

**ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ФРАНКА**

Факультет прикладної математики та інформатики
кафедра дискретного аналізу та інтелектуальних систем

ДИПЛОМНА РОБОТА

на тему:

Deep learning в технологіях комп'ютерного зору

Студента 4 курсу, групи ПМі-45
напряму підготовки Августюк Р. С.

Керівник ас. Коркуна Н. М.

Національна шкала _____

Кількість балів: _____

Оцінка: ECTS _____

Львів – 2023

Зміст

ВСТУП	3
СУТЬ КУРСОВОЇ РОБОТИ	4
Розділ 1. Комп'ютерний зір та нейронні мережі	4
1.1 Комп'ютерний зір	4
1.2 Архітектура штучної нейронної мережі.....	5
1.3 Математична модель штучної нейронної мережі.....	7
1.4 Згорткова нейронна мережа.....	8
1.5 Рекурентна нейронна мережа	9
1.6 Згортково Рекурентна нейронна мережа	11
Розділ 2. Модель згорткової нейронної мережі	12
2.1 Архітектура згорткової нейронної мережі.....	12
Розділ 3. Модель LSTM рекурентної нейронної мережі	15
3.1 Архітектура LSTM.....	15
3.2 Модель LSTM	16
Розділ 4. Модель Згортково рекурентної нейронної мережі	19
4.1 Архітектура Згортково рекурентної нейронної мережі.....	19
Розділ 5. Програмна реалізація	22
5.1 Обладнання та програмне забезпечення	22
5.2 Опис бази даних	22
5.3 Опис програмної реалізації	22
Висновки	27
Список використаної літератури	28

ВСТУП

Однією з галузей, яка зараз розвивається найшвидше, є комп'ютерний зір. Використання глибокого навчання відкриває нові шляхи та можливості для створення систем автоматизованого розпізнавання об'єктів, аналізу зображень та візуальної інтерпретації.

Розробка потужних інтелектуальних моделей, які здатні автоматично виконувати складні завдання обробки зображень, є одним із ключових аспектів використання глибокого навчання в технологіях комп'ютерного зору. Глибокі нейронні мережі, особливо ті, що побудовані на архітектурі згорткової нейронної мережі, показали свою успішність у вирішенні проблем візуального аналізу, таких як класифікація об'єктів, локалізація об'єктів, визначення орієнтації та розпізнавання складних зображень. При використанні з великими наборами даних ці моделі можуть досягти надзвичайного рівня точності.

Глибоке навчання в технологіях комп'ютерного зору також забезпечує здатність до автоматичного вивчення нових розпізнавальних ознак та адаптації до різних умов зйомки. Це підвищує адаптивність і універсальність систем комп'ютерного зору, дозволяючи використовувати їх у багатьох галузях промисловості, включаючи робототехніку, безпілотні автомобілі, безпеку та багато іншого. Глибокі моделі мають високу стійкість до шуму та змін у вхідних даних, що має велике значення для застосування у практичних ситуаціях.

Інтеграція глибокого навчання в технології комп'ютерного зору також змінює спосіб розробки систем. Автоматизовані алгоритми, засновані на глибокому навчанні, замінюють звичайні методи обробки зображень, які використовують попередньо визначені правила та евристики. Для цього необхідна значна кількість навчальних даних, потужні обчислювальні можливості та відповідні інструменти для навчання та

впровадження моделі. Однак у міру розвитку технологій і поширення даних і обчислювальної потужності глибоке навчання стає все більш популярним і значущим у сфері технологій комп'ютерного зору.

Основна мета даної роботи – показати, що алгоритми, які використовують глибокі нейронні мережі можуть бути застосовані до реально існуючих проблемам класифікації та ідентифікації. Для цього розглянемо завдання розпізнавання тексту, тобто, розпізнавання образів із вхідного зображення у текстовий формат.

СУТЬ КУРСОВОЇ РОБОТИ

Розділ 1. Комп'ютерний зір та нейронні мережі

1.1 Комп'ютерний зір

Комп'ютерний зір досліджує інструменти та методи використання комп'ютерів для розпізнавання та аналізу візуальних даних. Він об'єднує компоненти машинного навчання та обробки зображень для автоматичного виявлення, класифікації та інтерпретації візуальних даних.

Використовуючи зображення та відео, комп'ютерний зір може ідентифікувати об'єкти, обличчя, рухи, знаки та інші візуальні елементи. Він використовується в багатьох галузях промисловості, включно з охороною здоров'я, транспортом, безпекою. Наприклад, в медицині він може допомагати виявляти ознаки хвороб на медичних зображеннях, а в автомобільній промисловості - розпізнавати дорожні знаки та покажчики для систем автоматичного водіння.

Пошук і вилучення значущих даних із великих наборів даних, що містять зображення, є одним із основних завдань комп'ютерного зору. Це досягається за

допомогою різноманітних алгоритмів обробки зображень, включаючи фільтрацію, видалення шуму, визначення контурів та інших візуальних особливостей. Після обробки зображення можна класифікувати, тобто можна ідентифікувати його зміст або особливі характеристики.

У майбутньому комп'ютерний зір може мати значний вплив на наше повсякденне життя. Він може використовуватись для автоматизації рутинних задач, виявлення та запобігання шахрайству, поліпшення якості медичної діагностики та багато іншого. Продовження досліджень та розвиток комп'ютерного зору можуть принести нові можливості та переваги у різних галузях технологій та життя людей.

1.2 Архітектура штучної нейронної мережі

Машинне навчання та нейронні мережі являються одним з основних напрямків розвитку комп'ютерного зору в задачах розпізнавання образів. Нейронні мережі, засновані на біологічній структурі людського мозку, перевищують за своєю обчислювальною спроможністю інші алгоритми машинного навчання.

Основна ідея, втілена в архітектурі нейронної мережі, полягає в імітації щільно з'єднаних між собою клітин головного мозку за допомогою комп'ютера. При цьому, отримана система має мати здатність досить легко навчатись для розпізнавання образів чи прийняття будь-яких рішень, подібно до того, як ці дії здійснюються людиною. Однією з головних переваг штучних нейронних мереж є їх можливість діяти згідно з попереднім досвідом, не вимагаючи додаткового програмування, що робить їх схожими з людським мозком.

Основною перевагою використання нейронних мереж є можливість їх застосування для отримання інформації з великих за обсягом або неточних даних, виявлення основних ознак та тенденцій, які не можуть бути розпізнані людиною або іншими методами глибокого навчання. Навчена нейронна мережа може виступати як

"експерт" для тієї категорії даних, на яку вона була підготовлена. Більше того, структура подібного типу надалі може допомогти прогнозувати нові результати.

Штучна нейронна мережа складається з десятків, тисяч, іноді навіть мільйонів нейронів, об'єднаних у шари. Так, вхідний шар складається з нейронів, призначених для прийняття різних форм інформації від зовнішнього світу, яка надалі оброблятиметься нейронною мережею. З протилежного йому боку розташований шар вихідних нейронів, що реагують на інформацію, що надходить. Між розглянутими шарами розташовується кілька проміжних шарів, які містять нейрони, що обробляють інформацію (рис 1.1).

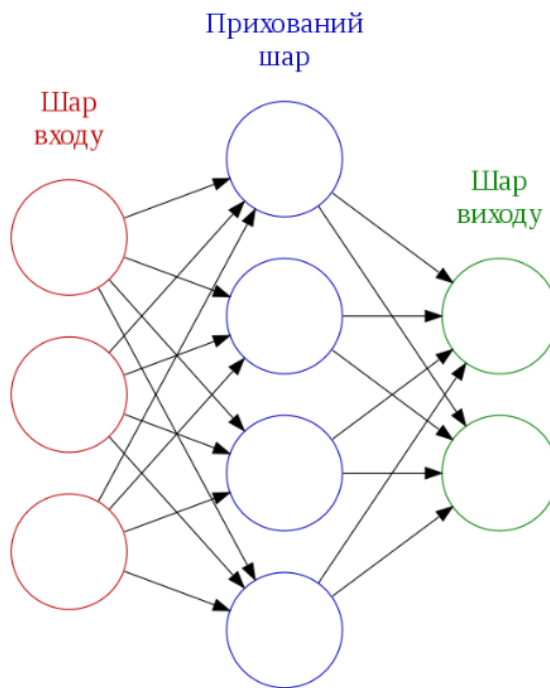


Рисунок 1.1. Шари нейронної мережі

Нейрони між шарами мережі з'єднані за допомогою спеціальних зв'язків, іменованих вагами, які можуть бути як позитивними (у випадку, якщо попередній нейрон збуджує наступний за ним), так і негативними (якщо попередній нейрон гальмує наступний за ним).

1.3 Математична модель штучної нейронної мережі

Далі, під штучною нейронною мережею розумітимемо впорядковану трійку елементів (N, V, w) . Тут N – множина нейронів мережі, $V = \{(i, j) | i, j \in N\}$ – множина, елементи якої є зв'язками між нейронами i та j . $w: V \rightarrow \mathbb{R}$, $w(i, j)$ або ж w_{ij} – вагова функція між нейронами i та j .

При розгляді конкретного нейрону j , видно, що до даного нейрону приєднано достатня кількість зв'язків, які ведуть до інших нейронів мережі, тобто, передаючих сигнал нейрону j . До нейрону j функція активації доставляє виходи $o_{i1}, o_{i2}, \dots, o_{in}$ нейронів i_1, i_2, \dots, i_n , які мають зв'язок з j , і приєднує до них відповідні ваги $w_{i,j}$ тим самим формуючи вихід нейрону net_j . Таким чином, вхід нейрона – це результат деякої функції.

Виходячи з наданих вище роздумів, можна сформулювати визначення входу конкретного нейрону мережі.

Нехай $I = \{i_1, i_2, \dots, i_n\}$ – множина нейронів, таких, що $\forall z \in \{1, \dots, n\}: \exists w_{iz,j}$. Тоді вхід нейрону net_j обчислюється як (рис1.2):

$$net_j = f_{prop}(o_{i1}, o_{i2}, \dots, o_{in}, w_{i1,j}, w_{i2,j}, \dots, w_{in,j}).$$

Часто в якості функції f_{prop} використовують суму вигляду:

$$net_j = \sum_{i \in I} (o_i * w_{i,j}).$$

Реакцію кожного нейрона на сигнал, що надійшов, будемо визначати з допомогою функції активації, а під станом активації (збудженості) a_j нейрону j будемо розуміти результат даної функції. Також розглянемо граничне значення b_j , яке визначається для кожного конкретного нейрона j , як максимальне значення градієнта

функції активації. З біологічної точки зору, граничне значення є величиною, при якій нейрон починає збуджуватись.

Таким чином, функція активації нейрона j має вигляд (рис.1.2)

$$a_j = \text{fact}(\text{net}_j, a_{j-1}, b_j)$$

Вихідна функція нейрона j обчислює значення, що передаються на вхід решті нейронів, приєднаних до j . З математичної точки зору вихід нейрона o_j являє собою результат впливу деякої функції f_{out} (рис.1.2):

$$f_{out}(a_j) = o_j.$$

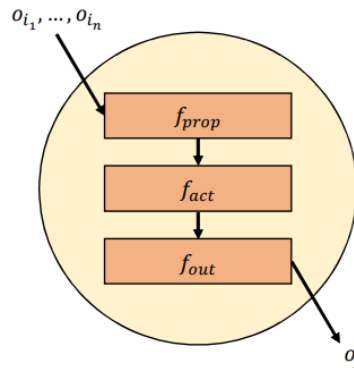


Рисунок 1.2. Обробка даних нейроном j .

1.4 Згорткова нейронна мережа

Згорткова нейронна мережа є однією з найяскравіших і найзначніших форм штучних нейронних мереж. Вона була вперше запропонована Я.Лекуном і Й. Бенгуа. Вчені досліджували зорову кору головного мозку кішки, яка містила в собі карти місцевих рецептивних полів, що зменшувалися в міру деталізації об'єкта, на який дивиться тварина. Я.Лекун та Й.Бенгуа дійшли висновку, що математичну модель поведінки зорових рецепторів кішки можна побудувати, спираючись на наступні етапи обробки зображень:

- згортка вихідного зображення об'єкта за допомогою кількох невеликих фільтрів;
- агрегування (об'єднання) отриманих на попередньому кроці ключових ознак;
- повторення попередніх кроків (згортки, а потім агрегування) до тих пір, поки на виході не вийде достатня кількість ознак вхідного зображення;
- використання моделі повнозв'язкового шару для отримання вирішення конкретного завдання.

Таким чином, згорткова нейронна мережа завдяки своїй архітектурі здебільшого використовується для вирішення складних графічних завдань розпізнавання та класифікації зображень.

Згорткова нейронна мережа має ряд значних переваг, завдяки яким вона обходить за обчислювальною потужністю алгоритми розпізнавання минулих десятиліть.

Насамперед алгоритми, що використовують згорткові нейронні мережі, інваріантні до різних спотворень, таких як поворот камери або нерівномірний розподіл світла на зображенні, горизонтальному або вертикальному зсувам та іншим. Також, згорткова нейронна мережа не вимагає виділення великої кількості пам'яті для зберігання вилучених у процесі роботи ознак. Ще однією перевагою використання згорткової нейронної мережі є досить велика швидкість навчання, яка досягається завдяки зменшенню кількості параметрів. Таким чином, продуктивність згорткової нейронної мережі перевищує в декілька разів продуктивність інших нейронних мереж, що використовуються в задачах по розпізнаванню.

1.5 Рекурентна нейронна мережа

Згорткові нейронні мережі добре підходять для класифікації зображення та відео, але у них є велика проблема, вони не вміють запам'ятовувати. Читаючи книгу чи статтю люди розуміють кожне слово на основі розуміння попередніх слів.

Якщо ми поставимо перед собою задачу, розпізнати кожен кадр фільму і використаємо, наприклад, згорткові нейронні мережі, це не буде ефективно, адже така архітектура не дозволяє використовувати інформацію про розпізнавання попередніх кадрів щоб розпізнати наступний.

Рекурентні нейронні мережі в свою чергу здатні на це. Завдяки особливій будові, в вигляді послідовно з'єднаних блоків, ця нейромережа здана до запам'ятовування.(рис.1.3).



Рисунок 1.3 Рекурентна нейронна мережа з петлею

Саме наявність зворотного зв'язку передавати інформацію з одного шару мережі в інший.

Подібна будова може виглядати заплутано, але в ній немає нічого складного. Рекурентна нейронна мережа можна розглядатися як кілька копій однієї нейронної мережі, кожна з яких передає інформацію наступному. Якщо ми розгорнемо петлю то ми побачимо, що подібна архітектура нейронної мережі найкраще підходить для роботи з списками та послідовностями (Рис 1.4).

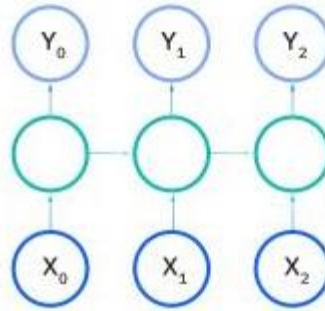


Рисунок 1.4 Розгорнута рекурентна нейронна мережа

Завдяки такій будові, дані нейронні мережі вже багато використовуються в таких задачах як розпізнавання зображення, переклад, розпізнавання мовлення та інших.

1.6 Згортково Рекурентна нейронна мережа

Згортково Рекурентна нейронна мережа (Convolutional Recurrent Neural Network або CRNN) є одним з представників нейромереж глибокого навчання, яка комбінує в собі зворотній зв'язок рекурентних нейронних мереж, та згорткові шари. Ця архітектура була розроблена для розв'язання завдань розпізнавання та опису послідовностей, зокрема для розпізнавання тексту на зображеннях.

Подібна архітектура була створена адже звичайні методи розпізнавання тексту мають критичні недоліки. На відміну від звичайного розпізнавання об'єктів на зображенні, розпізнавання писаного тексту являється задачею розпізнавання послідовностей. Згорткові нейронні мережі працюють з фіксованим розміром вхідного зображення та не можуть працювати з послідовностями. Виходячи з цього подібні нейронні мережі не можуть застосовуватись для розпізнавання різних слів, адже у, наприклад, слові "I" 1 символ, а у слові "Network" 7 символів. Рекурентні нейронні мережі навпаки, прекрасно працюють з послідовностями, але процес

попередньої обробки зображення який перетворює вхідний об'єкт в послідовність ознак, є важливим.

Таким чином згортково рекурентна нейронна мережа має велику кількість переваг над іншими нейромережами. Дана архітектура дозволяє навчатись безпосередньо з послідовності міток, не потребуючи детальних анотацій. Має такі ж властивості як згорткова нейромережа в навчанні безпосередньо з зображення, без необхідності попередньої обробки або локалізації компонентів. Як і рекурентна нейронна мережа здатна будувати послідовності міток.

Розділ 2. Модель згорткової нейронної мережі

2.1 Архітектура згорткової нейронної мережі

Згорткові нейронні мережі є одним з найважливіших класів теорії глибокого машинного навчання для вирішення задач комп'ютерного зору та розпізнавання. Вони складаються з декількох, шарів обробки, які як правило чергуються, кожен з яких містить, як лінійні, і нелінійні оператори. Сьогодні використання згорткових нейронних мереж є одним з основних методів для вилучення ознак з аудіо, відео та текстових даних.

Дана архітектура складається з трьох основних видів шарів: згортковий шар, агрегувальний шар і вихідний шар (найчастіше повнозв'язний). Шари згорткової нейронної мережі розташовані один за одним: спочатку згортковий шар, а потім агрегувальний, за останнім згортковим шаром слід вихідний шар (Рис.2.1). Згортковий та агрегувальний шари вважаються шарами двомірної розмірності, а вихідний шар, як правило, є вектор з простору \mathbb{R}^1 . У згортковій нейронній мережі

кожен двомірний шар має кілька рівнів. Кожен рівень є двомірним масивом. Вихід кожного рівня надалі називатимемо картою ознак.

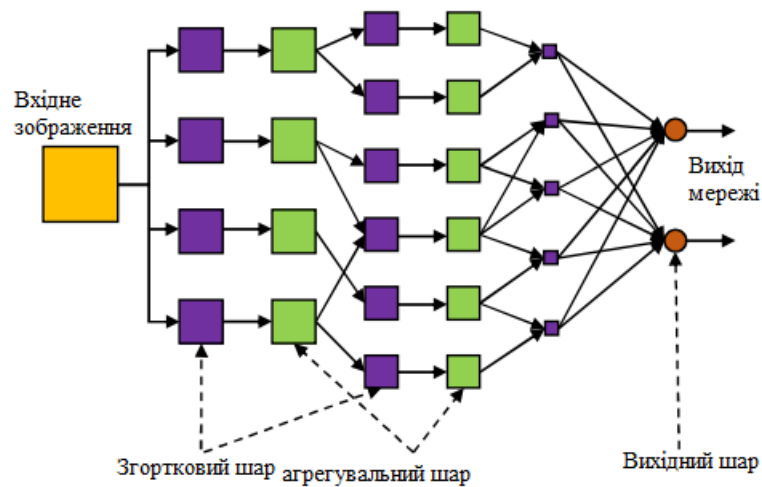


Рисунок 2.1. Архітектура згорткової нейронної мережі

Основною ідеєю використання згорткового шару є застосування математичної операції згортки (фільтра) до зображення. Згортка – це двовимірна матриця коефіцієнтів. Вхід такого фільтра – це фрагмент двовимірного зображення, а вихід – кілька (рис.2.2). Перевага використання подібного роду фільтрів полягає в наступному: число на вході тим більше, чим більше елемент зображення схожий на застосований до нього фільтр. Отже, використання операції згортки допомагає отримати на виході зображення, кожен піксель якого буде відповідати мірі подоби шматочка зображення на фільтр. Іншими словами, ми отримуємо карту ознак.



Рисунок 2.2. Приклад використання двохвимірної згортки для формування карти ознак

Агрегування використовується у згорткових нейронних мережах для комбінування інформації з різних нейронів або областей зображення в одне значення, що представляє загальну характеристику. Цей процес зазвичай відбувається шляхом обчислення суми, середнього значення або максимуму вхідних сигналів. Агрегування допомагає зменшити розмірність та узагальнити інформацію, забезпечуючи більш компактну та репрезентативну подачу даних для подальшого аналізу та прийняття рішень.

Повнозв'язний шар застосовується в кінці роботи згорткових нейронних мереж, адже після роботи шарів згортки і агрегування ми отримуємо вектор даних. Кожен нейрон цього шару містить функцію активації, або сигмоїдну функцію, або функцію ш=гіперболічного тангенса.

Розділ 3. Модель LSTM рекурентної нейронної мережі

3.1 Архітектура LSTM

Одна з особливостей рекурентних нейронних мереж є те, що вони здатні використовувати знання попередніх значень, наприклад в задачі розпізнавання відео, інформація про попередній кадр використовується для обробки поточного. Але з часом в такій архітектурі відбувається розривання зв'язку більш нових даних з старішими. Так, наприклад, якщо ми хочемо передбачити останнє слово в абзаці рекурентна нейромережа зможе використати тільки знання останніх речень, бо про те що говорилося в перших реченнях система втратила знання. Дану проблему вирішує LSTM.

Довга короткочасна пам'ять (Long short-time memory, або LSTM) – архітектура рекурентних нейронних мереж спеціально розроблена для навчання довгострокових залежностей, розроблена Зеппом Хозрайтером та Юргеном Шмідхубером.

LSTM як і будь яка рекурентна нейромережа має форму з'єднаних модулів. В простій архітектурі звичайних рекурентних нейромереж використовується всього один шар функції, в структурі LSTM цих слоїв 4 і вони особливо взаємодіють один з одним(Рис 3.1).

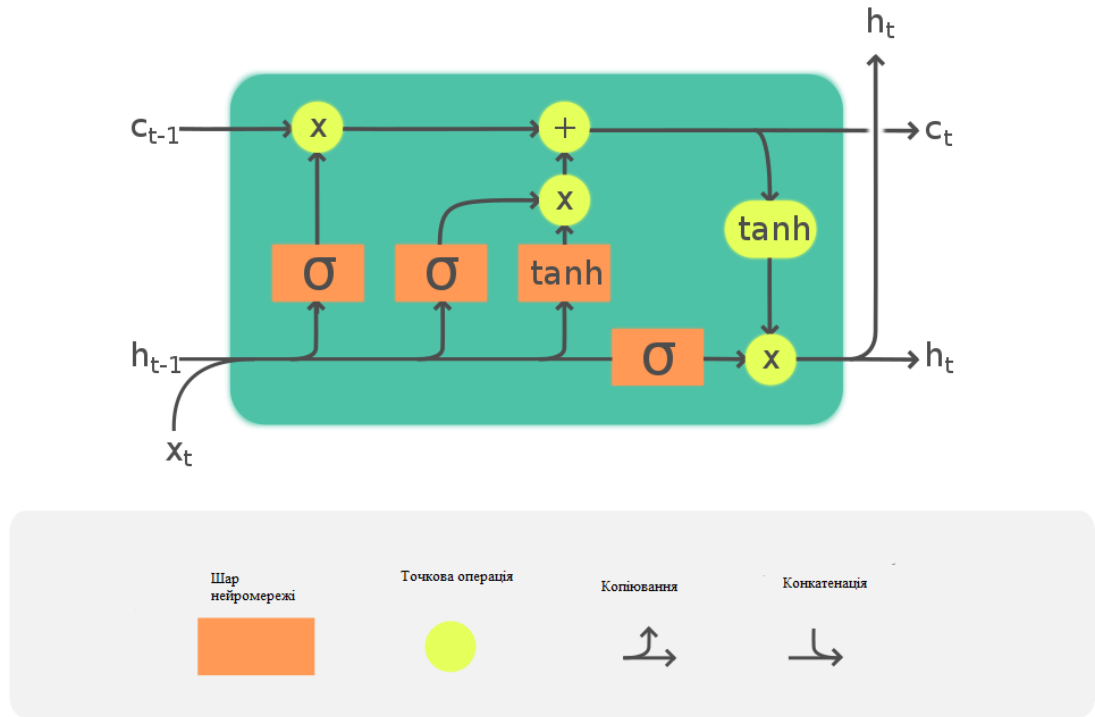


Рисунок 3.1 Модуль LSTM з чотирма шарами

3.2 Модель LSTM

Основним компонентом LSTM являється стан комірки (Рис 3.2). Він виглядає як конвеєр, що проходить через кожен блок мережі, інформація в якому практично не піддається змінам.



Рисунок 3.2 Стан комірки

Кількість інформації в стані комірки LSTM регулює за допомогою 3 фільтрів. Кожен з фільтрів використовується для вирішення, при яких умовах та чи інша частина інформації може пройти далі, а яка видалиться.

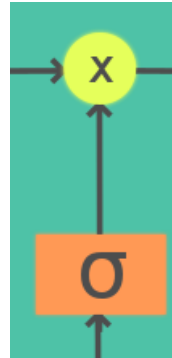


Рисунок 3.3 Фільтр LSTM

Фільтр повертає числа в діапазоні $[0;1]$, ці числа позначають частину кожного блоку інформації яку треба пропустити, де 1 означає, що потрібно пропустити все, а 0 – нічого.

При своїй роботі LSTM виконує 4 кроки. Спочатку, модель вирішує яку інформацію потрібно видалити з стану комірки, а яку залишити. Цю операцію виконує “фільтр забування”(Рис. 3.4). Він дивиться на h_{t-1} та x_t і виводить значення в діапазоні $[0;1]$ для кожного числа з стану комірки C_{t-1} .

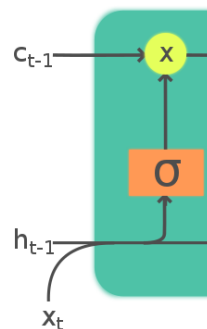


Рисунок 3.4 Фільтр забування

Далі “шар вхідного фільтру” визначає які значення необхідно оновити. Потім “tanh-шар” буде нових значень кандидатів \tilde{C}_t . Ці значення кандидати додаються в наш стан комірки (Рис 3.5).

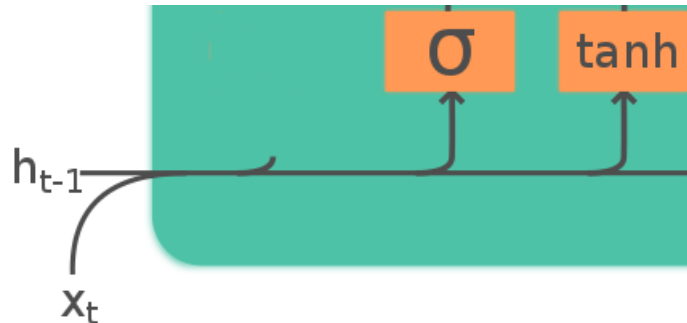


Рисунок 3.5 Шар вхідного фільтру та tanh-шар

Після цього ми заміняємо стан комірки C_{t-1} на C_t . Для цього ми множимо старе значення на f_t , завдяки цьому LSTM забуває те що необхідно, а потім додаємо $i_t * \tilde{C}_t$, що стає новими значеннями-кандидатом (Рис. 3.6).

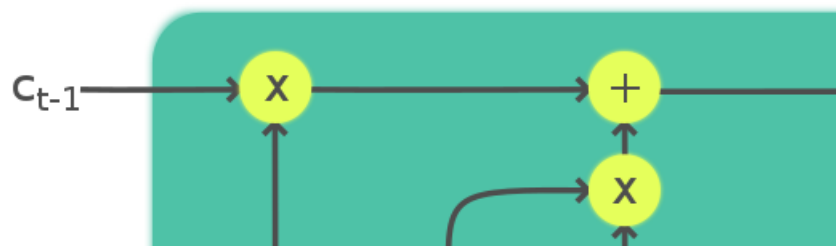


Рисунок 3.6 Запис нових значень

Останнім кроком вирішується яка інформація буде отримана на виході. Вихідні дані будуються на стані комірки, до яких використовуються певні фільтри. Спочатку ми використовуємо фільтр який вирішує, яку інформацію з стану комірки ми будемо виводити, потім значення проходять через tanh-шар через що ми отримуємо значення в діапазоні від -1 до 1, в кінці кінців це перемножується з вихідними значеннями фільтру (Рис. 3.7).

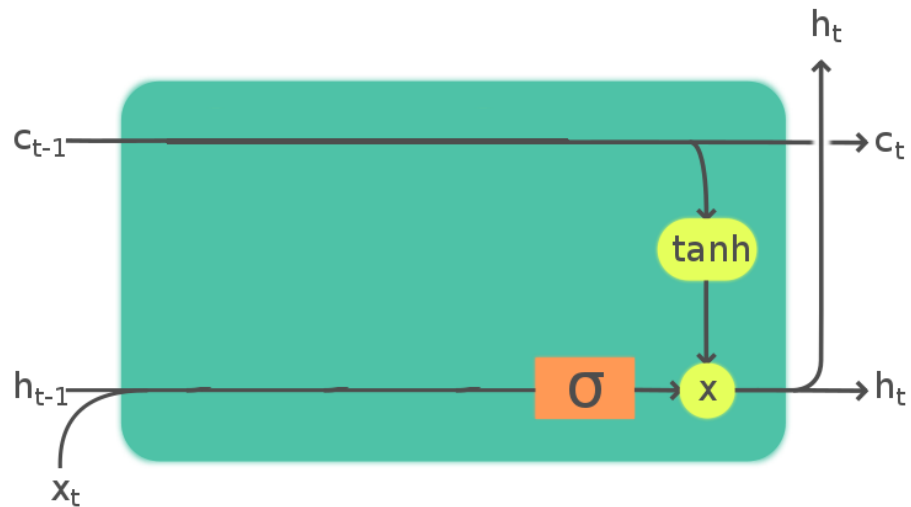


Рисунок 3.7 Отримання вихідних даних

Розділ 4. Модель Згортково рекурентної нейронної мережі

4.1 Архітектура Згортково рекурентної нейронної мережі

Згортково рекурентна нейронна мережа (Convolutional Recurrent Neural Network або CRNN) побудована з 3 частин. Згортковий шар – виділяє послідовність ознак. Рекурентний шар – приймає вихід зі згорткового шару та прогнозує кожен кадр. Шар транскрипції – приймає вихід рекурентного шару і перетворює прогнози в послідовність міток(Рис 4.1).

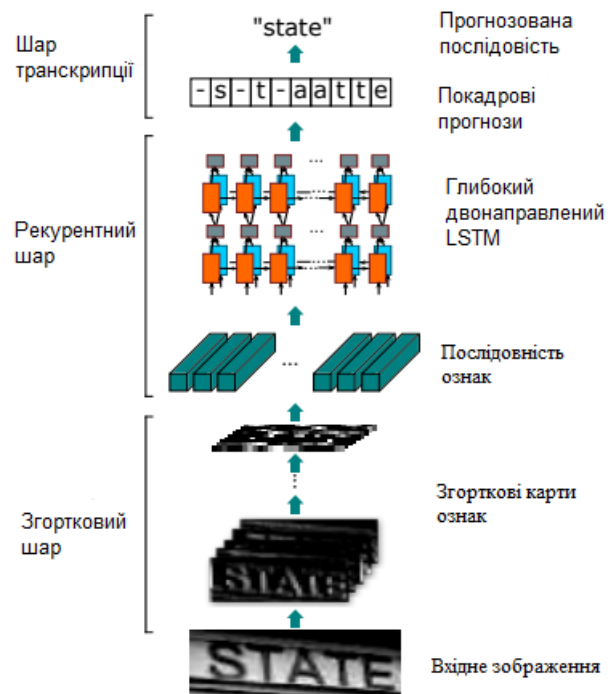


Рисунок 4.1 Будова CRNN

Згортковий шар CRNN включає в себе тільки шар згортки та шар агрегування звичайної згорткової нейронної мережі, в той час як вихідний шар нам не потрібен. Перед початком роботи всі зображення підганяються до однакової висоти, після чого з зображення утворюється карти ознак. Після чого з цих карт виділяються вектори ознак, при чому всі вони генеруються зліва на право по стовпчиках, через що кожен i -тий вектор ознак відповідає конкаєнації i -тих стовпчиків карти ознак. Оскільки шари згортки та шари агрегування інваріативні до перекладу, кожен стовпчик карт ознак відповідає прямокутній частині зображення і вони розташовані в однаковому порядку(Рис 4.2).

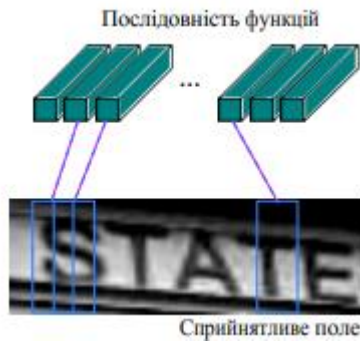


Рисунок 4.2 Збіг стовпчиків з частиною зображення

Рекурентний шар побудований з двонаправлених глибоких LSTM. Як розглядалось в розділі 3 “Модель LSTM рекурентної нейронної мережі” дану архітектуру корисно використовувати для прогнозування і розпізнавання рукописного тексту, бо вона використовує пам’ять про минулих значень при обробці, але для кращої продуктивності можна використовувати ще майбутні значення для передбачення. Склавши стандартний LSTM який прямує з ліва на право та LSTM який прямує з права на ліво, утворюється двонаправлений LSTM, а якщо їх декілька об’єднати то утворюється глибокий двонаправлений LSTM(Рис 4.3).

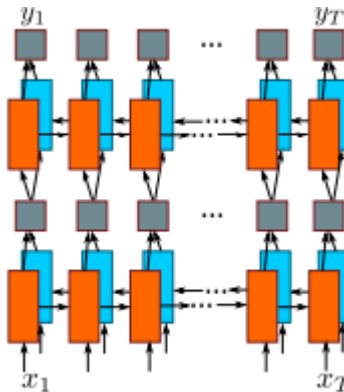


Рисунок 4.3 Будова глибокого двонаправленого LSTM

На шарі транскрипції архітектура приймає покадрові передбачення з рекурентного шару та перетворює їх в послідовність міток. По своїй суті процес транскрипції це знаходження найбільш ймовірної послідовності міток для покадрових

передбачень. На цьому шарі ми використовуємо коннекціоністську часову класифікацію (Connectionist Temporal Classification або CTC). Її роботу можна описати як: береться послідовність міток, береться список всіх міток, у випадку даної роботи це англійський алфавіт з розділяючими символами, з додатковою пустою міткою, після чого виконується функція відображення. Під час цього процесу ми перетворюємо послідовність міток, наприклад “-mm-i--lll-k--”, в слово “milk”.

Розділ 5. Програмна реалізація

5.1 Обладнання та програмне забезпечення

Програмна реалізація була виконана на ПК з процесором QuadCore Intel Core i5-8300H, 3900 MHz (39 x 100). Розробка моделі згортковоо рекурентної нейронної мережі виконувалась за допомогою мови програмування Python, використовуючи бібліотеки OpenCV, Keras, Numpy, Pandas, Sklearn, Tensorflow та інших необхідних.

5.2 Опис бази даних

Для навчання розробленої системи була використана база даних IAM Handwriting Database, а саме версія з підписаними та ізольованими рядками тексту, яких налічує 13 353 штуки.

5.3 Опис програмної реалізації

Першим кроком, підготував датасет на якій буде навчатись нейронна мережа, а саме зробив датафрейм таблицю з назвою зображення та її міткою(Рис 5.1).

0	a01-000u-00	A MOVE to stop Mr. Gaitskell from
1	a01-000u-01	nominating any more Labour life Peers
2	a01-000u-02	is to be made at a meeting of Labour

Рисунок 5.1 Приклад даних в таблиці

Потім використав цю таблицю для завантаження та обробки, а саме масштабував їх до розміру 800x64, повернув на 90 градусів по часовій стрілці, та використав бінарізацію(Рис 5.2).



Рисунок 5.2 Приклад бінарізованого зображення

Сама модель нейронної мережі складається з вхідного шару Conv2D з 64 фільтрами розміром (5, 5), що використовує функцію активації LeakyReLU з коефіцієнтом $\alpha=0.01$. Вхідні зображення мають розмірність (800, 64, 1). Далі застосовується шар MaxPooling2D з розміром вікна (2, 2), який зменшує розмірність зображення.

Наступні шари Conv2D з 128 фільтрами розміром (5, 5) і (3, 3) також використовують активацію LeakyReLU і шари MaxPooling2D з різними розмірами вікон для подальшого зменшення розмірності.

Після цього застосовується шар BatchNormalization, який нормалізує активації попереднього шару. Далі слідують два шари Conv2D з 256 фільтрами розміром (3, 3) і активацією LeakyReLU, а потім шар MaxPooling2D з розміром вікна (2, 2).

Після цього застосовується шар Conv2D з 512 фільтрами розміром (3, 3) і активацією LeakyReLU, а потім шар MaxPooling2D з розміром вікна (1, 2). Знову застосовується шар BatchNormalization.

Далі йде ще один шар Conv2D з 512 фільтрами розміром (3, 3) і активацією LeakyReLU, а потім шар MaxPooling2D з розміром вікна (1, 2).

Після цього слідує шар Reshape, який змінює форму тензора на (100, 512).

Потім модель містить два шари Bidirectional LSTM з 256 прихованими одиницями, які повертають послідовності результатів.

В кінці моделі застосовується шар Dense з кількістю вихідних нейронів, рівною довжині алфавіту плюс один (для символу порожнечі). Для активації використовується функція softmax, яка видає ймовірності для кожного символу з алфавіту(Рис 5.3).

```
model = Sequential()

model.add(Conv2D(64, (5, 5), padding='same', activation=LeakyReLU(alpha=0.01), input_shape=(800, 64, 1)))
model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(128, (5, 5), padding='same', activation=LeakyReLU(alpha=0.01)))
model.add(MaxPooling2D((1, 2)))

model.add(Conv2D(128, (3, 3), padding='same', activation=LeakyReLU(alpha=0.01)))
model.add(MaxPooling2D((2, 2)))
model.add(BatchNormalization())

model.add(Conv2D(256, (3, 3), padding='same', activation=LeakyReLU(alpha=0.01)))

model.add(Conv2D(256, (3, 3), padding='same', activation=LeakyReLU(alpha=0.01)))
model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(512, (3, 3), padding='same', activation=LeakyReLU(alpha=0.01)))
model.add(MaxPooling2D((1, 2)))
model.add(BatchNormalization())

model.add(Conv2D(512, (3, 3), padding='same', activation=LeakyReLU(alpha=0.01)))
model.add(MaxPooling2D((1, 2)))

model.add(Reshape((100, 512)))

model.add(Bidirectional(LSTM(256, return_sequences=True)))
model.add(Bidirectional(LSTM(256, return_sequences=True)))

model.add(Dense(len(alphabet) + 1, activation='softmax'))
```


Рисунок 5.3 Модель нейронної мережі

Розділив дані на 80% навчальних, 20% валідаційних, а навчальні дані розділив на 75% навчальних, 25% тренувальних, та почав тренування нейронної мережі. Тренування мало відбуватись на 75 епохах з розміром пакету в 64, але використана функція EarlyStopping яка зупинила навчання на 44 епосі. Після завершення навчання нейронна мережа має відсоток помилкових символів на рівні 9.88%, це можна побачити якщо подивитися на статистику навчання нейронної мережі (Рис 5.4).

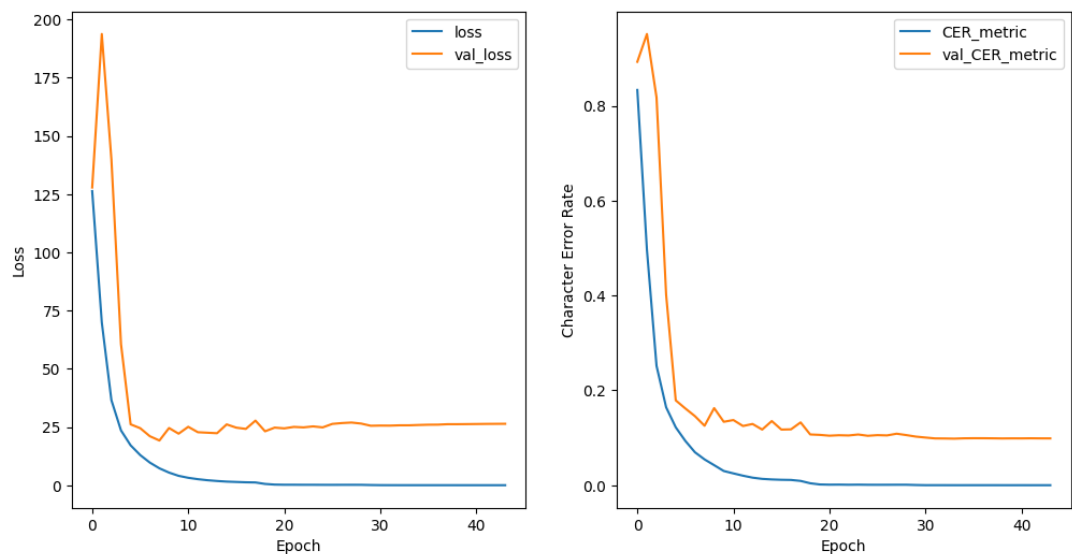


Рисунок 5.4 Статистика навчання

Перевірів роботу нейронної мережі на декількох прикладах з валідаційної вибірки(Рис 5.5).

ment by the Duke of Wellington . - mont by the Duke of Wellington .

ment by the Duke of Wellington,

the history of Anglesey's unceasing search for an - the history of Anglesey's unceasing search for an

the history of Anglesey's unceasing search for an

Above the coals the smoking fragment - thove tho coals the smoling fragment

Above the coals the smoking fragment

Рисунок 5.5 Приклад роботи нейронної мережі

Висновки

Дипломна робота зосереджувалася на використанні глибокого навчання у технологіях комп'ютерного зору, зокрема на створенні Згортково рекурентної нейронної мережі(CRNN). Ця робота демонструє потужність deep learning для розпізнавання тексту в зображеннях і вирішення завдань оптичного розпізнавання символів.

Результати дослідження підтверджують ефективність CRNN у виконанні завдань комп'ютерного зору, пов'язаних з текстовими даними. Мережа поєднує в собі переваги згорткових шарів для витягування особливостей зображення та рекурентних шарів для моделювання послідовностей. Це дозволяє досягти високої точності і швидкодії розпізнавання тексту на різних мовах та типах шрифтів.

У технологіях глибокого навчання комп'ютерного зору є багато можливостей для прогресу. За допомогою цієї технології тепер вирішуються складні завдання обробки зображень, які раніше було важко або неможливо автоматизувати. Продовження досліджень у галузі глибокого навчання сприятиме появі нових методів і моделей, що забезпечать ще більшу точність, ефективність та швидкість роботи систем комп'ютерного зору.

Список використаної літератури

1. An introduction to Neural Networks. [Електронний ресурс] - 2021. - [Cited 2021, 30 August.]. Available from: <http://www.explainthatstuff.com/introduction-to-neural-networks.html>
2. Keras: Python deep learning API [Електронний ресурс]. - 2015. - [Cited 2022, 13 May.]. - Available from: <https://github.com/keras-team/keras>
3. Understanding Convolutions. [Електронний ресурс]. - 2014. - [Cited 2014, 13 July .]. - Available from: <http://colah.github.io/posts/2014-07-Understanding-Convolutions/>
4. A Novel Connectionist System for Unconstrained Handwriting Recognition (2008)./ Graves Alex, Liwicki Marcus, Fernandez Santiago, Bertolami Roman, Bunke Horst, Schmidhuber Jürgen.
5. Introduction to Recurrent Neural Network [Електронний ресурс]. - Available from: <https://www.geeksforgeeks.org/introduction-to-recurrent-neural-network/>
6. Deep learning | Introduction to Long Short Term Memory [Електронний ресурс]. - Available from: <https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/>
7. Maximal Figure-of-Merit Embedding for Multi-Label Audio Classification (2008)/Ivan Kuhanov, Ville Hautamaki, Kong Aik Lee
8. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition (2017)/ Baoguang Shi, Xiang Bai, Cong Yao