

Katalóg požiadaviek

Rekonštrukcia top kvarkov

Richard Mészáros

Martin Slimák

Magdaléna Kozubaľová

Veronika Benková

Obsah

1 Úvod

1.1 Účel tohto katalógu požiadaviek

1.2 Rozsah využitia systému

1.3 Slovník pojmov

1.4 Odkazy a referencie

1.5 Prehľad nasledujúcich kapitol

2 Všeobecný popis

2.1 Perspektíva systému

2.2 Funkcie systému

2.3 Charakteristika používateľa

2.4 Všeobecné obmedzenia

2.5 Predpoklady a závislosti

3 Špecifické požiadavky

3.1 Funkčné požiadavky

3.2 Kvalitatívne požiadavky

3.3 Požiadavky rozhrania

Úvod

1.1 Účel tohto katalógu požiadaviek

Tento dokument slúži na špecifikovanie a súhrn požiadaviek od zadávateľa pre aplikáciu Rekonštrukcia Top Kvarkov. Bol vytvorený na základe komunikácie so zadávateľom projektu. Účel tohto dokumentu je zabezpečenie správnosti a úplnosti dohodnutých požiadaviek od zadávateľa. Taktiež dokument je záväzný pre obe strany.

1.2 Rozsah využitia systému

Cieľom projektu je vytvorenie systému, ktorý bude pomocou neurónovej siete priradovať jety ku kvarku z ktorého vznikol. Ďalším cieľom je porovnanie výstupu z KIFitter-u a neurónových sietí vo webovom rozhraní prostredníctvom dát z databázy. Webové rozhranie bude slúžiť na komunikáciu s databázou, serverom atlas23 a interakciou s používateľom.

1.3 Slovník pojmov

Kvark - podľa štandardného modelu časticovej fyziky elementárne častice, z ktorých sa skladajú hadróny (teda napríklad protóny a neutróny).

Top kvark – je kvark tretej generácie s elektrickým nábojom $+(2/3)e$. Je to najťažší zo všetkých známych elementárnych častíc.

Jet - úzky kužeľ hadrónov a iných častíc produkovaných hadronizáciou kvarku alebo gluónu v experimente s fyzikou častíc alebo experimentom s ťažkými iónmi.

KIFitter – knižnica na kinematickú montáž pomocou pravdepodobnosti. Je primárne vyvinutý pre prípad rekonštrukcie top kvarku, ale dá sa ľahko upraviť tak, aby vyhovoval aj iným procesom.

Neurónová sieť – výpočtový model, zostavený na základe abstrakcie vlastností biologických nervových systémov.

Hlboká neurónová sieť – výpočtový model, zostavený na základe abstrakcie vlastností biologických nervových systémov s viacerými vrstvami.

Tensorflow – bezplatná a otvorená softvérová knižnica pre dataflow a diferencovateľné programovanie v rôznych úlohách. Je to matematická knižnica a používa sa tiež na aplikácie strojového učenia, ako sú neurónové siete.

KERAS – vysokoúrovňové API pre neurónové siete, schopné bežať na vrchole Tensorflow. Umožňuje rýchle experimentovanie prostredníctvom vysokoúrovňového, ľahko použiteľného, modulárneho a rozširiteľného API.

ATLAS23 – Server

TTbar event- zrážka top kvakových párov

konfiguračný súbor – súbor obsahujúci všetky potrebné parametre na tréning a následne používanie neurónovej siete.

Procesorový čas: čas potrebný na vykonanie výpočtov na procesore.

1.4 Odkazy a referencie

1. ROOT Framework na analýzu dát:
<https://root.cern.ch/>
2. Ako čítať strom z ROOT:
<https://root.cern.ch/how/how-read-tree>
3. KL Fitter knižnica:
<https://github.com/KLFitter/KLFitter>
–
4. LWTNN knižnica:
– <https://github.com/lwttn/lwttn>
–

1.5 Prehľad nasledujúcich kapitol

V nasledujúcich kapitolách sa čitateľ oboznámi s funkciami ,
perspektívami a obmedzeniami systému, taktiež so špecifickými
požiadavkami.

2. Všeobecný popis

Implementácia neurónovej siete a optimalizácia metódy KL Fitter slúžiacej na rekonštrukciu top kvarkových párov.

2.1 Perspektíva systému

V tomto systéme bude mať užívateľ 2 možnosti výberu rekonštrukcie tt-bar eventov :

1. cez hlboké neurónové siete

2. cez KL Fitter

Tieto možnosti si bude užívateľ môcť vybrať vo webovom rozhraní. Vo webovom rozhraní sa bude dať spustiť tréning neurónovej siete, pre ktorú sa bude môcť vo webovom rozhraní nastaviť parametre. Systém bude využívať aj databázu ktorá bude uložená na serveri ATLAS23.

Po spracovaní dát metódami na rekonštrukciu top kvarkov sa výsledky z týchto metód uložia do databázy. Výstupom budú dáta v grafickej a písomnej podobe ktoré budú reprezentovať účinnosti použitých metód.

2.2 Funkcie systému

V tomto systéme bude mať používateľ možnosť vybrať si KL Fitter alebo neurónovú sieť na rekonštrukciu top kvarkov. Vstupom bude súbor typu root , ktorý sa prekonvertuje pomocou nástroja na prácu so súbormi typu root - ROOT na vhodný dátový typ, s ktorým vie systém pracovať. V prípade, že si používateľ vyberie neurónovú sieť , systém vytvorí skonvertovaný konfiguračný súbor , ktorý je potrebný pre tréning neurónovej siete. Natrénovanú neurónovú sieť bude systém vyhodnocovať a následne výsledky tréningu uloží do databázy.

2.3 Charakteristika používateľa

Tento systém je určený pre fyzikálnych výskumníkov a študentov, ktorí pracujú s top kvarkami a jetmi. V systéme budú existovať dva druhy používateľov : administrátor, učiteľ a študent , pričom nebudú mať rovnaké práva. Učiteľ bude mať možnosť okrem vyššie uvedených funkcií systému aj pridávať práva na prístup aj iným používateľom. Študent bude môcť využívať len obmedzené funkcie systému, ktoré nemodifikujú systém. Administrátor sa bude líšiť od učiteľa iba možnosťou odoberaním práv.

2.4 Všeobecné obmedzenia

Tento systém potrebuje server, databázu a pripojenie na internet.

2.5 Predpoklady a závislosti

1. úložný priestor na serveri
2. internet
3. server na ktorom bude bežať program

Špecifické požiadavky

3.1 Funkčné požiadavky:

KLFitter: Systém ponúkne implementovanú metódu KLFitter na rekonštrukciu top kvarkov.

Hlboká neurónová sieť: Ďalšou metódou na rekonštrukciu top kvarkov je pomocou hlbokaj neurónovej siete ktorá bude natrénovaná podľa konfiguračných súborov.

Databáza: Výsledky z vyššie uvedených metód algoritmov budú uložené v databáze. Databáza okrem uložených dát z metód bude obsahovať aj informáciu o dĺžke procesorového času ktorý bol potrebný na vykonanie metód.

Porovnávanie: Po natrénovaní a použití neurónovej siete (DNN) aplikujeme metódu KLFitter a porovnáme výsledky z oboch metód, čo bude slúžiť ako forma kontroly správnosti metódy DNN. Výsledky tohto porovnávania sa zaznamenajú do grafu a textového súboru, aby sa dali jednoducho skontrolovať.

Spúšťanie: Systém sa dá spustiť dvoma spôsobmi. Prvou možnosťou je nahranie súboru typu ROOT do webového rozhrania. Druhá možnosť je spustenie cez príkazový riadok v linuxe.

Premenné: Vo web interface sa budú dať okrem nahrania súboru aj ručne zvoliť alebo zmeniť potrebné premenné použité v systéme. Ide o premenné, ktoré by štandardne boli vo vstupnom súbore.

Human readable: Musíme zabezpečiť, aby bol systém human readable, a teda aby sme vedeli výsledky zhodnotiť aj ručne.

Katalóg požiadaviek

Vstup: Vstup budeme musieť upraviť pomocou už implementovaného nástroja na prácu so súbormi typu root. Ide o nástroj ROOT. Tento proces je nevyhnutný pre prácu s knižnicou KERAS.

Trénovanie: Výstup z tréovania DNN treba prekonvertovať aby bol použiteľný v C++ (LWTNN [4])

3.2 Kvalitatívne požiadavky:

Procesy: Systém by mal vedieť spustiť viac procesov (vstupov) naraz.

Stabilita: Systém by nemal padať.

Rýchlosť: Systém by mal pracovať čo najefektívnejšie, avšak pre veľké vstupy sa doba bežania odhadnúť nedá. (tzn. systém môže bežať niekoľko hodín, dní... ale nevieme to vopred odhadnúť). Aj napriek tomu bude potrebné optimalizovať algoritmy KLFittera, aby bežal rýchlejšie.

3.3 Požiadavky rozhrania:

Výber metódy: V rozhraní sa musí byť možnosť výberu, či chceme robiť rekonštrukciu cez DNN alebo KLFlitter

Heslo: Rozhranie bude zaheslovane

Katalóg požiadaviek

Stav: Rozhranie bude zobrazovať, na koľko percent je proces dokončený

Katalóg požiadaviek