35. Souhlasím, že genetický algoritmus je nejznámější případ evolučního algoritmu, ale rozhodně není nejpoužívanější vzhledem k mnoha problémům překonaným např. diferenciální evolucí.

Kap. 9.6 končí nedokončenou větou…

Poslední bod zadání „Demonstrujte jeho činnost na řešení optimalizačních úloh, které jsou popsány daty uloženými v databázi dotazované prostřednictvím SparkSQL.“ Student odbyl příkladem generovaným v kódu programu, tedy vůbec neřešil otázku předávání dat mezi externí databází a

algoritmem prostředky Sparku. Také přehlédl, že nelze jakkoli srovnávat jednotlivé algoritmy, které jsou ze své podstaty stochastické, na základě jediného běhu jediného příkladu.

Definice funkce combine vypadá zvláště a je s podivem, že mohly být získány výsledky pro porovnání:

```
static double[] Combine(double[] parent1, double[] parent2, double probability)
{
 return Combine(parent1, parent2, probability, false);
}
```

V práci jsou i drobné pravopisné chyby, jako „spark“, „Apache spark“ apod.

**Celkové posouzení práce a zdůvodnění výsledné známky:**

Práce je plná drobných nepřesností a poslední bod zadání byl splněn opravdu jen částečně.

**Vyhodnocení kontroly textu práce pomocí systému pro odhalování plagiátu:**
Práce není plagiátem.

**Otázky k obhajobě:**
Proč nebylo naplněno zadání ohledně načítání testovacích příkladů?

**Práci doporučuji k obhajobě.**

**Navržená výsledná známka: E**

**V** Pardubicích**, dne 22. srpna 2025**

_____

podpis