МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кручок Сергій Ігорович

УДК 004.75:004.853

ЗАСОБИ ПРОГНОЗУВАННЯ ТА БАЛАНСУВАННЯ НАВАНТАЖЕННЯ У РОЗПОДІЛЕНИХ СИСТЕМАХ, НА ПЛАТФОРМІ NODE.JS

Спеціальність 8.050103

«Інженерія програмного забезпечення»

АВТОРЕФЕРАТ

дисертації на здобуття освітньо-кваліфікаційного рівня
магістр

Київ – 2016

Робота виконана на кафедрі автоматизації проектування енергетичних процесів та систем НТУУ «КПІ» Міністерства освіти і науки України.

Науковий керівник: кандидат технічних наук, доцент
Смаковський Денис Сергійович
доцент кафедри автоматизації проектування
енергетичних процесів і систем НТУУ «КПІ» (м. Київ)

Рецензент: кандидат технічних наук, старший викладач
Баранюк Олександр Володимирович

Захист відбудеться __ червня 2016 р., на засіданні ДЕК кафедри АПЕПС НТУУ „КПІ” аудиторія _____

З дисертацією можна ознайомитись у методичному кабінеті кафедри АПЕПС НТУУ „КПІ”, аудиторія .

Реферат підготовлено та представлено до розгляду „__” _____2016 р.
Робота рекомендована до захисту „__” _____ 2016 р.

Завідуючий кафедрою АПЕПС НТУУ „КПІ”,
доктор технічних наук, професор
Лук’яненко С. О.

Відповідальний за випуск магістрів
кафедри АПЕПС НТУУ «КПІ»,
кандидат технічних наук, доцент Гагарін О. О.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. У зв’язку з тенденцією до енергозбереження актуальною постає задача раціонального використання ресурсів розподілених систем. Часто через потребу забезпечення заданого рівня якості обробки запитів користувачів у години-пік для веб-застосунків, виділяють додаткові ресурси, які у не години-пік залишаються ввімкненими, що негативно впливає на енергоефективність такої розподіленої системи. Використання прогнозування навантаження для розподілених систем дозволяє більш енергоефективно використовувати ресурси, не зменшуючи якість надання послуг користувачам.

Ця задача є актуальною і у разі використання хмарних ресурсів для розміщення розподіленої системи. У цьому випадку використання прогнозування навантаження дозволяє зменшити фінансові витрати на хмарні ресурси, за рахунок їх раціональної оренди, при забезпечені заданого рівня якості надання послуг.

Зв’язок роботи з науковими програмами, планами, темами

Дисертаційна робота магістра виконувалась у НТУУ "КПІ" у відповідності з планом наукових досліджень кафедри АПЕПС.

Метою дослідження є виявлення нових підходів до збільшення енергоефективності розподілених систем та розробка засобу для балансування та прогнозування навантаження.

Для реалізації поставленої мети були сформульовані наступні **завдання дослідження**, що визначили логіку дослідження та його структуру:

- проаналізувати існуючі методи балансування навантаження;
- дослідити можливості використання методів машинного навчання для прогнозування навантаження;
- здійснити програмну реалізацію засобу балансування та прогнозування навантаження, на платформі Node.js.

Об’єктом дослідження є програмне забезпечення прогнозування та балансування навантаження.

Предмет дослідження - програмне забезпечення прогнозування та балансування навантаження у розподілених системах на платформі Node.js.

Методи дослідження: розв'язання поставлених задач виконувалось на базі положень прикладної математики, зокрема:

- методів балансування навантаження для розподілу запитів;
- методів машинного навчання для прогнозування навантаження;
- засобів комп'ютерного моделювання для перевірки результатів.

Наукова новизна одержаних результатів. Найбільш суттєвими науковими результатами магістерської дисертації є:

- удосконалено спосіб балансування навантаження шляхом використання машинного навчання для прогнозування навантаження, що дозволило збільшити енергоефективність розподілених систем;
- набуло подальшого розвитку застосування машинного навчання для прогнозування часових рядів.

Практичне значення одержаних результатів визначається тим, що запропонований метод збільшення енергоефективності та відмовостійкості розподілених систем при заданій якості надання послуг зменшує коштовність оперування розподіленою системою.

Апробація результатів дисертації

Основні положення роботи доповідались і обговорювались на :

1. XIV Міжнародній науково-практичній конференції аспірантів, магістрантів і студентів «Сучасні проблеми наукового забезпечення енергетики» (м. Київ, 18-21 квітня 2016 р).

Публікації. Наукові положення дисертації опубліковані у 1 роботі.

1. Кручок, С. Засоби балансування та пронозування навантаження у розподілених системах, на платформі Node.js / С. І. Кручок, Д. С. Смаковський // Сучасні проблеми наукового забезпечення енергетики. Матеріали XIV Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, присвяченої 85 річчю теплоенергетичного факультету, м. Київ, 18–21 квітня 2016 р. У 2 т. – К. : НТУУ «КПІ», 2016. – Т. 2. – С. 116.

Ключові слова. МАШИННЕ НАВЧАННЯ, ШТУЧНІ НЕЙРОННІ МЕРЕЖІ, ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ, ЧАСОВИЙ РЯД, БАЛАНСУВАННЯ НАВАНТАЖЕННЯ, РОЗПОДІЛЕНІ СИСТЕМИ.

ОСНОВНИЙ ЗМІСТ РОБОТИ

У **вступі** охарактеризовано актуальність дисертаційної роботи, розписано основні питання, що розглядалися в дослідженні та напрям дослідження.

У **першому розділі** оглянута предметна область та постановка задачі. Зокрема, описані задачі та принципи балансування навантаження, а також хмарних обчислень.

Балансування навантаження це розподіл завдань між декількома мережевими ресурсами з метою оптимізації використання ресурсів, скорочення часу обслуговування запитів, горизонтального масштабування кластера, а також забезпечення відмовостійкості.

Статичні алгоритми розподіляють завдання на обчислювальні ресурси, виходячи з завантаження обчислювальних ресурсів у момент надходження завдання, і в якості критерію балансування вибирається рівномірне навантаження ресурсів.

Динамічні алгоритми вирішують задачу розподілення завдань на основі поточної інформації про завантаження всіх доступних обчислювальних ресурсів.

Як результат динамічні алгоритми виконують розподіл завдань коректніше, оскільки для призначення завдання на обчислювальний ресурс окрім статичної інформації про характеристики ресурсів, яка зберігається в інформаційній системі, використовується поточна інформація про реальне завантаження обчислювальних ресурсів, що надходить в інформаційну систему від постачальників у режимі реального часу (надходять усі зміни в значеннях параметрів оцінки стану ресурсів).

Підкреслено, що для покращення енергоефективності та відмовостйкості розподілених систем необхідний балансувальник навантаження з превентивною, а не реакційною, логікою роботи.

Сформульовані завдання дослідження, а саме:

- проаналізувати існуючі методи балансування навантаження;
- дослідити можливості використання методів машинного навчання для прогнозування навантаження;
- здійснити програмну реалізацію засобу балансування та прогнозування навантаження, на платформі Node.js.

У **другому розділі** описано машинне навчання, зокрема модель штучної нейронної мережі та топологія довгої короткочасної пам’яті.

Машинне навчання це розділ штучного інтелекту, має за основу побудову та дослідження систем, які можуть самостійно навчатись з даних. Наприклад, система машинного навчання може бути натренована на електронних повідомленнях для розрізняння спам і не спам-повідомлення. Після навчання вона може бути використана для класифікації нових повідомлень електронної пошти на спам та не-спам папки.

Штучні нейромережі є електронними моделями нейронної структури мозку, який головним чином навчається з досвіду. Природній аналог доводить, що множина проблем, які поки що не підвладні розв'язуванню наявними комп'ютерами, можуть бути ефективно вирішені блоками нейромереж.

Існують три загальні парадигми навчання: "з вчителем", "без вчителя" (самонавчання) і змішана. У першому випадку нейромережа має у своєму розпорядженні правильні відповіді (виходи мережі) на кожен вхідний приклад. Ваги налаштовуються так, щоб мережа виробляла відповіді як можна більш близькі до відомих правильних відповідей. Навчання без вчителя не вимагає знання правильних відповідей на кожен приклад навчальної вибірки. У цьому випадку розкривається внутрішня структура даних та кореляція між зразками в навчальній множині, що дозволяє розподілити зразки по категоріях. При змішаному навчанні частина ваг визначається за допомогою навчання зі вчителем, у той час як інша визначається за допомогою самонавчання.

Довга короткочасна пам'ять (ДКЧП) це архітектура рекурентних нейронних мереж, вперше опублікована у 1997 році Зеппом Хохрайтером та Юргеном Шмідгубером. Як і більшість рекурентних нейронних мереж, мережа ДКЧП є

універсальною в тому сенсі, що за достатньої кількості вузлів мережі вона може обчислювати будь-що, що може обчислювати звичайний комп'ютер, за умови, що вона має належну матрицю вагових коефіцієнтів, що може розглядатися як її програма.

На відміну від традиційних рекурентних нейронних мереж, мережа ДКЧП добре підходить для навчання з досвіду з метою класифікації, обробки або передбачення часових рядів в умовах, коли існують дуже великі часові затримки невідомої тривалості між важливими подіями. Це є однією з головних причин чому ДКЧП в численних застосуваннях перевершує альтернативні рекурентні нейронні мережі, приховані марковські моделі та інші методи навчання послідовностей.

Застосування ДКЧП включають:

- прогнозування часових рядів;
- розпізнавання мовлення;
- навчання ритму;
- написання музики;
- навчання граматики;
- розпізнавання рукописного тексту;
- розпізнавання людських дій;
- керування роботами.

У **третьому розділі** проведено аналіз існуючих підходів допрограмої оптимізації енергоефективності та запропоновано розроблений метод прогнозування навантаження.

Цікавим прикладом створення програмної інфраструктури більш енергоефективною є Facebook’s Autoscale. Вона була розгорнута у виробничих кластерах і вже продемонструвала значну економію енергії. Основна ідея Autoscale полягає у тому, що замість чисто round-robin підходу, балансування навантаження буде концентрувати навантаження на сервер, поки він не має принаймні робоче навантаження середнього рівня. Якщо загальне навантаження низьке, наприклад близько опівночі, балансувальник навантаження буде використовувати тільки підмножину серверів. Інші сервери можуть бути залишені працюючими у холостому

режимі або використовуватися для інших робочих навантажень.

Маючи інформацію про попередню історію навантаження на розподілену систему, яка включає в себе кількість користувачів та їх джерело походження за часовими проміжками, можна використати машинне навчання, а саме математичну модель штучної нейронної мережі, для виявлення залежностей по навантаженню на розподілену систему та використанню навченої нейронної мережі для прогнозування кількості користувачів через заданий час.

Це дозволить більш енергоефективно використовувати власні ресурси, та за необхідності використовувати раціонально хмарні ресурси, не зменшуючи якість надання послуг користувачам. Наприклад вірним шляхом у разі прогнозованого збільшення навантаження на більше, ніж вистачає власних ресурсів, ініціалізувати запуск хмарних ресурсів які зможуть обробити додаткову кількість запитів від користувачів, що сприятиме відмовостійкості.

Запропонована модель штучної нейронної мережі має наступні типи входів:

- на який час вперед спрогнозувати навантаження;
- час доби;
- день тижня;
- кількість всіх користувачів в даний момент та в декількох попередніх часових проміжках;
- відсоток користувачів із HTTP заголовком referer в даний момент та в декількох попередніх часових проміжках.

Всі значення нормовані від 0 до 1.

Важливим є поділ помилки прогнозування на два типи помилок:

- помилки першого роду (альфа помилки, помилкова спрацьовування), коли була спрогнозована більша кількість користувачів, ніж виявилась насправді;
- помилку другого роду (бета помилки, пропуск події), коли була спрогнозована менша кількість користувачів, ніж виявилась насправді.

Ці два типи помилок мають різні наслідки для розподіленої системи – помилка першого роду призводить до зменшення енергоефективності, в той час як помилка другого роду призводить до перевантаження розподіленої системи і як

наслідок зменшення якості обслуговування (обробки запитів) користувачів нижче за заданий рівень. В даному випадку помилка другого роду є більш критичною.

У **четвертому розділі** описано програмну реалізацію засобу прогнозування та балансування навантаження і результати моделювання. По програмній реалізації була розглянута платформа Node.js, засоби реалізації програмної системи, архітектура та інструкція з використання.

Програмний продукт написаний на мові програмування JavaScript на платформі Node.js. Програмна система не проектувалась під конкретне сімейство операційних систем, завдяки крос-платформеності платформи Node.js.

Node.js це крос-платформене середовище виконання, з відкритим джерельним кодом, для розробки серверних веб-застосунків. Він має подійно-орієнтовану архітектуру здатну до асинхронного введення та виводу. Цей конструкторський вибір спрямований на оптимізацію пропускної здатності і масштабованості в веб-застосунках з великою кількістю операцій введення та виводу, а також для веб-застосунків реального часу, наприклад в режимі реального часу комунікаційних програм і браузерних ігор.

Архітектура програмної системи спроектована як middleware для балансування навантаження.

Складовими загальної архітектури (рисунок 1) є:

- пул серверів;
- клієнт;
- балансувальник навантаження;
- прогнозувальник навантаження;
- скрипти для запуску та зупинки серверів.

Складовою middleware є балансувальник навантаження та прогнозувальник навантаження. Клієнтське програмне забезпечення, серверне програмне забезпечення та скрипти для запуску та зупинки серверів пишуться розробниками прикладного забезпечення самостійно.

Варто особливо відзначити, що така архітектура програмного засобу має не проксуючий балансувальник навантаження. Цей підхід дозволяє уникнути

високонавантаженої точки відмови та сприяє ефективному використаню серверної мережевої інфраструктури.

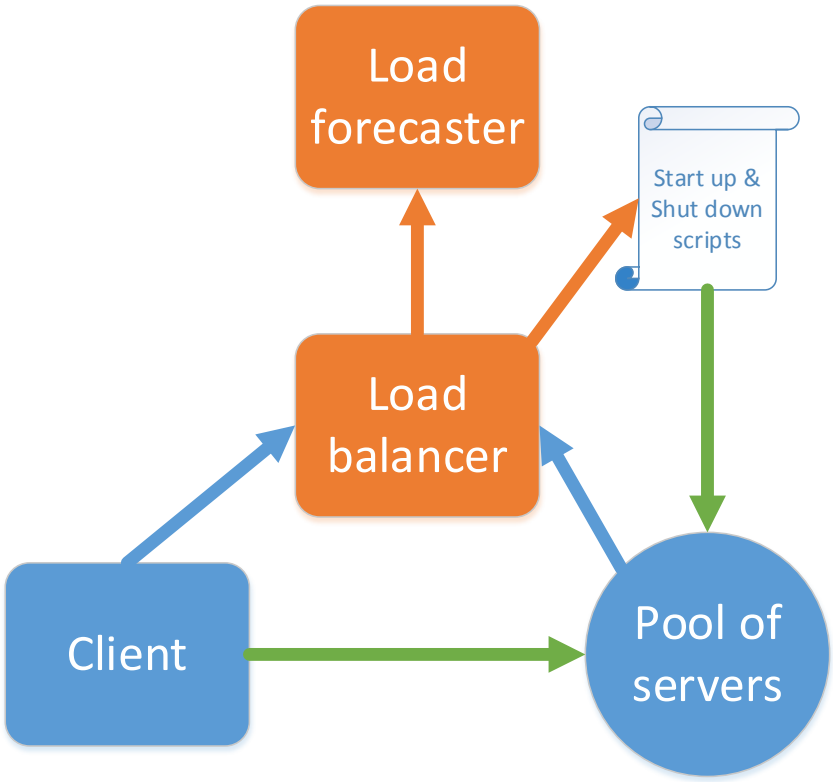
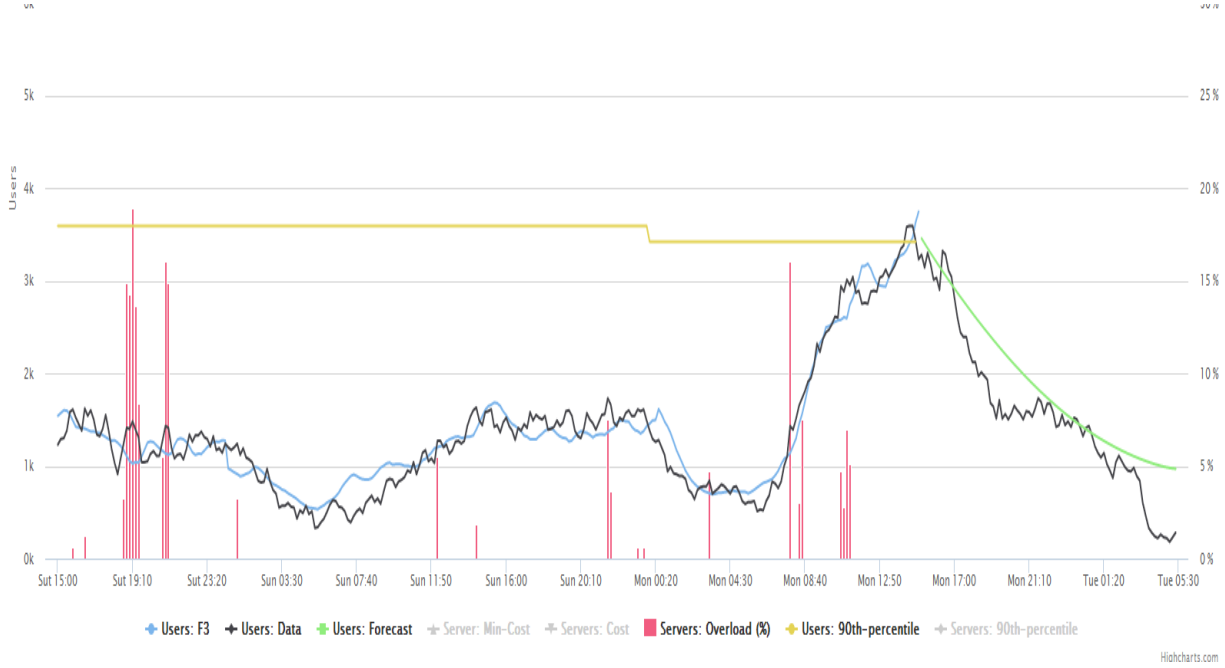


Рисунок 1 – Загальна архітектура

Оскільки немає потреби проксувати трафік від балансувальника навантадення до пулу серверів, така архітектура дозволяє мати географічно розділений балансувальинк навантаження та пули серверів. У разі, якби був використаний проксуючий балансувальник навантаження, був би негативний вплив на рівень якості обробки запитів користувачів через затримку від передачі запиту від балансувальника навантаження до одного із серверів із пулу серверів, та передачі відповіді оберненим шляхом.

Саме це дозволяє піти шляхом, коли досягається енергоефективне використання власних ресурсів та у разі прогнозованого збільшення навантаження на більше, ніж вистачає власних ресурсів, ініціалізується запуск хмарних ресурсів, які можуть бути географічно віддалені, та які зможуть обробити додаткову кількість запитів від користувачів, що сприятиме відмовостійкості.

Наведений рисунок демонструє результат моделювання



У процесі експлуатації розробленого засобу отримується нова інформація про навантаження на розподілену систему, згідно з якою штучна нейрона мережа довчається для врахування можливих змін у поведінці користувачів, та відповідно залежностей по навантаженню на розподілену систему.

ВИСНОВКИ

В результаті роботи над магістерською дисертацією було проведено аналіз огляду предметної області, який показав, що існуючі методи балансування навантаження переважно не спрямовані на енергоефективність та діють реакційно, лише реагуючи на вже існуючу ситуацію, що не дозволяє у повній мірі використовувати розподілену систему енергоефективно при забезпечені відмовостійкості. В результаті досліджень було виявлено, що потрібний засіб із превентивною логікою роботи, що зможе діяти на основі спрогнозованого навантаження, із архітектурою не проксуючого (тобто, без пропуску трафіку через один пристрій) балансувальника навантажувача, тож необхідно розробляти, вдосконалювати та застосовувати програмне забезпечення прогнозування та балансування навантаження.

На основі аналізу предметної області, та визначення мети роботи, були сформульовані завдання досліджень.

Для визначення можливостей вирішення задачі оптимізації енерговитрат та покращення відмовостійкості, проведено аналіз існуючих методів. Недоліком є залишення серверних ресурсів увімкнутими навіть якщо вони не використовуються, та не здатність завчасно підготувати серверні ресурси для їх використання через реакційний підхід замість превентивного підходу.

Для усунення існуючих недоліків було розроблено новий підхід, що склав наукову новизну роботи:

- удосконалено спосіб балансування навантаження шляхом використання машинного навчання для прогнозування навантаження, що дозволило покращити енергоефективність розподілених систем;
- набуло подальшого розвитку застосування машинного навчання для прогнозування часових рядів.

На основі запропонованих методів та моделей був розроблений засіб для балансування навантаження у розподілених системах, який за рахунок використання прогнозування краще за існуючи аналоги.

Користувачами програмної системи можуть бути розробники веб-сайтів/веб-застосунків із розподіленою системою обробки великої кількості запитів, де вимога до часу на виконання кожного запиту задана у порядку сотень чи десятків мілісекунд, та швидше. Програмне забезпечення має бути впроваджено як middleware для балансування навантаження.

Вірним шляхом є енергоефективне використання власних ресурсів та у разі прогнозованого збільшення навантаження на більше, ніж вистачає власних ресурсів, ініціалізація запуску хмарних ресурсів які можуть бути географічно віддалені, та які зможуть обробити додаткову кількість запитів від користувачів, що сприятиме відмовостійкості.

Дослідження, які були проведені під час виконання магістерської дисертації, показали, що на певних етапах розробки моделі виникали деякі ідеї, на які у майбутньому слід було б зупинитись. Наприклад, цікавим є питання про введення

додаткового входу у моделі нейронної мережі з ціллю задання бажаного співвідношення між помилками першого та другого роду, що може, наприклад, збільшити відмовостійкість за рахунок зменшення енергоефективності, або навпаки. Зважаючи на вищезазначене, напрямок робіт в цій області вбачається перспективним.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

1. Кручок, С. Засоби балансування та пронозування навантаження у розподілених системах, на платформі Node.js / С. І. Кручок, Д. С. Смаковський // Сучасні проблеми наукового забезпечення енергетики. Матеріали XIV Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, присвяченої 85 річчю теплоенергетичного факультету, м. Київ, 18–21 квітня 2016 р. У 2 т. – К. : НТУУ «КПІ», 2016. – Т. 2. – С. 116.