Классификация тональности текста с использованием гибридной свёрточной нейронной сети с параллельными и последовательными соединениями между слоями.

Роман Пелещак , Василий Литвин, Иван Пелещак, Андрей Худый, Зориана Рыбчак, Соломия Мушаста.

Львовский Политехнический Национальный Университет, Улица Степана Бандеры, 12, Львов, 79000, Украина.

Анотация.

Анализ тональности текстов является актуальной проблемой в области обработки естественного языка, которая часто решается с помощью сверточных нейронных сетей. Однако большинство из этих моделей CNN фокусируются только на изучении локальных функций, игнорируя глобальные особенности. В данной статье для анализа тональности текста предлагается гибридная сверточная нейронная сеть с параллельно-последовательными связями между слоями и со слоя максимального вытягивания, полученного из матрицы исходного текста. Предлагаемый гибридный сверточная нейронная сеть извлекает текстовые объекты, используя параллельно подключенный сверточный блок. Затем нейронная сеть классифицирует объекты и объединяет эти объекты с исходными текстовыми объектами. Модель предлагаемой нейронной сети способна изучать как локальные, так и глобальные особенности коротких текстов и обладает меньшим временем сходимости и вычислительными ресурсами по сравнению с параллельной DenseNet. Гибридная сверточная нейронная сеть с параллельными последовательными соединениями между слоями обладает более высокой эффективностью классификации тонов текста в 6 различные базы данных по сравнению с базовыми моделями CNN, TextCNN, FastText, DPCNN.

Ключевые слова: Тональность текста, классификация, сверточная нейронная сеть.

1. Введение

Проблемы обработки естественного языка становятся все более важными в связи с постоянно растущим объемом информации в Интернете и необходимостью ориентироваться в этой информации.

Задачи, которые широко используются при обработке естественного языка, включают классификацию текста, создание чат-ботов или генерацию ответов на вопросы пользователей, машинный перевод с одного языка на другой, распознавание языка, правописание, идентификацию частей речи в предложении и их аннотацию, переписывание текста информация для создания веб-контента. Помеченный набор данных, содержащий текстовые документы и их метки, используется для обучения классификатора.

Классификация текстов широко используется в тональном анализе (Imdb, классификация обзоров Yelp), анализе фондового рынка, для автоматических ответов на электронные письма. Методы, основанные на углубленном обучении нейронных сетей, стали современной практикой наряду с классическими алгоритмами интеллектуального анализа текста. Для решения задач классификации текста используются следующие архитектуры нейронных сетей: рекуррентная нейронная сеть, иерархическая сеть внимания и сверточная нейронная сеть [1].

Классификация документов - это процесс отнесения документов к определенной категории в зависимости от их содержания. Классификация текстов необходима для решения следующих задач: персонификация в рекламе; разделение сайтов по тематическим каталогам; борьба с вводящей в заблуждение рекламной корреспонденцией (спамом); распознавание тона текста, т.е. определение цвета эмоций в тексте.

Целью работы является разработка новой модели архитектуры сверточной нейронной сети с параллельными и последовательными связями между слоями для классификации тональности текстов с повышенной эффективностью.

2. Обзор литературы.

Большинство современных алгоритмов машинного обучения фокусируются на описании признаков объектов, поэтому все документы

преобразуются в реальное пространство признаков. Есть идея, что слова отвечают за принадлежность документа к определенному классу, и в текстах одного класса будет много похожих слов. Наиболее известные способы преобразования текста в пространство объектов основаны на статистической информации о словах. Каждый объект (текст) преобразуется в вектор, длина которого равна количеству слов во всех образцах текстов.

Существует три основные стратегии обнаружения особенностей при анализе тональности текста: модель набора слов [1], модель встраивания слов [2, 3] и модель графовой сети [4]. Инжир. На рис. 1 представлена структура исследования анализа тональности текста в соответствии с тремя основными методами, используемыми для выявления признаков.

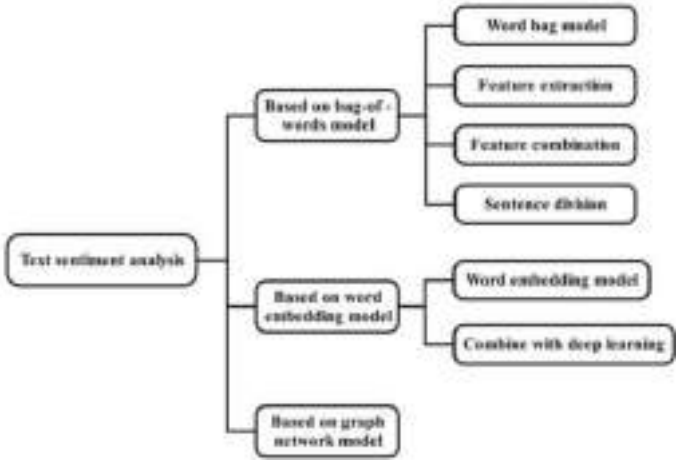

Рисунок 1: Анализ тональности текста в соответствии с тремя основными методами, используемыми для выделения

Основное предположение модели Bag-of-words заключается в том, что порядок слов в документе не важен, и набор документов можно рассматривать как простой выбор пар "документ - слово" (d, w), где $d \in D$, где $D = \{d_i \dots, d_i\}$– набор текстовых документов; $w \epsilon \mathrm{W_d}$, где $\mathrm{W_d} = (\mathrm{w_i}, \dots, \mathrm{W_{n_d}})$ – последовательность слов, $\mathrm{n_d}$ – длина документа d. Все документы представлены в виде матрицы $\mathrm{T} = (\mathrm{t})_{\mathrm{d,w}}$, каждая строка которого соответствует определенному документу или тексту, а каждый столбец соответствует определенному слову. Элемент $\mathrm{t_{d,w}}$ соответствует количеству вхождений слова w в документ d.

Давайте предположим, что предложение представляет собой синтаксически упорядоченный набор слов. Оно состоит из m слов. Каждое слово кодируется с помощью 1 в предложении – m кодировании, т.е. каждое слово w будет соответствовать вектору длины m. Составляющая этого вектора, соответствующая закодированному слову, равна 1, в позиции, которая равна

порядковому номеру размещения слова в предложении, и 0 – во всех остальных позициях. Поэтому была разработана модель набора слов в методах обнаружения признаков, таких как тегирование частей речи, POS-тегирование, POST-тегирование и n-граммовые тегирующие фразы.

Пометка части речи также известна как грамматическая пометка или идентификация частей речи. Это процесс определения слова в тексте по принадлежности к определенной части речи, основанный на его определении и контексте – на его связи с родственными словами во фразе, предложении или абзаце.

Наиболее популярным способом преобразования текста в вектор является модель Bag-of-words & TF-IDF [5]. Как и в модели Bag-of-words, все документы представлены в виде матрицы $T = (t)_{d,w}$. Но элемент $t_{d,w}$ соответствует функции TF IDF (w, d, D) слова $w \epsilon W_d$ в документе $d \epsilon D$.

Определения 1. TF-IDF – это статистический показатель, используемый для оценки важности слова в контексте документа. Рассчитывается по формуле:

$$\text{TF-IDF}(w, d, D) = \text{TF}(w, d) \times \text{IDF}(w, D).$$

где TF – частота слова, которая оценивает важность слова wi в пределах определенного документа

$$TF(w,d) = \frac{n_i}{\sum_{i=1}^{m} n_i},$$

n_i– вхождения слова i в документ.

$\sum_{k=1}^{m} n_k$ – общее количество слов в документе

IDE - обратная частота документа. Учет IDF снижает важность часто используемых слов.

$$IDF(w,|D|) = \log \frac{|D|}{|d_i \supset w_i|},$$

|D| – количество документов в корпусе.

$|d_i \ni W_i|$ - – количество документов, в которых встречается слово wi.

Часто информация в тексте обозначается не только отдельными словами, но и последовательностью слов, то есть словосочетаниями и фразеологизмами. В данном случае модель Bag of Ngrams & TF-IDF используется для получения языковых особенностей при преобразовании

текста в вектор. N-граммы [6] - это последовательности из N слов, в которых одно слово зависит от нескольких других. N-грамм - это показатель того, что данные из N слов связаны в задачах классификации текста.

Модель пакета из N граммов и TF-IDF аналогична модели пакета из слов и TF-IDF, только вектор признаков для каждого документа, за исключением слов TF-IDF, содержит TF-IDF всех последовательностей из N слов.

$$TF-IDF(w,d,D,N)=TF(w,d)\times IDF(w,D)\cup TF(N,d)\times IDF(N,D). \quad (4)$$

$$TF(N,d)\times IDF(N,D)=\frac{N_g}{\sum_{g=1}^{M}N_g}\times\log\frac{|D|}{|d_g\supset P_g|}. \quad (5)$$

N_g – вхождения g, N-грамм в документе.

$\sum_{g=1}^{M} N_g$ – общее количество $N-$ грамм в документе; в случае $M < m$

$|d_g \ni P_g|$ – количество документов, в которых N-грамм P_g

POS-тегирование и N-gram были объединены в [7, 8]. Эта модель может повысить точность классификации в соответствии с результатами экспериментов. Однако при анализе тональности коротких текстов [9, 10] было обнаружено, что эта модель не может достичь удовлетворительной точности, поскольку в коротких текстах состав предложений довольно произвольный. Следовательно, применять маркировку частей речи для их анализа нецелесообразно [9].

Существует три основных способа анализа тона коротких текстов. Они основаны на словарном запасе, традиционном машинном обучении и глубоком обучении. Модель анализа тональности текста, основанная на словарном запасе, позволяет получить сентиментальную направленность текстов путем вычисления и оценки текстов по словам с сентиментальной информацией. Модель основана на традиционном машинном обучении, не зависит от словарного запаса и обладает способностью самостоятельно изучать сентиментальные особенности текстов. Модель анализа текстовых настроений основана на глубоком обучении, позволяет более продвинутым и невыразимые сентиментальные особенности текстов [11]. Следовательно, особенности, полученные с помощью модели анализа тональности текста, основанной на глубоком обучении, являются абстрактными и их трудно выразить четко.

Что касается дифференциации в предложении, авторы [12] разработали систему классификации, которая может идентифицировать слова,

передающие сентиментальную полярность. Авторы [13] использовали алгоритм классификации, разработанный с использованием оценки эмоций SentiWordNet, и добились значительных улучшений производительности на шести наборах данных оценки. SentiWordNet - это словарь мнений, который присваивает три оценки настроения каждому синтаксису WordNet: позитивность, негативность и объективность. Некоторые исследования показали, что использование SentiWordNet для оценки настроения слова и добавление его в качестве функции может повысить точность анализа ключевых слов [7, 14]. Приведенный выше обзор литературы показывает, что классификация текста, основанная на модели набора слов, может дать лучшие результаты, если характеристики слова получены должным образом.

Однако модель "мешка слов" имеет и некоторые недостатки, поскольку она не учитывает порядок слов в предложении, т.е. синтаксис, и не может передать глубинные семантические особенности и семантические сочетания.

Анализ тональности текста, основанный на модели встраивания слов, решает проблему модели пакета слов, если мы применяем вектор слов в многомерном пространстве признаков [15].

Методы представления слов с использованием вектора фиксированной длины (когда его длина равна количеству слов, используемых в выборке) являются наиболее известными [16]. Каждый вектор состоит из нулей 0 и единиц 1.

Word2Vec - это технология [17], ориентированная на статистическую обработку больших массивов текстовой информации. Word2Vec собирает статистику о встречаемости слов в данных, удаляет наименее и/или наиболее распространенные слова, затем решает проблему уменьшения размерности с помощью методов нейронной сети и создает компактные векторные представления слов заданной длины.

В этом случае Word2Vec максимизирует косинусное сходство между векторами слов, которые встречаются в сходных контекстах, и минимизирует косинусное сходство между словами, которые не встречаются вместе.

Косинусное сходство измеряет сходство между двумя векторами. Косинусное сходство между векторами $\overline{A}$ и $\overline{B}$ рассчитывается по формуле:

$$similarity = \cos(\theta) = \frac{\overline{A} \cdot \overline{B}}{\|\overline{A}\| \cdot \|\overline{B}\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}. \quad (6)$$

Следует отметить, что для реализации технологии Word2Vec для преобразования слова в вектор можно использовать две разные архитектуры нейронных сетей: непрерывный пакет слов и Skip gram.

Модель встраивания слов основана на принципе "удаленного подобия" и имеет функцию сглаживания. Еще одним преимуществом модели встраивания слов является то, что это метод обучения без учителя. Доказано, что модель встраивания слов может обладать большим количеством семантических и грамматических особенностей , чем модель набора слов. Это преимущество позволяет модели встраивания слов достигать очень хороших результатов в различных задачах обработки естественного языка; [2, 15] разработали метод QVEC для измерения оценки эффективности представления характеристик различных моделей анализа текста. Полученные результаты показывают, что для векторной оценки текста в формате 300D QVEC тональность, основанная на модели встраивания слов, выше, чем в других моделях. В последние годы для анализа текста с большей производительностью используется комбинация модели встраивания слов и модели углубленного обучения. Авторы [18] разработали алгоритм обучения встраиванию слов, который объединяет векторы слов с RNN и может быть хорошо применен для распознавания речи. Авторы [19, 20] комбинируйте векторы слов с долговременной кратковременной памятью (LSTM) для достижения большей эффективности. Хотя CNN, разработанный авторами [1], имеет только один сверточный слой, его классификация эффективность намного выше, чем в обычном алгоритме классификации машинного обучения. Однако этот метод не может выделить свертки в больших текстах. В 2017 году авторы [21] выявили зависимости в больших текстах путем углубления сети. Авторы [22] представили структуру, аналогичную DenseNet, используя сокращения между верхним и нижним блоками свертки, так что более крупные объекты могут быть получены из комбинаций меньших объектов. Однако в модели использовался свернутый ядро определенного размера, которое перемещается от начала текста к концу, создавая карту объектов. Авторы [23] представили метод обучения на небольшой выборке для классификации текста. Авторы [24, 25] разработали модель текстовой стеганографии путем объединения текста с сокрытием информации и добились благоприятных результатов. Авторы [26] использовали временные функции нескольких объектов на основе LSTM для обнаружения спама. Результаты, полученные в работе [26], показали эффективность анализа тональности в длинных текстах.

Параллельная плотная сеть предложена в [27] на основе традиционных тесно связанных сверточных сетей для реализации тонального анализа короткого текста. В частности, в этой статье предлагаются два новых модуля извлечения признаков на основе DenseNet и многомасштабной сверточной

нейронной сети. Эта модель способна извлекать как локальные, так и глобальные объекты с коротким текстом, комбинируя выходные данные и объекты, извлеченные с помощью блока параллельного извлечения объектов, а затем отправляя объединенные объекты в окончательный классификатор.

Принцип анализа тональности текста, основанный на модели bag-of-words, заключается в том, чтобы поместить все слова в один пакет, так называемый word bag. Когда в предложении появляется слово, позиция этого слова в векторе равна 1, а позиция других слов равна 0. В этом случае слова в предложении расположены не по порядку. Таким образом, модель набора слов была разработана в методах распознавания признаков, таких как пометка части речи (POS) и пометка фраз N-граммами. Пометка части речи, также известный как грамматическая маркировка, это процесс маркировки слов в тексте (корпусе) как соответствующих определенным частям. N-граммовая маркировка фраз основана на том факте, что одно слово зависит от нескольких других слов. При обозначении слова это слово обычно сочетается с предыдущим словом.

Технология GloVe [28] позволяет получить соответствующий вектор фиксированной длины для каждого слова в текстовых данных, используя статистическую информацию о слове в данных.

Пусть размер словаря равен V. Все слова, найденные в данных, пронумерованы $\overline{1,V}$ Формируется матрица совпадения слов $X \in R^{VxV}$, где x_{ij} – указывает, сколько раз слово i используется в контексте слова j. Слово a встречается в контексте слова b, если между ними есть часть текста, содержащая не более девяти слов.

Давайте отметим $X_i = \sum_{k=i} x_{ik}$ (сумма строки i). Тогда вероятность того, что слово j встречается в контексте слова i, равна $P_{ij} = P(j|i) = \frac{x_{ij}}{X_i}$.

Следует отметить, что если слово встречается в контексте слова k чаще, чем просто слово встречается в контексте слова k, то $\frac{P_{ik}}{P_{ik}} > 1, \frac{P_{ik}}{P_{ik}} < 1$.

Давайте создадим функцию F (w_i , w_j , $\widehat{w_k}$) это показывает, какое из слов i или j с большей вероятностью встречается в контексте слова k. w_i , w_j , $\widehat{w_k}$ – векторное представление слов i, j и k.

$$F\left(w_i, w_j, \tilde{w}_k\right) = \frac{P_{ik}}{P_{jk}}.$$

Авторы модели перчаток предложили использовать

$$F\left(\left(w_i - w_j\right)^T \hat{u}_k\right) = \frac{F\left(w_i^T \hat{w}_k\right)}{F\left(w_j^T \hat{w}_k\right)}$$

$$F\left(w_i^T \hat{w}_k\right) = P_{ik} = \frac{x_{ik}}{X_i}.$$

Затем вы можете выбрать F(x)=exp(x) как функцию F и выбираем вектор w_i то

$$w_i^T \hat{w}_k = \log\left(P_{ik}\right) = \log\left(x_{ik}\right) - \log\left(X_i\right).$$

Теперь, учитывая, что $\log(X_i)$ исправлено, мы переписываем проблему следующим образом

$$w_i^T \hat{w}_k + b_i + \tilde{b}_k = \log\left(x_{ik}\right)$$

$$b_i + \tilde{b}_k = \log\left(X_i\right).$$

В результате авторы используют функцию потерь J и корректируют модель с помощью алгоритма AdaGrad [4].

Функция f(x) должен соответствовать следующим требованиям: f(0)=0; f(x) – не уменьшается; f(x) – относительно мал при больших значениях x.

Авторы использовали следующее:

$$f(x) = \begin{cases} \left(\frac{x}{x_{max}}\right)^{\alpha}, x < x_{max} \\ 1, x > x_{max} \end{cases}.$$

Параметры были выбраны эмпирическим путем:

$$\alpha = \frac{3}{4},\ x_{max} = 100.$$

3. Постановка задачи

Задача классификации текстовой информации формулируется следующим образом: пусть в категории имеется конечное число классов $\bar{C} = \{c_1, c_2, \ldots . c_m\}$ и конечный набор документов $\bar{D} = \{d_1, d_2, \ldots . d_m\}$ и неизвестная целевая функция f который определяет соответствие для каждой пары (документ, класс) f: $\bar{D}$ x $\bar{C}$ → {0,1}. Задача состоит в том, чтобы найти функцию f_0 которая максимально приближена к целевой функции f , то есть

предоставляется минимальная ставка min -||f-f_0|| в евклидовом пространстве. Функция f_0 называется классификатором.

Под тональностью текста понимается эмоциональная лексика и эмоциональная оценка, данная автором по отношению к объекту. Анализ тональности текста имеет большое практическое значение: оценка качества товаров и услуг на основе отзывов пользователей интернет-ресурсов; профилактика экстремизма и терроризма; анализ фондовых рынков и прогнозирование волатильности (изменчивости) финансовых активов.

Основная задача анализа тональности текста - выявить идеи в тексте и определить их свойства. Мнения бывают двух типов: сравнение мнений и прямое мнение. Прямое мнение содержит высказывание автора об объекте. Формальное определение мнения описывается как кортеж из 4 элементов

$$\bar{K} = \{o(p), e(f), t, h\}$$

Где o(p) оценка ориентации или полярности тональности; e(f) сущность или признак - объект тональности или его свойство f; t время – окончание предложения; h - держатель – субъект тональности (автор).

Тональность текста оценивается как нейтральная, отрицательная или положительная.

В общей статистике волатильность - это показатель, характеризующий колебания временных рядов или тенденций изменения рыночных цен и доходов с течением времени; составление текстов с заранее заданными эмоциональными характеристиками.

Существуют различные типы классификации текста: субъективная; объективная; многомасштабная, т.е. классификация в соответствии с многоуровневой шкалой и классификация в соответствии с двоичной шкалой. В этой статье для решения этой проблемы используется фреймворк машинного обучения Keras и язык программирования Python.

4. Архитектурная модель разветвленной сверточной нейронной сети с параллельными и последовательными соединениями.

Архитектура новой гибридной сверточной нейронной сети (рис. 2) состоит из блока сверточной нейронной сети с параллельными и последовательными соединениями между слоями и слоя максимального вытягивания, который получается из матрицы исходного текста $\overline{x_0} = (x_i, x_2, \dots x_m)$ mod; длина m; где $x_i \in R^d, i = 1,2, \dots, m$ Вы можете

использовать разные ядра для свертки предложения и получения различных признаков в предлагаемой сверточной нейронной сети. Полученные признаки объединяются с матрицей максимального вытягивания, полученной из $\overline{x_0}$ с помощью блока сверточной нейронной сети. Эти признаки классифицируются с использованием классификатора MLP. Следует отметить, что предлагаемая новая гибридная сверточная нейронная сеть отличается от сжатой нейронной сети Dense Net [27, 29]. В частности, эта модель имеет меньшее время сходимости и не требует многократных итераций обучения (меньше вычислительных ресурсов) из-за отсутствия плотной блочной сети DenseNet, которая используется для классификации тональности текстов [30, 31].

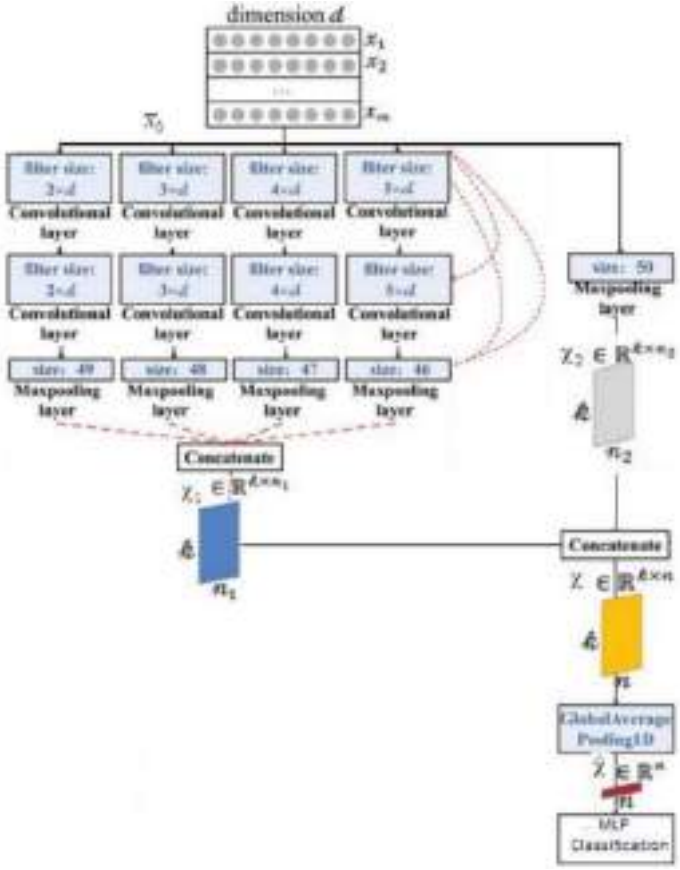

Рисунок 2: Структура гибридной сверточной нейронной сети параллельными и последовательными соединениям между слоями.

Текст вводится на вход разветвленной сверточной нейронной сети (рис. 2).

$$\overline{x}_0 = (x_1, x_2, .., x_m)_{m \times d}, \; x_i \in R^d, \; i = 1, 2, ..., m$$

с длиной m. Эта нейронная сеть (рис. 2) состоит из двух частей: блока сверточной нейронной сети с параллельными и последовательными

соединениями между слоями и слоя максимального вытягивания. Блок сверточной нейронной сети состоит из слоев с разными размерами окон, которые соединены параллельно по столбцам и последовательно по строкам структуры. Входные данные каждого сверточного слоя в столбце состоят из суммы выходных данных всех предыдущих слоев. Параллельная текстовая матрица $\overline{x_0}$ наносится на входе в свернутые слои 5 x d, 4 x d, 3 x d, 2 x d. Для классификации тональности текста мы будем использовать усредненные общие

признаки, которые могут быть получены путем объединения двух характеристик (признаков, полученных из блока сверточной сети с размерами $\widehat{X_1}$ и от максимального вытягивающего слоя $\widehat{X_2}$ из-за среднемирового значения. Каждая подсеть свертки используется для майнинга объектов с использованием различных сочетаний слов, в зависимости от размера ядер. В частности, для ядра 5 x d для определения функций используется комбинация из 5 слов. Аналогично, матрица входного текста представлена на основе подсети с ядрами 2 x d, 3 x d , 4 x d.

Входная текстовая матрица $\overline{x_0}$ вводится в свернутые слои определенного размера 5 x d, 4 x d , 3 x d для майнинга функций.

$$\begin{aligned} \hat{y}_{15} &= f_{5\times d}(\overline{x}_0) \\ \hat{y}_{14} &= f_{4\times d}(\overline{x}_0) \\ \hat{y}_{13} &= f_{3\times d}(\overline{x}_0) \\ \hat{y}_{12} &= f_{2\times d}(\overline{x}_0) \end{aligned} \quad (8)$$

$\widehat{y_{15}}$, $\widehat{y_{14}}$, $\widehat{y_{13}}$, $\widehat{y_{12}}$ – матрица объектов после первого слоя сверточного преобразования с размерами ядер 5 x d, 4 x d , 3 x d , 2 x d. После этого исходная входная текстовая матрица объединяется с матрицами признаков после преобразования свертки, и мы получаем новые входные текстовые матрицы $\widehat{y_{15}}$, $\widehat{y_{14}}$, $\widehat{y_{13}}$, $\widehat{y_{12}}$.

$$\begin{aligned} \hat{x}_{15} &= Cat([\overline{x}_0, \hat{y}_{15}]) \\ \hat{x}_{14} &= Cat([\overline{x}_0, \hat{y}_{14}]) \\ \hat{x}_{13} &= Cat([\overline{x}_0, \hat{y}_{13}]) \\ \hat{x}_{12} &= Cat([\overline{x}_0, \hat{y}_{12}]) \end{aligned} \quad (9)$$

Мы вводим новые матрицы входного текста $\widehat{x_{15}}$, $\widehat{x_{14}}$, $\widehat{x_{13}}$, $\widehat{x_{12}}$ во втором сверточные слои с ядрами 5 x d, 4 x d , 3 x d , 2 x d.

$$\begin{aligned} \hat{y}_{25} &= f_{5\times d}(\hat{x}_{15}) \\ \hat{y}_{24} &= f_{4\times d}(\hat{x}_{14}) \\ \hat{y}_{23} &= f_{3\times d}(\hat{x}_{13}) \\ \hat{y}_{22} &= f_{2\times d}(\hat{x}_{12}) \end{aligned} \quad (10)$$

Чтобы получить новые матрицы признаков, мы выполняем следующие операции по объединению матриц:

$$\begin{aligned}\hat{x}_{25} &= Cat\left(\left[\bar{x}_0, \hat{y}_{15}, \hat{y}_{25}\right]\right)\\ \hat{x}_{24} &= Cat\left(\left[\bar{x}_0, \hat{y}_{14}, \hat{y}_{24}\right]\right)\\ \hat{x}_{23} &= Cat\left(\left[\bar{x}_0, \hat{y}_{13}, \hat{y}_{23}\right]\right)\\ \hat{x}_{22} &= Cat\left(\left[\bar{x}_0, \hat{y}_{12}, \hat{y}_{22}\right]\right)\end{aligned} \quad (11)$$

После преобразований свертки выполняется операция вытягивания для получения новых матриц признаков.

$$\begin{aligned}\hat{x}^{(1)} &= h_{46}\left(\hat{x}_{25}\right)\\ \hat{x}^{(2)} &= h_{47}\left(\hat{x}_{24}\right)\\ \hat{x}^{(3)} &= h_{48}\left(\hat{x}_{23}\right)\\ \hat{x}^{(4)} &= h_{49}\left(\hat{x}_{22}\right)\end{aligned} \quad (12)$$

$\hat{x}^{(1)}, \hat{x}^{(2)}, \hat{x}^{(3)}, \hat{x}^{(4)}$ – новые матрицы признаков, полученные после операции вытягивания h_{46}, h_{47}, h_{48}, h_{49}

После этого новые матрицы признаков объединяются для получения матрицы многомасштабного блока для интеллектуального анализа сверточных признаков.

$$\hat{\chi}_1 = Cat\left(\left[\hat{x}^{(1)}, \hat{x}^{(2)}, \hat{x}^{(3)}, \hat{x}^{(4)}\right]\right). \quad (13)$$

Функция Cat описывает объединение матриц $\hat{x}^{(1)}, \hat{x}^{(2)}, \hat{x}^{(3)}, \hat{x}^{(4)}$ Матрица объектов из слоя максимального пула размерности 50 описывается формулой

$$\hat{\chi}_2 = h_{50}\left(\bar{x}_0\right). \quad (14)$$

Мы объединяем функции матрицы $\hat{X}_i$ и $\hat{X}_2$ использование функции Cat для получения матрицы общих признаков $\hat{X}$

$$\hat{\chi} = Cat\left(\hat{\chi}_1, \hat{\chi}_2\right). \quad (15)$$

И выполнить операцию одномерного глобального усреднения $\hat{X}$ чтобы получить окончательную матрицу признаков %

$$\%= g\left(\hat{\chi}\right). \quad (16)$$

Функция g представляет собой одномерное усредненное слияние. После этого мы представляем конечную матрицу признаков классификатору нейронной сети (MLP) для классификации тональности текста.

5. Компьютерный эксперимент

Для компьютерного эксперимента были отобраны шесть различных наборов данных, которые разделены на различные категории тональности текста. Они включают:

 Набор данных GameMultiTweet, этот набор состоит из 12780 частей, которые разделены на трипкатегории, состоящие из 3952 - 915 - 7913 частей.

 Набор данных SemEval, этот набор состоит из 7967 частей, которые разделены на три категории,nсостоящие из 2964 - 1151 - 3852 частей.

 Набор данных SS-Tweet, этот набор состоит из 4242 частей, которые разделены на три категории,nсостоящие из 1953 - 1336 - 953 частей.

 Набор данных AG News, этот набор состоит из 127 600 частей, которые разделены на четыре категории , состоящие из 31,900 - 31,900 - 31,900 - 31,900 частей.

 Набор данных R8, этот набор состоит из 4203 частей, которые разделены на восемь категорий, состоящих из 1392 - 241 - 2166 - 20 - 162 - 0 - 72 - 150 частей.

 Yahoo! Набор данных ответов, этот набор состоит из 350 000 частей, которые разделены на десять категорий, состоящих из 23726 - 35447 - 31492 - 35252 - 35546 - 25787 - 25787: 81571 - 23961 - 28706 - 28482 частей.

Все эти наборы данных были случайным образом разделены на три части: 70% обучающего набора, 15% проверочного набора и 15% тестового набора. Статистика по каждому набору приведена в таблице 1

Таблица 1

Информация о наборе данных

Dataset	Train	Validation	Test	Categories	Avg. length
GameMultiTweet	8964	1917	1917	3	26
SemEval	5577	1195	1195	3	31
SS-Tweet	2870	636	636	3	29
AG News	89320	19140	19140	4	45
R8	2943	630	630	8	66
Yahoo! Answers	245000	52500	52500	10	112

Наша модель сравнивается с другими моделями:

 Модель CNN, состоящая из трех сверточных слоев, в которых сверточные ядра имеют

одинаковый размер.

 Модель TextCNN, предложенная в статье [1].

 Модель быстрого текста, предложенная в статье [17].

 Модель DPCNN, предложенная в статье [21]

В нашем исследовании предложение было преобразовано в матрицу размером 150x300 с помощью word2vector. Были заданы некоторые параметры, такие как использование оптимизатора adam и установка скорости обучения на 0,001, коэффициента отсева на 0,2 и веса потери L2 на 10-8. Размер пакета модели составлял 50, а количество эпох - 5. Если потери не уменьшались в течение 10 последовательных периодов, обучение прекращалось. В модели встраивания слов перед обучением использовалось встраивание слов 300D word2vector.

Таблица 2

Сравнение нашей модели с другими

Model	GameMultiTweet	SemEval	SS-Tweet	AG News	R8	Yahoo! Answers
CNN	73,5	60,5	50,2	85,6	92,3	47,3
TextCNN	77,5	62,7	51,1	88,9	94,4	49,5
FastText	78,3	63,8	51,4	88,5	96,1	39,8
DPCNN	75,6	47,5	43,2	87,1	88,5	47,5
Our Model	78,5	66,0	52,4	89,7	98,1	51,6

Результаты, представленные в таблице 2, показывают, что наша модель достигла более высокой точности, чем ее аналоги.

6. Выводы

Разработана гибридная сверточная нейронная сеть для анализа тональности текста. Она состоит из сверточного блока параллельных и последовательных соединений между слоями и слоя максимального вытягивания, полученного из матрицы исходного текста.

Показано, что такая гибридная сверточная нейронная сеть анализирует текстовые объекты, используя сверточный блок. Затем она анализирует и классифицирует объекты, комбинируя эти объекты с исходными текстовыми объектами.

Было обнаружено, что модель гибридной сверточной нейронной сети имеет меньшее время сходимости и вычислительный ресурс по сравнению с параллельной плотной сетью.

Было доказано, что гибридная сверточная нейронная сеть с параллельными и последовательными соединениями между слоями обеспечивает более высокую эффективность классификации тональности текста в 6 различных базах данных GameMultiTweet, SemEval, SS-Tweet, AG News, R8, Yahoo! Ответы по сравнению с другими базовыми моделями.

7. Ссылки

[1] Ким Ю. Сверточные нейронные сети для классификации предложений. Материалы конференции 2014 г. Конференция по эмпирическим методам обработки естественного языка, 2014, стр. 1746-1751. URL-адрес: https://arxiv.org/abs/1408.5882

[2] Дэниел Джурафски, Джеймс Х. Мартин. Обработка речи и языка, 2021. URL-адрес: https://web.stanford.edu /~юрафский/slp3

[3] П. Лю, Х. Цю, Х. Хуан. Рекуррентная нейронная сеть для классификации текста с многозадачностью познающий. В работе. IJCAI, Нью-Йорк, США, 2016, стр. 2873-2879. URL-адрес: https://arxiv.org/abs/1605.05101

[4] Л. Яо, К. Мао, Ю. Луо. Графовые сверточные сети для классификации текстов. В работе. AAAI, Гавайи, США, 2019, стр. 7370-7377. URL: https://arxiv.org/abs/1809.05679

[5] Зулкарнайн, Царица Дви Путри. Интеллектуальные транспортные системы (ИТС): систематический обзор с использованием подхода обработки естественного языка (NLP). Гелийон, 2021, том 7, e08615. doi: 10.1016/j.heliyon.2021.e08615

[6] Дж. М. Ченло, Д. Э. Лосада. Эмпирическое исследование особенностей предложений для классификации субъективности и полярности. Информационные науки, 2014, Том 280, стр. 275-288. DOI:10.1016/j.ins.2014.05.009

[7] С. Приянка, Д. Гупта. Определение наилучшей комбинации функций для анализа настроений в отзывах клиентов. В процессе. ICACCI, Майсур, Индия, 2013, стр. 102-108. DOI:10.1109/ICACCI.2013.6637154

[8] Э. Кулумпис, Т. Уилсон, Дж. Мур. Анализ настроений в Twitter: хорошие, плохие и боже мой! В работе. ICWSM, Барселона, Испания, 2011,

стр. 538-541. URL-адрес: https://ojs.aaai.org/index.php/ICWSM/article/view/14185

[9] С. Сун, Х. Лю, А. Абрахам. Тегирование части речи в Твиттере с использованием скрытой предварительной классификации Модель Маркова. В процессе. IEEE SMC, Сеул, Южная Корея, 2021, стр. 1118-1123. DOI:10.1109/ICSMC.2012.6377881

[10] К. душ Сантуш, М. Гатти. Глубокие сверточные нейронные сети для анализа настроений коротких текстов. В трудах КОЛИНГА. Дублин, Ирландия, 2014. стр. 69-78. URL-адрес: https://aclanthology.org/C14-1008

[11] В. Яньнянь, С. Цюнь, С. Jiquan, бойы Х., Murtadha A., Zhanhuai литий г. машинного обучения для Аспект-уровень анализа настроений, 2019. Адрес: https://arxiv.org/abs/1906.02502

[12] Д. Тан, Ф. Вэй, Б. Цинь, Л. Донг, Т. Лю и др. Совместная система сегментации и классификации для анализа настроений. В Proc. EMNLP, Доха, Катар, 2014, стр. 477-487. DOI:10.3115/v1/D14-1054

[13] Ф. Х. Хан, С. Башир и У. Камар. ТОМ: Платформа для сбора мнений в Twitter с использованием гибридной схемы классификации. Системы поддержки принятия решений, 2014, Том 57, стр. 245-257. DOI:10.1016/j.dss.2013.09.004

[14] У. Чамлертват, П. Бхаттаракосол, Т. Рунгкасири, С. Харуечайясак. Получение информации о потребителях из Twitter с помощью анализа настроений. Журнал Universal Computer Science, 2012, Том.

18, стр. 973-992. URL: https://www.semanticscholar.org/paper/Discovering-Consumer-Insight из-Twitter-через-Chamlertwat-Bhattarakosol/b32c462e6a5821c62c852bb42a8730eff880f8cd

[15] Юлия Цветкова, Манаал Фаруки, Ван Линг, Гийом Лэмпл, Крис Дайер. Оценка Векторных представлений слов путем выравнивания в подпространстве. Институт языковых технологий Университет Карнеги-Меллона. Питтсбург, Пенсильвания, США, 2021. URL: https://aclanthology.org/D15-1243.pdf

[16] Томас Миколов, Кай Чен, Грег Коррадо, Джеффри Дин. Эффективная оценка представлений Word в векторном пространстве, 2014. URL: https://arxiv.org/abs/1301.3781

[17] А. Джоулин, Э. Грейв, П. Бояновски и Т. Миколов. Набор приемов для эффективной классификации текстов. в сборнике. EACL, Валенсия, Испания, 2017, стр. 427-431. URL: https://arxiv.org/abs/1607.01759

[18] С. Комбринк, Т. Миколов, М. Карафит и Л. Бурже. Рекуррентное языковое моделирование на основе нейронных сетей при распознавании встреч. В работе. INTERSPEECH, Флоренция, Италия, 2011, стр. 2877–2880. URL: https://www.semanticscholar.org/paper/Recurrent-Neural-Network-Based Языковое моделирование в Kombrink-Mikolov/b4fc91e543ec868658cde6170f1e59c33292e595

[19] Дж. Ченг, Х. Чжан, П. Ли, С. Чжан, З. Дин и др. Изучение анализа настроений в микроблогах тексты для опроса общественного мнения о китайских общественных деятелях. Applied Intelligence, 2016, Том 45, стр. 429–442. DOI:10.1007/s10489-016-0768-0

[20] М. Сандермайер, Р. Шлтер, Х. Ней. Нейронные сети LSTM для моделирования языка. В сборнике. INTERSPEECH, Портленд, США, 2012, стр. 194-197. DOI:10.21437/Interspeech.2012-65

[21] Р. Джонсон, Т. Чжан. Сверточные нейронные сети глубокой пирамиды для категоризации текста. В Proc. ACL, Ванкувер, Канада, 2017, стр. 562-570. DOI:10.18653/v1/P17-1052

[22] С. Ван, М. Хуан, З. Дэн. Тесно связанный CNN с многомасштабной функцией "внимание к тексту" классификация. В работе. IJCAI, Стокгольм, Швеция, 2018, стр. 4468-4474. URL-адрес: https://www.semanticscholar.org/paper/Densely-Connected-CNN-with-Multi-scale-Feature-for Ван-Хуан/35f0b854901dc6c5a69b271637d302f7db49b79a

[23] Л. Ян, Ю. Х. Чжэн, Дж. Цао. Обучение с помощью нескольких кадров для классификации коротких текстов. Мультимедийные инструментыи приложения, 2018, Том 77, стр. 29799-29810. DOI:10.1007/s11042-018-5772-4

[24] Л. Сян, С. Ян, Ю. Лю, К. Ли, С. Чжу. Новая лингвистическая стеганография, основанная на генерации текста на уровне символов. Математика, 2020, Том 8, стр. 1558. DOI:10.3390/math8091558

[25] З. Ян, С. Чжан, Ю. Ху, З. Ху, Ю. Хуан. VAE-Stega: лингвистическая стеганография, основанная на вариационном автокодере. IEEE Transactions on Information Forensics and Security, 2021, Том. 16, стр. 880-895. DOI:10.1109/TIFS.2020.3023279

[26] Л. Сян, Г. Го, К. Ли, К. Чжу, Дж. Чен и др. Обнаружение спама в обзорах с использованием многомерных временных функций на основе lstm. Интеллектуальная автоматизация и мягкие вычисления, 2020, Том 26, стр. 1375–1390. DOI:10.32604/iasc.2020.013382

[27] Луци Янь, Цзиньхань, Иши Юэ, Лю Чжан, Яннань Цянь. Анализ настроений коротких текстов Основан на параллельной DenseNet.

Компьютеры, материалы и континуум, 2021, Том 69, стр. 51-65. DOI:10.32604/cmc.2021.016920

[28] Пеннингтон Дж. Гловч. Глобальные векторы для представления слов. EMNLP, 2014, стр. 1532-1543. URL: https://nlp.stanford.edu/pubs/glove.pdf.

[29] Г. Хуан, З. Лю, Л. Ван Дер Маатен, К. К. Вайнбергер. Плотно связанные сверточные сети. в Proc. CVPR, Гавайи, США, 2017, стр. 4700-4708. URL-адрес: https://arxiv.org/abs/1608.06993

[30] Василюк А., Басюк Т. Особенности построения системы управления промышленной средой. Материалы 5-й Международной конференции по компьютерной лингвистике и интеллектуальным системам (COLINS 2021), Львов, Украина, 2021, Том 2870, стр. 1011-1025. URL: http://ceur ws.org/Vol-2870/paper76.pdf

[31] Басюк Т., Василюк А. Подход к созданию системы визуализации онтологий предметной области. Материалы 5-й Международной конференции по компьютерной лингвистике и интеллектуальным системам (COLINS 2021), Львов, Украина, 2021, Том 2870, стр. 528-540. URL: http://ceur ws.org/Vol-2870/paper39.pd