Хмельницький національний університет
Факультет програмування та комп'ютерних і телекомунікаційних систем
Кафедра комп'ютерних наук та інформаційних технологій

ДИПЛОМНА РОБОТА

МАГІСТРА

на тему *Система прогнозування продажів сервіснис послуг в системах обслуговування*

Галузь знань *12 – Інформаційні технології*
Шифр і назва галузі знань

Спеціальність *122 – Комп'ютерні науки*
Шифр і назва спеціальності

Виконав: студент *2* курсу, група *КНм-19-1* *М.С. Кузьмінський*
Підпис Ініціали, прізвище

Керівник: *к.т.н., доцент кафедри КНІТ* *Е.А. Манзюк*
Підпис Ініціали, прізвище

Нормоконтроль: *к.т.н., доцент кафедри КНІТ* *Р.О. Багрій*
Підпис Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КНІТ, д.т.н., професор *.О.В.Бармак*
Підпис Ініціали, прізвище

7 *12* 2020 р.

Хмельницький 2020

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет *програмування та комп'ютерних і телекомунікаційних систем*
Кафедра *комп'ютерних наук та інформаційних технологій*
Освітній ступінь *магістр*
Галузь знань *12 – Інформаційні технології*
Спеціальність *122 – Комп'ютерні науки*

ЗАТВЕРДЖУЮ
Завідувач кафедри *комп'ютерних наук та інформаційних технологій*

_____
(підпис)
д.т.н., професор О.В. Бармак
«*7*» *9* *2020* року

ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ МАГІСТРА

1. Тема дипломної роботи магістра: *«Система прогнозування продажів сервіснис послуг в системах обслуговування»*

2. Завдання видано студенту *Кузьмінський Михайло Сергійович*
(прізвище, ім'я, по батькові)

3. Керівник роботи *к.т.н., доцент Манзюк Едуард Андрійович*
(прізвище, ім'я, по батькові)

4. Затверджені наказом університету від «*9*» *9* *2020* р. № *22*

5. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи – розробка системи прогнозування продажів сервіснис послуг в системах обслуговування. Об'єктом дослідження є методи аналізу та прогнозування отримання інформації та її обробки.

Предметом дослідження є групованні данні за показниками ефективності прикладної області дослідження.

Реферат

Дипломна робота магістра присвячена розробці системі прогнозування продажів сервісних послуг в системах обслуговування.

Актуальність теми. В магістерській роботі було розроблено інформаційну систему аналізу систему прогнозування завантаженості та продажів сервісних послуг. Ця система дозволяє враховувати сучасні тенденції на ринку та робити прогнози використовуючи методи та підходи машинного навчання.

Метою дослідження є розробка інформаційної системи аналізу та дослідження впливу основних факторів на функціонування галузі сервісних послуг.

Для досягнення зазначеної мети поставлені наступні **задачі**:

- показати, що використання методів машинного навчання дозволяє поліпшити роботу системи сервісних послуг;
- провести дослідження впливу ознак на необхідні параметри інформаційної системи;
- провести аналіз відомих методів та підходів в предметній області дослідження

При цьому передбачається розв'язок таких **підзадач**, як

- обробка та очищення даних;
- відбір ознак та проведення аналізу їх якісних показників;
- розробка методів прогнозування;
- застосування моделей машинного навчання;
- тестування ефективності методів застосування машинних моделей на практичних даних;
- програмна реалізація системи прогнозування функціонування галузі.

Об'єктом дослідження є методи аналізу та прогнозування отримання інформації та її обробки.

Предметом дослідження є груповані данні за показниками ефективності прикладної області дослідження.

Оцінка запропонованих архітектур машинного навчання є основним внеском цього дослідницького проекту. Набір даних, який використовується в цій роботі, розглядався за кількома наборами, таким чином використовуючи відгуки про регіони і враховуючи той факт, що люди можуть мати різні стандарти, які можуть впливати на рейтинг.

Під час пошуку всіх експериментальних таблиць (точність, час навчання/генерації) під час використання набору даних модель Doc2VecC перевершує в порівнянні з деякими іншими моделями, такими як Bag-of-words або Word2VecC, оскільки вона має найнижчий показник похибки та найкращу точність (на підмножині тестового набору Semantic-Syntactic World Relationship). Єдине питання щодо цього методу полягає в тому, що час навчання трохи вище, ніж очікувалося. Важливою частиною Doc2VecC є узагальнення даних, яке враховує словам, які не часто зустрічаються і полегшує представлення документів із середнім показником використання вивченого слова. В експериментальній частині можна помітити, що цей метод дає хороші результати.

Однією з найбільш важливих частин цієї роботи є хороший набір даних, з усією наявною інформацією. Якщо моделі отримали хороший результат, з більшою частиною негативних відгуків, наприклад, 80% відгуків у цьому наборі даних мають оцінку значимості між 80-100, що зробило його складним для досягнення кращих результатів.

Результати показують, що методи машинного навчання можуть бути використані для класифікації ресторанів з точки зору санітарних проблем, а результати з використанням тексту досить надійні. Однак, це складно працює для боротьби з підробленими відгуками. Однак робота не говорить про те, щоб перевірити коментарі клієнтів, а довести, що методи класифікації можуть допомогти оцінювачам отримати хороші результати.

Достовірність результатів забезпечується проведенням всебічного оцінювання та порівняння ефективності різних методів.

Найважливішим внеском роботи є експерименти з даними в соціальних мережах, а також методи машинного навчання, які використовуються для обробки цих даних. Відгуки дуже корисні для цього типу досліджень, що стосуються громадського здоров'я ресторанів. Користувачі додатків знаходяться в певних місцях концентрації і їхні дані кластерізуються відповідно до розташування. Додатковою перевагою набору даних є той факт, що всі відгуки англійською мовою, яка є найбільш поширеною. Механізм концентрації, дозволяє застосовувати модель для атрибута ваги трьох рівнів (тобто, на рівні огляду, рівень речення і рівень слова). Найбільшою перевагою є можливість моделі приділяти увагу до окремих слів або наборів слів, і в той же час орієнтація на можливість інтерпретувати результати візуалізації ваги, застосовної до відповідного рівня. Результати також дають уявлення про те, як моделі можуть бути поліпшені, коли частини збалансовані, тоді результат кращий. Саме це аспект зменшує розходження між продуктивністю моделі точністю.

Модель, є хорошим підходом до використання з усіма типами проблем з визначенням тексту, а не тільки з онлайн-відгуками. Можливість адаптувати модель до інших повсякденних ситуацій, проста і корисна в інших споріднених сферах.

Практична значимість дослідження полягає в тому, що проведені дослідження можуть бути застосовні в сфера оцінювання впливу сукупності параметрів на ефективність функціонування предметної області.

Кластерна карта сервісних закладів допомагає візуалізувати локації та фільтрувати ресторани на основі типів кухні. Точка визначає оцінки і включає в себе описи вибраних ресторанів. На тепловій карті видно щільність ресторанів на основі вибору кухні. Це показує, що площа має більшу кількість ресторанів і може бути корисним для ділових людей, щоб приймати обґрунтовані рішення про те, де відкрити нові ресторани на основі типів ресторанів, які вже є.

Нарешті, ця програма може бути корисною для людей, щоб фільтрувати базу даних на кухнею, рейтингами та класом інспекції. Люди які хочуть піти поїсти з конкретними критеріями можуть фільтрувати ресторани і відвідувати свої улюблені ресторани на основі найвищих оцінок як для відгуків, так і для класів інспекції.

Апробація дипломної роботи.

Основні положення і результати роботи опубліковані в збірнику наукових праць – Кузьмінський М. С. Система прогнозування продажів сервіснихпослуг в системах обслуговування / М. С. Кузьмінський, Е. А. Манзюк/ / Збірник наукових праць за матеріалами Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук - 2020» Хмельницький, 2020, – С.157-158.

Структура та обсяг роботи. Дипломна робота магістра складається з завдання, реферату, змісту, вступу, 4 розділів, висновків, переліку посилань із 15 найменувань та додатків. Загальний обсяг дипломної роботи магістра становить 71 сторінок, з них 70 сторінок основного тексту та 1 сторінки додатків, в роботі наведено 25 рисунків.

Ключові слова: нейронна мережа, класифікація, інформаційна система.

Зміст

Вступ

Систематичний збір даних інспекції охорони здоров'я має важливе значення для оцінки ресторану, а також є основою для запобігання ряду захворювань на основі отруєння. Для цих та інших цілей інспектори повинні писати санітарні звіти, що містять місцезнаходження та інші дані ресторану, разом з текстовими описами з причин, які в є основі оцінки ресторану, і оцінки інспекції здоров'я.

Аналіз ресторанів також передбачає рекомендації. Yelp - це платформа, яка дозволяє людям класифікувати та давати свою думку про місця проведення (наприклад, такі як бари та ресторани), які вони відвідали. Основна мета Yelp полягає в тому, щоб інформувати нових клієнтів, якщо місце, яке вони ніколи не відвідували, має хорошу рекомендацію чи ні, на основі інформації, наданої іншими клієнтами. Зокрема, автоматична класифікація даних Yelp, пов'язаних з санітарними звітами, дозволить людям краще знати, чи має місце хорошу оцінку чи ні, і вирішити, чи заслуговує це місце свого часу і грошей.

За даними центру з контролю і профілактики захворювань, близько одного з шести клієнтів (48 мільйонів людей) захворіли, 128 000 госпіталізовані, а 3000 помирають від харчових захворювань.

Yelp використовується широким колом людей в сучасному світі для обміну своїм досвідом, а також пошуку будь-якого бізнесу (ресторанів, перукарні і т.д.) деталей. Санітарний стан ресторану є важливим фактором для клієнтів, урядів і самих ресторанів. Цей проект має на меті передбачити рекомендацію закладу на основі наявних функцій.

Прогнозування санітарного стану ресторану для кожного ресторану (на основі наявних даних про функцію) може допомогти уряду вирішити ризик ресторану. Це забезпечить громадське здоров'я та безпеку. Yelp може співпрацювати з ресторанами, щоб надати рекомендації щодо поліпшення своїх

ресторанів і потреб користувачів. Це може забезпечити взаємну вигоду як для yelp, так і для ресторанів.

Актуальність теми. В магістерській роботі було розроблено інформаційну систему аналізу систему прогнозування завантаженості та продажів сервісних послуг. Ця система дозволяє враховувати сучасні тенденції на ринку та робити прогнози використовуючи методи та підходи машинного навчання.

Метою дослідження є розробка інфрмаційної системи аналізу та дослідження впливу основних факторів на функціонування галузі сервісних послуг.

Для досягнення зазначеної мети поставлені наступні **задачі**:

– показати, що використання методів машинного навчання дозволяє поліпшити роботу системи сервісних послуг;

– провести дослідження впливу ознак на необхідні параметри інформаційної системи;

– провести аналіз відомих методів та підходів в предметній області дослідження

При цьому передбачається розв'язок таких **підзадач**, як

– обробка та очищення даних;

– відбір ознак та проведення аналізу їх якісних показників;

– розробка методів прогнозування;

– застосування моделей машинного навчання;

– тестування ефективності методів застосування машинних моделей на практичних даних;

– програмна реалізація системи прогнозування функціонування галузі.

Об'єктом дослідження є методи аналізу та прогнозування отримання інформації та її обробки.

Предметом дослідження є групованні данні за показниками ефективності прикладної області дослідження.

Оцінка запропонованих архітектур машинного навчання є основним внеском цього дослідницького проекту. Набір даних, який використовується в цій роботі, розглядався за кількома наборами, таким чином використовуючи відгуки про регіони і враховуючи той факт, що люди можуть мати різні стандарти, які можуть впливати на рейтинг.

Під час пошуку всіх експериментальних таблиць (точність, час навчання/генерації) під час використання набору даних модель Doc2VecC перевершує в порівнянні з деякими іншими моделями, такими як Bag-of-words або Word2VecC, оскільки вона має найнижчий показник похибки та найкращу точність (на підмножині тестового набору Semantic-Syntactic World Relationship). Єдине питання щодо цього методу полягає в тому, що час навчання трохи вище, ніж очікувалося. Важливою частиною Doc2VecC є узагальнення даних, яке враховує словам, які не часто зустрічаються і полегшує представлення документів із середнім показником використання вивченого слова. В експериментальній частині можна помітити, що цей метод дає хороші рещультати.

Однією з найбільш важливих частин цієї роботи є хороший набір даних, з усією наявною інформацією. Якщо моделі отримали хороший результат, з більшою частиною негативних відгуків, наприклад, 80% відгуків у цьому наборі даних мають оцінку значимості між 80-100, що зробило його складним для досягнення кращих результатів.

Результати показують, що методи машинного навчання можуть бути використані для класифікації ресторанів з точки зору санітарних проблем, а результати з використанням тексту досить надійні. Однак, це складно працює для боротьби з підробленими відгуками. Однак робота не говорить про те, щоб певірити коментарі клієнтів, а довести, що методи класифікації можуть допомогти оцінювачам отримати хороші результати.

Достовірність результатів забезпечується проведенням всебічного оцінювання та порівняння ефективності різних методів.

Найважливішим внеском роботи є експерименти з даними в соціальних мережах, а також методи машинного навчання, які використовуються для обробки цих даних. Відгуки дуже корисні для цього типу досліджень, що стосуються громадського здоров'я ресторанів. Користувачі додатків знаходяться в певних місцях концентрації і їхні дані кластерізуються відповідно до розташування. Додатковою перевагою набору даних є той факт, що всі відгуки англійською мовою, яка є найбільш поширеною. Механізм концентрації, дозволяє застосовувати модель для атрибута ваги трьох рівнів (тобто, на рівні огляду, рівень речення і рівень слова). Найбільшою перевагою є можливість моделі приділяти увагу до окремих слів або набірів слів, і в той же час орієнтація на можливість інтерпретувати результати візуалізації ваги, застосовної до відповідного рівня. Результати також дають уявлення про те, як моделі можуть бути поліпшені, коли частини збалансовані, тоді результат кращий. Саме це аспект зменшує розходження між продуктивністю моделі точністю.

Модель, є хорошим підходом до використання з усіма типами проблем з визначенням тексту, а не тільки з онлайн-відгуками. Можливість адаптувати модель до інших повсякденних ситуацій, проста і корисна в інших споріднених сферах.

Практична значимість дослідження полягає в тому, що проведені дослідження можуть бути застосовні в сфера оцінювання впливу сукупності параметрів на ефективність функціонування предметної області.

Кластерна карта сервісних закладів допомагає візуалізувати локації та фільтрувати ресторани на основі типів кухні. Точка визначає оцінки і включає в себе описи вибраних ресторанів. На тепловій карті видно щільність ресторанів на основі вибору кухні. Це показує, що площа має більшу кількість ресторанів і може бути корисним для ділових людей, щоб приймати обґрунтовані рішення про те, де відкрити нові ресторани на основі типів ресторанів, які вже є.

Нарешті, ця програма може бути корисною для людей, щоб фільтрувати базу даних на кухнею, рейтингами та класом інспекції. Люди які хочуть піти

поїсти з конкретними критеріями можуть фільтрувати ресторани і відвідувати свої улюблені ресторани на основі найвищих оцінок як для відгуків, так і для класів інспекції.

Розділ 1
Аналіз та концепція прогнозування завдань предметної області

1.1 Опис предметної області

Нова цифрова модель епідеміології здоров'я, яка використовує підхід на основі даних до харчових захворювань, показує обнадійливі результати.

Система повідомлення про харчове отруєння напрочуд низькотехнологічна; для більшості департаментів охорони здоров'я, якщо хочете повідомити про проблему в ресторані, можете по електронній пошті або зателефонувати в доповіді. Наприклад, у Нью-Йорку є форма для подання скарг.

Дослідницька група в Google і Гарвардській школі громадського здоров'я протестувала новий спосіб виявлення харчових захворювань швидше і точно за допомогою комбінації пошукових запитів і даних про місцезнаходження [2]. «Виявлення харчової хвороби в масштабі в реальному часі» була опублікована в Digital Medicine в листопаді 2018 року. У статті пояснюється, як побудувати модель на основі даних для виявлення ресторанів, які, ймовірно, мають порушення санітарного кодексу.

Команда побудувала модель машинного навчання під назвою Finder, яка призначена для прогнозування харчової хвороби в режимі реального часу. Команда з Google і Гарварду використовувала анонімний агрегований веб-пошук і дані про місцезнаходження, щоб з'ясувати, які ресторани мають порушення безпеки відносно харчових продуктів, які можуть отруїти. Цей далекоглядний метод має потенціал для заміни загального підходу, який використовує після звіти від окремих споживачів і двічі на рік перевірки ресторанів департаментами охорони здоров'я [4].

Виявлення небезпечних ресторанів раніше: По-перше, Finder шукає пошукові запити, які свідчать про те, що людина має харчове отруєння. Модель використовує машинне навчання, щоб визначити всі способи, якими симптоми харчового отруєння описуються користувачами пошуку Google. Наступним

кроком буде пошук ресторанів, які відвідують ці особи, використовуючи знеособлену історію місцезнаходжень [4]. Ці дані з журналів пошуку та розташування надходять від користувачів, які вирішили поділитися своїми даними про місцезнаходження.

Щоб відфільтрувати шум, притаманний пошуковим запитам, команда дослідження описує "збереження конфіденційності під наглядом машинного класифікатора", який вони використовували. Цей метод враховує результати запиту, в результаті якого користувач здійснив вибір, і зміст сторінок, які користувач переглядав за результатами пошуку.

Щоб оцінити потужність Finder, дослідницька група протестувала систему близько чотирьох місяців. Інспекторам було надано список ресторанів для відвідування, які включали деякі ідентифікатори Finder. Потім інспектори оглянули ресторани, щоб виявити порушення санітарного кодексу. Під час дослідження департаменти охорони здоров'я продовжували свої звичайні графіки перевірок.

Дослідження включало чотири набори даних:

- всі інспекції ресторану не спонукали Finder (базовий).
- регулярні планові перевірки (рутинні).
- перевірки, викликані скаргами (скаргою).
- перевірки, рекомендовані Finder (finder).

Було 5880 перевірок під час дослідження, з 71 викликаним аналізом Finder. Тест для аналізу машинного навчання був, длятого чи краще працює програма, ніж стандартні протоколи департаменту охорони здоров'я при виявлення небезпечних ресторанів [4].

Близько половини ресторанів, які позначили Finder, були небезпечними при огляді. У базовій групі перевірок 25% ресторанів були небезпечними. Finder зробив кращий аналіз з визначення ресторанів в категорії низького ризику, ніж у категорії високого ризику.

Команда дослідників також порівняла результати перевірок, рекомендованих Finder, з перевірками, викликаними скаргами клієнтів. Оскільки багато клієнтів ресторанів є туристами, кількість скарг низька в цьому місті; з цієї причини, ця частина аналізу включала скарги тільки локально. Ресторани, ідентифіковані Finder, швидше за все, отримали небезпечне позначення. Дослідники прийшли до висновку, що Finder був більш надійним, ніж окремі скарги, тому що підхід машинного навчання об'єднує інформацію від численних людей, які їли в одному ресторані.

Finder також уникає упередженості відгуку, яка може вплинути на системи звітності на основі скарг. Відгук упередженості відбувається тоді, коли людина не пам'ятає попередніх подій або переживань точно або провокує деталі у зв'язку з плином часу. Наприклад, упереджений відгук є ризиком, коли людина отримує харчове отруєння один тиждень і робить скаргу в департаменті охорони здоров'я на наступному тижні. Досвід, який людина мала з моменту відвідування конкретного ресторану і плин часу, можливо, вплинув на пам'ять людини про ресторан, про який йде мова.

Ця модель машинного навчання є прикладом епідеміології цифрової перевірки санітарного стану. Марсель Салате, професор Швейцарського федерального технологічного інституту, описав різницю між цим новим підходом і традиційними методами виявлення причин захворювань у популяцій. Замість того, щоб інспектор охорони здоров'я ходить від дверей до дверей, щоб запитати людей про джерела їжі або води, ця цифрова версія здоров'я використовує дані, отримані за межами системи громадського здоров'я. На прикладі Finder джерело даних – це пошукові запити, а не особисті опитування [6].

Департаментам охорони здоров'я було важко зробити перехід до підходу, керованого даними, до безпеки харчових продуктів, принаймні, виходячи з досвіду застосування. У 2014 році люди в Чиказькому департаменті інновацій і технологій побудували алгоритм, схожий на Finder. Він використовував

загальнодоступні дані, щоб передбачити, які ресторани, швидше за все, порушують санітарні норми, виходячи з інформації від раніше зафіксованих порушень. Такий підхід також використовував технології прогнозування захворювань у соціальних мережах для цільових перевірок. Це спрацювало: алгоритм виявив порушення приблизно за 7,5 днів до звичайної інспекції.

Метою проекту було зробити його легким для інших департаментів охорони здоров'я . Спочатку тільки ще один департамент охорони здоров'я протестував нову систему. Початкова перешкода - зміна стандартного підходу до перевірок ресторанів, занадто висока для широкого прийняття.

Автори дослідження з Google і Гарварду заявили, що департаменти охорони здоров'я не мають достатньої кількості інспекторів, щоб зробити більш широкий тест рекомендацій Finder.

Модель Finder все ще знаходиться на стадії дослідження і не доступна публічно для департаментів охорони здоров'я на даний момент. Автори дослідження стверджують, що дані інших пошукових систем, які включають історію місцезнаходжень, можуть створювати подібні алгоритми і, можливо, генерувати порівнянні результати [4].

Томер Шекель, старший менеджер з продуктів в Google, сказав, що команда працює з Гарвардською школою громадського здоров'я та іншими агентствами, щоб продовжити дослідження в цій галузі. Шекель сказав, що дослідницька група шукає інші проблеми громадського здоров'я, які можуть бути вирішені з цифровим підходом до епідеміології [4].

1.2 Мотивація та завдання

Ця робота являє собою розробку методу автоматичної класифікації відгуків, наданих клієнтами на Yelp, для того, щоб віднести оцінку інспекції охорони здоров'я до кожного закладу.

Основними цілями проекту є виявлення гігієнічних проблем і порушень санітарних норм в ресторанах з використанням інформаційних технологій, таких як методи машинного навчання, для текстової класифікації. Проект досліджував механізми вишукування тексту на основі глибоких нейронних мереж, які мають передові надійні моделі, здатні отримати хороші результати з точки зору точності аналізу класів і одночасно механізмів, які дозволяють інтерпретувати отримані результати. З впровадженням методів машинного навчання в області обробки природної мови, представлення текстів, які будуть оброблятися, більше не базується на розріджених векторах, що кодуються в двійкову форму наявності слів та/або n-грам слів. Замість цього, представлення здійснюється через щільні вектори (наприклад, вбудовування слів), які фіксують семантичне значення слів. Ця робота оцінювала застосування сучасних методів глибокого навчання, на специфікації с текст проблеми прогнозування оцінки перевірки здоров'я.

1.3 Методологія досліджень

Для навчання та оцінки моделей обробки тексту використано набір даних, зібраний з Yelp що було випущено в виклик для дослідників, запропонованих Yelp. Порівняємо дані, засновані на рецидивних нейронних мережах (RNN) або згортках нейронних мереж (CNN).

Головна мета цієї роботи полягала в тому, щоб зрозуміти, як ці методи автоматичної класифікації будуть виконуватися над цими даними.

На першому етапі роботи особливу увагу було приділено набору даних Yelp. Довелося зробити перевірку на відгуки, так як деякі відгуки не були дані в очікуваному форматі, в той же час співвідносились до кожного огляду установи з відповідною оцінкою інспекції охорони здоров'я. Наступний етап полягав у вивченні фундаментальних понять та пов'язаних робіт щодо подібних проблем з порівняння тексту. Є багато цікавих попередніх досліджень, які дають високу якість результатів з точки зору точності автоматичної точності класифікації. Ідеї

з кількох попередніх публікацій, вирішення інших типів проблем з інформативною діяльністю і які описували інноваційні механізми, засновані на глибоких нейронних мережах, були прийняті до уваги і згодом включені в архітектуру мережі, яка була запропонована. Експерименти брали участь у двох нейронних мережевих архітектурах (наприклад, Convolutional Neural Networks (CNN) і Recurrent Neural Networks (RNN)), з метою оцінки відносних переваг архітектур моделі. Порівняємо обидві ці моделі і проаналізуємо, які з них можуть досягти кращих результатів, з двома показниками регресії під назвою Root Mean Square Error (RMSE) і Mean Absolute Error (MAE).

У цій моделі дані є векторним представленням слова (наприклад, вбудовування слів), які надходять у звичайні шари, проходять через об’єднування шарів (наприклад, максимальне об'єднання) і надалі досягає повністю з'єднаного шару, щоб зробити передбачення. Для того, щоб оцінити результати, використано регресійні показники для порівняння досягнутого прогнозу з бажаним балом. Середня квадратична помилка вимірює середню величину похибки і визначає правило підрахунку оцінки. Середня помилка дасть більшу вагу великим помилкам, які пізніше будуть повторно вилучатись для більш кращого результату. Середня помилка, також вимірює середню величину похибки в наборі прогнозів, не розглядаючи їх. Це середнє значення над тестовою вибіркою абсолютних d_i між прогнозом і фактичним спостереженням, де всі окремі відмінності мають однакову вагу.Основний набір, містить 58306 відгуків. Також відомо, що кожен огляд містить в середньому приблизно 106,5 слів і в середньому оцінку інспекції охорони здоров'я 80,30.

1.4 Постановка задачі

Метою дослідження є розробка інфрмаційної системи аналізу та дослідження впливу основних факторів на функціонування галузі сервісних послуг.

Для досягнення зазначеної мети поставлені наступні задачі:

– показати, що використання методів машинного навчання дозволяє поліпшити роботу системи сервісних послуг;

– провести дослідження впливу ознак на необхідні параметри інформаційної системи;

– провести аналіз відомих методів та підходів в предметній області дослідження

При цьому передбачається розв'язок таких підзадач, як

– обробка та очищення даних;

– відбір ознак та проведення аналізу їх якісних показників;

– розробка методів прогнозування;

– застосування моделей машинного навчання;

– тестування ефективності методів застосування машинних моделей на практичних даних;

– програмна реалізація системи прогнозування та функціонування галузі.

Висновок до розділу 1

Проаналізовані основні напрямки з предметної області та відповідним набором даних. Показано важливість області дослідження та актуальність вибряного напрямку. Вказано основні критеріїї, які можна покласти в основу подальших досліджень та окреслено оціночні критарії щодо майбутніх результатів.

Розділ 2
Розробка інформаційної технології

2.1 Мережеві моделі для обробки тексту

У цьому підрозділі представити дві концепції, повторювані нейронні мережі (RNN) і згорткові нейронні мережі (CNN), два потужних типи нейронних мереж, які передають інформацію через серію математичних операцій. Основною метою цієї концепції є обробка даних з послідовними залежностями, такими як представлення текстових документів (векторів), а також додавання можливості розпізнавати шаблони і приховані залежності в послідовних даних. У свою чергу, це робить їх досить корисними для серійні прогнози. Ці мережі дають нам можливість звести до мінімуму помилки у виводі (результатах), безпосередньо всі вхідні параметри. У нас є механізм зворотного зв'язку, відомий як зворотне розповсюдження, яке несе вага нейронів. RNN має тип мережі, яка працює рекурсивно. На додаток до початкового введення, отриманого в кожному нейроні, RNN також включає в себе, в кожному вузлі, прихований стан, який визначив останню особливість в послідовній серії. Цей стан потім поєднується з новим входом для створення нового стану. Така дія називається back-propagation і виконується на етапі навчання моделі.

Ці мережі є потужними моделями навчання і діляться на два види архітектур нейронної мережі: мережі пересилання каналів. Архітектура RST включає в себе мережі з повністю з'єднаними шарами, які діють як класифікатори, включаючи бінарні та багатоцільові проблеми з шарами, а також більш складні проблеми структурованих прогнозування. Ця модель має можливість легко інтегрувати попередньо навчені вбудовування слів, щоб розвинути кращу точність класу. Як бачили раніше, ці мережі працюють з довільними розмірами даних. Щоб протистояти цьому, структура кодування може бути вектором ширини, який може передати іншому статистичному учителю для подальшої обробки. Друга архітектура має можливість працювати з

послідовностями і деревами, зберігаючи структурну інформацію і в той же час виробляючи більш сильні результати для моделювання мови. Перш ніж обговорювати структуру мережі більш глибоко, важливо помітити, як представлені функції.

Нейронна мережа має функцію, яка часто використовується як NN(x), яка приймає вектор x, який представляє (наприклад) слова, як вхідні дані і прогнозує один або кілька векторів виводу, які відповідають вхідному сигналу. Вхідний вектор поширюється через мережу (шар за шаром), поки він не досягне вихідного вузла. Нейронна мережа з одним прихованим шаром може бути представлена

$$y = g(g'(xW^1 + b^1)W^2 + b^2) \tag{2.1}$$

У попередньому рівнянні x є вектором входів і y вектором виходів. Матриця W^1 і вектор b^1 представляють, відповідно, матрицю ваг і вектор зміщення rst шару, в той час як W^2 і b^2 - матрицю ваги і вектор зміщення другого шару. Нарешті, функцію g можна розглядати як функцію активації.

У цій техніці представляємо їх як щільний вектор, а це означає, що кожна основна особливість вбудована в d розмірний простір і представлена як вектор в цьому просторі. У короткому вступі ця модель пов'язує вектор з одним словом / функцією окремо, все в одному вимірному просторі. Коли розріджений режим ставить всі слова / функції в одному векторі, розмір значно збільшується. Ця модель допомагає нам поліпшити крок навчання, оскільки він буде відповідати рівним особливостям з подібними векторами, тому інформація завжди поділяється між одними і тим же функціями.

Оскільки однаковий розмірний простір (щільні вектори), такі можуть бути корисними, коли у нас є слова, що з'являються з певною регулярністю. Наприклад, можемо мати документ d і спостерігати слово добре кілька разів під час тренувального кроку, але спостерігаємо тільки слово погано іноді. За

допомогою цього нового методу вивчений вектор для добра може бути близьким до вектора від поганого, що дозволяє моделі обмінюватися статистичною інформацією між двома подіями з подібними функціями. Важливо відзначити, що ці функції є частиною фреймворку нейронної мережі. Нарешті, можемо структурувати систему класифікації класів на основі архітектури RST (корм-вперед нейронної мережі). Потім можемо NLP витягти набір функцій, які мають відношення до класу виводу, і для кожної функції можемо пов'язати вектор. Зрештою, можемо об'єднати всі вектори інтересу в вхідний вектор x.

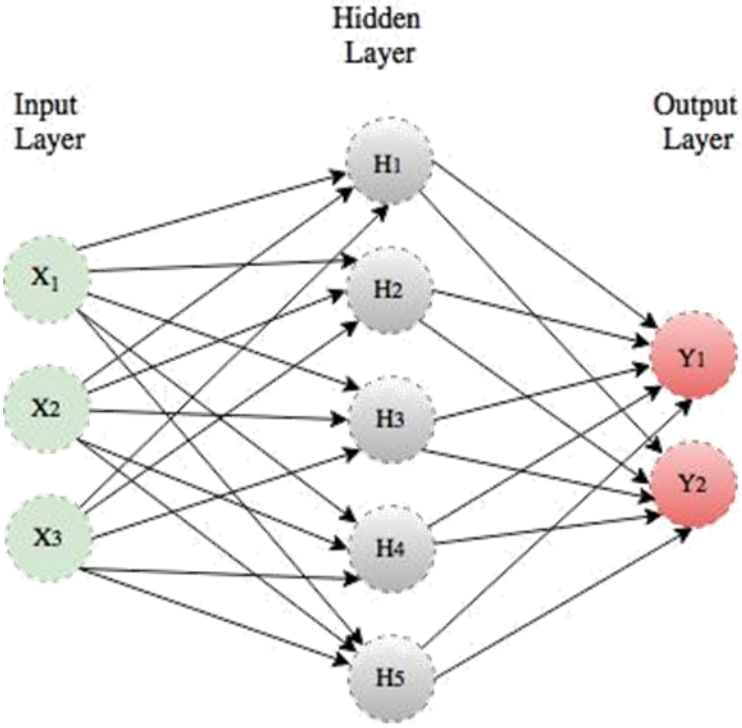


Рисунок 2.1 - Адаптована мережа, без включення умов аналізу [5]

Як правило, коли працюємо з мовними даними, дуже часто маємо входи як слова, речення та документи. “Безперервний мішок слів” в мережах подачі вперед має деякі обмеження з представленням, яке досить обмежене, і змушує розпорядження порядком функцій. У RNN представлення має довільно розміром структуровані входи в векторі і в той же час звертає увагу до властивостей вхідного сигналу. У нейронних мережах (RNN) можемо спостерігати чотири

змінні, які складають функції. Починаючи з вхідного, тобто впорядкований список векторів $X_{i:n}$ разом з початковим державним вектором S_0, і повертає список замовлень головних векторів $S_{i:n}$, а також впорядкований список векторів виводу $Y_{i:n}$ (відповідний стан $S_{i:n}$), як бачимо на рисунку 2.2. Вихідний вектор Y_i потім використовується для подальшого прогнозування. Математично кажучи, у є рекурсивна функція r, яка приймає два входи, вектор X_i і державний вектор S_{i1}, це об'єднане призводить до нового державного вектора S_i. Щоб співставивши цей новий вектор стану, нам потрібна нова функція, яка створює вихідний вектор Y_i. Цей RNN представлений:

$$\begin{aligned} \mathrm{RNN}(S_{i-1}, X_{i:n}) &= S_{i:n}, Y_{i:n} \\ \mathrm{S_i} &= \mathrm{r}(S_{i-1}, X_i) \\ \mathrm{Y_i} &= \mathrm{o}(S_i) \end{aligned} \tag{2.2}$$

Наведена вище презентація відповідає рекурсивній RNN, але для наших вхідних послідовностей у нас є краща з розблокуванням рекурсії для розміру вхідних послідовностей.

У цьому випадку, наприклад, S4 представлений:

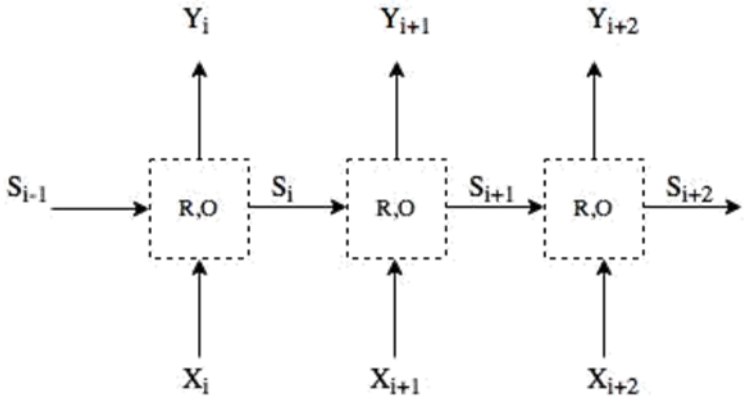


Рисунок 2.2 - Нерозгорнута повторювана нейронна мережа [4]

$$\begin{aligned} \mathrm{S}_4 &= \mathrm{r}(\mathrm{r}(\mathrm{r}(\mathrm{r}(S_0, X_1), X_2), X_3), X_4) \\ \mathrm{r}(S_1, X_2) &= \mathrm{r}((S_0, X_1), X_2) \end{aligned} \tag{2.3}$$

Докладні відомості про функції r і o, допомагають ввести довгострокову короткострокову пам'ять (LSTM), певний тип вузла RNN.

2.3 Мережа аналізу

Ці мережі мають справу з тимчасовими залежностями, які необхідно вирішити. Вирішенням цієї проблеми є метод зворотного розповсюдження, який часто використовується на етапі навчання для перепрограмування ваги кожного нейрона. Зрештою, це дає нам найкраще значення для виходу. Основною метою даного методу є оптимізація навального виходу мережі. Результат вектора, коли він надходить додатковий шар РНН, порівнюється з бажаним виходом за допомогою функції, яка обчислює помилку. Після цього отримаємо розрахункову похибку до першого шару та оновлюємо вагу кожного нейрона, щоб досягти мінімальної втрати. Таке можливо з математичними функціями, які імітують логічні елементи. Кожний pipeline вирішує, які частини нового вхідного вектора повинні бути написані і які частини поточної пам'яті слід забути.

$$\begin{aligned} \mathrm{s_j} &= \mathrm{R_{LSTM}}(s_{j-1}, x_j) = [c_j; h_j] \\ \mathrm{c_j} &= c_{j-1} \times f + g \times i \\ \mathrm{h_j} &= \tanh(c_j) \times o \end{aligned} \tag{2.4}$$

де C_J є компонентом пам'яті;

h_j - прихованим державним компонентом.

Механізм елементів складається з трьох компонентів: один для управління вхідними елементами; інший для вихідних елементів; і коли заборонені забуті елементи. Починаючи з забутих елементів f, що є першими елементами, де вектори проходять через систему контролювання. Цей підхід отримує попередній вихід h_{t-1} і вхідний вектор x_t на кроці t, об'єднує їх і передає результат через функцію g. Ця нова функція буде діяти як функція активації, яка має результати між 0 (відкинути стару інформацію) і 1 (стара інформація повинна розглядатися).

$$\begin{aligned} \mathrm{f} &= \sigma(x_j W^{xf} + h_{j-1} W^{hf}) \\ \mathrm{g} &= \tanh(x_j W^{xg} + h_{j-1} W^{hg}) \end{aligned} \tag{2.5}$$

Після цього переходимо до вхідних елементів, які контролюють, скільки нової пам'яті буде зберігатися в стані комірки. В цьому напрямку функція визначає, наскільки нова пам'ять знаходиться в попередньому стані і яке значення буде оновлено. Після цього вхідні елементи генерують нову пам'ять з усіма оновленнями і проходять через функцію активації tanh для створення нового вектора (кандидати, c_j і h_j).

$$\mathrm{i} = \sigma(x_j W^{xi} + h_{j-1} W^{hi}) \tag{2.6}$$

Нарешті, у нас є вихідні елементи, щоб визначити, що буде вироблятися. Результат стану комірки C_j множиться на вихід функції сигмовиду і результат фінальної інформації на вихід, функцію o і вихід y_j.

$$\begin{aligned} \mathrm{o} &= \sigma(x_j W^{xo} + h_{j-1} W^{ho}) \\ \mathrm{y_j} &= \mathrm{O_{LSTM}}(s_j) = h_j \end{aligned} \tag{2.7}$$

У своєму завдані будемо використовувати Gated Recurrent Unit (GRU) Рисунок 2.3, як хорошу альтернативу LSTM. Перевагою цього нового методу є зменшення кількості елементів.

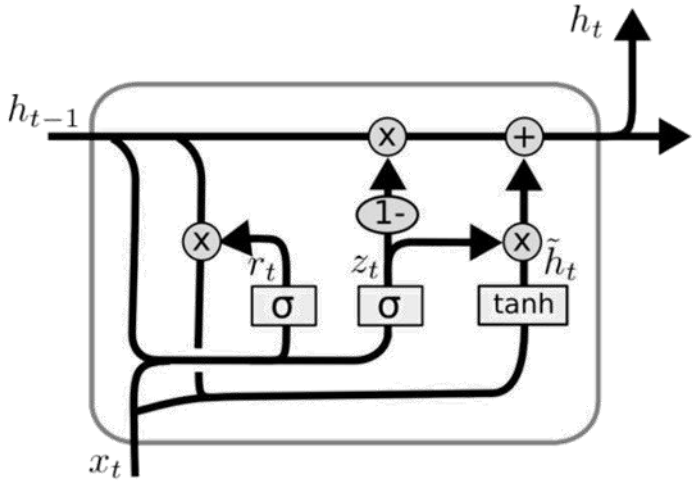


Рисунок 2.3 - Схема повторюваного блоку закритих елементів [4]

$$
\begin{aligned}
\mathrm{h_t} = \mathrm{h_{GRU}}(h_{t-1}, x_t) &= (1 - z_t) \times h_{t-1} + z_t \times \tilde{h}_t \\
\mathrm{z_t} &= \sigma(x_t W^{xz} + h_{t-1} W^{hz}) \\
\mathrm{r_t} &= \sigma(x_t W^{xr} + h_{t-1} W^{hr}) \\
\tilde{\mathrm{h}}_\mathrm{t} &= \tanh(x_t W^{xs} + (h_{t-1} \times r_t) W^{hg}) \\
\mathrm{y_t} &= \mathrm{O_{GRU}}(h_t) = h_y
\end{aligned}
\tag{2.8}
$$

Кожний pipeline має значення:

– Обчислити наступний прихований стан (h_t) з огляду на попередній прихований стан (h_{t-1}) і поточний вхід (x_t) за допомогою двох елементів.

– Оновлення елементів (z_t): Скільки інформації з минулого стану зберігається і скільки нової інформації додається. Визначається на основі інтерполяції попередньої функції та наступної.

– Скидання елементів (r_t): Наскільки минула держава сприяє новій державі-кандидату (h_t). Використовується для керування доступом до попереднього стану та обчислення запропонованого оновлення.

Щоб завершити цей розділ про нейронні мережі, пояснемо другу мережу, Convolutional Neural Network (CNN). Важливо відзначити, що наступні функції є частиною нейронної мережевої бази.

Є деякі початкові подібності з повторюваними нейронними мережами. Входи двох мереж можуть бути однаковими. Наприклад, у цій роботі буду використовувати область вбудовування слів, щоб представити одне або кілька слів у векторах. CNN використовує ці вектори, оскільки вони будуть передавати деякі шари, які застосовували згортки до локальних функцій. Вирішимо навчити просту CNN з одним шаром згортки на вершині векторів слів, отриманих від нейронної мовної моделі.

Як бачимо на рисунку 2.4, архітектура CNN починається з k-вимірного вектора слова, що відповідає і-го слову в реченні. Кожен вектор має відмінну довжину, відповідно до кількості слів, які хочемо представляти. Згортковий шар, це операція, яка включає в себе інтревал, який застосовується до вікна h слів для того, щоб виробляти нову функцію. Ця нова функція являє собою початковий вектор слова, в невеликій карті функцій (вектор розміру).

Наприклад, функція c_i генерується з потоку слів $x_{i:i+h-1}$ за:

$$c_i = f(w \cdot x_{i:i+h-1} + b) \tag{2.9}$$

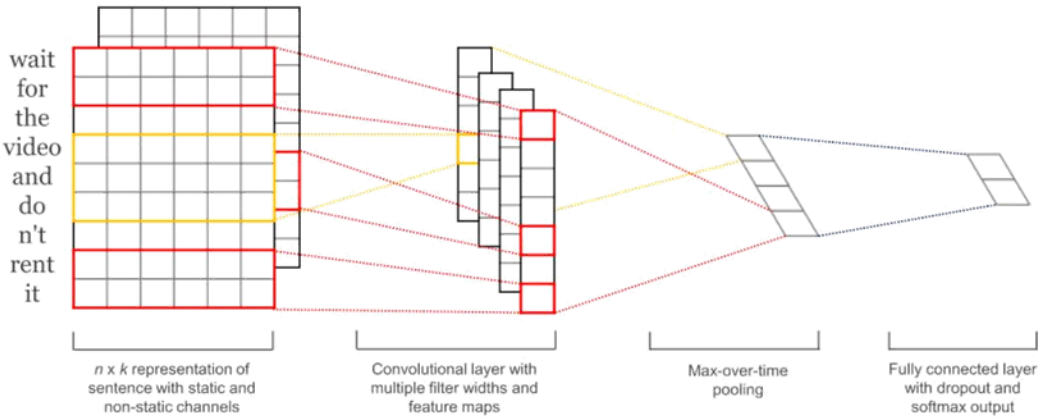


Рисунок 2.4 - Архітектура згортки нейронної мережі, з опису, наданого [8]

Де b є терміном зміщення, а f - це нелінійна функція, така як гіперболічний тангенс. Цей фільтр застосовується до кожного можливого вікна слів у реченні для створення карти функцій. Наступний шар - це макс-об'єднуючий шар, який наноситься над картою функцій і приймає найвище значення, яке може представляти кожну карту функцій в одному векторі. В основному, застосували фільтри по мережі, щоб зменшити довжину вектора, який представляє вхід. Модель використовує кілька елементів (з різними розмірами блоків) для отримання численних функцій. Ці функції, які утворюють макс-об'єднуваючий шар, передаються до повністю з'єднаного шару, який дає результат, вихідний вектор.

Ці мережі мають деякі варіації, оскільки можемо додати згорткові шари. Наприклад, проект буде слідувати підходу мережі, яка має два згорткових шари з кроком два (тобто нова карта функцій представлена в половині вектора), і макс-опитування шару, щоб сформувати пірамідальну мережу.

Нарешті, для досягнення кращих результатів буду використовувати дві регресійні метрики, які мають можливість кількісно оцінити читабельну кількість помилок при порівнянні досягнутого прогнозу з бажаним прогнозом. Коренева середня квадратна помилка (RMSE - Root Mean Squared Error) вимірює

середню величину похибки і дає правило підрахунку очок. RMSE дасть більшу вагу великим помилкам, які пізніше будуть повторно видалятись в подальшому використанні. Mean Absolut Error (MAE), також вимірює середню величину похибки в наборі прогнозів, не розглядаючи їх напрямок. Це середнє значення над тестовою вибіркою абсолютних значень між прогнозом і фактичним спостереженням, де всі окремі залежності мають однакову вагу і регресійні показники даються:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_i - \hat{y}_i)^2} \tag{2.10}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_i - \hat{y}_i| \tag{2.11}$$

де y_i є фактичним або бажаним значенням виводу, а $\hat{y}_i$ – прогнозованим значенням. Результат, отриманий цим методом, являє собою середнє абсолютне значення помилок.

2.3 Методи навчання для реплікації

В останньому підрозділі говоримо про інспекцію охорони здоров'я на основі деяких відгуків, зроблених клієнтами (наприклад, в Yelp або nEmesis System). У цій частині визначимо елементи, пов'язані з методами для текстових класів. Для початку, є підхід, який пояснює, як згорткові нейронні мережі можуть бути застосовані до текстових класів.

Основною метою цього розділу є введення нової концепції для текстових класів, глибокої мережі Convolutional Neural Networks (DPCNN), в даному випадку з 15 шарами ваги, які позначають шість наборів даних. За допомогою

цієї архітектури можемо звести до мінімуму складність мереж. Крім того, можна підвищити точність класифікації тексту і категорії тем, не маючи складних нейронних мереж і великих обсягів навчальних даних. Згортка нейронної мережі (CNN) - це мережа пересилання каналів (з згорткових шарів для перетворення обсягів даних (наприклад, тексту або зображення) у вектор з об'єднанням шарів.

Одним з основних проблем між CNN і RNN є спосіб, яким вони обробляють слова.

RNN має лише рекурсивні з'єднання, тоді як CNN може оброблятися паралельно. Останній здатний знизити складність, коли навчальні набори даних викликають обчислювальні проблеми. Визначили цю нейронну мережу і знайшли архітектуру, яка може досягти найкращої точності, збільшуючи глибину. Вдалося це зробити, створивши архітектуру піраміди, яка проходить через блоки згортки і шар вибірки знову і знову.

2.4 Структура мережі

Структура DPCNN, схожа на перший шар вбудовування тексту з подібними або різними словами. Далі йде згорток блоків з двома шарами згортки і ярлик, переплетений з об'єднуючими шарами з максимумом кроку два для вибірки вниз. Це відбувається з кількістю функцій карти, які роблять обчислення часу для кожного згорткового шару вдвічі. Нарешті, останній шар об'єму перетворює внутрішні дані в один вектор. Коли говоримо, що виконуємо максимальне об’єднування з розміром три і крок два, це означає, що кожен шар виробляє репрезентативний вектор документа, беручи максимум три внутрішніх вектори. Ця вибірка робить внутрішнє представлення кожного документа половиною його розміру, оскільки представляємо, наприклад, три області тексту, що перекриваються, якщо у нас є ще один можливий текст. У DPCNN є ярликові з'єднання з попередньою активацією, а це означає, що пропустили деякі шари згортки з попередньою активацією. Такий підхід відноситься до активації, що

робиться до зважування, а не після, як звичайно. Модель має простий спосіб вирішення проблеми розмірності, вирішуючи ці прохідні дані саме так, як це є, залежно від кількості карт, змінених між шарами.

Після цього пояснення про архітектуру DPCNN, зосередимось на першому кроці цієї мережі, який є перетворенням слова в вектор (слово вбудовування). Обчислюємо функцію $W_{x + b}$ для кожного удаваного слова документа. x являє собою k-слово регіону, W представляє ваги і b представляє ухил. Останні дві змінні навчаються з параметрами інших шарів. Для області k-слова для змінної x маємо три типи простого представлення. RST — це послідовне представлення вхідних даних об'єму одномірних векторів; другий - вектор мішка слів; третій і останній — це вектор введення мішка з n-gram (наприклад, бі і триграми, що містяться в слові). Це відноситься до розміру регіону k = 1, і всі вони стають вбудовуваними словами. Для регіону вбудовування шарів з розміром k >1 це означає, що він шукає складні слова, на відміну від мережі з декількома шарами (погана оптимізація).

Точність була неякісно визначена при використанні системи без нагляду. Вони розподіляють область тексту як вид-1, і його прилеглі регіони як вид-2. Після цього вони використали навчальний набір для нейронної мережі одного прихованого шару, з метою прогнозування перегляду-2 знаючи (навчальний) вид-1, тобто вхід прихованого шару. Ця схема без нагляду функції вбудовування покращує точність.

Кращі результати від DPCNN без нагляду бі-вбудовування ("бі" означає два погляди). Для word2vec найкращою отриманою моделлю є мережа ієрархічної структури, яку обговоримо в наступній статті. На закінчення дослідимо, як вирішити проблему глибоких CNN на рівні слів з великими наборами даних навчання. Модель глибокої пірамідальної згортки нейронної мережі має низьку обчислювальну складність і може представляти великі асоціації в тексті з великою точністю.

Протестований метод розділив модель на дві частини: перший має ієрархічну структуру документів; а другий має два рівні слова і рівня речення. Останній клас має важливий зміст при побудові представлення документа.

Використаємо мережу ієрархічної структури, яку можна розділити на дві згідно структури документа.

Представлення документів будується на основі представлення речень, а потім об'єднувати ці речення в представленні документа. Після цього процесу визничимо, що кожне слово може мати відмінну інформативну вагу, тому створемо значення важливості слова і речення в залежності від контексту, який вставляється. Наприклад, якщо є речення зі словом ідеально підходить в огляді (наприклад, Yelp), це може сприяти позитивному коментарю. Таким чином, це слово має більш сильну інформацію для класифікації.

Механізм, що використовується, є послідовним кодером на основі GRU для відстеження стану послідовності без використання окремих комірок пам'яті. В основному, цей механізм контролює, як інформація оновлюється до певного стану, в певний час, і обчислює новий стан на основі попереднього.

Декодуємо слова і речення в векторне представлення і використаємо ієрархічну структуру для прогресивної побудови вектора документа. Кодер Word, який отримує як введення вектор слова, який проходить метод вбудовування (bidirectional GRU). Наступним кроком є значення слова, яке представляє вагу слова. Не всі слова однаково сприяють представленню значення речення, тому запровадили механізм вилучення важливих слів. Після цього вилучення обчислюємо вектор речення, додаючи представлення цих інформативних слів. Він слідує за кодером речення, який має ці вектори речення як вхідні дані, і отримує вектор документа в аналогічному процесі. Нарешті, є вага речення, щоб відзначити речення, які є підказками, які використовуються при правильному класифікації документа. Вектор документа використовується як ознака для класифікації документів (за допомогою функції softmax).

Аналізуючи результати та порівнюючи з іншими методами, можна зробити висновок, що запропонована ієрархічна модель уваги має найкращу продуктивність у всіх наборах даних. Для менших наборів даних модель перевершує інші методи. Нейронні мережеві методи мають перевагу перед іншими традиційними методами (наприклад, n-грамами, SVM + Bi-grams, BOW, TF-IDF) для широкомасштабної реплікації класів тексту.

Висновки до розділу 2

Однією з найбільш важливих частин цієї роботи є хороший набір даних, з усією наявною інформацією. Якщо моделі отримали хороший результат, з більшою частиною негативних відгуків, наприклад, 80% відгуків у цьому наборі даних мають оцінку значимості між 80-100, що зробило його складним для досягнення кращих результатів.

Результати показують, що методи машинного навчання можуть бути використані для класифікації ресторанів з точки зору санітарних проблем, а результати з використанням тексту досить надійні. Однак, це складно працює для боротьби з підробленими відгуками.

Розділ 3
Побудова моделі системи прогнозування продажів

3.1 Представлення текстових документів

У цьому розділі досліджуються елементи, які допоможуть чітко зрозуміти переваги методу. Вводяться поняття, пов'язані з представленням документів з використанням, вбудовування слів і глибоких нейронних мереж для текстових класів.

Класифікація тексту за допомогою контрольованого навчання вимагає використання методів представлення тексту, щоб бути входом до алгоритмів навчання. Загальний підхід передбачає перетворення текстових документів у вектори, які їх представляють. Кожна позиція одного такого вектора повторно визначає частоту зразка с слова в документі.

Можемо використати векторну модель як алгебраїчної моделі, яка представляє текстову інформацію як вектори. Крім того, складові цих векторів представляють актуальність кожного терміну зі словника, який використовується в збірнику документів. Одним з можливих методів, які можемо використовувати для зважування термінів є TF-IDF. Частота терміну (TF) визначає, скільки разів термін x присутній в документі d. Документ, в якому згадується даний термін частіше, має більше спільного з цим терміном, і тому повинен отримувати більш високий бал. Тому призначаємо вагу для кожного терміну в документі, який залежить від кількості входжень терміну в документі.

$$\mathrm{TF}(\mathrm{t},\mathrm{d}) = \sum_{x \in d} \mathrm{fr}(\mathrm{x},\mathrm{t}) \tag{3.1}$$

У попередній формулі fr(x, t) є простою функцією:

$$\mathrm{fr}(x,t) = \begin{cases} 1 & \text{if } x = t \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

Зворотна частота документів (IDF) виходить з терміну, який зустрічається занадто часто в колекції документів, розділяючи загальну кількість документів d, на кількість документів, які містять термін x. Ідея цього полягає в тому, щоб зменшити вагу TF терміну, на фактор, який збільшується з частотою збору терміну.

$$\mathrm{IDF_t} = \log\left(\frac{N}{DF_t}\right) \tag{3.3}$$

У цьому рівнянні змінна N - це загальна кількість документів, а змінна DFt - це кількість документів, де з'являється термін x.

Нарешті, можна сказати, що TF-IDF представляє, наскільки важливим є термін x для документа d в колекції. Об'єднуємо два визначення, щоб генерувати складену вагу для кожного терміну в кожному документі. Схема зважування TF-IDF призначає термін на вагу в документі d, наведеному таким рівнянням:

$$\text{TF-IDF}_{t,d} = \mathrm{TF_{t,d}} \times \mathrm{IDF_t} \tag{3.4}$$

Щоб побудувати векторну модель представлення колекції документів, потрібно створити словник з термінами, присутніми у всіх документах. Потім у нас є словник індексації, щоб могли перетворити тестовий документ, встановлений у векторному просторі, де кожен термін вектора індексований як наш індексний словник. При використанні представлень Vector Space Model стандартним способом кількісної подібності між двома документами є

обчислення косинусної схожості їх векторних представлень. Для двох векторних представлень V(d1) і V(d2) схожість козина представлена:

$$\text{sim}(d1, d2) = \frac{\text{V(d1)}.\text{V(d2)}}{\|\text{V(d1)}\| \times \|\text{V(d2)}\|} \tag{3.5}$$

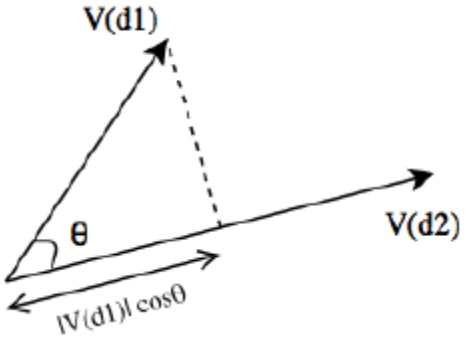


Рисунок 3.1 - Проекція вектора V (d1) у вектор V (d2)

З цієї формули бачимо, що чисельник являє собою точку між векторами V(d1) і V(d2), тоді як вектор є результатом їх евклідової довжини. Евклідова довжина вектора V(d) $\|\text{V(d)}\| = \sqrt{\sum_{i=1}^{M} \text{V}_i^2(d)}$.

Ця формула $\text{V(d1)}.\text{V(d1)} = \|\text{V(d1)}\| \, \|\text{V(d2)}\| \cos\theta$, представляє проекцію вектора V(d1) у вектор V(d2).

Нарешті, можемо осягнути схожість косину між двома векторами (або двома документами на Vector Space). Ця міра обчислює косинус кута між ними. Взяли до уваги тільки TF IDF кожного документа, однак, додаємо кут між документами. Векторна модель має обмеження, які призводять до того, що слово вбудовує кращий метод. Довгі документи погано представлені, оскільки вони мають погані значення подібності та документи з подібним контекстом, але словник не буде зв'язаний, що призведе до помилкового негативного збігу. Вбудовування слів - це набір методів моделювання мови та вивчення функцій у обробці природної мови, де слова або фрази зі словника співставляються з

векторами реальних чисел, щоб захопити з ними стільки семантичних, контекстних та ієрархічних.

Word2vec - це алгоритм, який складається з двох методів, а саме CBOW (Безперервний мішок слів) і модель Skip-gram, як показано на рисунку 3.2. Техніка RST, CBOW заснована на дослідженні всіх слів в “одній сумці” і без врахування порядку слів, які не в проекції. Крім того, вводяться прогнозування слова, а також вхідних даних, де критерій навчання полягає в правильній класифікації слова.

Друга методика схожа на попередню, однак замість того, щоб передбачати поточні слова на основі контексту, вона намагається класифікувати слово на основі іншого слова в тому ж реченні. Точніше, є слово як вхід до класифікатора з безперервним проекційним шаром. До того ж він може передбачати слова до і після поточного слова між рівнем. Якщо хочемо підвищити якість результатів виводу (вектори слів), повинні збільшити діапазон і одночасно, обчислювальна складність зростає. Складність навчання цієї архітектури пропорційна Q = C (D + D log2(V)), де C - це максимальна відстань слів. Таким чином, якщо виберемо C = 5, для кожного навчального слова виберемо випадковим чином число R в діапазоні <1; C>, а потім використовувати R слова з історії та R слів з майбутнього поточного слова як правильні мітки.

На останній темі цього підрозділу поясню концепцію n-grams на основі [ngr]. Ця модель замість того, щоб обчислити ймовірність слова з огляду на всю його історію, можемо приблизно до історії всього за останні кілька слів. Наприклад, припустимо, що історія h це дуже важливо, розглядаючи багато разів, і хочемо знати ймовірність того, що наступне слово теза:

$$P(\text{тезис} \mid \text{кількість згадування}) \qquad (3.6)$$

Одним з типів цієї моделі є біграмові моделі, які приблизні до ймовірності слова, даного всім попереднім словам $P(w_n|w_1^{n-1})$, використовуючи лише ймовірність попереднього слова $P(w_n|w_{n-1})$.

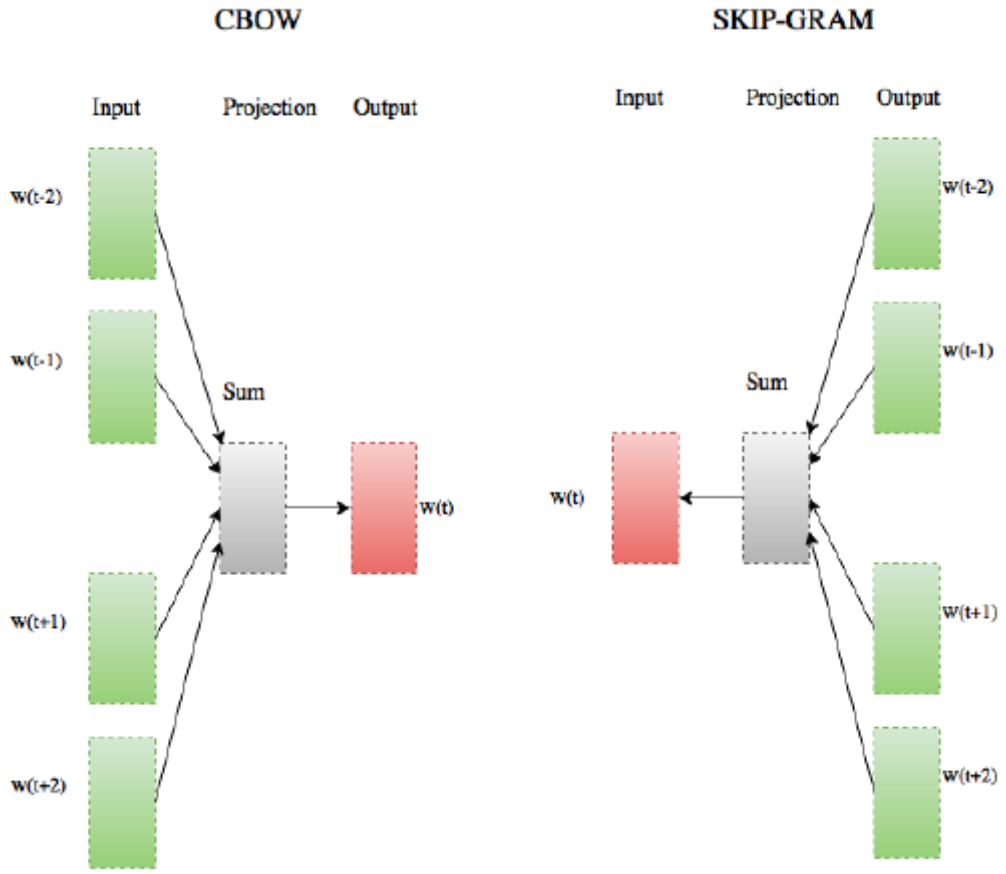


Рисунок 3.2 - Cbow і SKIP-GRAM архітектури [5]

Замість того, щоб обчислити ймовірність P(тезис | кількість згадування), метод приблизний до нього з імовірністю.

$$P(\text{тезис} \mid \text{частота}) \tag{3.7}$$

З огляду на біграмові припущення щодо ймовірності окремого слова, можемо обчислити ймовірність повної послідовності слів.

3.2 Обробка даних

Основною метою було вивчення інформації, що міститься в соціальних мережах, з метою зробити публічні перевірки та відповідні розкриття інформації більш доступними. Вивчили одну статистичну модель, яка порівнює текстові сигнали в оглядах ресторанів і записи інспекції гігієни від Департаменту громадського здоров'я. Департамент громадської інспекції охорони здоров'я зазвичай ділиться своїми записами, щоб допомогти клієнтам вирішити, який ресторан не порушив санітарні норми, тим самим даючи покровителям запевнення, що заклад є безпечним. Крім того, в деяких місцях ресторани зобов'язані розмістити свої оцінки інспекції. Ці ресторани мають свої оцінки на основі громадян, які відвідали.

У цій роботі представлено емпіричне дослідження, щоб продемонструвати, якщо відгуки мають реальну цінність для прогнозування перевірок санітарного стану. З точки зору бази даних, автори використовували записи громадської інспекції і додали базу даних Yelp. За допомогою цієї інформації вивчили кореляцію між штрафними балами і кількома статистичними відгуками, розділивши цю частину на три підрозділи: кількості відгуків, частоти і їх середньої довжини. Після цього вони створили концепцію настроїв відгуків, яка поєднувала в ресторанах середні рейтингові бали і негативні відгуки з 3 або менше зірок. Крім того, це корелювало з новими відгуками клієнтів. Останній елемент це оманливість відгуків, це коли вони є підробленими відгуками, що, в даний час, є надзвичайно поширеною проблемою. Розглянули кореляцію між порушеннями гігієни і ступенем обману. Таким чином, почали з дисперсії рейтингів огляду, що є мірою, яка показує форму думок про розподіл (двомодальні). Інша міра заснована на лінгвістичних образах, які розраховували обсяг оманливих відгуків, збираючи набір підроблених і правдивих відгуків, створюючи таким чином загальну картину.

Іншим способом було відгуки, які полягали в усуненні всіх відгуків, які знаходяться занадто далеко від середнього рейтингу огляду (на основі певної дельти). Наприклад, при розгляді ресторану, який має середню оцінку 3,9 зірки (діапазон зірок від 1 до) і вибираючи дельту 2, беремо до уваги всі рейтинги між 1,9 і 5 зірками. Результати їхнього досвіду базуються на розгляді особливостей специфікацій, таких як думки клієнтів та мета-даних ресторану. RST враховує середній рейтинг відгуків і зміст відгуків в n-грамах, а другий - на основі місця розташування (поштовий індекс), типу кухні (наприклад, португальської) і інспекції - історія.

Результати були об'єднані для детального класування. Визначено, що збільшення штрафного балу інспекції як порогу призводить до кращої точності. Точність зміни від найгіршої до найкращої інформації мета-даних - це середній рейтинг огляду (57,52%), кухня (66,18%) та місцеположення (67,32%), а найбільш е екстенсивними, інформативними особливостями є історія інспекції (72,22%) і текстовий контент (уніграм + двограм, 82,68%).

Останній елемент цього підрозділу намагається підійти до частої проблеми в суспільстві, яка називається профілактикою харчових захворювань на основі баз даних соціальних мереж. Численні ресторани не могли ідентифікувати джерела зараженої їжі. Крім того, була найбільша перешкода, яка полягає в тому що дозволяється ресторанам не мати інспекцій департаменту охорони здоров'я протягом цілого року.

Цей метод вирішує подібну проблему: визначення, якщо твіт вказує на харчову хворобу чи ні. Створено короткий словник з ключовими словами, як негативними, так і позитивними, і кожен з них несе певну вагу, яка відповідає цьому слові, що з'являється у твіті. Наприклад, можна розглянути твіт "Мій шлунок обурився", nEmesis, який отримує цей твіт і перевіряє словник, щоб отримати вагу своїх слів. Структура словника схожа на таблицю з двома аргументами "Особливість | Вага", де аргумент rst - це слово (наприклад, шлунок), а другий - вага (наприклад, 1.04635, якщо слово є позитивною рисою).

Після цього nEmesis може додати рахунок в ресторан, де користувач спочатку твітнув.

Для аналізу вибрано тип класів з позначками харчової хвороби, розділяючи їх на дві категорії: одну з відомих харчових хвороб патогенів хвороби та іншу для неспецифічним агентам. Вони також зробили свої дослідження на основі цього випадку (блювота, діарея, біль у животі є одним з прикладів поширених симптомів).

Інспектори nEmesis зобов'язані провести стандартну, рутинну перевірку, якщо деякі користувачі вказали на це. Використовували процес аналізу повідомлень, щоб сформулювати деякі питання (наприклад, якщо автор твіту мав розлад шлунка), і пвідомив в короткому опитуванні. Потім вони надіслали це опитування робітникам і заплатили по одному центу за кожен оцінюваний твіт. Кожен працівник позначив твіт з трьома можливими відповідями, і в підсумку, у них були всі твіти, позначені думкою працівників.

Навчальна частина складається з гіпер-площини, яка розділяє позитивні та негативні точки даних на основі проблеми квадратичної оптимізації за допомогою методів стохастики градієнтного спуску. Ця модель прогнозує, якщо твіт вказує на харчову хворобу, з навчальним набором з 8000 твітів, незалежно позначених працівниками, як описано вище, використовуючи n-грами як функції.

3.3 Методи навчання для тексту

В останньому підрозділі говоримо про результати аналізу деяких відгуків, зроблених клієнтами (наприклад, в Yelp або nEmesis System). Дослідимо кращий процес представлення документів навчальної бази, яка називається Document Vector (Doc2VecC). Doc2VecC - це процес, який створює словобудування з великою продуктивністю, здатною тренувати мільярди слів на годину, а також генерувати представлення невидимих документів.

Представлення документа робиться під час навчання кроку і формується в на основі вбудовування всіх слів в документ. Ця постановка, має деякі якості, такі як:

Складність моделі залежить тільки від розміру словникового запасу, маючи таку ж структуру моделі як Word2VecC.

Створення нового узагальнення, залежної від даних, що сприяє нечастим словам.

Тільки включно з усередненням слова.

Векторне представлення може збігатися з аналізом настроїв та визначити ставлення відгуку щодо даної теми на основі даного текстового коментаря.

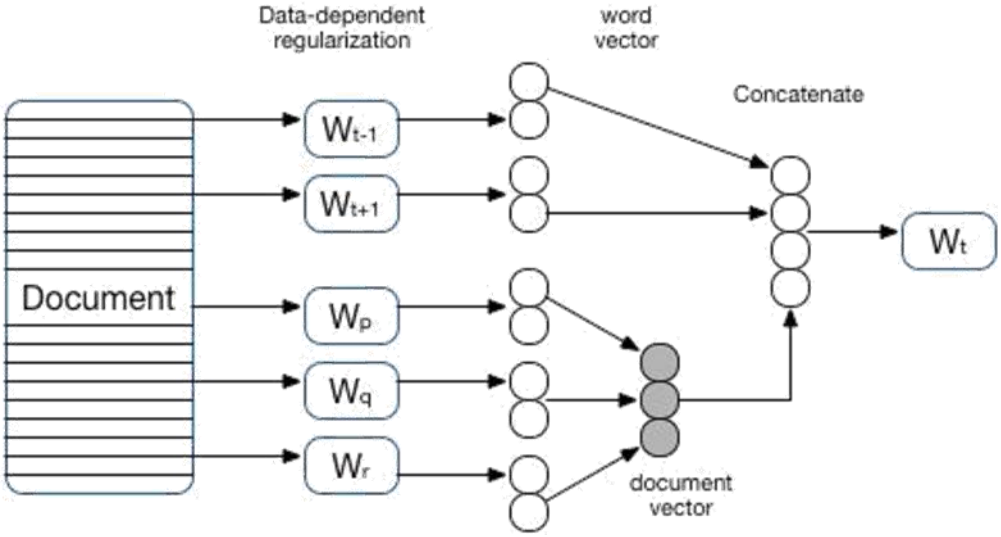


Рисунок 3.3 - Адаптована рамкова структура Doc2VecC [5]

Ідея представлення документів як середнього значення вбудовування слів походить від інших творів, які показали, що додавання або віднімання вбудовування слів може класифікувати синтаксичні та семантичні регулярності речень. Архітектура Doc2VecC складається з вхідного шару, де є сусідні слова, щоб забезпечити локальний контекст, і важливого слова (слова з більш високою вагою) для представлення всього документа. Також є проекційний шар, де слововектор (складається з сусідніх слів) і вектор документа об'єднати і зробити

середнє значення слова вбудовування. Нарешті, у нас є вихідний шар, щоб передбачити слово мети. Щоб поліпшити свою продуктивність вирішено змінити оригінальний документ, випадково видаливши деякі слова під час навчального кроку, і представляти документ, використовуючи тільки слова, що залишилися (з їх вектором вбудовування слів).

Імовірність дотримання слова w_t з огляду на його локальний контекст C_t, а також глобальний контекст x~, використовуючи Doc2VeC є (T - це довжина документа):

$$P(w^t|\mathbf{c}^t,\tilde{\mathbf{x}}) = \frac{\exp(\mathbf{v}_{w^t}^\top(\mathbf{U}\mathbf{c}^t + \frac{1}{T}\mathbf{U}\tilde{\mathbf{x}}))}{\sum_{w'\in V}\exp(\mathbf{v}_{w'}^\top(\mathbf{U}\mathbf{c}^t + \frac{1}{T}\mathbf{U}\tilde{\mathbf{x}}))}, \quad (3.8)$$

де U_c^t є локальним контекстом, а T_1 U_x ~ є глобальним контекстом.

Висновки до розділу 3

Під час пошуку всіх експериментальних таблиць (швидкість помилок, точність, час навчання/генерації) і використання набору даних модель Doc2VecC перевершує в порівнянні з деякими іншими моделями, такими як Bag-of-words або Word2VecC, оскільки вона має найнижчий показник похибки та найкращу точність (на підмножині тестового набору Semantic-Syntactic World Relationship). Єдине питання щодо цього методу полягає в тому, що час навчання трохи вище, ніж очікувалося. Найкращою частиною Doc2VecC є узагальнення даних, яке сприяє нечастим словам і полегшує представлення документів із середнім показником вбудовування вивченого слова. В експериментальній частині можна помітити, що цей метод дає хороші результати.

Розділ 4

Дослідження ефективності методів прогнозування

4.1 Чисельні результати проведених досліджень

Набір даних містить дані про порушення з кожної повної або спеціальної перевірки програми, проведеної до трьох років до останньої перевірки ресторанів і кафетерій в активному статусі на дату запису (дата витягування даних). Під час перевірки призводить до 1 порушення, значення для пов'язаних полів повторюються для кожного додаткового запису порушення. Заклади однозначно ідентифікуються за номером camis (ідентифікатор запису). Тисячі ресторанів починають бізнес і щороку виходять з бізнесу; до набору даних входять лише ресторани в активному стані.

Записи також включені для кожного ресторану, який подав заявку на отримання дозволу, але ще не був перевірений, і для перевірок, що призвело до відсутність порушень. Заклади з датою перевірки 1/1/1900 – це нові заклади, які ще не отримали перевірку. Ресторани, які не отримали порушень, представлені одним рядком і закодовані як такі, що не мають порушень за допомогою поля дія.

Оскільки цей набір даних складено з кількох великих адміністративних систем даних, він містить деякі нелогічні значення, які можуть бути результатом помилок введення або передавання даних. Дані також можуть бути відсутні.

Цей набір даних та інформація на веб-сайті департаменту охорони здоров'я надходять з одного джерела даних. Веб-сайт департаменту охорони здоров'я знаходиться тут: http://www1.nyc.gov/site/doh/services/restaurant-grades.page.

Набір даних надав історичні оцінки для кожного ресторану, а також найсвіжіших оцінок. Але рейтинги також розбиваються на критичні і не критичні категорії. Так що єдиний рейтинг матиме два ряди, один за критичні

порушення, а інший з некритичні порушення. Тому видалили інформацію щодо опису/типу порушення та видалили дублікати.

Оскільки набір даних містить історичну інформацію, також створено унікальний ключ для кожного окремого ресторану. Створимо зміну, що поєднує назву ресторану, будівлю, вулицю та поштовий індекс (наприклад, DARKHORSE 17 MURRAY STREET 10007).

4.2 Система рейтингу

Рахунок від 0 і вгору. Низька оцінка хороша, оскільки це вказує на відсутність порушень. Кожне порушення має певне значення, тому оцінка 0 означає, що немає порушень. Існує три типи порушень:

Загальне — наприклад, не правильно визначено кухонне обладнання— мінімум 2 бали

Критично — наприклад, подача сирої їжі, такої як салат, без належного очищення — мінімум 5 балів

Небезпека для здоров'я населення - наприклад, не в змозі тримати їжу при правильній температурі - 7 балів

Остаточний клас базується на сумі всіх балів.

Від 0 до 13 заробляє A

Від 14 до 27 заробляє B

28 або більше заробляє C

Розбивка оцінок можна побачити нижче у відсотковому вираженні, згруповані за останнім класом і всіма оцінками.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
pd.set_option("precision", 2)
pd.options.display.float_format = '{:20,.2f}'.format
```

```
df = pd.read_csv("Restaurant_Grades.csv")

# Replace spaces with underscores
df.columns = df.columns.str.replace("CUISINE DESCRIPTION","CUISINE")
df.columns = df.columns.str.replace(" ","_")

# Clean up some of the names
df["DBA"].replace("'","", inplace=True) # Remove apostrophe
df["DBA"].replace(" ?\(.+\)", "", regex=True, inplace=True) # Remove values in
parenthesis
df["DBA"].replace(" ?#.*", "", regex=True, inplace=True) # Remove # followed by
some string

# Convert dates to datetime
df.GRADE_DATE = pd.to_datetime(df.GRADE_DATE, format="%m/%d/%Y")
df.RECORD_DATE = pd.to_datetime(df.RECORD_DATE, format="%m/%d/%Y")

# Create a unique key based on restaurant
df["KEY"]          =          df[['DBA',          'BUILDING',          "STREET",
"ZIPCODE"]].astype(str).apply(lambda x: ' '.join(x), axis=1)
print("num    ratings:    {}    num    unique    restaurants:    {}".format(len(df),
len(df.KEY.unique())))
# num ratings: 186185 num unique restaurants: 24607

#
df =
df[["KEY","DBA","BORO","CUISINE","SCORE","GRADE","GRADE_DATE","RECORD_DATE"]].dr
op_duplicates()
df = df.sort_values(["KEY","GRADE_DATE"],ascending=[True,False])
```

GRADE	count all	perc all	count most recent	perc most recent
A	70,856.00	87.25	22,017.00	89.47
B	6,721.00	8.28	1,357.00	5.51
C	1,466.00	1.81	259.00	1.05
Not Yet Graded	2.00	0.00	2.00	0.01
P	1,193.00	1.47	nan	nan
Z	972.00	1.20	972.00	3.95

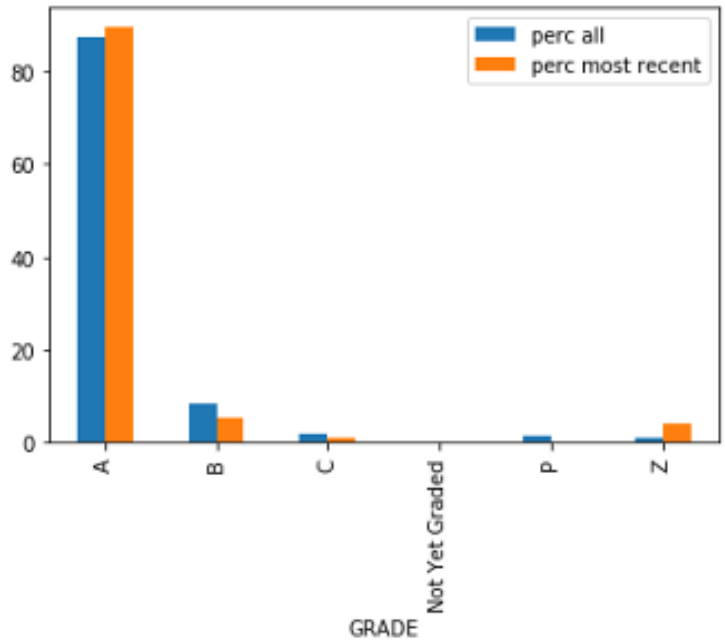


Рисунок 4.1 – Розподіл по рейтингам

Більшість оцінок, менше, 2% перевірок в результаті класу С.

```
gb_all = df.groupby("GRADE").GRADE.agg(["count"])
gb_all["perc"] = gb_all / gb_all.sum() * 100

gb_recent = df.drop_duplicates("KEY").groupby("GRADE").GRADE.agg(["count"])
gb_recent["perc"] = gb_recent / gb_recent.sum() * 100

gb_all = gb_all.join(gb_recent,lsuffix=" all", rsuffix=" most recent")#

gb_all[["perc all", "perc most recent"]].plot(kind="bar")
gb_all.style.format("{:,.2f}")
```

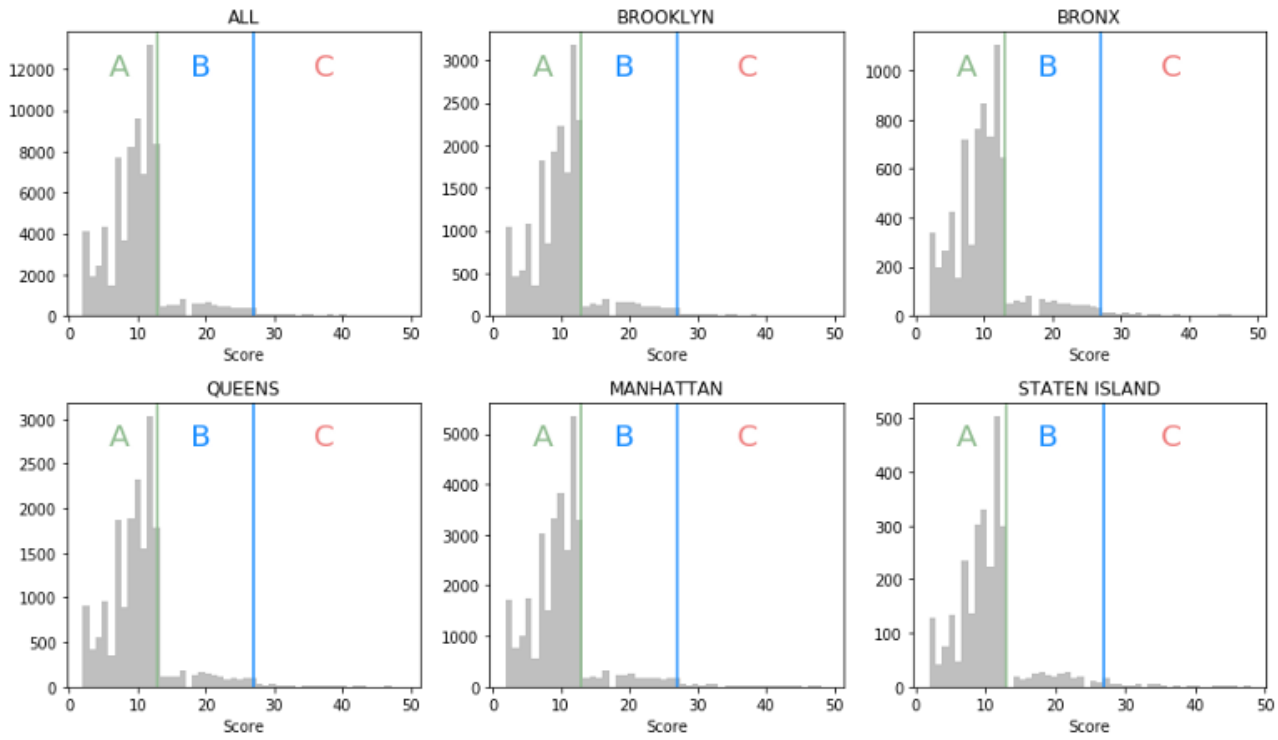


Рисунок 4.2 – Рейтинг по районам

Хороше представлення на 14 пунктів, при зрізі від А до Б. Два типи перевірок привели до відповідного класу: початкові перевірки, за які ресторан заробляє А, і повторні перевірки, які призвести до А, В або С.

Ресторан має два шанси заробити А в кожному циклі огляду. Якщо він не заробляє А на першому огляді, він втратив. Інспектор повертається в ресторан неанонсованим, як правило, протягом місяця, щоб оглянути його знову і повторний огляд оцінюється. Якщо клас є В або С, ресторан отримає картку класу і картка класу, очікує на розгляд. Він може розмістити будь-яку карту, поки вона не має можливості бути розглянутою.

Поки ресторан не має огляду, він вказаний як ще не оцінений на відповідному веб-сайті.

4.3 Розподіл рейтингу згідно інспекції

Таким чином, не -А оцінки спочатку не повідомляються, і ресторан має ще один шанс. Це знижує ймовірність оцінки В і С, про які повідомлялося. Але велика кількість балів прямо на відсіканні від А до Б. Припускаємо, що інспектори виявляють відхилення при наближені до порогу відсікання. Якщо він близький, інспектори дають ресторану вигоду від отримання вищого класі, оскільки клас є найбільш помітним компонентом для замовника. Це також може бути наслідком кількості балів, призначених кожному порушенню. Наприклад, критичні порушення становилять мінімум 5, а небезпека для здоров'я населення - мінімум 7. Таким чином, оцінка 10 або 12 повинна бути більш поширеною, що і бачимо.

```
f, axes = plt.subplots(nrows=1,ncols=3,figsize=(15,5))

_ = axes[0].hist(df.GRADE_DATE.dt.dayofweek, bins=np.arange(8)-0.5, rwidth=0.95)
axes[0].set_title("Days of week (0 is Monday)")
_ = axes[1].hist(df.GRADE_DATE.dt.day, bins=31)
axes[1].set_title("Day of month")
_    =    axes[2].hist(df[df.GRADE_DATE.dt.year    <    2018].GRADE_DATE.dt.month,
bins=np.arange(1,14)-0.5, rwidth=0.9)
axes[2].set_title("Month")

f.tight_layout()
```

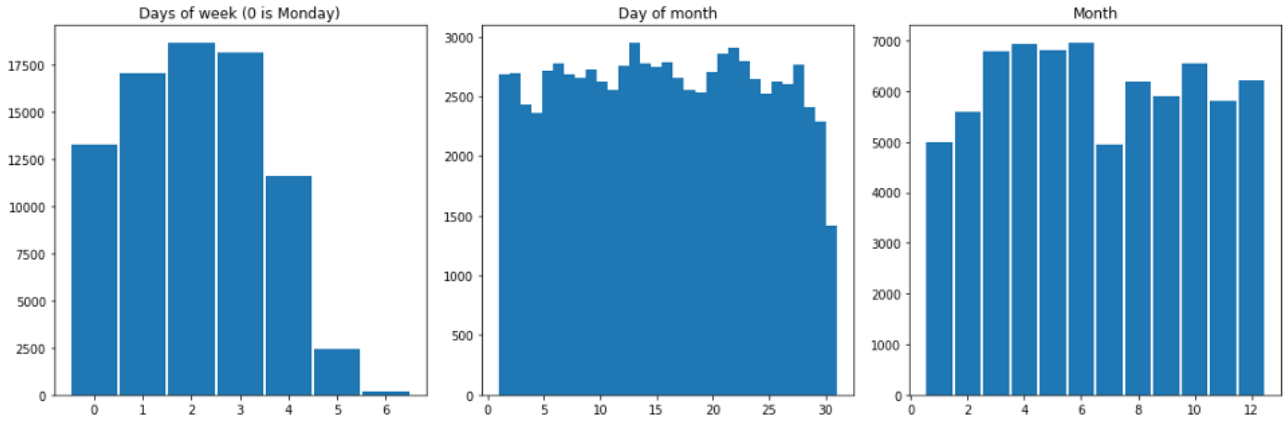


Рисунок 4.3 – Інспекція закладів

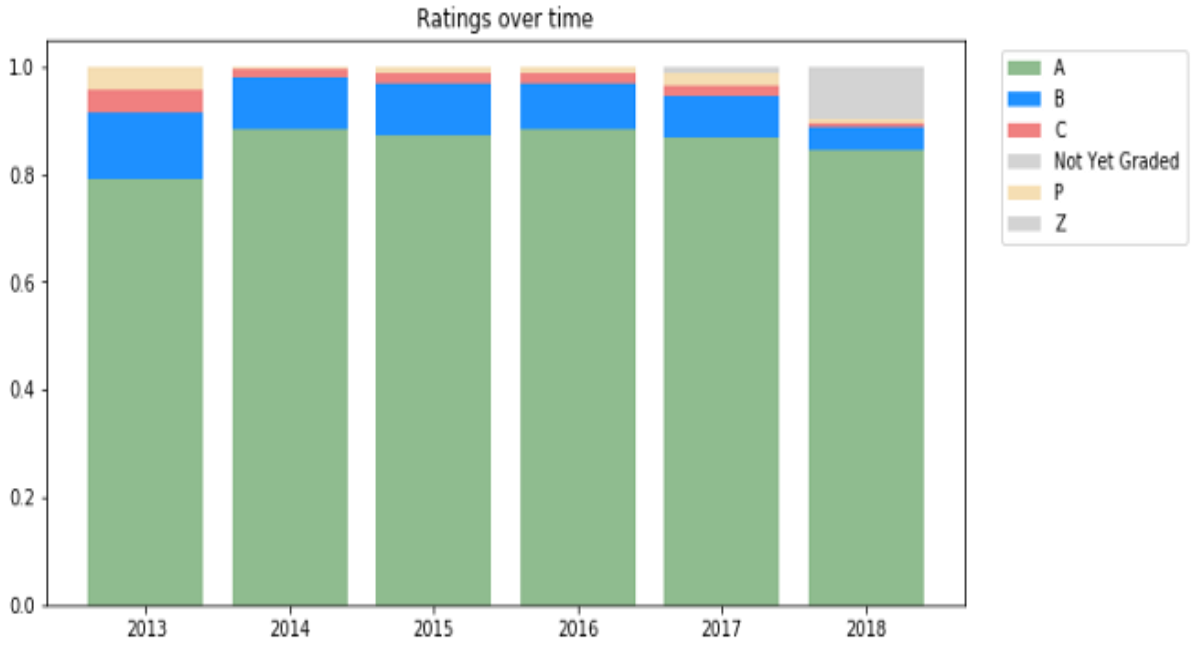


Рисунок 4.4 – Інфляція в класі протягом багатьох років

Перший рік мав лише ~ 80% рейтингу A, але розмір вибірки становить лише близько 100. Рейтинги B і C стабільно тримаються на рівні близько 12% і 4% відповідно. До цих пір в цьому році є багато Z або ще не оцінені.

```
f, ax = plt.subplots(figsize=(10,5))

df["YEAR"] = df.GRADE_DATE.dt.year

gb = df[["YEAR", "GRADE","SCORE"]].groupby(["YEAR","GRADE"]).agg("count")
gb["perc"] = gb / gb.sum(level=0)

for year in df.YEAR.unique():
    bottom = 0
    for grade in sorted(df.GRADE.unique()):
        perc      =      gb[(gb.index.get_level_values(0)      ==      year)      &
(gb.index.get_level_values(1) == grade)].perc
        if len(perc) > 0:
```

```
        if len(perc) > 0:
            perc = perc.values[0]
            ax.bar(year, perc, bottom=bottom, color=colors[grade])
            bottom += perc

ax.legend(sorted(df.GRADE.unique()), bbox_to_anchor=(1.25,1), loc="upper right")
ax.set_title("Ratings over time")
```

Набір даних має неузгоджені назви для опису кухні (наприклад, ручки 16 іноді класифікуються як американські та інші часи як морозиво). У цьому випадку використав найпоширеніший опис кухні. Також виключимо заклади з менш ніж 50 рейтингами в цілому. Нижча оцінка краща (менше порушень).

```
most_common_cuisine = df.groupby(["DBA"])["CUISINE"].agg(lambda x:
x.value_counts().index[0])
df = df.drop("CUISINE",axis=1)
df = df.join(most_common_cuisine, on="DBA")
# Calculate how many times each restaurant chain was graded
num_score_dba = df.groupby("DBA")[["SCORE"]].count()
num_score_dba.columns = ["NUM_SCORE_DBA"]
mean_score_dba = df.groupby("DBA")[["SCORE"]].mean()
mean_score_dba.columns = ["MEAN_SCORE_DBA"]
mean_score_dba = df.groupby("DBA")[["SCORE"]].median()
mean_score_dba.columns = ["MED_SCORE_DBA"]
max_score_dba = df.groupby("DBA")[["SCORE"]].max()
max_score_dba.columns = ["MAX_SCORE_DBA"]
min_score_dba = df.groupby("DBA")[["SCORE"]].min()
min_score_dba.columns = ["MIN_SCORE_DBA"]
std_score_dba = df.groupby("DBA")[["SCORE"]].std()
std_score_dba.columns = ["STD_SCORE_DBA"]
for field in [num_score_dba, mean_score_dba, min_score_dba, max_score_dba,
std_score_dba]:
    df = df.join(field, on="DBA")
# Update the dataframe with percentage breakdown of each grade
grade_dba = df.groupby(["DBA","GRADE"]).agg({'GRADE': 'count'})
grade_dba = grade_dba.groupby(level=0).apply(lambda x: x / float(x.sum()))
```

```
for grade in grade_dba.index.get_level_values("GRADE").unique():
    _grade_dba = grade_dba[grade_dba.index.get_level_values("GRADE") == grade]
    _grade_dba.index = _grade_dba.index.droplevel(level="GRADE")
```

CUISINE	DBA	GRADE_A	GRADE_C	MED_SCORE_DBA	MAX_SCORE_DBA	SCORE
American	APPLEBEE'S	1.00	0.00	8.00	13.00	92
	PANERA BREAD	1.00	0.00	10.00	13.00	60
	SHAKE SHACK	0.97	0.00	9.50	18.00	60
	BOSTON MARKET	0.96	0.01	10.00	20.00	71
	CHECKERS	0.94	0.00	10.00	26.00	111
	AMC THEATRES	0.91	0.02	9.50	45.00	56
Bagels/Pretzels	AUNTIE ANNE'S PRETZELS	1.00	0.00	7.00	13.00	64
CafÃ©/Coffee/Tea	STARBUCKS	0.99	0.00	5.00	27.00	954
	STARBUCKS COFFEE	0.99	0.00	6.50	17.00	104
	VIVI BUBBLE TEA	0.90	0.02	8.00	38.00	58
Caribbean	GOLDEN KRUST CARIBBEAN BAKERY & GRILL	0.88	0.03	10.00	46.00	182
Chicken	POPEYES LOUISIANA KITCHEN	0.94	0.00	9.00	32.00	257
	KFC	0.92	0.01	10.00	36.00	132
	KENNEDY FRIED CHICKEN	0.88	0.02	10.00	51.00	285
	CROWN FRIED CHICKEN	0.83	0.01	10.00	43.00	208
Donuts	DUNKIN DONUTS	1.00	0.00	7.00	13.00	89
	DUNKIN' DONUTS	0.96	0.01	8.00	41.00	1489
	DUNKIN' DONUTS, BASKIN ROBBINS	0.95	0.00	9.00	36.00	431
Hamburgers	WHITE CASTLE	1.00	0.00	7.00	13.00	76
	FIVE GUYS FAMOUS BURGERS AND FRIES	0.99	0.00	9.00	20.00	73
	WENDY'S	0.97	0.00	9.00	21.00	151
	MCDONALD'S	0.97	0.01	9.00	62.00	710
	BURGER KING	0.92	0.01	9.00	36.00	275
	BAREBURGER	0.90	0.01	9.50	27.00	70
Ice Cream, Gelato, Yogurt, Ices	CARVEL ICE CREAM	0.95	0.00	9.00	23.00	109
Mexican	CHIPOTLE MEXICAN GRILL	0.95	0.01	9.00	41.00	261
Pancakes/Waffles	IHOP	0.96	0.01	10.50	41.00	74
Pizza	LITTLE CAESARS	0.97	0.00	9.00	21.00	117
	PIZZA HUT	0.95	0.02	7.00	30.00	64
	PAPA JOHN'S	0.94	0.00	8.00	25.00	130
	DOMINO'S	0.81	0.05	10.00	53.00	306
Salads	CHOP'T	1.00	0.00	9.00	13.00	56
	JUST SALAD	0.96	0.00	9.00	21.00	70
Sandwiches	PRET A MANGER	0.99	0.00	8.00	14.00	154
	LE PAIN QUOTIDIEN	0.99	0.00	8.00	13.00	131
	SUBWAY	0.95	0.00	9.00	46.00	1079
Sandwiches/Salads/Mixed Buffet	AU BON PAIN	0.94	0.00	9.00	26.00	121
	POTBELLY SANDWICH WORKS	0.93	0.03	9.50	38.00	58
Soups & Sandwiches	HALE & HEARTY SOUP	0.98	0.00	8.50	18.00	86

Рисунок 4.5 – Ресторани, які мають найкращий рейтинг для своєї кухні

Рейтинги хороші, але вважаємо, що це очікується для великих послідовностей. Пам’ятаємо що високий бал поганий. Деякі заклади, такі як ніколи не отримували публічного класу, крім A. Інші групи, дуже непослідовні і часто погані (тільки 81% отриманих оцінок є As і 5% є Cs). Тим не менш, McDonald's має сумнівні причини мати найвищий бал 62 у великих ресторанах.

Один із способів розглянути це питання через швидкість зміни. Швидкість зміни - це ймовірність переходу від одного стану до іншого (або перебування в тому ж стані). На основі ймовірності перехід від одного рейтингу до іншого.

```
max_num_ratings = max(df.groupby("KEY").size())
columns = [idx for idx in range(max_num_ratings)]
columns.insert(0, "KEY")
df_rest = pd.DataFrame(columns=columns)

for key in df.KEY.unique():
    df_key = df[df.KEY == key]
    new_row = {col: "NA" for col in columns}
    new_row = {"KEY": key}
    for idx, (k, v) in enumerate(df_key.iterrows()):
        new_row[idx] = v.GRADE

    df_rest = pd.concat([df_rest, pd.DataFrame(new_row, index=[0])],
ignore_index=True)

df_rolls = pd.DataFrame(columns=[1,2])
for c1 in range(max_num_ratings - 2):
    c2 = c1 + 1
    df_rest_valid = df_rest[(~df_rest[c1].isna()) & (~df_rest[c2].isna())]
    df_roll = pd.concat([df_rest_valid[c1], df_rest_valid[c2]], axis=1)
    df_roll.columns = [1,2]
    df_rolls = pd.concat([df_rolls, df_roll], ignore_index=True)
```

```
states = ["A","B","C","P","Z"]
df_roll_rates = pd.DataFrame(np.zeros([5,5]), columns=states, index=states)
for s1 in states:
    for s2 in states:
        num_match = sum((df_rolls[1] == s1) & (df_rolls[2] == s2))
        num_all = sum(df_rolls[1] == s1)
        if num_all > 0:
            df_roll_rates.loc[s2,s1] = num_match / num_all

df_roll_rates.columns.name = "from"
df_roll_rates.index.name = "to"

df_roll_rates * 100
```

from / to	A	B	C	P	Z
A	89.49	71.40	68.35	74.43	0.00
B	6.50	18.66	18.89	18.86	0.00
C	1.37	4.16	5.88	3.10	0.00
P	1.44	2.48	3.40	0.00	0.00
Z	1.20	3.30	3.48	3.60	0.00

Рисунок 4.6 – Зміна рейтингу з плином часу для окремих ресторанів

Близько 90% часу рейтинг A знову отримає винагороду A на наступному огляді. Існує 71% ймовірність того, що ресторан B отримає нагороду A в наступному рейтингу, а 68% шансів, що ресторан C отримає нагороду A. Це повинно бути добре, хто їсть в ресторанах B і C.

Визначимо за класом, так як, що погано оцінені ресторани отримують огляд частіше.

```
#
df["NEXT_GRADE_DATE"] = df.GRADE_DATE.shift()

# Remove next grade date for most recent grades
df.loc[df.drop_duplicates("KEY").index,"NEXT_GRADE_DATE"] = pd.NaT
```

```
df["TIME_AT_GRADE"] = df.NEXT_GRADE_DATE - df.GRADE_DATE

f, ax = plt.subplots(figsize=(10,5))

for grade, color in colors.items():
    days_at_grade = df[(~df.TIME_AT_GRADE.isna()) & (df.GRADE == grade)].TIME_AT_GRADE.dt.days
    if days_at_grade.size > 0:
        ax.hist(days_at_grade, color=color, label=grade, alpha=0.5, bins=100, weights=np.zeros_like(days_at_grade) + 1. / days_at_grade.size)
ax.legend()
f.tight_layout()
```

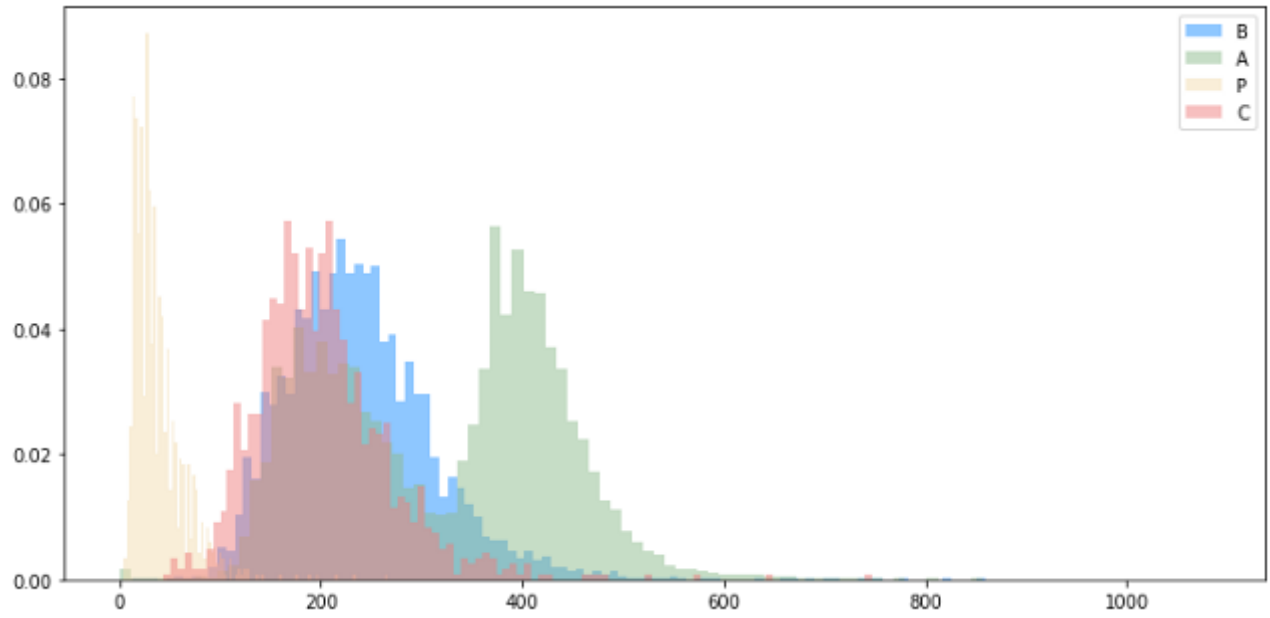


Рисунок 4.7 – Як часто ресторани залишаються в заданому класі

Це не відразу очевидно з вищесказаного, але оцінки A мають двомодальний розподіл.

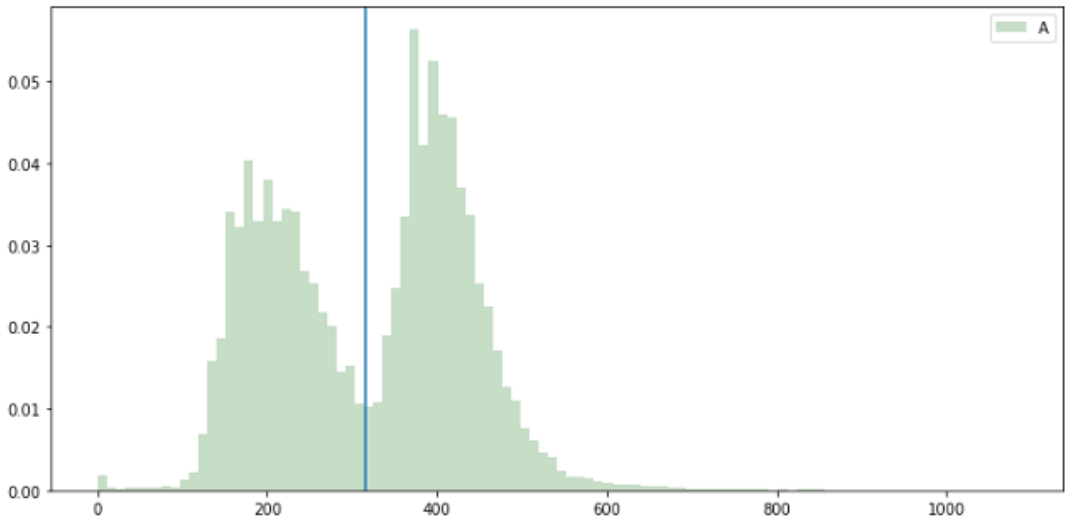


Рисунок 4.8 – Зміна рейтингу ресторанів класу A

Витратили деякий час, намагаючись з'ясувати, що є причиною, але не знайшли те задовольняє відповідь. Розподіл кількості днів з класом був послідовним протягом багатьох років і протягом декількох місяців. Вони, як правило, двомодальні над усіма типами кухні і через райони.

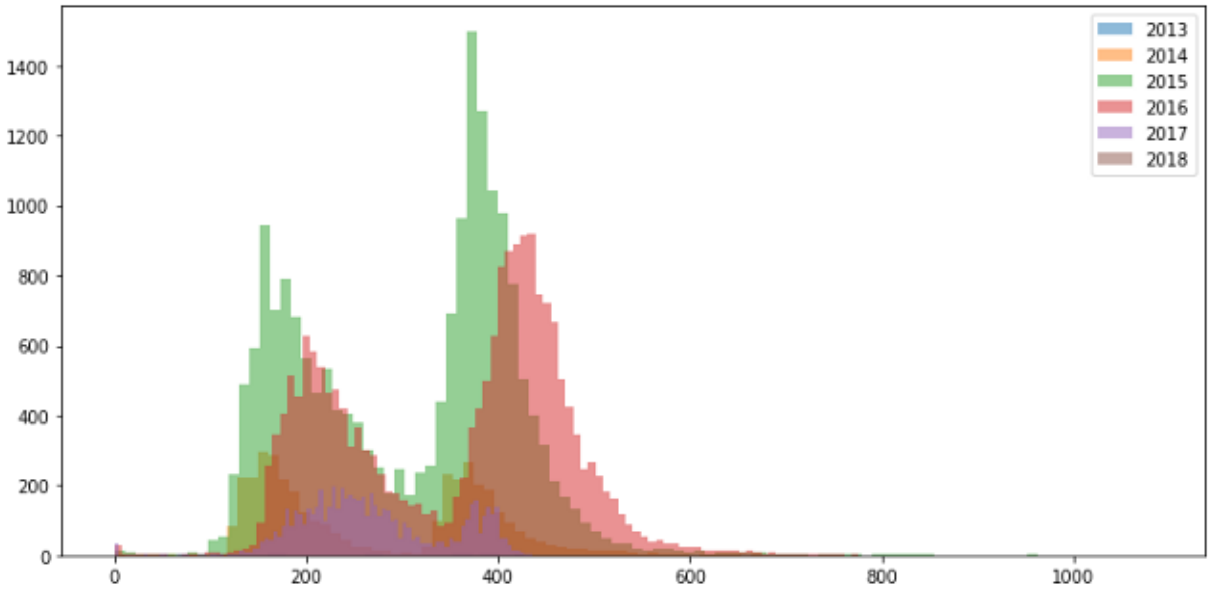


Рисунок 4.9 – Двомодальний розподіл рейтингу

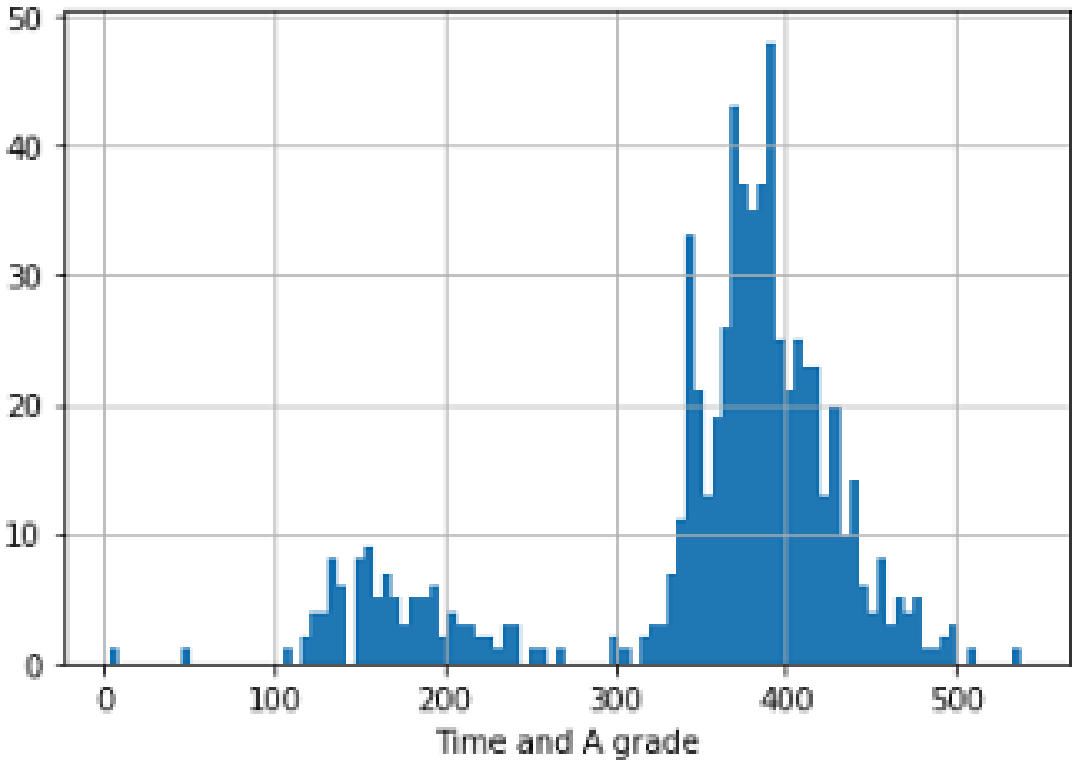


Рисунок 4.10 – Двомодальний розподіл Starbucks

Це дещо корелює з тим, як часто ресторан був оцінений, (наприклад, ресторани, такі як Starbucks, майже всі на вершині). Можливо не вистачає інформації, але припускаємо, що це пов'язано з деякими оцінками, які не надається. Як уже згадувалося раніше, якщо ресторан отримує клас, відмінний від B, оцінка йде непідтвердженою, і ресторани отримує ще один шанс на A протягом місяця. В іншому випадку на цій градації, яка має непідтверджені причини ресторану, щоб отримати огляд частіше. Це пояснює, чому добре оцінені ресторани отримують оцінки рідше. Навіть якщо подивитеся на ресторани, які ніколи не були удостоєні класу нижче A, двомодальний розподіл існує, але це може бути тільки тому, що класи, які не є A, непідтверджені.

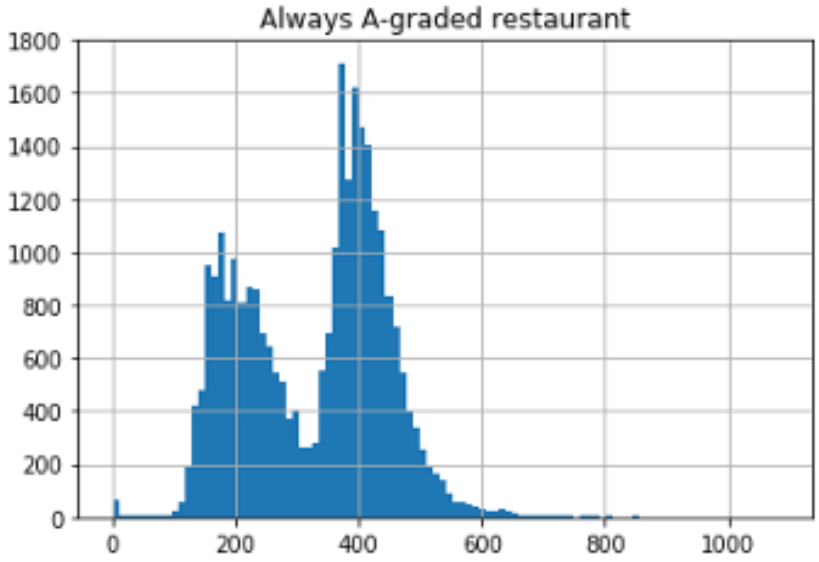


Рисунок 4.11 – Двомодальний розподіл закладів класу А

Навіть ті, хто має ідеальні оцінки, є двомодальними.

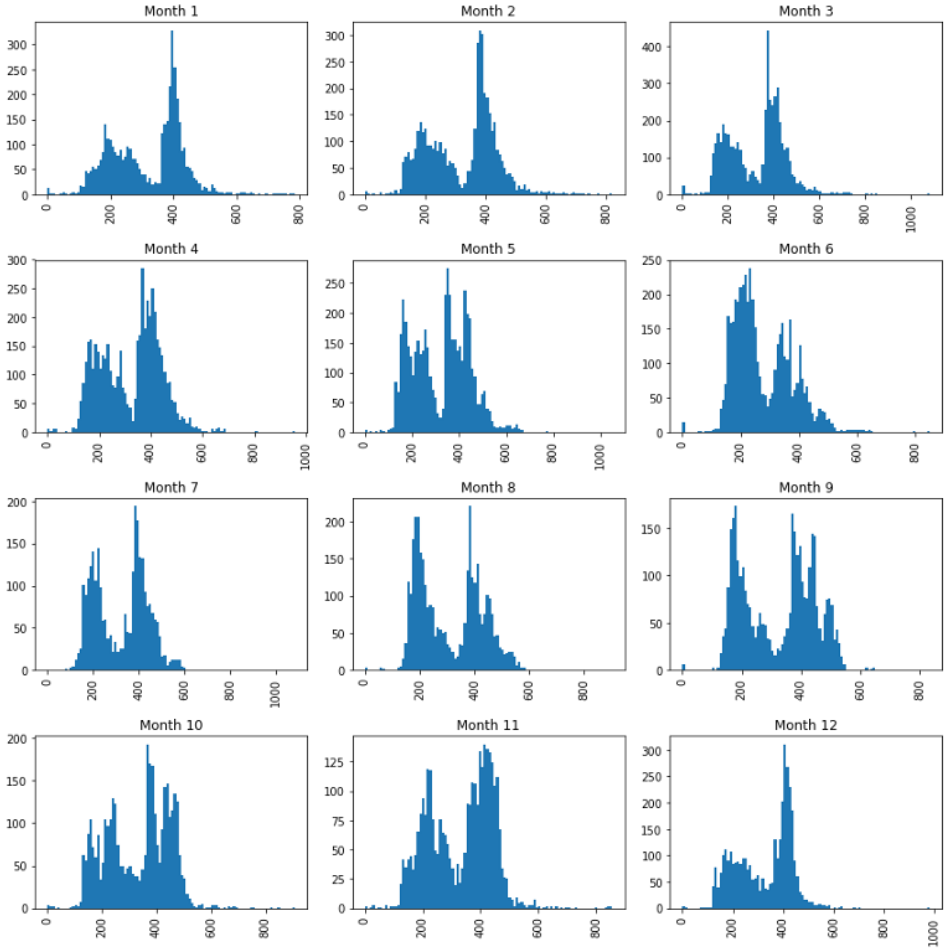


Рисунок 4.12 –Двомодальні протягом декількох місяців

4.4 Відгуки та оцінки інспекції

Клас свідчить про найвищі оцінки для здоров'я та безпеки, так що можна відчувати себе в безпеці про їжу там. Але не обов'язково те, що буде дуже хороша їжа і ввічливе обслуговування. Щоб дізнатися це, слід звернутися до відгуків ресторану. Для цього проекту подивимось на простий аналіз даних і візуалізацію оглядів ресторанів та даних інспекційних балів, щоб з'ясувати, чи є якась кореляція між ними. Дані також покажуть, які типи кухонь і які місця в Нью-Йорку, як правило, залучають більше оцінок.

В даний час бізнес-огляди, рейтинги та оцінки є прийняттям рішень для будь-якого бізнесу, щоб виміряти їх якість, популярність і майбутній успіх. Для ресторанів рейтинги, гігієнічні дані та чистота мають важливе значення. Популярний сайт для відгуків, Yelp, пропонує безліч індивідуальних оцінок для ресторанів. Департамент охорони здоров'я та психічної гігієни Нью-Йорка (DOHMH) щорічно проводить неанонсовані перевірки ресторанів. Вони перевіряють, чи відповідає харчова обробка, температура їжі, особиста гігієна працівників та контроль за ресторанами відповідно до гігієнічних норм. Процес підрахунку очок і оцінювання можна знайти проводиться досить ретельно.

Рейтинги ресторанів та інформація про місцезнаходження, що використовується в цьому проекті, походять від API Yelp. Інспекційні дані були завантажені з веб-сайту відкритих даних Нью-Йорка. Об'єднаємо yelp ресторани огляд даних і інспекційних даних і видалимо NA рядків, які немають ні оцінка інспекції або огляди. Також перепризначимо оцінку огляду в категоріях A, B і C, оскільки ця міра широко використовується і етикетка на ресторанах. Були й інші оцінки, в першу чергу P або Z, або якась версія класу в очікуванні, яку ігноруємо в нашому аналізі тут. Ресторани з рахунком від 0 до 13 очок заробляють A, ті, хто має від 14 до 27 балів, отримують B і ті, хто має 28 або більше C.

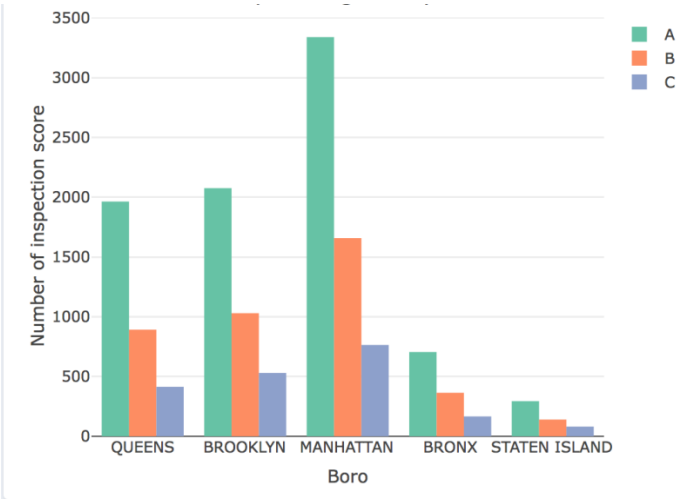


Рисунок 4.12 – Рейтинги інспекції від бюро

Рисунок 4.13 – Рейтинги інспекції по кухні

Дані показують, що A є найбільш часто призначена класом інспекції для ресторанів всіх типів у всіх місцях. Намалюємо графіки, щоб візуалізувати оцінки інспекції та рейтинги на основі типу бюро та кухні.

Що стосується місця розташування, то ця ділянка показує, що Манхетен має найбільшу кількість ресторанів з усіма оцінками в порівнянні з іншими. Це

очевидно, оскільки він має найбільшу кількість ресторанів в цілому. Стейтен-Айленд має найнижчу кількість ресторанів з оцінками A, B і C серед усіх.

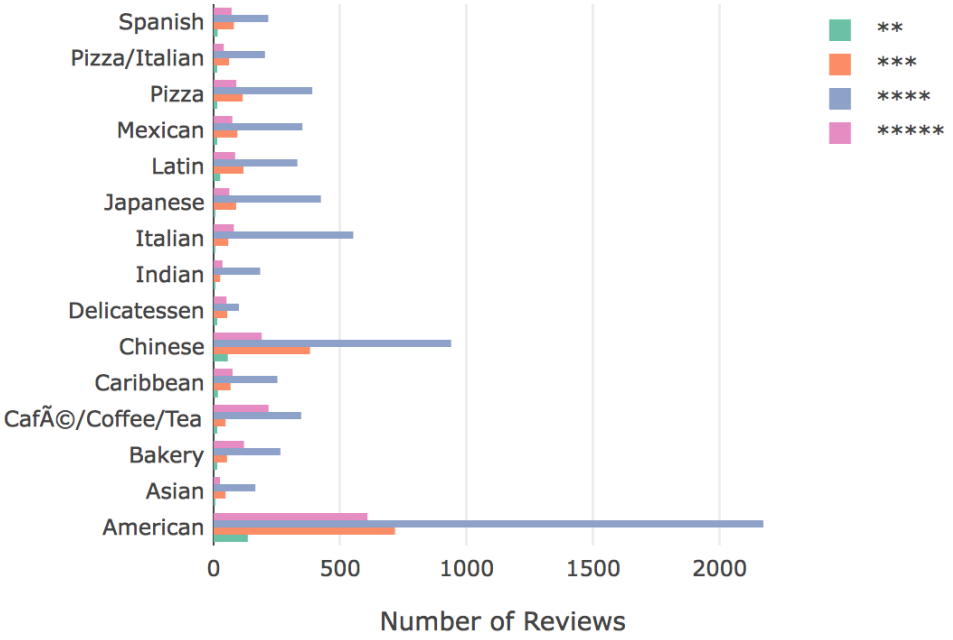


Рисунок 4.14 – Рейтинги по кухні

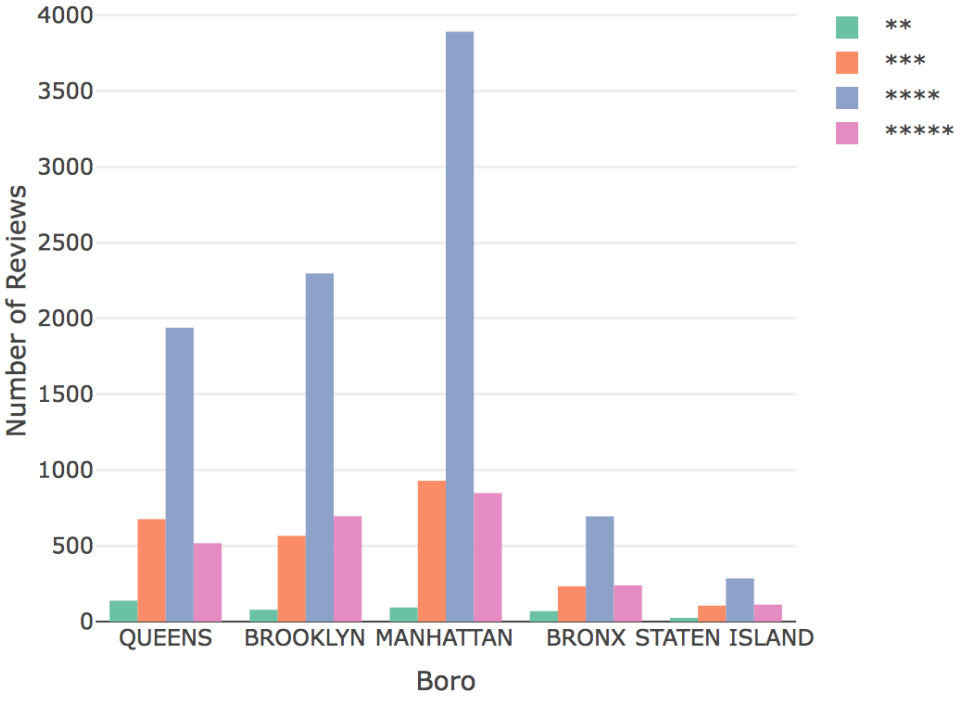


Рисунок 4.15 – Рейтинги по районам

Що стосується типів кухні, то в розрізі кухонь показані перші 15 ресторанів з найбільшою кількістю підрахунків на основі кухні. Це свідчить про те, що американська кухня має найбільшу кількість класів А в порівнянні з іншими. Це свідчить про те, що американські ресторани більше зосереджені на гігієніу та чистоті в порівнянні з іншими типами ресторанів.

Сюжет огляду вказує на те, що більшість ресторанів досягають найвищого рейтингу 4 зірок. Знову ж таки, Манхеттен має найбільшу кількість ресторанів з рейтингом чотири зірки в той час як Стейтен-Айленд має найнижчу кількість ресторанів з високим рейтингом. Це також показує, що майже всі райони мають низьку кількість 2-зіркових ресторанів. Крім того, огляди кухні сюжет вказує на те, що американська кухня, як правило, мають найвищий рейтинг в порівнянні з іншими кухнями. Причинами може бути більше американських ресторанів під цією категорією, то інші.

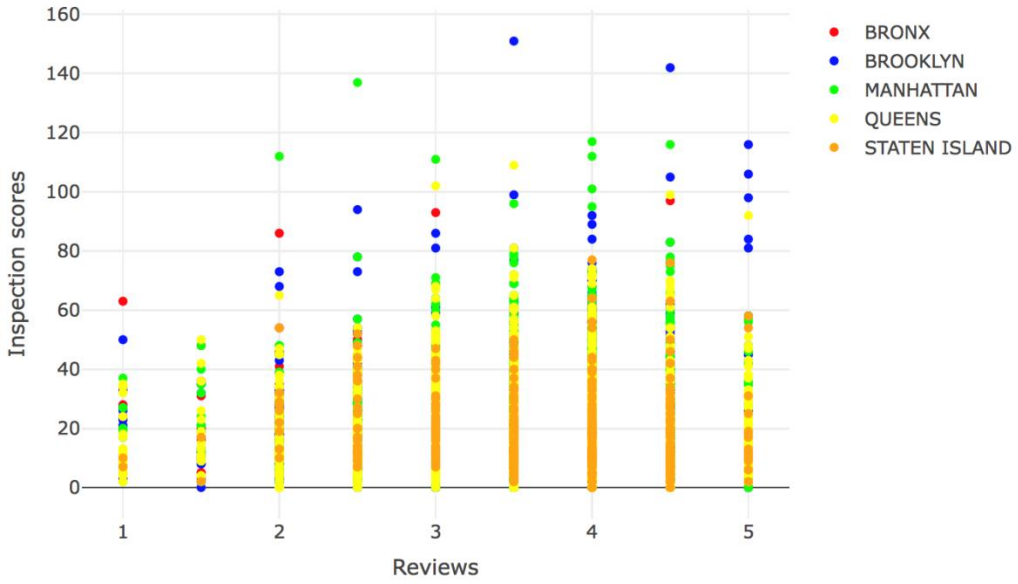


Рисунок 4.16 – Розподіл по районам

Точкові дані показують взаємозв'язок між оцінкою огляду та рейтингом. Це вказує на те, що немає прямої чіткої кореляції між двома змінними. Це досить часто для ресторану з оцінкою класу С для досягнення оцінки 4-5 зірок в огляді.

Також є можливість знайти ряд оцінок класу А для ресторанів, які мають тільки 1-2 зірки. Це може бути тому, що до тих пір, поки їжа смачна, люди будуть добре оцінювати ресторан, тому що вони не приділяють дуже багато уваги чистоті та гігієнічним питанням.

Розкид даних також показує, що хоча деякі ресторани підтримують дуже високий рівень чистоти і гігієнічних умов харчування, вони не отримують хороших оцінок, які можуть бути пов'язані з поганим обслуговуванням або менше, ніж смачна їжа.

Можемо зробити подальший аналіз по обидва боки ресторанів, аналізуючи коментарі оглядів і намагаючись з'ясувати, чому деякі ресторани мають хороші відгуки, але низький показник огляду і навпаки. Для цього потрібні додаткові дані про відгуки коментарів та подальший аналіз за допомогою NLP.

Рисунок 4.17 – Бали по районам

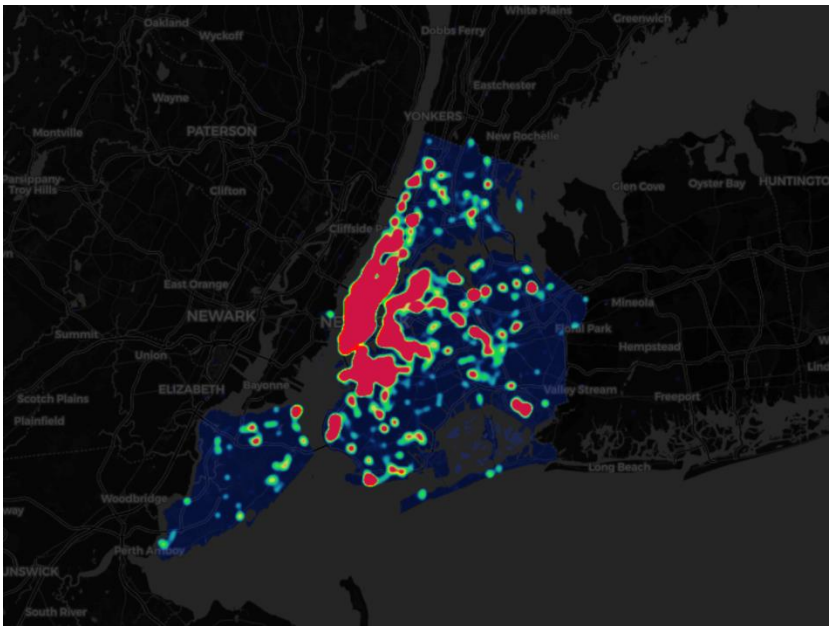


Рисунок 4.18 – Кластери зосередження закладів

Висновки до розділу 4

Кластерна карта ресторанів Нью-Йорка допомагає візуалізувати локації та фільтрувати ресторани на основі типів кухні. Точка позначає оцінки і включає в себе описи вибраних ресторанів. На тепловій карті видно щільність ресторанів на основі вибору вибору кухні. Це вказує на те, яка територія має більшу кількість ресторанів. Це може бути корисним для ділових людей, щоб приймати обґрунтовані рішення про те, де відкрити нові ресторани на основі типів ресторанів, які вже є.

Нарешті, ця програма може бути корисною для людей, щоб фільтрувати базу даних на кухню, рейтингами та класом інспекції. Люди хочуть піти поїсти з конкретними критеріями можуть фільтрувати ресторани і відвідувати свої улюблені ресторани на основі найвищих оцінок як для оцінок, так і для класів інспекції.

Загальні висновки

Найважливішим внеском роботи є експерименти з даними в соціальних мережах, а також методи глибокого навчання, які використовуються для обробки цих даних. Відгуки від Yelp дуже корисні для цього типу досліджень, що стосуються громадської гігієни ресторанів. Користувачі додатків Yelp знаходяться переважно в містах. Додатковою перевагою набору даних є той факт, що всі відгуки англійською мовою яка є найбільш поширеною. Механізм ваги, впроваджений у нейронній мережі, дозволяє створити модель для атрибута вага значимості на трьох рівнів (тобто, на рівні огляду, рівень речення, і рівень слова). Найбільшою перевагою є можливість моделі приділяти параметр значимості до окремих слів або набору слів, і в той же час орієнтація на можливість інтерпретувати результати візуалізації ваги. Результати також дають цікаве уявлення про те, як нейронні мережі можуть бути поліпшені, коли обидві частини збалансовані, результати будуть кращими.

Модель, є хорошим підходом до використання з усіма типами проблем з визначенням тексту, а не тільки з онлайн-відгуками. Є можливість адаптувати модель до інших повсякденних ситуацій надання оцінки.

Перелік посилань

1. Yoav Goldberg. A primer on neural network models for natural language processing. Journal of Arti cial Intelligence Research, 57, 2016.

2. Yoon Kim. Convolutional neural networks for sentence класифікації. arXiv preprint arXiv:1408.5882v2, 2014.

3. Adam Sadilek, Henry Kautz, Lauren DiPrete, Brian Labus, Eric Portman, Jack Teitel, and Vin-cent Silenzio. Deploying nemesis: Preventing foodborne illness by data mining social media. Proceedings of the Twenty-Eighth Association for the Advancement of Arti cial Intelligence Conference on Innovative Applications (IAAI-16), 2016.

4. Rie Johnson and Tong Zhang. Deep pyramid convolutional neural networks for text categoriza-tion. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2017.

5. Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, lex Smola, and Eduard Hovy. Hierarchical attention networks for document класифікації. In Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics, 2016.

6. Minmin Chen. E cient vector representation for documents through corruption. arXiv preprint arXiv:1707.02377, 2017.

7. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schutze. An Introduction to Infor-mation Retrieval. Cambridge University Press, 2009.

8. Tomas Mikolov, Kai Chen, Greg Corrado, and Je rey Dean. E cient estimation of word repre-sentations in vector space. CoRR, abs/1301.3781, 2013.

9. T. Chai and R. R. Draxler. Root mean square error or mean absolute error arguments against avoiding rmse in the literature. Geoscienti c Model Development Discussions, 2014.

10. Mikel Joaristi, Edoardo Serra, and Francesca Spezzano. Identifying health-violating restaurants with online reviews. In Proceedings of the IEEE-ACM International Conference on Advances in Social Networks Analysis and Mining, 2016.

11. Jun Seok Kang, Polina Kuznetsova, Michael Luca, and Yejin Choi. Where not to eat? improving public policy by predicting hygiene inspections using online reviews. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013

12. Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. arXiv preprint arXiv:1705.02364, 2017.

13. Victoria Zayats and Mari Ostendorf. Conversation Modeling on Reddit Using a Graph-Structured LSTM. Empirical Methods in Natural Language Processing., 2018.

14. Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. CoRR, abs/1412.6980, 2014.

15. Shashank Uppoor and Shreyas Pathre Balakrishna. Predicting restaurant health inspection penalty score from yelp reviews. Association for Computing Machinery ISBN 978-1-4503-2138-9, 2016.

Додатки

```
f, ax = plt.subplots(figsize=(10,5))

df["YEAR"] = df.GRADE_DATE.dt.year

gb = df[["YEAR", "GRADE","SCORE"]].groupby(["YEAR","GRADE"]).agg("count")
gb["perc"] = gb / gb.sum(level=0)

for year in df.YEAR.unique():
    bottom = 0
    for grade in sorted(df.GRADE.unique()):
        perc = gb[(gb.index.get_level_values(0) == year) &
(gb.index.get_level_values(1) == grade)].perc
        if len(perc) > 0:
            perc = perc.values[0]
            ax.bar(year, perc, bottom=bottom, color=colors[grade])
            bottom += perc

ax.legend(sorted(df.GRADE.unique()), bbox_to_anchor=(1.25,1), loc="upper right")
ax.set_title("Ratings over time")
```

```
df["TIME_AT_GRADE"] = df.NEXT_GRADE_DATE - df.GRADE_DATE

f, ax = plt.subplots(figsize=(10,5))

for grade, color in colors.items():
    days_at_grade = df[(~df.TIME_AT_GRADE.isna()) & (df.GRADE ==
grade)].TIME_AT_GRADE.dt.days
    if days_at_grade.size > 0:
        ax.hist(days_at_grade, color=color, label=grade, alpha=0.5, bins=100,
weights=np.zeros_like(days_at_grade) + 1. / days_at_grade.size)
ax.legend()
f.tight_layout()
```

```
most_common_cuisine = df.groupby(["DBA"])["CUISINE"].agg(lambda x:
x.value_counts().index[0])
df = df.drop("CUISINE",axis=1)
df = df.join(most_common_cuisine, on="DBA")
# Calculate how many times each restaurant chain was graded
num_score_dba = df.groupby("DBA")[["SCORE"]].count()
num_score_dba.columns = ["NUM_SCORE_DBA"]
mean_score_dba = df.groupby("DBA")[["SCORE"]].mean()
mean_score_dba.columns = ["MEAN_SCORE_DBA"]
mean_score_dba = df.groupby("DBA")[["SCORE"]].median()
mean_score_dba.columns = ["MED_SCORE_DBA"]
max_score_dba = df.groupby("DBA")[["SCORE"]].max()
max_score_dba.columns = ["MAX_SCORE_DBA"]
min_score_dba = df.groupby("DBA")[["SCORE"]].min()
min_score_dba.columns = ["MIN_SCORE_DBA"]
std_score_dba = df.groupby("DBA")[["SCORE"]].std()
std_score_dba.columns = ["STD_SCORE_DBA"]
for field in [num_score_dba, mean_score_dba, min_score_dba, max_score_dba,
std_score_dba]:
    df = df.join(field, on="DBA")
# Update the dataframe with percentage breakdown of each grade
grade_dba = df.groupby(["DBA","GRADE"]).agg({'GRADE': 'count'})
grade_dba = grade_dba.groupby(level=0).apply(lambda x: x / float(x.sum()))


states = ["A","B","C","P","Z"]
df_roll_rates = pd.DataFrame(np.zeros([5,5]), columns=states, index=states)
for s1 in states:
    for s2 in states:
        num_match = sum((df_rolls[1] == s1) & (df_rolls[2] == s2))
        num_all = sum(df_rolls[1] == s1)
        if num_all > 0:
            df_roll_rates.loc[s2,s1] = num_match / num_all

df_roll_rates.columns.name = "from"
df_roll_rates.index.name = "to"

df_roll_rates * 100
```

```
max_num_ratings = max(df.groupby("KEY").size())
columns = [idx for idx in range(max_num_ratings)]
columns.insert(0, "KEY")
df_rest = pd.DataFrame(columns=columns)

for key in df.KEY.unique():
    df_key = df[df.KEY == key]
    new_row = {col: "NA" for col in columns}
    new_row = {"KEY": key}
    for idx, (k, v) in enumerate(df_key.iterrows()):
        new_row[idx] = v.GRADE

    df_rest    =    pd.concat([df_rest,    pd.DataFrame(new_row,    index=[0])],
ignore_index=True)

df_rolls = pd.DataFrame(columns=[1,2])
for c1 in range(max_num_ratings - 2):
    c2 = c1 +  1
    df_rest_valid = df_rest[(~df_rest[c1].isna()) & (~df_rest[c2].isna())]
    df_roll = pd.concat([df_rest_valid[c1], df_rest_valid[c2]], axis=1)
    df_roll.columns = [1,2]
    df_rolls = pd.concat([df_rolls, df_roll], ignore_index=True)
```

УДК 004.4

Кузьмінський М. С., Манзюк Е. А.

Хмельницький національний університет

СИСТЕМА ПРОГНОЗУВАННЯ ПРОДАЖІВ СЕРВІСНИХ ПОСЛУГ В СИСТЕМАХ ОБСЛУГОВУВАННЯ

Розроблено інформаційну систему аналізу систему прогнозування завантаженості та продажів сервісних послуг. Ця система дозволяю враховувати сучасні тенденції на ринку та робити прогнози використовуючи методи та підходи машинного навчання. Найважливішим внеском роботи є експерименти з даними в соціальних мережах, а також методи глибокого навчання, які використовуються для обробки цих даних.

The information system of the analysis of system of forecasting of loading and sales of service services is developed. This system allows you to take into account current market trends and make predictions using the methods and approaches of machine learning. The most important contribution of the work is the experiments with data in social networks, as well as the methods of deep learning that are used to process this data.

Систематичний збір даних інспекції охорони здоров'я має важливе значення для оцінки ресторану, а також є основою для запобігання ряду харчових захворювань. Для цих та інших цілей інспектори повинні писатися санітарні звіти, що містять місцезнаходження та інші дані ресторану, разом з текстовими описами з причин, які в є основі оцінки ресторану, і оцінки інспекції здоров'я.

Аналіз ресторанів також передбачає рекомендації. Yelp - це платформа, яка дозволяє людям класифікувати та давати свою думку про місця проведення (наприклад, такі як бари та ресторани), які вони відвідали. Основна мета Yelp полягає в тому, щоб інформувати нових клієнтів про місце, яке вони ніколи не відвідували та не має хорошої оцінки класифікації чи має, на основі інформації, наданої іншими клієнтами. Зокрема, автоматична класифікація даних Yelp, пов'язаних з санітарними звітами, дозволить людям краще знати, чи є має якесь місце хорошу оцінку чи ні, і вирішити, чи заслуговує це місце свого часу і грошей.

Класифікація тексту за допомогою контрольованого навчання вимагає використання методів представлення тексту, щоб бути входом до алгоритмів навчання. Загальний підхід передбачає перетворення текстових документів у вектори, які їх представляють. Кожна позиція одного такого вектора повторно визначає частоту зразка слова в документі.

Можемо використати векторну модель (VSM) як алгебраїчну модель, яка представляє текстову інформацію як вектори. Крім того, складові цих векторів представляють актуальність кожного терміну та вмість у словнику, який

використовується в збірнику документів. Одним з можливих методів, які можемо використовувати для зважування термінів є TF-IDF (Частота терміну до зворотної частоти документів). Частота терміну (TF) визначає, скільки разів термін x присутній в документі d. Документ, в якому згадується даний термін частіше, має більше спільного з цим терміном, і тому повинен отримувати більш високий бал. Тому призначаємо вагу для кожного терміну в документі, який залежить від кількості входжень терміну в документі.

Рейтинги ресторанів та інформація про місцезнаходження, що використовується в цьому проекті, походять від API Yelp. Інспекційні дані були завантажені з веб-сайту відкритих даних Нью-Йорка. Об'єднаємо yelp ресторани огляд даних і інспекційні дані і видалити NA рядки, які немають ні оцінки інспекції або відгуки. Також перепризначимо оцінку огляду в категоріях A, B і C, оскільки ця міра широко використовується як етикетка на ресторанах. Були й інші оцінки, в першу чергу P або Z, або якась версія класу з очікування, яку ігноруємо в нашому аналізі тут. Ресторани з рахунком від 0 до 13 очок заробляють A, ті, хто має від 14 до 27 балів, отримують B і ті, хто має 28 або більше C.

Нарешті, ця програма може бути корисною для людей, щоб фільтрувати базу даних на кухню, рейтинги та клас інспекції. Люди хочуть піти поїсти з конкретними критеріями можуть фільтрувати ресторани і відвідувати свої улюблені ресторани на основі найвищих оцінок як для оцінок, так і для класів інспекції.

Результати показують, що методи глибокого навчання можуть бути використані для класифікації ресторанів з точки зору санітарних проблем, а завдання з класифікацією тексту дуже надійні та придатні для вирішення цих завдань. Звичайно, це складно для боротьби з підробленими відгуками, але ця робота не про те, що потрібно просто вірити в коментарі клієнтів, а довести, що методи класифікації можуть допомогти оцінювачам досягти безпеки закладів з використанням оцінки інспекції та відгуків про заклади.

Модель машинного навчання є підходом для використання з усіма типами текстових документів, а не тільки з онлайн-відгуками. Це говорить про можливість адаптувати модель до інших практичних завдань з використанням текстової інфомації та має вагоме практичне застосування.

Перелік посилань

1. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schutze. An Introduction to Information Retrieval. Cambridge University Press, 2009.
2. Tomas Mikolov, Kai Chen, Greg Corrado, and Je rey Dean. E cient estimation of word representations in vector space. CoRR, abs/1301.3781, 2013.
3. T. Chai and R. R. Draxler. Root mean square error or mean absolute error arguments against avoiding rmse in the literature. Geoscienti c Model Development Discussions, 2014.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

МАГІСТЕРСЬКА РОБОТА

Система прогнозування продажів сервісних послуг в системах обслуговування

Розробив ст. гр. КНм-19-1:
Кузьмінський М.С.

Хмельницький - 2020

В магістерській роботі було розроблено інформаційну систему аналізу систему прогнозування завантаженості та продажів сервісних послуг. Ця система дозволяє враховувати сучасні тенденції на ринку та робити прогнози використовуючи методи та підходи машинного навчання.

Метою дослідження є розробка інформаційної системи аналізу та дослідження впливу основних факторів на функціонування галузі сервісних послуг.

Для досягнення зазначеної мети поставлені наступні задачі:

- показати, що використання методів машинного навчання дозволяє поліпшити роботу системи сервісних послуг;
- провести дослідження впливу ознак на необхідні параметри інформаційної системи;
- провести аналіз відомих методів та підходів в предметній області дослідження

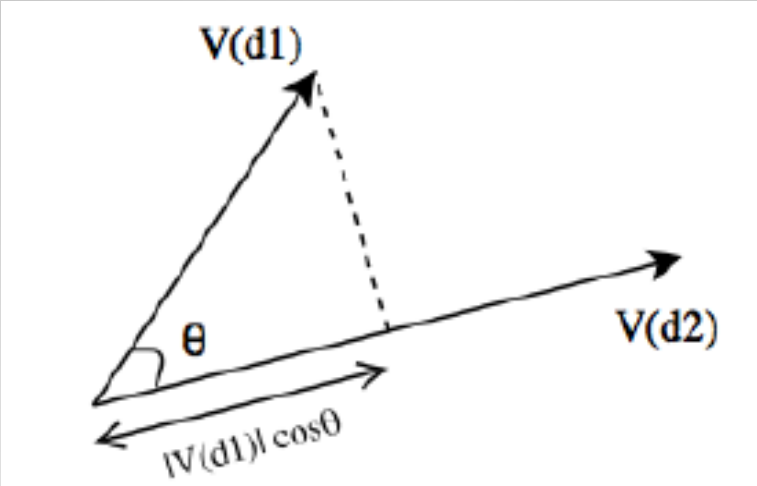
Об'єктом дослідження є методи аналізу та прогнозування отримання інформації та її обробки.
Предметом дослідження є груповані данні за показниками ефективності прикладної області дослідження.

Оцінка запропонованих архітектур машинного навчання є основним внеском цього дослідницького проекту. Набір даних, який використовується в цій роботі, розглядався за кількома наборами, таким чином використовуючи відгуки про регіони і враховуючи той факт, що люди можуть мати різні стандарти, які можуть впливати на рейтинг.

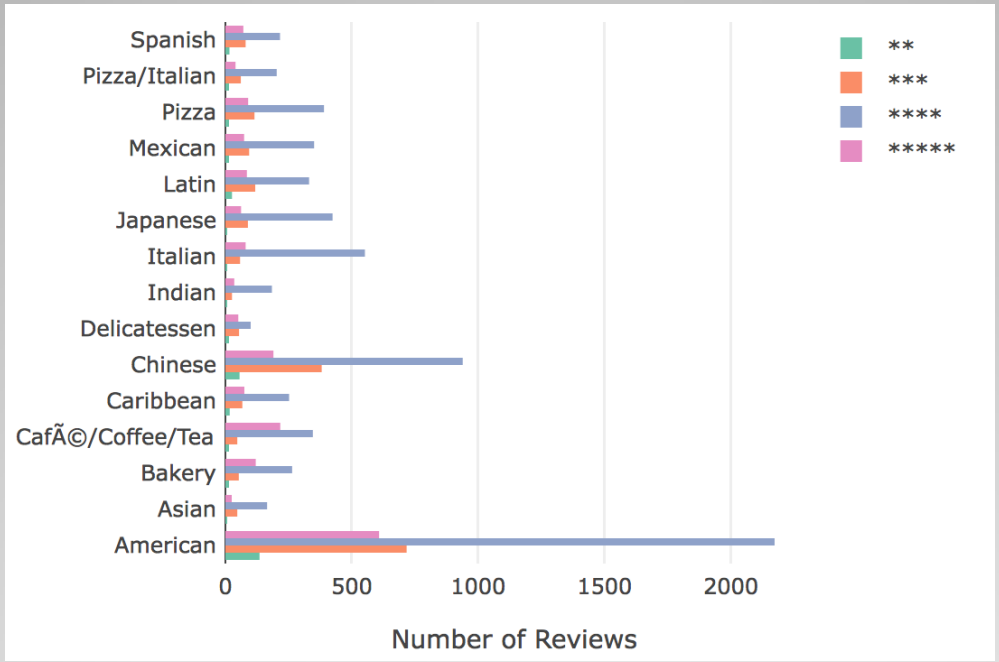
Під час пошуку всіх експериментальних таблиць (точність, час навчання/генерації) під час використання набору даних модель Doc2VecC перевершує в порівнянні з деякими іншими моделями, такими як Bag-of-words або Word2VecC, оскільки вона має найнижчий показник похибки та найкращу точність (на підмножині тестового набору Semantic-Syntactic World Relationship). Єдине питання щодо цього методу полягає в тому, що час навчання трохи вище, ніж очікувалося. Важливою частиною Doc2VecC є узагальнення даних, яке враховує словам, які не часто зустрічаються і полегшує представлення документів із середнім показником використання вивченого слова.

Результати показують, що методи машинного навчання можуть бути використані для класифікації ресторанів з точки зору санітарних проблем, а результати з використанням тексту досить надійні. Однак, це складно працює для боротьби з підробленими відгуками. Однак робота не спрямована на те, щоб перевірити коментарі клієнтів, а щоб довести, що методи класифікації можуть допомогти оцінювачам отримати хороші результати.

$$\text{sim}(d1, d2) = \frac{\text{V(d1)}.\text{V(d2)}}{\|\text{V(d1)}\| \times \|\text{V(d2)}\|}$$



Проекція вектора V (d1) у вектор V (d2) (Косинусна міра подібності)



Рейтинги по кухні, тобто розподіл по уподобанням клієнтів

Висновки

Внеском роботи є експерименти з даними в соціальних мережах, а також методи машинного навчання, які використовуються для обробки цих даних. Відгуки досить корисні для цього типу досліджень, що стосуються громадської гігієни ресторанів та взагалі рейтинг. Додатковою перевагою набору даних є той факт, що всі відгуки англійською мовою яка є найбільш поширеною. Механізм ваги, впроваджений у нейронній мережі, дозволяє створити модель для атрибута вага значимості на трьох рівнів (тобто, на рівні огляду, рівень речення, і рівень слова). Найбільшою перевагою є можливість моделі приділяти параметр значимості до окремих слів або набору слів, і в той же час орієнтація на можливість інтерпретувати результати візуалізації ваги. Результати також дають цікаве уявлення про те, як вони можуть бути поліпшені, коли частини збалансовані, результати будуть кращими.

Дякую за увагу

Mon Nov 30 16:54:02 EET 2020, Петровський Сергій Степанович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 4%**

	Документ		Сумарний збіг по Базі Даних	
ID: 81733 Назва: Система прогнозування продажів сервіснис послуг в системах обслуговування Додано в БД: 2020-11-30 Автора: Кузьмінський Михайло Сергійович Керівники: Манзюк Е.А. Консультанти: Опоненти:	Символи	Лексеми	Символи	Лексеми
	63662	531	696 (1%)	7 (1%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РІШЕННЯ КАФЕДРИ **КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ** ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: **Система прогнозування продажів сервісних послуг в системах обслуговування**
Автор: **Кузьмінський М. С.**
Спеціальність: **122 Комп'ютерні науки**
Науковий керівник: **к.т.н. доцент Манзюк Е.А.**

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту.	**відповідає**
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	-
3	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	-
4	Інше:	-

Підтвердження: Виявленні запозичення не є плагіатом так як є широко вживаними поняттями предметної області і складають 2.1%.

01.11.2020
Дата

Підпис керівника

Підпис завідувача кафедри

Хмельницький національний університет
Факультет програмування та комп'ютерних і телекомунікаційних систем
Кафедра комп'ютерних наук та інформаційних технологій

ВІДГУК ОПОНЕНТА

на дипломну роботу магістра

Магістра *гр. КНм-19-1* Кузьмінського Михайла Сергійовича

На тему: Система прогнозування продажів сервісних послуг в системах обслуговування

1. Актуальність і значення теми

Розроблено інформаційну систему аналізу прогнозування завантаженості та продажів сервісних послуг. Ця система дозволяє враховувати сучасні тенденції на ринку та робити прогнози використовуючи методи та підходи машинного навчання.

2.Оцінка якості та достовірності проведених досліджень

Робота проведена із належним забезпеченням якісної оцінки. Експериментальні дослідження проведені із забезпеченням необхідних показників достовірності.

3. Оцінка запропонованих заходів та пропозицій, практичної цінності та ефективності

Оцінка запропонованих архітектур машинного навчання є основним внеском цього дослідницького проекту. Набір даних, який використовується в цій роботі, розглядався за кількома наборами, таким чином використовуючи відгуки про регіони і враховуючи той факт, що люди можуть мати різні стандарти, які можуть впливати на рейтинг.

4. Загальний висновок та оцінка

Показано важливість області дослідження та актуальність вибраного напрямку. Вказано основні критерії, які можна покласти в основу подальших досліджень та окреслено оціночні критерії щодо майбутніх результатів. За своєю структурою, практичними цінностями, поставленій меті та вирішеними задачами робота відповідає вимогам вищої школи і вимогам, що пред'являються до освітньо-кваліфікаційного рівня «магістр».

Робота заслуговує на оцінку « задовільно ».

Опонент Мацевочук Н.Я., к.т.н., доцент кафедри технічного забезпечення та агроінженерії.