УДК 517.518.23

Анализ тональности текстов на основе методов машинного обучения.

Нгуен Зы Хоан

Московский физико-технический институт (государственный университет)

Аннотация: Рассматривается проблема автоматической классификации текстов по тональности, описываются методы машинного обучения для решения этой проблемы. Приводится описание алгоритмов классификации: наивного Байесовского классификатора, метода ближайших соседей. Рассматриваются способы векторного представления документов обучающей и тестовой выборок, а также функции весов. Для каждого сочетания параметров рассчитываются метрики эффективности. Для оценки работы алгоритмов используется метод перекрестной проверки. По результатам проверок, выбирается сочетание векторной модели языка, функции весов и классификатора с наибольшей оценкой точности.

Ключевые слова: классификатор, машинное обучение, анализ тональности.

Введение

Активное развитие в настоящее время социальных сетей, блогов и форумов привело к увеличению интереса, как со стороны научного сообщества, так и со стороны многих организаций к задаче автоматического анализа мнений пользователей Интернета по различным вопросам (отношение к товарам, услугам, событиям, высказываниям). Одной из основных проблем при анализе мнений является классификация текстов по тональности. Тональностью текста называется эмоциональная оценка, выраженная в тексте по отношению к некоторому объекту, и определяется тональностью составляющих его лексических единиц и правилами их сочетания. В простейшем случае классификация текстов по тональности осуществляется на два класса, обозначающие позитивные и негативные эмоциональные оценки.

1. Подходы к определению тональности текстов.

Методы классификации текстов по тональности могут быть разделены на следующие классы [1]:

1. На основе правил с использованием шаблонов [2]. Подход заключается в генерации правил, на основе которых будет определяться тональность текста. Для этого текст разбивается на слова или последовательности слов (n-grams). Затем полученные данные используются для выделения часто встречающихся шаблонов, которым присваивается положительная или отрицательная оценка. Выделенные

шаблоны применяются при создании правил вида «ЕСЛИ *условие*, ТО *заключение*»;

2. Машинное обучение без учителя[3]. Данный подход основан на идее, что наибольший вес в тексте имеют термины, которые чаще встречаются в этом тексте и в то же время присутствуют в небольшом количестве текстов всей коллекции. Выделив данные термины и определив их тональность, можно сделать вывод о тональности всего текста;
3. Машинное обучение с учителем [4]. В этом подходе требуется наличие обучающей коллекции размеченных в рамках эмотивного пространства текстов, на базе которой строится статистический или вероятностный классификатор (например, байесовский);
4. Гибридный метод [5]. Данный подход сочетает все или несколько из рассмотренных выше принципов и заключается в применении классификаторов на их основе в определенной последовательности.

2. Обзор литературы

Котельников Е.В. и Клековкина М.В. в своем исследовании [6] представили методы автоматической обработки текстов на основе методов машинного обучения. В ходе своего исследования они пытались отыскать оптимальный вариант векторной модели представления текстов и наилучшего классификатора. Исследователи проанализировали несколько векторных моделей на основе подхода TF.IDF (TF — Term Frequency, IDF — Inverse Document Frequency). Были рассмотрены несколько методов машинного обучения, в том числе наивный Байесовский классификатор, метод опорных векторов (Support Vector Machine или SVM), метод ключевых слов и т.д. Экспериментальным путем было выявлено, что наилучшие результаты дает бинарная модель с косинусной нормализацией без обучения и метод, комбинирующий использование ключевых слов и SVM.

В исследовании Tang C. и соавт. [7] предлагаются два метода типа обучение с учителем для выявления характеристик продукта или услуги и классификации этих характеристик для составления краткого резюме отзывов потребителей. Результаты их работы показывают, что каждый метод хорош по-своему, это зависит от конкретной метрики эффективности. Работа этих методов обычно достигает более 70% точности.

Как было указано выше, исследователи часто комбинируют подходы для достижения наилучших результатов. Например, научная работа Васильева В.Г., Давыдова С. и Худяковой М.В. [8] использует лингвистический подход, дополенный методами машинного обучения для коррекции отдельных правил классификации путем обучения.

Большинство исследователей в своих работах полагаются на лингвистический подход, и этому есть разумное объяснение. Алгоритмы, основанные на правилах, дают более точные результаты, так как работа этих методов тесно связана с семантикой слов, в отличие от методов машинного обучения, оперирующих со статистикой и теорией веороятности. Но, как уже было упомянуто, лингвистический подход обладает рядом серьезных недостатков.

Лингвистический подход может предоставить относительно точные результаты, будучи реализованным для научных или журнальных статей или других, грамматически верных текстов. Принимая во внимание тот факт, что одно из главных применений анализа тональности – бизнес-разведка, становится понятно, что интернет-сообщество не может обеспечить исследователей грамматически правильными текстами, или даже текстами без орфографических ошибок. В этой связи, не стоит и говорить о грамматике и стиле письма рядовых пользователей социальных сетей.

Кроме того, подход, основанный на правилах, сильно привязан к конкретному языку. В связи с вышесказанным, в настоящем исследовании рассматривается подход, основанный на методах машинного обучения с учителем.

3. Исследование и построение решения задачи

Решение этой задачи можно разбить на этапы:

1) Выбор метрик для оценки эффективности алгоритмов.
2) Выбор признаков, по которым будет осуществляться классификация.
3) Выбор и реализация нескольких алгоритмов классификации.
4) Создание обучающей выборки.
5) Проверка эффективности алгоритмов.

3.1. Метрики эффективности

В качестве метрик правильности классификации текстов были выбраны точность (precision) и полнота (recall). Точность в пределах класса – это доля текстов, действительно принадлежащих данному классу, относительно всех текстов, причисленных классификатором к этому классу. Полнота системы – отношение числа найденных классификатором текстов, принадлежащих классу, к числу всех текстов этого класса в тестовой коллекции.

В результате классификации текстов рецензий тестовой выборки, к классу позитивных (положительных) рецензий правильно отнесены TP текстов, неправильно – FP, к классу негативных (отрицательных) рецензий правильно были отнесены TN текстов, неправильно – FN. Иными словами:

- TP — истинно-положительное решение.
- TN — истинно-отрицательное решение.
- FP — ложно-положительное решение.
- FN — ложно-отрицательное решение.

Тогда, относительно класса позитивных рецензий, точность *Precision* и полнота *Recall* определяются следующим образом:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

3.2. Векторная модель языка

Вектор признаков – это алгебраическая модель представления текстов:

$$d_j = (w_{1j}, w_{2j}, \ldots, w_{nj}),$$

где d_j - векторное представление документа j, w_{ij} – вес термина i в документе j и n – количество всех терминов в выборке. Для полноценного понимая принципа векторной модели текста, необходимо объяснить, как именно рассчитываются веса векторов. Существует несколько базовых функций весов. Pang и соавт. в своем исследовании [9] выявили, что бинарная функция взвешивания векторов более эффективна. Это значит, что наличие термина в документе важнее, чем его частота. Бинарные векторы представлены как последовательность нулей и единиц: если конкретный термин из словаря выборки встречается в тексте – вес термина будет равен 1, иначе – 0. Частотные векторы формируются на основе количества вхождений определенного термина в классе документов.

3.3. Классификаторы

3.3.1. Наивный Байесовский классификатор

Наивный Байесовский классификатор (Naïve Bayes Classifier, NBC) является одним из примеров использования методов векторного анализа. Данная модель классификации базируется на понятии условной вероятности принадлежности документа *d* классу *c*.

NBC – один из самых часто используемых классификаторов, из-за сравнительной простоты в имплементации и тестировании. В то же время, наивный Байесовский классификатор демонстрирует не худшие результаты, по сравнению с другими, более сложными классификаторами.

В основе наивного Байесовского классификатора лежит теорема (или формула) Байеса:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Для данной модели, документ – это вектор: $d = \{ w_1 , w_2 ,..., w_n \}$, где w_i - вес i-ого термина, а n – размер словаря выборки. Таким образом, согласно теореме Байеса, вероятность класса c для документа d будет:

$$P(c \mid d) = \frac{P(c)P(d \mid c)}{P(d)}$$

Таким образом, вычисляется условная вероятность для всех классов.

Наиболее вероятный класс c^*, которому принадлежит документ d тот, при котором условная вероятность принадлежности документа d классу с максимальна:

$$c^* = \arg\max_c P(c \mid d)$$

По теореме Байеса:

$$c^* = \arg\max_c P(\mathrm{d} \mid \mathrm{c}) * \mathrm{P(c)}$$

и, так как

$$d = \{w_1, w_2, ..., \mathrm{w}_n\}$$

то

$$c^* = \arg\max_c \mathrm{P}\{w_1, w_2, ..., \mathrm{w}_n \mid \mathrm{c}\} * \mathrm{P(c)}$$

Знаменатель может быть опущен, так как для одного и того же документа d вероятность $P(d)$ будет одинаковой, а это значит, что ее можно не учитывать.

Для наивного Байесовского классификатора определено существенное допущение – предполагается, что все признаки $x_1 , x_2 ,..., x_n$ документа d независимы друг от друга. Из-за этого допущения модель и получила название «наивная». Это очень серьезное упрощающее допущение и, в общем случае, оно неверно, но наивная Байесовская модель демонстрирует неплохие результаты, несмотря на это [10]. Предполагается так же, что позиция термина в предложении не важна. Как следствие, условную вероятность $\mathrm{P}\{w_1, w_2, ..., \mathrm{w}_n \mid \mathrm{c}\}$ для признаков $x_1 , x_2 ,..., x_n$, можно представить как

$$P(w_1 \mid c) * (w_2 \mid c) * ... * (w_n \mid c) = \prod_i P(w_i \mid c_j)$$

Таким образом, для нахождения наиболее вероятного класса для документа $d = \{w_1, w_2, ..., \mathrm{w}_n\}$ с помощью наивного Байесовского классификатора, необходимо посчитать

условные вероятности принадлежности документа *d* для каждого из представленных классов отдельно и выбрать класс, имеющий максимальную вероятность:

$$C_{NB} = \arg\max_c [P(c_j) * \prod_i P(w_i \mid c_j)]$$

Теперь необходимо оценить $P(c_j)$ и $P(w_i \mid c_j)$. Оценить вероятность класса несложно : $P(c_j)$ является отношением количества докуметов класса j в обучающей выборке к общему количеству документов в выборке.

$$P(c) = \frac{D_c}{D},$$

где D_c - количество документов класса c, а D – общее количество документов в выборке.

Для оценки условных вероятностей для признаков, используется формула:

$$\hat{P}(w_i \mid c_j) = \frac{count(w_i, c_j)}{\sum_{w \in V} count(w, c_j)},$$

где $P(w_i \mid c_j)$ определяется как отношение количества терминов w_i в классе c_j общему количеству терминов в этом классе. V – словарь обучающей выборки.

Существует небольшая проблема, связанная с этой формулой. Если в тестовом наборе встретится слово, которое не встречается в наборе обучающих документов, то вероятность $P(w_i \mid c_j)$ этого слова для любого из классов будет равна нулю. Поскольку $P(d \mid c_j) \approx \prod_i (w_i \mid c_j)$, то и вероятность принадлежности документа любому из классов также будет равна нулю, что, конечно, неправильно. Для решения этой проблемы обычно используют так называемое аддитивное сглаживание. Идея сглаживания заключается в том, что к частотам появления всех терминов из словаря искуственно добавляется единица. Получается, что термины, которые не присутствовали в документах обучающей выборки, получают незначительную, но не нулевую вероятность появления и, тем самым, дают возможность определить документ в какой-либо из классов.

$$\hat{P}(w_i \mid c) = \frac{count(w_i, c) + 1}{\sum_{w \in V} (count(w, c) + 1)} = \frac{count(w_i, c) + 1}{(\sum_{w \in V} (count(w, c)) + |V|}$$

Здесь $|V|$ - количество слов в словаре обучающей выборки.

Как было сказано выше, документ представляется в виде вектора для классификации: $d = \{w_1, w_2, ..., w_n\}$, где $w_1, w_2, ..., w_n$ - веса для каждого из терминов словаря выборки. w_i может быть количеством вхождений термина x_i в документ d, или же может быть задано бинарно. Для бинарного вектора число вхождений термина w_i не имеет значения, важен лишь факт появления w_i в документе d.

Можно заметить, что для относительно больших текстов, вероятность $P(c_j \mid d)$ представляет собой произведение большого количества очень маленьких дробей. Для того чтобы избежать потери точности, можно заменить произведение вероятностей суммой логарифмов вероятностей.

3.3.2. Метод ближайших соседей

Метод ближайших соседей – еще один алгоритм классификации текстов. Для его реализации нужна обучающая выборка размеченных рецензий. Для определения класса рецензии из тестовой выборки, нужно определить расстояние от вектора этой рецензии до векторов из обучающей выборки. Определить k объектов обучающей выборки, расстояние до которых минимально (k задается экспертом или выбирается согласно оценкам эффективности). Класс входного вектора – это класс, которому принадлежат больше половины из соседних k векторов. В качестве функции расстояния было выбрано Евклидово расстояние:

$$\rho(x, x') = \sqrt{\sum_{i}^{n} (x_i - x_i')^2}$$

3.4. Создание обучающей выборки

Для создания решений, основанных на методах машинного обучения, требуется размеченная обучающая выборка. От подбора выборки напрямую зависит качество классифицирующего алгоритма. Для максимизации значений оценок эффективности, обучающая выборка составляется из текстов той же предметой области, для которой будет применяться классификатор.

3.5. Тестирование работы алгоритма

Для тестирования алгоритмов определение тональности текстов рецензий был использован метод перекрестной проверки. Процедура перекрестной проверки выполняется следующим образом:

1) Фиксируется множество разбиений обучающей выборки на собственно обучающую и тестовую.
2) Для каждого разбиения происходит обучение алгоритма на обучающей подвыборке и тестирование на тестовой.
3) Результатом перекрестной проверки алгоритма являются средние значения оценок эффективности для тестовых подвыборок.

4. Эксперименты и результаты

Коллекция обучающих текстов содержит отзывы пользователей по фильмам[11].

Таблица 1. Наивный Байесовский классификатор

Векторная модель	**Униграммы**		**Биграммы**	
Веса векторов	**Точность, %**	**Полнота, %**	**Точность, %**	**Полнота, %**
Бинарные векторы	80,1	84,5	83,7	92,5
Частотные векторы	82,2	83,4	85,2	90,4

Таблица 4. Метод k ближайших соседей (для k = 3)

Векторная модель	**Униграммы**		**Биграммы**	
Веса векторов	**Точность, %**	**Полнота, %**	**Точность, %**	**Полнота, %**
Бинарные векторы	70,6	75,5	69,3	80,5
Частотные векторы	68,7	78,5	67,8	92,1

Наивный Байесовский классификатор показал очень хорошие результаты, наиболее эффективной конфигурацией в плане точности оказалось сочетание биграмм и частотной функции взвешивания. Были получены довольно низкие результаты для метода k ближайших соседей.

5. Заключение

В ходе работы была реализована программа анализа тональности текстов на основе методов машинного обучения.

Были решены следующие задачи:

1. Изучена проблема анализа тональности, проанализированы подходы ее решения.
2. Реализовано два алгоритма классификации текстов по тональности (наивный Байесовский классификатор и алгоритм k ближайших соседей).
3. Выбраны метрики и вычислены оценки эффективности алгоритмов.

4. Проведено тестирование эффективности алгоритмов методом перекрестной проверки.
5. Для каждого алгоритма было выбрано сочетание векторной модели языка (биграммы, униграммы) и функции весов (бинарная, частотная), которое дало наилучший показатель точности на данной выборке.

Для повышения оценок эффективности алгоритмов, предположительно, следует дополнить их элементами лингвистического анализа. Исследования [8] показывают, что лучшие результаты достигаются путем комбинирования лингвистического и статистического подходов.

Литература

1. *Prabowo R., Thelwall M.* Sentiment analysis: A combined approach // Journal of Informetrics, Vol. 3, No. 2. (April 2009), pp. 143-157.
2. *Liu H. MontyLingua*: An end-to-end natural language processor with common sense, 2004. Available at <http://web.media.mit.edu/hugo/montylingua> (accessed 1 February 2005).
3. *Turney P.* Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews // Proceedings of ACL-02, 40th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2002, pp. 417–424.
4. *Joachims T.* Making large-scale SVM learning practical // In B. Schölkopf, C. J. C. Burges, & A. J.Smola (Eds.), Advances in kernel methods: support vector learning, 1999. The MIT Press.
5. *König A. C. & Brill, E.* Reducing the human overhead in text categorization // In Proceedings of the 12th ACM SIGKDD conference on knowledge discovery and data mining, August 20–23, 2006, pp. 598–603.
6. *Котельников Е.В., Клековкина М.В.* Автоматический анализ тональности текстов на основе методов машинного обучения. РОМИП 2011.
7. *Tang, X. Yang, C., Wong, Y., Wei C.* Understanding Online Consumer Review Opinions with Sentiment Analysis using Machine Learning // Pacific Asia Journal of the Association for Information Systems. – 2010. - № 3(2). – С. 73-89.
8. *Худякова М.В., Давыдов С., Васильев В.Г.* Классификация отзывов пользователей с использованием фрагментных правил. РОМИП 2011.
9. *Pang L.* Thumbs up? Sentiment Classification using Machine Learning Techniques // Proceedings of EMNLP (2002).
10. *Domingos P. & Pazzani, M.* On the optimality of the simple Bayesian classifier under zero-one loss // Machine Learning. - 1997. - № 29. – С. 103-137.
11. http://ai.stanford.edu/~amaas/data/sentiment/