УДК 338.27
DOI 10.25688/2312-6647.2018.18.4.1

С.С. Голубев,
Е.П. Дюндик,
Е.В. Скубрий

Методы и модели решения задач классификации при инновационном прогнозировании научно-технологического развития с использованием интеллектуального анализа «больших» данных

В статье рассмотрены инструменты интеллектуального анализа «больших» данных для выявления ключевых трендов научно-технологического инновационного развития отраслей промышленности, выделяются алгоритмы, реализующие методы и модели решения задач классификации данных в процессе разработки прогнозов научно-технологического развития, на основе которых необходимо делать выводы и принимать обоснованные решения.

Ключевые слова: отрасли промышленности; интеллектуальный анализ; классификация данных; методы; алгоритм.

Анализ зарубежного опыта формирования, сбора, аналитической обработки, представления, актуализации и мониторинга технологических достижений в части анализа «больших» данных из открытых источников информации показал, что на сегодняшний день системы интеллектуального анализа больших массивов данных, позволяющие интегрировать разнородные неструктурированные данные и извлекать из них значимые выводы, носят узкоспециализированный характер, а разработка таких систем остается капитало-трудоемкой задачей.

Развитие современных методов и инструментов интеллектуального анализа «больших» данных для выявления ключевых трендов научно-технологического инновационного развития отраслей промышленности сосредоточено

на пересечении областей искусственного интеллекта, машинного обучения, статистики, компьютерной лингвистики и онтологического инжиниринга. Особенно выделяются алгоритмы, реализующие методы и модели решения задач классификации данных.

Классификация — структурирующее рассматриваемое множество явлений в совокупность отдельных классов, отражающих важные свойства этих явлений. Также этот термин применяется к задачам отнесения отдельных объектов к заранее заданным классам [1].

Пусть имеется множество объектов, описанных с помощью множества признаков и разделенных на классы. *Обучающей выборкой* является конечное множество объектов, для которых известно, к каким классам они относятся. Для остальных объектов принадлежность к классам неизвестна. Говорят, что алгоритм *классифицирует* произвольный объект из исходного множества, если позволяет определить номер и\или наименование (*label)* класса, к которому относится этот объект.

Формализованная постановка задачи классификации:

X — множество описаний объектов, или признаковое пространство такое, что $X = D_{f_1} \times ... \times D_{f_n}$.

Y — конечное множество идентификаторов классов.

$x = \left(f_1(x), ..., f_n(x)\right)$ — *признаковое описание* объекта $x \in X$, *признаком* называется отображение f: $X \to D_f$, D_f — множество допустимых значений признака. Для объектов обучающей выборки $X^m = \{(x_1, y_1), ..., (x_m, y_m)\}$ известны значения зависимости y^* : $X \to Y$.

Требуется построить алгоритм $a : X \to Y$, способный классифицировать $\forall x \in X$.

Основные типы признаков:

– *бинарный*, $D_f = \{0, 1\}$;

– *номинальный*, D_f — конечно;

– *порядковый*, D_f — конечно и упорядочено;

– *количественный* — совпадает с X.

Вероятностная постановка задачи классификации описана в [2]. Пусть множество всех пар *объект-признак* есть вероятностное пространство $X \times Y$ с неизвестной вероятностной мерой P. В соответствии с мерой P сгенерирована обучающая выборка $X^m \{(x_1, y_1), ..., (x_m, y_m)\}$. Требуется построить алгоритм $a : X \to Y$, способный классифицировать произвольный объект $x \in X$.

Рассмотрим методы классификации, основанные на правилах.

Деревья решений (*decision trees*) — это класс непараметрических методов машинного обучения с учителем, ориентированных на решение задач классификации

и регрессии. В основе метода лежит модель иерархически выстроенных решающих правил, то есть дерева, которое классифицирует объекты, последовательно применяя решающие правила в зависимости от результата, выданного предыдущим правилом. Первое правило, применяемое к объекту, является корнем дерева, а правила в листьях определяют принадлежность объекта к классу. С теоретико-графовой точки зрения каждый узел дерева решений, не являющийся листом, взвешивается признаком классифицируемого объекта, листья — идентификаторами классов, а каждая ветвь — множеством значений, которое может принимать признак.

Одно из главных преимуществ деревьев решений — это простота интерпретации. По построению, всякому пользователю понятно, на каком основании дерево решений классифицировало тот или иной пример. В некоторых случаях дополнительным плюсом является отсутствие параметров модели. Но деревья решений неустойчивы к переобучению и обеспечивают невысокую точность классификации по сравнению с другими, более продвинутыми алгоритмами классификации, например нейронными сетями или ансамблевыми методами. Случайный лес (*random forest*), по существу, является ансамблем деревьев и позволяет существенно повысить точность классификации по сравнению с одним деревом. Заметим, что при этом теряется простота интерпретируемости результатов работы алгоритма.

Каждое дерево решений можно представить набором правил (решающие правила), если следовать от корня дерева к листьям. Заметим, что решающие правила можно порождать и на основе данных без построения дерева. В статье [3] проведен обзор алгоритмов построения наименьшего возможного набора решающих правил, согласующихся с обучающей выборкой. Слишком большое количество порожденных решающих правил обычно свидетельствует о том, что алгоритм пытается запомнить данные, а не обнаружить закономерности в них, и часто ведет к проблеме переобучения.

Преимущества и недостатки решающих правил в целом те же, что и у деревьев решений — отличная интерпретируемость, возможность учета экспертных знаний данной предметной области, представленных также в виде правил, но невысокая точность классификации.

Методы интеллектуального анализа данных, составляющие основу программного обеспечения системы, связывают нечеткие деревья принятия решений, генетический алгоритм, использованный для усовершенствования алгоритма оптимизации, с данными, полученными об агентах в многоагентной системе.

Классификация по запросу, или «ленивая» классификация (*lazy classification*), — это подход к решению задачи классификации, при котором обучающая выборка обрабатывается с учетом текущего запроса, то есть с учетом свойств того объекта, который нужно классифицировать. Современные реализации этого подхода могут быть достаточно универсальными и применимыми к различным алгоритмам классификации.

Алгоритм построения «ленивых» деревьев решений *LazyDT*, по сравнению с обычными деревьями решений, имеет преимущества:

– решающие правила получаются намного короче и, как следствие, лучше трактуются;

– при ограниченной обучающей выборке многие алгоритмы построения деревьев решений сталкиваются с проблемой сильной фрагментации.

В алгоритмах типа *C*4.5 и *ID*3 [4] на каждом шаге построения дерева выбирается лучшее разбиение на основе среднего улучшения какого-либо критерия.

Поскольку выбор делается на основе усредненного значения критерия, для некоторых дочерних ветвей он может быть отрицательным. Для объектов, которые попадают в такой путь в дереве, дальше разбиение приводит только к избыточной фрагментации данных. В алгоритме построения «ленивых» деревьев решений для каждого тестового объекта строится свой путь в дереве решений, что позволяет избежать лишней фрагментации данных.

На каждом шаге алгоритма выбирается разбиение, приводящее к максимальному уменьшению энтропии целевого класса.

Алгоритм LazyDT:

Вход: X — обучающая выборка, t — объект из тестовой выборки.

Выход: y_t — предсказанная метка целевого класса для объекта t.

1. Если все объекты в X имеют одну и ту же метку l, вернуть.

2. В противном случае выбрать признак A, пусть a — значение признака A у объекта t. Пусть X' — подмножество обучающих объектов со значением признака A, равным a. Применить алгоритм для X'.

Альтернативой деревьям решений являются классификаторы, построенные по ассоциативным правилам (ассоциативные классификаторы, *Eager Associative Classifier — EAC*). По качеству классификации они часто превосходят деревья решений. Классификаторы находят правила с глобальным максимумом критерия, например критерием прироста информации. Стоит отметить, что количество порождаемых ассоциативных правил может быть очень большим.

Модификацией ассоциативных классификаторов являются классификаторы, работающие по запросу на основе правил (*Lazy Associative Classification — LAC*). В этом случае правила порождаются не по всей обучающей выборке, а только по тем признакам, которые наблюдаются в тестовом объекте, то есть число правил существенно уменьшается. Порождаются только правила, посылка которых содержится в признаках классифицируемого тестового объекта.

Алгоритм «ленивой» классификации данных со сложной структурой (то есть комплексными признаками) на основе аппарата узорных структур (*pattern structures*) был предложен в [5].

Основное отличие заключается в том, что алгоритм может работать на любых данных, для которых задано:

– описание объекта (это может быть как множество признаков, так и последовательности, интервалы или графы);

– операция пересечения описаний (этот метод применим для интервальных данных (например, в задаче кредитного скоринга), последовательностей, и данных, представленных графами (например, в задаче прогнозирования токсичности химических веществ)).

Для этого метода решена задача порождения хорошо интерпретируемых наборов классифицирующих правил для данных со сложной структурой.

Мощный и развитый метод интеллектуального анализа данных — нейронные сети (*neural networks*) — исторически является особенно распространенным и проработанным для решения задач.

Приведем краткий обзор методов и базовых понятий, не вдаваясь в фундаментальные подробности. Заметим, что прекрасное базовое введение в классическую теорию нейронных сетей приведено в работе [6].

Единица обработки информации в нейронной сети называется нейроном. Согласно исторически сложившейся модели нейронов, выделяют три основных ее элемента (см. рис. 1):

– набор синапсов (*synapse*), или связей (*connecting link*), каждый из которых характеризуется весом w_{k_j} (*weight*) и силой (*strength*);

– сумматор (*adder*) для вычисления линейной комбинации входных сигналов (*linear combiner output*) x_j и синаптических весов нейрона:

$$u_k = \sum_{j=1}^{m} w_{k_j} x_j;$$

– функция активации (*activation function*), или функция сжатия (*squashing function*), для ограничения амплитуды выходного сигнала нейрона.

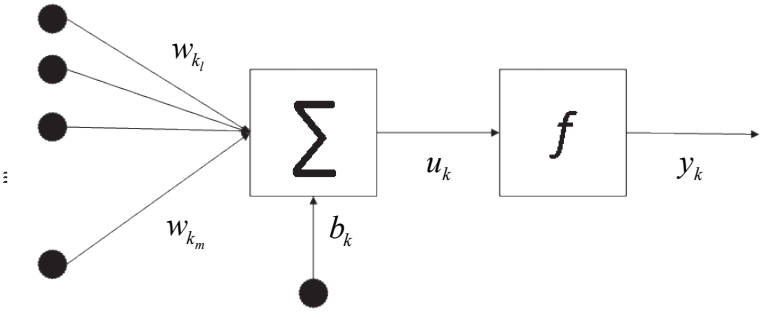

Рис. 1. Простейшая модель нейрона

Пороговый элемент b_k (*bias*) — величина, отражающая увеличение или уменьшение входного сигнала, подаваемого на функцию активации.

Выходной сигнал нейрона y_k:

$$y_k = f(u_k + b_k).$$

Достаточно полную подборку видов функций активации можно найти в [6]. Наиболее распространенными являются три типа:

– пороговая функция (функция единичного скачка) (*hard limit transfer function*) чаще применяется в нейронах, классифицирующих входы по двум категориям;

– линейная функция (*linear transfer function*) популярна для нейронов типа «адалайн» (*ADALINE*);

– сигмоидальная функция (*log-sigmoid transfer function*) применяется для многослойных сетей, в которых обучение проводится при помощи алгоритма обратного распространения ошибки (*backpropagation algorithm*) и от функции активации требуется свойство дифференцируемости.

Нейронная сеть — это сеть с конечным числом слоев, состоящая из однотипных элементов — аналогов нейронов с различными типами связей между слоями. Понятие архитектуры нейронных сетей тесно связано с нейронными слоями (*layer*), потому что в многослойных сетях нейроны располагаются по слоям. Также на архитектуру сети влияют применяемые алгоритмы обучения.

В *однослойной сети* (*single-layer network*) сначала располагается входной слой (*input layer*) источников, воздействие в которых передается на один (единственный) выходной слой (*output layer*), состоящий из нейронов. Передача воздействия идет от входного слоя к выходному, поэтому сети называют *сетями прямого распространения* (*feed-forward*).

Существуют сети, в которых участвуют сети, состоящие из одного или нескольких скрытых (*hidden)* слоев. Нейроны в этих слоях называются скрытыми нейронами (*hidden neuron*). Выходным слоем *многослойной нейронной сети* называется слой, выход которого совпадает с выходом всей сети.

Сеть, в которой присутствует хотя бы одна обратная связь (*feedback loop*), называется *рекуррентной*. Обратная связь подразумевает наличие передачи воздействия из части нейронов выхода на часть нейронов входов.

Обучение нейронной сети — это процесс, в котором свободные параметры нейронной сети настраиваются посредством моделирования среды, в которую эта сеть встроена. Тип обучения определяется способом подстройки этих параметров.

Последовательность действий в алгоритмах обучения нейронных сетей:

– в нейронную сеть поступают стимулы из внешней среды;

– изменяются свободные параметры нейронной сети;

– после изменения внутренней структуры нейронная сеть отвечает на последующие возбуждения иначе.

При обучении нейронных сетей применяют подход обучения как с учителем, так и без учителя. Обучение с учителем предполагает, что, кроме входных сигналов, известны также и ожидаемые выходные сигналы нейрона. Подбор весовых коэффициентов организуется так, чтобы они принимали значения максимально близкие к ожидаемым.

В обучении без учителя применяются конкурентные подходы, например *WTA* (победитель получает все), или *WTM* (победитель получает больше), а также методы, учитывающие корреляции между обучающими и выходными сигналами, например обучение по Хеббу.

Современные нейронные сети расширились методами машинного обучения, например глубоким обучением (*deep learning*), и сверточные (*convolution*) — нейронными сетями.

Глубокие нейронные сети (*deep neural networks*, *DNN*) — это нейронные сети прямого распространения, или многослойные персептроны с большим количеством скрытых слоев, веса которых полностью взаимосвязаны. Класс глубоких нейронных сетей также включает глубокие генеративные модели, или *глубокие сети доверия* (*deep belief network*, *DBN*). Глубокие сети доверия — вероятностные генеративные модели, состоящие из нескольких слоев скрытых вероятностных переменных.

По архитектуре и способам применения выделяют три класса глубоких нейронных сетей:

– генеративные и глубокие нейронные сети, обучающиеся без учителя;
– глубокие нейронные сети, обучающиеся с учителем;
– гибридные нейронные сети.

Глубокое обучение — подход к построению техник машинного обучения, при котором многоуровневая обработка информации в иерархических архитектурах обучения с учителем переносится на выявление признаков в обучении без учителя, сличение с образцом и классификацией.

К гибридным (*hybrid*) сетям относятся сети глубокого обучения, архитектура которых состоит одновременно из генеративных и дискриминативных моделей.

Рассмотрим конкретные примеры использования нейроновых сетей. В работе [7] предложен *ATR*-алгоритм (*automatic target recognition*), представляющий собой композицию нескольких методов машинного обучения, первым шагом которого является применение импульсной нейронной сети в модуле сегментации при обнаружении целей и дальнейшее распознавание, и классификация при помощи метода опорных векторов.

При решении задачи классификации и регрессии с помощью метода опорных векторов — *Support Vector Machines* (*SVM*) ставится цель разработать алгоритмически эффективные методов построения оптимальной разделяющей гиперплоскости в пространстве признаков большой размерности. Оптимальность понимается в смысле минимизации верхних оценок вероятности ошибки обобщения. Метод относится к бинарным классификаторам, но известны его расширения для задач классификации по нескольким классам.

Методом опорных векторов строится классифицирующая функция $F(x) = sign(\langle w, x\rangle + b)$, где $\langle sign(\langle w, x\rangle\rangle$ — скалярное произведение, w — нормаль к разделяющей гиперплоскости, b — вспомогательный параметр.

В один класс относят объекты, для которых $F(x) = 1$, а остальные с $F(x) = -1$ — в другой.

При решении задач классификации классификатор (алгоритм классификации) называется слабым, если его ошибка на обучающей выборке больше 0 % и меньше 50 %. В случае бинарной классификации говорят, что классификатор слабый, если он не намного лучше, чем простое случайное угадывание. Если ошибка классификатора на обучающей выборке может быть уменьшена до значения, сколь угодно близкого к 0 %, за полиномиальное время, то тогда классификатор называется *сильным*.

Использование методов и моделей решения задач классификации данных при интеллектуальном анализе «больших» данных в процессе разработки прогнозов научно-технологического развития отраслей промышленности необходимы для разработки и описания математических моделей и алгоритмов, на основе которых делаются выводы при принятии обоснованных решений.

Литература

1. *Mirkin B.* Core Concepts in Data Analysis: Summarization, Correlation and Visualization. Springer, 2011. 390 p.

2. *Вьюгин В.В.* Математические основы машинного обучения и прогнозирования. М.: МЦНМО, 2013. 304 с.

3. *Furnkranz J.* Separate-and-conquer rule learning // Artificial Intelligence Review. Vol. 13. 1999. Pp. 3–54.

4. *Quinlan J.R.* C4.5: Programs for Machine Learning. San Francisco: Morgan Kaufmann Publishers Inc., 1993.

5. *Kuznetsov S.O.* Scalable Knowledge Discovery in Complex Data with Pattern Structures // 5th International Conference Pattern Recognition and Machine Intelligence. 2013. Vol. 8251. Pp. 30–41.

6. *Hagan M.T., H.B.D., M.H.B., De Jesús O.* Neural Network Design. 2nd ed. Martin Hagan, 2014. 800 p.

7. *Neagoe V.E., Carata S.V., Ciotec A.D.* An advanced neural network-based approach for military ground vehicle recognition in SAR aerial imagery // Scientific Research and Education in the Air Force. Vol. 18. 2016. Pp. 41–48.

Literatura

1. *Mirkin B.* Core Concepts in Data Analysis: Summarization, Correlation and Visualization. Springer, 2011. 390 p.

2. *V'yugin V.V.* Matematicheskie osnovy' mashinnogo obucheniya i prognozirovaniya. M.: MCzNMO, 2013. 304 s.

3. *Furnkranz J.* Separate-and-conquer rule learning // Artificial Intelligence Review. Vol. 13. 1999. Pp. 3–54.

4. *Quinlan J.R.* C4.5: Programs for Machine Learning. San Francisco: Morgan Kaufmann Publishers Inc., 1993.

5. *Kuznetsov S.O.* Scalable Knowledge Discovery in Complex Data with Pattern Structures // 5th International Conference Pattern Recognition and Machine Intelligence. 2013. Vol. 8251. Pp. 30–41.

6. *Hagan M.T., H.B.D., M.H.B., De Jesús O.* Neural Network Design. 2nd ed. Martin Hagan, 2014. 800 p.

7. *Neagoe V.E., Carata S.V., Ciotec A.D.* An advanced neural network-based approach for military ground vehicle recognition in SAR aerial imagery // Scientific Research and Education in the Air Force. Vol. 18. 2016. Pp. 41–48.

S. S. Golubev,
E. P. Dundik,
E. V. Skubriy

Methods and Models for Solving Classification Problems in Innovative Forecasting of Scientific and Technological Development using the Intellectual Analysis of «Large» Data

In the article the tools of intellectual analysis of «large» data for revealing key trends of scientific and technological innovation development of industries are considered. The algorithms that implement methods and models for solving data classification problems in the process of developing forecasts of scientific and technological development on the basis of which it is necessary to draw conclusions and make informed solutions are shown in the article.

Keywords: industries; intellectual analysis; data classification; methods; algorithm.