Київський національний університет імені Тараса Шевченка

Факультет комп'ютерних наук та кібернетики

Звіт

до лабораторної роботи з курсу

Обробка природної мови

на тему:

"Машинний переклад"

Виконали

Студенти 2 курсу магістратури

Спеціальності 122 Комп'ютерні науки, групи ШІ-2

Зуєв А.М.

Коровай К.О.

Київ, 2021

1. Вступ

Метою даної роботи була розробка системи автоматизованого перекладу. Зважаючи на достатньо серйозні обмеження у часі розробки та відсутність ресурсів для серйозного тренування, розробити повноцінний перекладач виявилось практично неможливо. Тому було вирішено взяти як мету створити систему яка б могла перекладати невеликі розмовні фрази з початкової мови на цільову, виконуючи таким чином, роль розмовника.

Для досягнення цієї мети було поставлено наступні задачі:

- Дослідити існуючі методи та підходи машинного перекладу, обрати найбільш оптимальний.
- Дослідити наявні набори даних, знайти підходящий до обраної задачі.
- Виконати обробку даних.
- Виконати побудову моделі, її тренування та тестування.
- Проаналізувати отримані результати, зробити висновки.

Машинний переклад (МП) - одна з підгруп комп’ютерної лінгвістики. Це напрямок наукових досліджень, що пов'язаний з побудовою систем автоматизованого перекладу. Найпростіші системи машинного перекладу на початковому рівні виконують звичайну заміну слів з однієї мови на слова з іншої мови (target language), але, як правило, здійснений у такий спосіб переклад не є дуже якісним, адже для того щоб, повністю передати сенс речення, та знайти найспорідненіший аналог в «цільовій» (тобто потрібній перекладачу) мові, часто потрібно здійснювати переклад цілої фрази відразу.

Існують різні підходи для вирішення цієї проблеми:

- МП на основі правил (rule-based) - система машинного перекладу сформована на базі лінгвістичної інформації з одномовних (unilingual), двомовних (bilingual) чи багатомовних (multilingual) словників та граматичних правил вихідної мови та цільової мови.
- Статистичний МП - загальний підхід до вирішення проблеми перекладу, який заснований на пошуку найімовірнішого перекладу речення з використанням даних, отриманих з двомовної сукупності текстів.
- Нейронний МП - підхід заснований на методі глибокого засвоєння інформації (deep learning).
- Гібридний МП - метод, що заснований на об'єднанні та використанні сильних сторін зазначений методів (наприклад, статистична корекція після rule-based МП).

У даній роботі ми зосередилися на дослідженні та реалізації саме нейронних методів машинного перекладу, обравши в результаті обговорення класичну енкодер-декодер модель, та додавши так званий механізм уваги. Детальніше про архітектуру моделі у розділі 3.

2. Опис та обробка даних

Для тренування та перевірки якості роботи було вирішено використовувати паралельні корпуси текстів.

Паралельні тексти - це тексти з їх перекладом іншою мовою, вирівняні на певному рівні (речень, абзаців і т.д.). Підходи, які використовують паралельний корпус, намагаються витягти сильні перекладацькі відношення (translation relations) між двома мовами, на рівні слів або або на більш високому рівні (наприклад, рівень фрази). Ці перекладацькі відношення потім можна використовувати для перекладу запитів чи документів.

Було використано датасети з сайту manythings.org [1], де можна вільно знайти та завантажити обрані зразки текстів з Tatoeba Project [2]. Дані датасети гарно підходять для обраної задачі, оскільки вони містять співставлені пари “фраза”-”переклад фрази” з англійської на багато мов (вирівняні на рівні відносно коротких речень). При розробці виконувалося тестування на переклад з німецької на англійську утім тренування можна виконати для довільної пари мов (для перекладу не з англійської/на англійську спочатку необхідно одержати їхній датасет шляхом “об’єднання” датасетів цих мов з англійською).

Варто зауважити, що на даний момент, хоча система технічно і може обробляти мови слов’янської групи, результати будуть дещо гірші, адже у системі відсутні методи аналізу і тому форми одного слова вважатимуться різними словами, що призведе до значного зростання словника і як наслідок суттєвого погіршення точності.

Рисунок 2.1. Приклад зразків з англо-німецького датасету.

Після завантаження обраного набору даних, було виконано наступні кроки у якості передобробки даних:

1. З кожного речення видаляємо спеціальні символи (крім знаків пунктуації, які виділяємо додатковими пробілами і подаємо на токенізатор).
2. Додаємо маркер початку (<s>) та кінця (<e>) до кожного речення.
3. Створюємо словник мови, що відображає кожне слово на його індекс, та зворотній індексний словник (word → id, id → word відповідно).
4. Западимо (pad sequence) кожне речення максимальною довжиною.

На цьому етап передобробки ввжається завершеним.

3. Побудова моделі та її опис

Як уже було сказано, мережа представляє собою класичну схему “енкодер-декодер” до якої додано механізм attention.

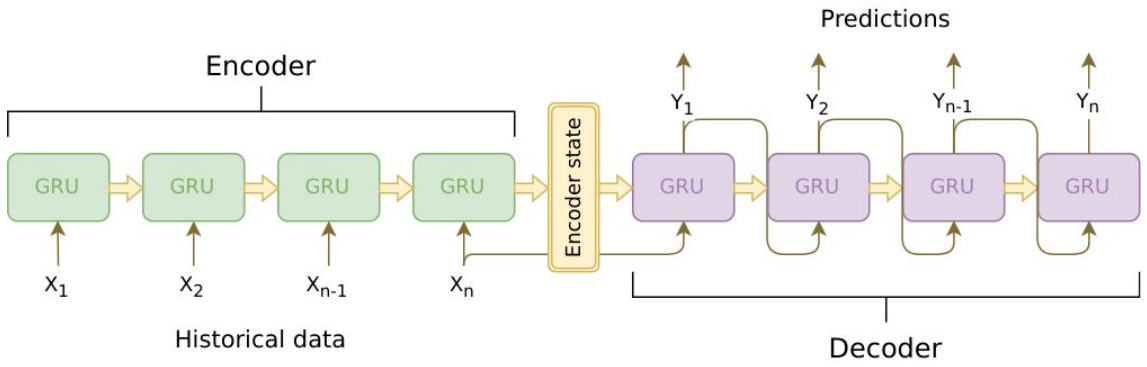

Рисунок 3.1. Загальна схема seq2seq моделі на основі енкодеру та декодеру.

Енкодер являє собою мережу яка подає на вхід рекурентного шару один за одним ембедінги слів вхідного речення та запам’ятовує приховані стани для подальшої роботи декодера. Видає на вихід множину своїх прихованих станів та вихід останнього кроку. Енкодер ітерується по вхідному реченні токен за токеном, на кожному кроці генеруючи вихідний вектор (output) та вектор прихованого стану (hidden state), після чого hidden state вектор передається на наступний крок, в той час як вихідний вектор записується та зберігається. Енкодер перетворює контекст, який він бачив у кожній точці послідовності, у набір точок у багаторозмірному просторі, який декодер використовуватиме для створення значущих результатів для даного завдання.

В основі нашого енкодера лежить багатошаровий вентильний рекурентний вузол (gated recurrent units), описаний Чо (Cho) та ін. у 2014 році [3].

Декодер являє собою мережу, у якій на певному кроці, спираючись на попередній крок та приховані стани енкодера будується масив ймовірностей рівний розміру вихідного алфавіту, що у даній позиції перекладу стоятиме слово з і-тим індексом. Вважається що у даній позиції просто стоятиме слово з максимальною з ймовірностей. Виконавши ітерерації до тих пір поки не буде отримано символ кінця рядка або одержано граничну довжину послідовності і отримаємо переклад.

Типовою проблемою так званого "ванільного" декодера seq2seq є те, що якщо ми покладаємось лише на контекстний вектор, щоб кодувати значення всієї вхідної послідовності, швидше за все, ми втратимо інформацію. Це особливо стосується випадків довгих послідовностей, що значно обмежує можливості нашого декодера.

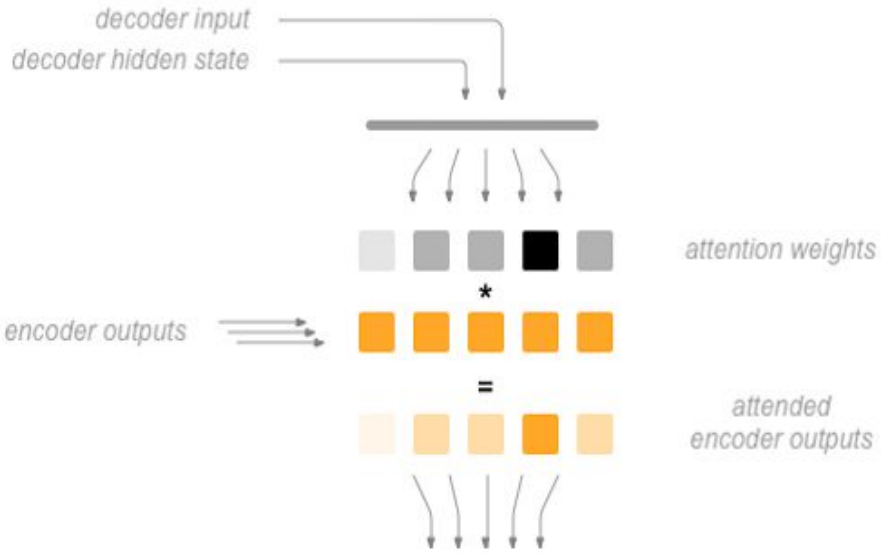

Рисунок 3.2. Візуалізація Шина Робертсона роботи механізму уваги (attention mechanism).

Для боротьби з цим, було створено «механізм уваги», який дозволяє декодеру "звертати увагу" на певні частини вхідної послідовності, а не використовувати весь фіксований контекст на кожному кроці (Bahdanau et al, 2014) [4]. На високому рівні, "увага" розраховується

за допомогою поточного прихованого стану декодера та виходів енкодера. Ваги уваги на виході мають ту саму форму (розмірність), що і вхідна послідовність, що дозволяє нам помножити їх на виходи кодера, даючи нам зважену суму, яка вказує на частини виводу кодера, на які слід звернути увагу. У загальному, цей підхід був чудово візуалізований Шином Робертсоном. Ця візуалізація наведена на рис. 3.2.

Згодом було створено ряд модифікацій описаного в оригінальній статті підходу, але в нашій роботі ми будемо використовувати саме запропонований Багданау механізм уваги, який ще називають локальним.

Тренування полягає у читанні та передобробці датасету та подальшому поданні датасету на систему “енкодер-декодер”. Після цього методом градієнтного спуска за допомогою оптимізатора Adam виконується підбір параметрів для отримання найкращого передбачення.

Для тестування береться фраза вхідної мови, виконується її передобробка та подається на систему “енкодер-декодер”. Після розкодування вихідних даних ми отримуємо бажаний переклад. Також на вихід моделі було додано додатковий grammar checker (LanguageTool [6]). Даний граммар чекер виконує роль пост обробки даних, виправляючи мінорні помилки (які можуть бути наявні у результаті через помилковий вхідний датасет, який нам залишалось взяти "as is", враховуючи відсутність знань з німецької мови). У подальшому як один в можливих варіантів покращення даної роботи, пропонується використати LanguageTool для початкового датасету, знайшовши та виправивши всі помилки у ньому, після чого вже подавати його на передобробку та вхід моделі.

4. Результати та приклади

Під час тренування було виконано певну кількість спроб на різних розмірах вхідних даних. Було помічено, що тренування на 100.000 та більше фраз не є фізично можливим на доступній машині. Результати для 50.000, 70.000 та 80.000 дають порівняно добрі результати на розповсюджених фразах. Спроби подавати на натреновану модель реальні тексти або просто фрази довші ніж фрази з реальної вибірки призводить або до дуже скороченого перекладу їх змісту або взагалі до малозв'язного тексту, що слабко пов'язаний з вхідними даними.

Приклади роботи розробленої системи наведено на рис.4.1.

Як бачимо, у більшості випадків ми маємо адекватний та коректний переклад. У деяких ситуаціях переклад помилковий (але відсоток таких випадків досить малий), що можна пояснити впливом обмежень по часу розробки та по наявним ресурсам.

Для подальшого розвитку моделі та покращення якості її роботи можна зробити кілька речей:

1. Збільшення розміру вхідних даних та мережі. На момент написання роботи не було можливим через відсутність технічних потужностей. В ідеалі, можна дозволити аналізувати значно довші речення ніж зараз (10-12 слів на моделі максимального розміру).
2. Виправлення можливих помилок у вхідному датасеті (наприклад, з використання LanguageTool) як попередній етап обробки даних.
3. Додавання певних способів лексичного аналізу для покращення вибору кандидатів (вибір на етапі декодування не

лише за ймовірністю, а й за відповідністю уже згенерованій частині речення).

4. Додавання дослідження зв’язків між реченнями для можливості генерації зв’язного тексту. Розклад речень на семантичні одиниці. Додаткова обробка "складних" слів у німецькій мові (тобто тих, що утворені злиттям двох або більше слів).
5. Додаткова обробка знаків пунктуації (додатковий аналіз замість простого подання знаків пунктуації на модель).
6. Заміна ембедінга індексів слів на інші значення меншої розмірності (наприклад використовуючи техніку word2vec). Дозволить прискорити тренування моделі та потенційно покращити результат.
7. Заміна локального механізму уваги на глобальний (модифікація, запропонована Luong et al. у 2015 р.).

Загалом у рамках поставленої задачі модель показує себе досить пристойно і переклад коротких фраз за змістом майже завжди співпадає з отриманим за допомогою Гугл Перекладача, що, на нашу думку, є досить гарним результатом за заданих обмежень.

```
>Ich bin Tom
Input : Ich bin Tom
Translating :  Ich bin Tom
Translated  -  i m tom .
Output: ('i m tom .', '<s> ich bin tom <e>')
Fixed : I m tom.
>Hallo!
Input : Hallo!
Translating :  Hallo!
Translated  -  hi .
Output: ('hi .', '<s> hallo ! <e>')
Fixed : Hi.
>guten morgen
Input : guten morgen
Translating :  guten morgen
Translated  -  see you tomorrow .
Output: ('see you tomorrow .', '<s> guten morgen <e>')
Fixed : See you tomorrow.
>guten Tag.
Input : guten Tag.
Translating :  guten Tag.
Translated  -  good afternoon .
Output: ('good afternoon .', '<s> guten tag . <e>')
Fixed : Good afternoon.
>das ist gut
Input : das ist gut
Translating :  das ist gut
Translated  -  this is good .
Output: ('this is good .', '<s> das ist gut <e>')
Fixed : This is good.
```

Рисунок 4.1. Приклад роботи розробленої системи. Input - вхідний рядок; Output - вихід моделі (пара "переклад"-"вхід моделі"); Translated - результат перекладу; Fixed - результат після застосування grammar checker на вихід перекладача.

5. Список використаних джерел

1. Tab-delimited Bilingual Sentence Pairs (selected sentence pairs from the Tatoeba Project). Режим доступу: http://www.manythings.org/anki/
2. Tatoeba Project. Open collaborative multilingual "sentence dictionary". Created by Trang Ho, Allan Simon. Режим доступу: https://tatoeba.org/
3. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
4. Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
5. Luong, M. T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025.
6. LanguageTool: free online proofreading service. Режим доступу: https://languagetool.org/