УДК 004.89

АНАЛІЗ МЕТОДІВ ПЕРЕТВОРЕННЯ СЛОВА В ВЕКТОР ФІКСОВАНОЇ ДОВЖИНИ ДЛЯ ЗАДАЧІ КЛАСИФІКАЦІЇ ТЕКСТІВ

Петришин Сергій, Переродов Артемій

Вінницький національний технічний університет

Анотація

Розглянуто поняття векторизації слів. Наведено основні задачі, для яких використовується векторизація слів. Розглянуто такі методи перетворення слів у вектор як one-hot encoding, word2vec, glove.

Abstract

The concept of word vectorization is considered. The main tasks, for which the word vectorization is used, are given. The following algorithms for word conversion into vector like one-hot encoding, word2vec, glove are considered.

Вступ

Векторизація слова – це перетворення слова у вектор дійсних чисел фіксованої довжини. Задача перетворення слова у вектор є досить актуальною проблемою при обробці тексту. Наприклад, ця проблема постає при вирішення таких задач, як:

– класифікація текстів;

– аналіз емоційного забарвлення тексту;

– машинний переклад;

– підсумовування об’ємних текстів.

Аналіз основних методів векторизації слів

Основними методами, які використовуються для перетворення слів у вектор, є:

– one-hot encoding;

– word2vec;

– glove.

Метод one-hot encoding є найбільш простим методом векторизації слів. В даному методі кожне слово кодується за допомогою вектора фіксованої довжини, що дорівнює кількості використовуваних слів в вибірці. Кожен вектор складається з нулів і однієї одиниці. Недоліком цього методу є величезна довжина вектору слова, так як в основному вибірка слів складається з десятків, а то і сотень тисяч слів.

Word2vec – це метод розрахунку векторних представлень слів, який реалізує дві основні архітектури: [1]

– Continuous Bag of Words (CBOW);

– Skip-gram.

На вхід подається корпус тексту, а на виході виходить набір векторів слів. Знаходження зв'язків між контекстами слів згідно з припущенням, що слова, що знаходяться в схожих контекстах, мають тенденцію означати схожі речі, тобто бути семантично близькими. Більш формально завдання стоїть так: максимізація косинусної близькості між векторами слів (скалярний добуток векторів), які з'являються поруч один з одним, і мінімізація косинусної близькості між векторами слів, що не з'являються поруч один з одним. Поруч один з одним в даному випадку означає в близьких контекстах.

Косинусна міра близькості (косинусна схожість) - це міра подібності між двома векторами. Косинусна схожість між векторами A і B обчислюється за формулою [1]:

$$similarity = \cos\theta = \frac{A * B}{\|A\| \, \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

CBOW і Skipgram - це нейромережеві архітектури, які описують, як саме нейромережа «навчається» на даних і «запам'ятовує» представлення слів. Принципи у обох архітектур різні. Принцип роботи CBOW - передбачення слова по заданому контексті, а skip-gram навпаки - передбачення контексту по заданому слові. Skipgram модель працює повільніше, але зазвичай за допомогою неї досягається краща якість класифікації текстів.

Недоліком методу word2vec є те, що для навчання моделі word2vec хорошої якості потрібен дуже великий корпус текстів, але в мережі Інтернет доступні вже “навчені”, на великих об’ємах даних, моделі.

Ще одним методом, який використовується для векторизації слів є метод Glove. Нехай об’єм словника даних рівний *V*. Усі слова, які зустрічаються в даних нумеруються від *1* до *V*. Складається матриця слово-слово $X \in R^{V \times V}$, де x_{ij} – кількість слів, коли слово *i* зустрічається в контексті слова *j*. Позначимо $X_{ij} = \sum_k^V V_{ik}$ (сума *i*-ої стрічки). Тоді ймовірність того, що слово *j* зустрілось в контексті слова *i* дорівнює $P_{ij} = P(j|i) = \frac{x_{ij}}{X_i}$ [2].

Для того, щоб зрозуміти, яке слово *i* або *j* зустрінеться в контексті слова *k* слід побудувати функцію: [2]

$$F(w_i, w_j, \widehat{w_k}) = \frac{P_{ik}}{P_{jk}}$$

Автори метода Glove пропонують використовувати таку функцію [2]:

$$F\left((w_i - w_j)^T \widehat{w_k}\right) = \frac{F(w_i^T \widehat{w_k})}{F(w_j^T \widehat{w_k})}, \text{де}$$

$$F(w_i^T \widehat{w_k}) = P_{ik} = \frac{X_{ik}}{X_i}$$

Отже, вектор w_i повинен бути таким, щоб: [2]

$$w_i^T \widehat{w_k} = \log(P_{ik}) = \log(X_{ik}) - \log(X_i)$$

Недоліком цього методу, як і методу word2vec, є те, що для навчання моделі хорошої якості потрібен дуже великий корпус текстів. Також недоліком є складність обчислень.

В ході аналізу методів перетворення слова у вектор фіксованої довжини було розглянуто такі методи, як one-hot encoding, word2vec, glove. Для вирішення задачі класифікації текстів для перетворення слова в вектор був обраний метод word2vec, так як він забезпечує схожість векторів семантично близьких слів та має відносно невелику розмірність.

Список використаних джерел:

1. Метод векторизації слів - word2vec – [Електронний ресурс]. – Режим доступу: http://nlpx.net/archives/179
2. GloVe: Global Vectors for Word Representation – [Електронний ресурс]. – Режим доступу: https://nlp.stanford.edu/pubs/glove.pdf