

Актуальність роботи: Розробка Доменно-Специфікованої мови програмування для опису часових рядів

Автор: Глушко Богдан Сергійович, ІП-71

Науковий Керівник: ас. Очеретяний Олександр Костянтинович

Опис:

Робота присвячена створенню мови програмування, призначеної вирішенню задач обробки, аналізу та прогнозування часових рядів.

Аналіз останніх наукових робіт, присвячених роботі з часовими рядами, що створювалися в рамках дослідницьких установ по всьому світі, наглядно демонструє використання авторами класичних мов високого рівня у процесі роботи над часовими рядами. Попри повну можливість такого підходу, однак, відсутність доменно-специфікованого інструментарію для взаємодії призводить до збільшення часових витрат на його створення замість проведення власне досліджень, а також вимагає додаткової уваги відносно проектування архітектури мовних структур, що гарантуватимуть коректні результати, реалізацію необхідних структурних моделей а також засобів математичної трансформації. Окрім цього, відсутність єдиного стандарту опису часових рядів та взаємодії з ними призводить до складнощів розуміння робіт при вивченні науковою спільнотою - адже кожен синтаксис є унікальним та створеним на розсуд автора роботи.

Виходячи із усього вищесказаного, основними задачами поточної роботи стали:

- Створення стандартизованої доменно-специфікованої мови для аналізу та дослідження часових рядів
- Реалізація базового набору функціоналу для роботи, а саме:
 - Стандартні типи даних (вектор, часовий ряд, графік, помилка, файл)
 - Утилітарні методи (збереження та читання файлів, зчитування з потоку вводу, відображення на дисплеї тощо)
 - Методи графічної візуалізації (побудова графіків, відображення кількох графіків на одному просторі тощо)
 - Математичні методи взаємодії (арифметичні операції, експоненційне згладжування, прості трансформації часових рядів, декомпозиція)

- Стандартизація синтаксису мову для досліджень
- Розробка системи визначення та відслідковування помилок мови з унікальними кодами та системою деталізації.

У якості системи створення мови було обрано пакет розробки Racket, а також модуль зворотної підтримки для мов Python і Ruby. Перша використовується для взаємодії із побудовою графіків та кількох відображень на одній осі координат, а Ruby - для базових примітивів даних та їх спрощеної реалізації.

Глобально, мова складається із БНФ, Лексичної системи, Парсера та Інтерпретатора.

БНФ описує правила синтаксису мови у формі Бекуса-Наура, дозволяючи сформувані коректні правила написання коду мовою, проводячи паралелі із псевдокодом для розробки.

Лексер призначений для переведення мовного синтаксису на набір токенів, що надалі дозволяють побудувати лексичне дерево для коректної взаємодії із обробкою мовних конструкцій.

Парсер дозволяє власне сформувані синтаксичне дерево, а також розподілити мовні параметри у коректну структуру для подальшої взаємодії. Враховуючи, що парсер є класичним прикладом синтаксичного аналізатору, а запропонована граматика проекту може вважатися безконтактною, синтаксичне дерево можна вважати прямим, тобто таким, що рухається від головної вершини (точки ініціалізації) до коренів (значень повернення).

На основі сформованого дерева, інтерпретатор поетапно обробляє кожен програмний етап, або вузол. В результаті, мова транслюється у C-подібний синтаксичний зразок, що надалі передається в обробку компілятором Racket.

Наведемо приклад створення часового ряду, його збереження, та отримання базових даних у коді мови та трансляції необхідного БНФ-синтаксису.

```

-- Створюємо новий об'єкт часового ряду
time_series series1;
-- Заповнюємо його значеннями з файлу
create_from_file (series1 ; "data.txt")
-- Застосовуємо пошук зміщеного середнього з потенційним викликом
помилки
throwable (moving_average (series1_avg ; series1) ; "Transformation
Went Wrong!")
-- Створюємо об'єкт діаграми
chart_create (series1_chart ; series1_avg)
-- Зберігаємо об'єкт діаграми
chart_save (series1_chart ; "test_chart.png")
-- Відображаємо діаграму на екрані
chart_show (series1_chart)

```

У свою чергу, БНФ для подібного синтаксичного набору матиме наступний вигляд:

```

<time_series_name> ::= letter | word
<create_from_file> ::= "(" <time_series_name> ";"
-toolkit.file-<file_name> ")"
<error_name> ::= letter | word
<reason_name> ::= letter | word
<command_set> ::= & [1 | 1-10]
<throwable> ::= "(" <command_set> ";" <reason_message> ")"
<chart_name> ::= letter | word
<chart_show> ::= "(" <chart_name> ")"
<chart_create> ::= "(" <chart_name> ";" <time_stamp_name> ")"
<chart_save> ::= "(" <chart_name> ";" <file_name> ")"

```

На основі наведеного вище прикладу бачимо, що предмет дипломної роботи надає зручний та комфортний інструмент для взаємодії та дослідження різних природних аспектів тощо методом використання часових рядів та їх подальшої обробки, із швидкими та очевидними синтаксичними методами та компонентами DSL-мови.