УДК 004.852 DOI: 10.26102/2310-6018/2020.28.034

Классификация потоковых данных на основе байесовского критерия

Л.С. Ломакина, А.Н. Субботин
Нижегородский государственный технический университет им. Р.Е. Алексеева, Нижний Новгород, Россия

Аннотация: Рассматривается проблема классификации потоковых данных, поступающих из различных источников в случайные моменты времени. Это может быть поток данных, содержащих результаты измерения датчиков, расположенных в прибрежных зонах океанов, позволяющий оценивать параметры состояния экосистемы, и также поток текстов, получаемых, например, в письмах электронной почты и т. д. Интернет содержит большие объёмы неструктурированной информации, отсутствие организации которых делает работу с данными неудобной и ресурсоёмкой. Преодоление указанного недостатка является актуальной задачей. Классификация служит инструментом, позволяющим облегчить работу с неструктурированной информацией. Разработан алгоритм классификации потоковых данных на основе байесовского критерия. Построена математическая модель потоковых текстовых данных, позволяющая применять алгоритмы классификации текстов на естественном языке на потоковых данных. Предложена модификация наивного байесовского классификатора, использующая характеристику tf-idf как меру принадлежности терминов классам, позволяющая улучшить качество классификации. Классификатор был обучен с помощью машинного фонда русского языка. Разработано программное обеспечение, позволяющее извлекать потоковые текстовые данные из сети Интернет и производить классификацию разработанным алгоритмом в реальном времени.

***Ключевые слова*:** классификация, классификатор, поток данных, байесовский критерий, байесовский классификатор.

Для цитирования: Ломакина Л.С., Субботин А.Н. Классификация потоковых данных на основе байесовского критерия. *Моделирование, оптимизация и информационные технологии.* 2020;8(1). Доступно по: https://moit.vivt.ru/wp-content/uploads/2020/02/LomakinaSubbotin_1_20_1.pdf DOI: 10.26102/2310-6018/2020.28.1.034

Stream data classification based on bayesian criteria

L.S. Lomakina, A.N. Subbotin
Nizhny Novgorod State Technical University n.a. R.E. Alekseev, Nizhny Novgorod, Russia

Abstract: The paper describes the issue of stream data classification. Stream data is described as a set of objects arriving from different sources at random moments of time. It might be a stream of data containing ocean coastal area sensors measure information and describing the parameters of the ecosystem condition, as well, it might be a stream of texts acquired from incoming emails attachments, etc. The Internet contains vast volumes of unstructured information. The lack of organization makes data inconvenient and resource-intensive to work with. Addressing to such an issue considered to be a relevant problem. Classification provides an opportunity to make it easier to work with unstructured information. The paper describes the algorithm for stream data classification based on Bayesian criteria. Text stream data model is proposed. This model allows applying natural language text classification algorithms to stream data. Naive Bayes classifier modification using tf-idf measure for evaluating the

proximity of a classified document to a particular class that allows improving the classification quality is proposed. The classifier has been trained using the machine Fund of the Russian language. Software allowing text data stream extraction from the Internet and its classification using the proposed algorithm in real-time scale is proposed

Keywords: classification, data stream, naive Bayesian classifier, Bayesian criteria.

***For citation*:** Lomakina L.S., Subbotin A.N. Stream data classification based on Bayesian criteria. *Modeling, optimization and information technology*. 2020;8(1). Available by: https://moit.vivt.ru/wp-content/uploads/2020/02/LomakinaSubbotin_1_20_1.pdf DOI: 10.26102/2310-6018/2020.28.1.034 (In Russ.).

Введение

В настоящее время в связи с развитием цифровых технологий количество информации растёт экспоненциально. С течением времени растет число как потребителей, так и источников информации разного характера. В связи с этим, становятся все более актуальными проблемы, связанные с поиском, фильтрацией, рубрикацией, кластеризацией и аннотированием информации.

Интернет содержит большие объемы информации. Существуют миллионы сайтов, содержащих информацию самых разных направленности и характера. Такое разнообразие неизбежно влечет сложности работы с этой информацией, поскольку способы организации данных в Интернете по большому счету ничем строго не регламентированы. Очевидно, работа со структурированной информацией требует намного меньше ресурсов. Алгоритмы машинного обучения позволяют производить автоматизированную обработку данных с целью повышения ее структурированности. Произведение классификации в данном контексте позволяет классифицировать данные и тем самым многократно уменьшить время работы по сравнению с неструктурированными данными.

Текстовая информация на данный момент является одним из самых распространённым типом данных. Также, текстовые данные относительно просты для машинного представления, что упрощает получение и обработку текстовых данных на практике. Поэтому, в контексте данной работы будем работать именно с текстовыми данными. Однако, рассматриваемые положения применимы также и к классификации любых других объектов.

В данной работе в качестве входных данных мы рассматриваем текстовую информацию, которая подается на вход программного обеспечения перед выполнением алгоритмов классификации с течением времени. Входными данными могут считаться поток текстовой информации, а также обучающая коллекция – совокупность элементов (в данном случае текстов) с заведомо известными классами, к которым они относятся. Поток текстовой информации в данной работе представляется как совокупность текстовых сообщений, принимаемых в случайные моменты времени. В первых двух случаях массивы элементов подаются на вход единовременно, сразу же происходит их обработка, и выдаётся результат (обученный классификатор или качественная оценка работы алгоритма, выводимая в удобочитаемом виде). В случае, если входными данными является текстовый поток, классификация элементов будет производиться во время получения элементов.

В работе в качестве потока тестовых сообщений рассматривается совокупность новостных статей, создающихся в случайные моменты времени на определённых сайтах в сети Интернет. С помощью специального алгоритма отслеживаются новые текстовые статьи на этих сайтах, их содержимое получается, образуя таким образом элементы потока текстовых сообщений.

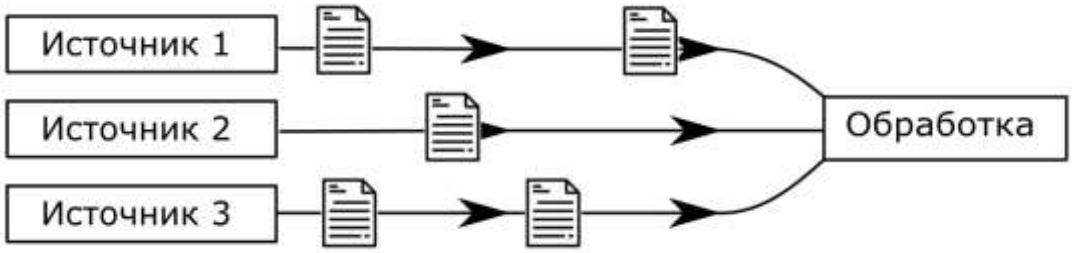

Рисунок 1 – Обработка потоков данных
Figure 1 – Data stream processing

Развитие программных средств позволяет получать информационные потоки из практически любых источников. Это может быть, к примеру, поток данных, содержащих измерения датчиков, расположенных в прибрежных зонах океанов, позволяющий оценивать многие параметры состояния данной экосистемы, также это может быть поток текстов, получаемых в качестве прикреплений в письмах электронной почты на определённом почтовом ящике, другими словами, алгоритм получения входных данных в качестве потока в данной работе можно применить практически под любые контексты, предусматривающие получение элементов из одного или нескольких источников в случайные моменты времени [1].

Задача классификации текстовых документов состоит в определении класса, к которому принадлежит тот или иной текстовый документ. Характеристика, по которой классифицируется текст может быть различной. К примеру: стиль повествования текста – художественный, технический, публицистический, неформальный и т. д.; степень орфографической правильности текста – классификация по оценкам: отлично, удовлетворительно, т. д.; однако самой актуальной и обсуждаемой является классификация по тематике текста. Именно по этой характеристике производится обработка текста в данной работе. Классом далее будет называться общность, к которой в результате классификации будет отнесён входной документ по тематическому признаку.

В связи с ростом количества текстовой информации все более актуальным становится такой объект, как информационный поток. Особую ценность при этом представляет не столько информация сама по себе, сколько смысл и идеи, заключенные в большом количестве документов.

Понятие потоков текстовой информации можно применить к тестовым сообщениям, документам, текстам, создающимся и распространяющимся во всевозможных сервисах, сайтах, социальных сетях сети Интернет.

Как отмечается в [3], одной из важных особенностей, отличающей алгоритмы анализа текстов и алгоритмы анализа текстовых потоков является высокая скорость поступающих данных. Тексты новостных статей, образующих текстовый поток создаются в случайные моменты времени, поэтому неизвестно какой объем информации должен быть обработан в определенные моменты времени. Это предъявляет алгоритмам работы с текстовыми потоками требования быстродействия и вынуждает отказаться от относительно сложных алгоритмов. Поэтому в задаче анализа и обработки потоков можно говорить о требовании к оперативности алгоритма, что в данном случае определяет способность к обработке данных по мере их поступления в реальном времени.

Теоретический анализ

Так как поток текста представляется как текстовые сообщения, получаемые в случайные моменты времени, имеет смысл говорить о модели текстового сообщения. Согласно [4], входные данные для алгоритма классификации – это не коллекция документов, а образы каждого документа:

$$\vec{D} = \{\vec{d_i}\}, i = \overline{1, |D|},$$

где $\vec{D}$ – это образ документа D.

Рассмотрим возможные модели текстовых сообщений. Самые распространенные из них это:

1) Мультимножества терминов текстов.

2) Векторная модель терминов текстов.

Здесь термин – это слово, находящееся в тексте по крайней мере один раз, за исключением часто используемых слов, не несущих значимой информации о тематике текста. Важно заметить, что слова, встречающиеся в разных падежах и склонениях, относятся к одному термину, т. е., слова ‘моделям’ и ‘моделей’ представляют собой разные формы одного термина – ‘модель’.

Использование модели мультимножеств подразумевает, что все термины, встречающиеся в документе, являются элементами пересекающихся множеств, сформированных по некоторым правилам.

В данной работе предлагается использовать более простую, векторную модель, в рамках которой документ представляется в виде вектора, содержащего значения, описывающие некоторые характеристики применяемых в документе терминов.

Тогда моделью описания документа с множеством терминов T будет вектор

$$\vec{d_l} = (d_{i1}, \dots, d_{i|T|})^T,$$

содержащий значения, описывающие частотные характеристики терминов документа.

В рассматриваемом алгоритме классификации используется байесовский критерий. Опишем применение этого критерия в задаче классификации текста подробнее. Согласно формуле Байеса, при условии, что произошло некоторое событие B, вероятность некоторого события A, статистически взаимосвязанного с событием B рассчитывается как:

$$P(B|A) = \frac{P(B)P(A|B)}{P(A)} \quad (1)$$

Допустим, имеется набор из n условий, вероятность наступления каждого из которых – p_i, $i = 1..n$. Примем выполнение некоторого условия из этого набора за событие B_i. Тогда, вероятность наступления события A, зависящего от этого набора условий, при условии наступления события B_i равна

$$P(B_i|A) = \frac{P(B_i)P(A|B_i)}{P(A)} \quad (2)$$

Если нам известны вероятности наступления каждого из событий B_i (вероятности до опыта, или априорные вероятности) и вероятность наблюдаемого события A, доставляющее информацию о реализованном событии B_i, вероятности $P(B_i)$ могут быть скорректированы с помощью формулы Байеса, т. е. мы можем вычислить апостериорную вероятность $P(B_i|A)$ события B_i. Пусть существует множество объектов D, и множество классов C. Тогда, если необходимо произвести классификацию нового

объекта, используя доопытную вероятность принадлежности этого объекта классу c_i апостериорная вероятность принадлежности объекта d_i классу c_j находится как:

$$P(c_j \mid d_i) = \frac{P(c_j)P(d_i|c_j)}{P(d_i)}, \quad (3)$$

где $P(d_i|c_j)$ – шанс встретить текст d_i среди текстов класса c_j.

Целью классификации будет является поиск самого подходящего класса для документа, т. е. поиск самой большой вероятности $P(c_j \mid d_i)$ [5].

Таким образом, процедура классификации сводится к отысканию класса cj, для которого упомянутое выражение будет иметь максимальное значение:

$$c_{res} = argmax\, P(c_j \mid d_i) = \arg\max \frac{P(c_j)P(d_i|c_j)}{P(d_i)}. \quad (4)$$

Таким образом, для документа d_1 будут произведены следующие расчеты:

$$P(c_1 \mid d_1) = \frac{P(c_1)P(d_1|c_1)}{P(d_1)},$$

$$P(c_2 \mid d_1) = \frac{P(c_2)P(d_1|c_2)}{P(d_1)},$$

…

$$P(c_n \mid d_1) = \frac{P(c_n)P(d_1|c_n)}{P(d_1)}, n = |C|.$$

Очевидно, что знаменатель имеет постоянное значение $P(d_1)$ и не влияет на максимальное значение.

Таким образом, формула (4) выражение приводится к виду:

$$c_{res} = \arg\max P(c_j)P(d_i|c_j) \quad (5)$$

Априорные вероятности классов $P(c_j)$ рассчитываются как отношение количества документов обучающей коллекции в классе c_j к общему числу документов коллекции:

$$P(c_j) = \frac{|c_j|}{|C|}$$

Чем популярней тема – тем больше документов этой темы в обучающей коллекции и тем больше вероятность, что очередной документ будет относится именно к ней.

$P(d_i|c_j)$ в данной формуле – вероятность того, что документ d_i принадлежит классу c_j. Чтобы оценить эту вероятность, будем использовать вероятности встретить термины, входящие в документ d_i в документах класса c_j. Рассматривая документ как сущность, содержащую термины без учёта их порядка и вероятностных взаимоотношений, мы можем оценить вероятность принадлежности документа d_i классу c_j как перемножение вероятностей вхождения в класс c_j терминов t_k, составляющих документ d_i.

$$P(d_i|c_j) = \prod_{k=1}^{|T|} P(t_k|c_j), t_k \in d_i$$

Здесь $P(t_k|c_j)$ – вероятность встретить термин t_k в документах класса c_j.

Таким образом, формула (5) приводится к виду

$$c_{res} = \arg\max\left(P(c_j)\prod_{k=1}^{|T|} P(t_k|c_j)\right), t_k \in d_i \quad (6)$$

Методика

В данной статье в качестве оценки вероятности принадлежности некоторого термина к определенному классу, предлагается использовать модифицированный алгоритм байесовской классификации, использующий меру term frequency – inverse document frequency (далее tf-idf) [5].

Мера tf-idf это произведение меры частотности термина и меры обратной частотности термина.

Частотность термина (term frequency, далее tf) – описывает частоту применения некоторого термина в тексте и описывается формулой

$$tf(t,f) = \frac{n_t}{\sum_k n_k},$$

где n_t – количество исходных словоформ термина в документе,

$\sum_k n_k$ – количества слов документа.

Обратная частота термина (inverse document frequency, далее idf) – описывает редкость применения термина в документах всей обучающей коллекции. Эта величина позволяет определить является ли термин часто используемым. В случае если является – его значение для задачи классификации мало, и мера tf-idf, соответственно, тоже будет малой [2]. Данная характеристика описывается формулой

$$idf(t,C) = \log\frac{|C|}{d_i \supset t},$$

где |C| – количество терминов в обучающей коллекции классификатора,

$d_i \supset t$ – количество документов, в которых встречается термин t.

Наивный Байесовский классификатор относится к алгоритмам машинного обучения с учителем, поэтому обучение классификатора – это неотъемлемая часть рассматриваемого алгоритма. Рассмотрим стадию обучения модифицированного байесовского классификатора. Данная стадия состоит из нескольких этапов:

1. Подготовка текста – удаление символов и неинформативных частей речи, стемминг всех документов коллекции.
2. Поиск и удаление общих стоп-слов.
3. Расчет априорных вероятностей для каждой тематики: чем популярней та или иная тематика, тем больше вероятность, что очередной документ принадлежит именно ей.
4. Расчет частотных характеристик терминов. Данный содержит 4 подэтапа.

4.1 Устранение слов с низкой частотой употребления в масштабе каждого документа.

4.2 Расчет частоты употребления терминов в масштабе документа.

4.3 Расчет частоты употребления терминов в масштабе всей коллекции.

4.4 Расчет характеристики tf-idf для каждого термина на основе ранее рассчитанных характеристик.

Рассчитанные в результате алгоритма обучения частотные характеристики являются мерами вероятности принадлежности терминов к определенным тематикам. Эти характеристики будут использоваться при классификации.

В результате алгоритма обучения рассчитываются меры принадлежности термина каждому из возможных классов. Эти расчеты применяются далее на стадии классификации.

Стадия классификации также состоит из нескольких этапов:

1. Нормализация текста документа.
2. Удаление общих стоп-слов из содержимого документа.
3. Расчет частотности терминов документа.
4. Расчет апостериорной вероятности принадлежности документа к каждому к классу по формуле (6). Рассчитанные частотные характеристики терминов используется как мера вероятности принадлежности каждого термина документа к каждому классу. В результате классом, которому принадлежит классифицируемый документ признается класс, для которого рассчитанное значение будет максимальным [6].

Результаты

В рамках работы было реализовано программное обеспечение [7], позволяющее выполнять рассматриваемый алгоритм, включающий обучение классификатора и алгоритм непосредственно классификации. В практической реализации в качестве обучающей выборки использовался массив статей с новостями из газет 90-х годов на русском языке, а классифицировались статьи новостных сайтов в реальном времени.

Для оценки качества результатов работы алгоритма была проведен эксперимент. В рамках данного эксперимента была произведена классификация документов из тестовой выборки коллекции текстовых документов с помощью двух алгоритмов наивной байесовской классификации: традиционного (НБК) и модифицированного (МНБК), с разными объемами обучающих выборок. Для оценки качества классификации использовались следующие критерии: полнота, точность и F1-мера.

Таблица 1 – Значения полноты, точности и F1-меры для алгоритмов, при максимально большой обучающей выборке
Table 1 – Recall, precision, F1-score for both algorithms, with maximum training collection size

	Полнота	Точность	F1-мера
НБК	0.6	0.6664	0.6348
МНБК	0.675	0.7385	0.7053

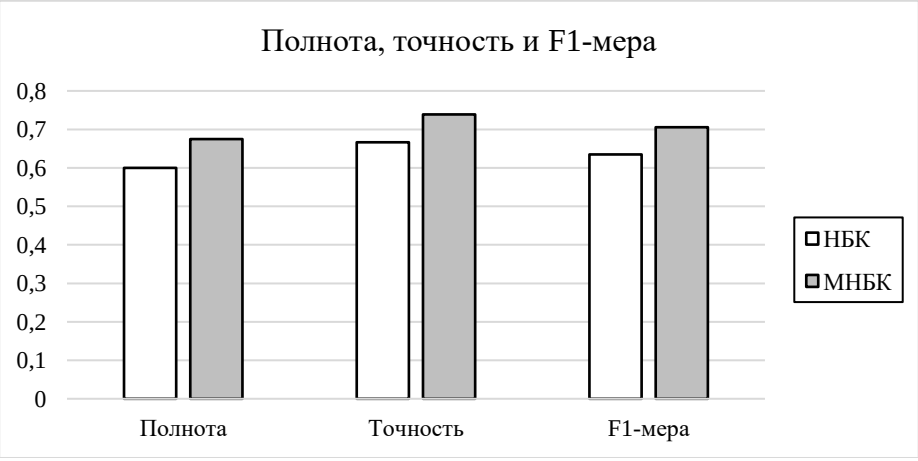

Рисунок 2 – Полнота, точность и F1-мера традиционного и модифицированного байесовских классификаторов (при максимальном объеме обучающей коллекции)
Figure 2 – Recall, precision, F1-score for traditional and modified Bayesian classifiers for maximum training collection size

Таблица 2 – Значения F1-меры для алгоритмов для выборок разных объемов
Table 2 – F1-score for different training collection sizes for traditional and modified Bayesian classifier

	НБК	МНБК
500	0.16	0.21
1000	0.24	0.286
2000	0.376	0.44
3000	0.506	0.573
4000	0.59	0.68
5000	0.6348	0.7053

Рисунок 3 – F1-мера при разных объёмах обучающей коллекции
Figure 3 – F1-score for different training collection sizes

Заключение

В работе описан алгоритм классификации потоковых данных с помощью байесовского критерия. Построена математическая модель потоковых текстовых данных, позволяющая применять алгоритмы классификации текста на естественном языке на потоковых данных. Предложена модификация наивного байесовского классификатора, использующая характеристику tf-idf как меру принадлежности терминов классам, позволяющая улучшить показатели классификации. Разработано программное обеспечение, позволяющее извлекать потоковые текстовые данные из сети Интернет, производить классификацию предложенным модифицированным алгоритмом в реальном времени.

ЛИТЕРАТУРА

1. Lomakina L.S., Subbotin A.N., Surkova A.S. Naïve Bayes Modification for Data Streams Classification. *Proceedings of the Thirteenth International MEDCOAST Congress on Coastal and Marine Sciences, Engineering, Management and Conservation (MEDCOAST 2017).* 2017;2:805-814.
2. Большакова Е.И., Клышинский Э.С., Ландэ Д.В., Носков А.А., Пескова О.В., Ягунова Е.В. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учеб. Пособие. М.: МИЭМ. 2011.
3. Gaber M.M., Zaslavsky A., Krishnaswamy S. A Survey of Classification Methods in Data Streams. Data Streams; Ed. by Aggarwal C.C. Springer US. 2007.

4. Berry M.W., Kogan J. Text Mining. Applications and Theory. Wiley. 2010.
5. Ломакина Л.С., Ломакин Д.В., Субботин А.Н. Байесовская классификация текстовых потоков. *Системы управления и информационные технологии.* 2016;4(66):60-64.
6. Субботин А.Н. Алгоритм классификации потоков текстовой информации на естественном языке. *Научно-технический вестник Поволжья*. 2020;1:18-21.
7. Ломакина Л.С., Ломакин Д.В., Субботин А.Н. Программа классификации потоков текстовых данных на основе байесовского подхода. Свидетельство государственной регистрации программы для ЭВМ № 2017611236, 31 октября 2016 г.

REFERENCES

1. Lomakina L.S., Subbotin A.N., Surkova A.S. Naïve Bayes Modification for Data Streams Classification. *Proceedings of the Thirteenth International MEDCOAST Congress on Coastal and Marine Sciences, Engineering, Management and Conservation (MEDCOAST 2017).* 2017;l.2:805-814.
2. Bolshakova E.I, Klishinskii E.S., Lande D.V., Noskov A.A., Peskova O.V., Yagunova E.V. Automatic processing of natural language texts and computer linguistics: educational material. M.: MIEM. 2011 (In Russ).
3. Gaber M.M., Zaslavsky A., Krishnaswamy S. A Survey of Classification Methods in Data Streams. Data Streams. Ed. by Aggarwal C.C. Springer US. 2007.
4. Berry M.W., Kogan J. Text Mining. Applications and Theory. Wiley. 2010.
5. Lomakina L.S. Lomakin D.V., Subbotin A.N. Text streams Bayesian classification. *Control systems and information technologies.* 2016;4(66):60-64 (In Russ).
6. Subbotin A.N. Algorithm for natural language text information classification. *Scientific and Technical Bulletin of the Volga Region*. 2020;1:18-21(In Russ).
7. Lomakina L.S., Lomakin D.V., Subbotin A.N. Program for classifying text data streams based on the Bayesian approach. *Certificate of state registration of a computer program № 2017611236,* October 31th, 2016.

ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATIONS ABOUT AUTHORS

Ломакина Любовь Сергеевна, доктор технических наук, профессор, профессор кафедры «Вычислительные системы и технологии», Нижегородский Государственный Технический Университет им. Р.Е. Алексеева, Нижний Новгород, Российская Федерация.
email: llomakina@list.ru

Lyubov S. Lomakina, Doctor Of Technical Science, Professor, Professor At «Computer Systems And Technologies» Department, Nizhny Novgorod State University N. A. R.E. Alekseev.

Субботин Артем Николаевич, аспирант, Нижегородский Государственный Технический Университет им. Р.Е. Алексеева, ООО «СВТЕКНН», инженер по разработке программного обеспечения, Нижний Новгород, Российская Федерация.
email: turnonmore@yandex.ru

Artem N. Subbotin, Graduate Student, Nizhny Novgorod State University N. A. R.E. Alekseev, «СВТЕКНН», LLC, SW Developer, Nizhny Novgorod, Russian Federation.