Внутренний цикл служит для реализации игровой логики и прорисовки объектов игры.

ЛИТЕРАТУРА

1. Эл Свейгард «Учим Python, делая крутые игры» — М.: Эксмо, 2018. — 418 с.

2. Сообщество Pygame : [Электронный ресурс] – Электронные данные. Режим доступа: https://www.pygame.org/news

3. Проект it-black.ru: [Электронный ресурс] – Электронные данные. Режим доступа: https://it-black.ru/

4. Свободная энциклопедия ВикипедиЯ: [Электронный ресурс] – Электронные данные. Режим доступа:https://ru.wikipedia.org

УДК 004.41

Магистрант Д.А. Радиванович
Науч. рук. проф. И. Г. Сухорукова
(кафедра программной инженерии, БГТУ)

ИСПОЛЬЗОВАНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ КЛАССИФИКАЦИИ ЭЛЕКТРОННОЙ КОРРЕСПОНДЕНЦИИ

Наиболее популярным и эффективным подходом для фильтрации электронной корреспонденции на данный момент является использование машинного обучения с учителем по различным признакам, основанными как на содержании сообщений, так и на свойствах отдельных профилей пользователей.

Машинное обучение – это область научного знания, имеющая дело с алгоритмами, «способными обучаться». Необходимость использования методов машинного обучения объясняется тем, что для многих сложных – «интеллектуальных» – задач очень сложно (или даже невозможно) разработать «явный» алгоритм их решения, однако часто можно научить компьютер обучиться решению этих задач. Одним из первых, кто использовал термин «машинное обучение», был изобретатель первой самообучающейся компьютерной программы игры в шашки А. Л. Самуэль в 1959 г [1]. Под обучением он понимал процесс, в результате которого компьютер способен показать поведение, которое в нее не было заложено «явно». Это определение не выдерживает критики, так как не понятно, что означает наречие ”явно”. Более точное определение дал намного позже Т. М. Митчелл: говорят, что компьютерная программа обучается на основе опыта E по отношению к некоторому классу задач T и меры качества P, если каче-

ство решения задач из T, измеренное на основе P, улучшается с приобретением опыта E [2].

На этапе классификации и оценки были протестированы пять алгоритмов: наивный байесовский классификатор (NaiveBayes classifier), метод k ближайших соседей (k-nearest neighbors algorithm, k-NN)), метод опорных векторов (SVM), дерево принятия решений (Decision tree), случайные леса (Random forest).

Для обучения было использовано 40% начальной выборки. В процессе оптимизации гиперпараметров для каждого классификатора использовалась кросс-валидация (CV-10). Кросс-валидация способствует минимизации риска переобучения, и, следовательно, смещения в оценке качества классификатора. Оптимизация проводилась для обоих групп признаков (каждый признак был нормализован и принимал значение в диапазоне [-1, 1]). Кроме этого для настройки SVM было отобрано 30% наиболее значимых признаков по критерию хи-квадрат, поскольку это более оптимально с точки зрения вычислительной нагрузки.

Далее приведены результаты оптимизации гиперпараметров с помощью полного перебора по сетке.

После оптимизации гиперпараметров каждый из алгоритмов оценивался на оставшихся 60% данных с использованием обоих групп признаков и CV-10. Также как и в случае поиска по сетке, каждый признак был нормализован, а для SVM использовалось 30% наиболее значимых признаков по критерию хи-квадрат.

В качестве метрик оценки классификаторов были выбраны полнота (R), точность (P) и F-мера. Метрика полноты представляет из себя отношение числа сообщений, корректно классифицированных как спам (True Positives) и общего числа спамовых сообщений (True Positives + False Negatives):

$$R = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (1)$$

Точность − это отношение числа сообщений, корректно классифицированных как спам (True Positives) и общего числа сообщений, классифицированных как спам (True Positives + False Positives):

$$P = \frac{T\,rueP\,ositives}{T\,rueP\,ositives + F\,alseP\,ositives} \quad (2)$$

F-мера может быть проинтерпретирована как гармоническое среднее между метриками точности и полноты:

$$F1 = \frac{2PR}{P+R} \quad (3)$$

Как видно из таблицы 1, классификаторы на основе дерева показали лучший результат. Random Forests обошел остальные классификаторы по показателю F-показателю F-меры. Наилучший известный мне результат в данной задаче был достигнут у Lee [3] и составляет 98% по той же оценке F-меры. Однако Lee использовал исторические признаки, что является ограничением в нашем случае, следовательно, результат можно считать вполне неплохим.

Таблица 1 — Сравнение показателей классификаторов

Классификатор	**Точность**	**Полнота**	**F-мера**
Naive Bayes	76%	76%	76%
k-NN	86%	86%	86%
SVM	86%	86%	86%
Decision Tree	89%	89%	89%
Random Forest	**93%**	**93%**	**93%**

В результате работы были исследованы 5 алгоритмов машинного обучения: наивный байесовский классификатор, метод k ближайших соседей, метод опорных векторов, решающее дерево и случайный лес. Лучший результат продемонстрировал алгоритм Random Forest - 93%. Наиболее высокий известный в данной задаче составляет 98%, однако в нем отсутствуют какие-либо ограничения на историчность признаков.

ЛИТЕРАТУРА

1. Mitchell,ThomasM.MachineLearning/ThomasM.Mitchell.— 1edition.—New York, NY, USA: McGraw-Hill, Inc., 1997.

2. Samuel, Arthur L. Some studies in machine learning using the game of checkers / Arthur L. Samuel // IBM JOURNAL OF RESEARCH AND DEVELOPMENT. — 1959. — P. 71–105

3. Lee,Kyumin.Sevenmonthswiththedevils:a long-term study of content polluters on twitter / Kyumin Lee, Brian David Eoff, James Caverlee // In AAAI Int'l Conference on Weblogs and Social Media (ICWSM. — 2011.