



имеют наиболее уникальные границы. Преобразования Хафа могут быть применены для поиска прямоугольных или круглых областей, с последующим поиском среди них памятных табличек, так как именно они, в основном, имеют заранее чётко известную и простую форму. Методы Виолы-Джонса и относительной яркости подходят для решения задачи в целом, то есть для поиска наиболее распространённых туристических объектов (памятников, монументов, зданий), но для каждого из методов необходимо составить обучающие выборки и обучать систему, кроме того каждый из методов имеет уникальный формат обучающих данных (шаблонов, матриц, коэффициентов обучения). Мы можем сделать вывод о том, что наибольшее количество преимуществ имеет метод Виолы-Джонса, дополнительным его преимуществом является наличие стандартной реализации в библиотеке OpenCV [6], что значительно облегчает его апробацию.

#### Литература

1. Азаренко, Д.С. Детектирование объекта на изображении и определение его смещения на двух различных изображениях [Текст] /Д.С. Петров //Искусственный интеллект. – 2013. – № 3. – С. 90–97.
2. Корнилов, Ф.А. Поиск прямоугольных контуров на изображениях с помощью преобразования Хафа [Текст] /Ф.А. Корнилов //Современные проблемы математики и её приложений. – 2014. – С. 192.
3. Копылов, И.Е. Определение марки автомобиля по видеок cadру с использованием модифицированного алгоритма Виолы-Джонса [Текст] / И.Е. Копылов, К.Л. Тассов //Наука и образование: электронное научно-техническое издание. – 2012. – №. 06. – С. 205–210.
4. Viola, P. Rapid object detection using a boosted cascade of simple features / P. Viola, M. Jones // Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. December 2001. Vol. 1. PP. 1063–6919.
5. Агарков, А.В. Поиск объектов на изображениях с помощью относительной яркости отдельных областей [Текст] / А.В. Агарков, Р.О. Кашин, К.С. Ивашко //Искусственный интеллект. – 2014. – № 4. – С. 180–185.
6. Bradski, G. Learning OpenCV: Computer vision with the OpenCV library / G. Bradski, A. Kaehler – O'Reilly. – 2008. – 580 с.

В.П. Клюев

#### АВТОМАТИЧЕСКАЯ ОБРАБОТКА БОЛЬШИХ ОБЪЕМОВ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ ДЕРЕВЬЕВ РЕШЕНИЙ

(Самарский национальный исследовательский университет  
имени академика С.П. Королёва)

В современном мире быстро развиваются технологии автоматического анализа больших объёмов данных. Это вызвано тем, что накопленную инфор-



мацию уже невозможно обработать вручную. Один из методов автоматического анализа данных – это деревья решений.

Построение дерева решений начинается с определения обучающей выборки. Пусть задано некоторое множество  $X$ , содержащее примеры, каждый из которых характеризуется  $n$  атрибутами. Каждый объект принадлежит некоторому классу  $C_k$ .

Возможны 3 ситуации:

1. Множество  $X$  состоит из объектов принадлежащих одному классу  $C_k$ . Тогда  $X$  – это лист, определяющий класс  $C_k$ ;
2. Множество  $X$  пустое. Тогда  $X$  – это снова лист, который принадлежит такому же классу, что и родитель.
3. Множество  $X$  состоит из объектов, относящихся к разным классам. Множество  $X$  следует разбить на подмножества. Для этого выбирается один из атрибутов, имеющий различные значения  $O_1, O_2 \dots O_m$ . Множество  $X$  разбивается на подмножества  $X_1, X_2 \dots X_m$ , где каждое подмножество  $X_i$  содержит все объекты, имеющие значение  $O_i$  для выбранного атрибута. Это процедура повторяется рекурсивно.

После этого возникает несколько вопросов.

Первый вопрос - это выбор атрибута, по которому нужно будет проводить разбиение множества. Это особенно актуально, потому что большинство из известных алгоритмов являются "жадными алгоритмами". Если один раз был выбран атрибут, и по нему было произведено разбиение на подмножества, то алгоритм не может вернуться назад и выбрать другой атрибут, который дал бы лучшее разбиение. Из этого видно, что на этапе построения нельзя сказать даст ли выбранный атрибут оптимальное разбиение или нет.

Второй вопрос - это проблема отсечения избыточных ветвей. Сам по себе вопрос тривиален. Если разбиение множества в узле ведет к листам одного класса, то этот узел можно считать листом этого класса. Но отсюда вытекает другой вопрос.

Третий вопрос – критерий прекращения. Если во множестве на 99 объектов класса  $C_1$  приходится 1 объект класса  $C_2$ , то можно считать этот узел листом класса  $C_1$ ? А если на 95 объектов класса  $C_1$  – 5 объектов класса  $C_2$ ? Другими словами, как определить является узел листом дерева или его нужно еще раз разбить на подмножества.

В настоящее время существует большое число алгоритмов реализующих деревья решений: CART, C4.5, NewId, IRule, CHAID, CN2 и другие. Самыми известными и популярными на сегодняшний день являются алгоритм CART и C4.5.

CART (Classification and Regression Tree) – это алгоритм построения бинарного дерева решений (каждый узел дерева имеет двух потомков). Алгоритм CART использует, так называемый, индекс Gini, который оценивает "расстояние" между распределениями классов.



$$Gini(u) = 1 - \sum_j p_j^2,$$

где  $u$  - текущий узел;  $p_j$  - вероятность класса  $j$  в узле  $u$ .

Сейчас существуют модификации алгоритма CART - алгоритмы IndCART и DB-CART.

C4.5 – алгоритм, который умеет работать с непрерывным целевым полем, поэтому он решает только задачи классификации. У этого алгоритма количество потомков у узла не ограничено. Алгоритм C4.5 – развитие алгоритма ID3. Он использует теоретико-информационный подход. Для выбора наиболее подходящего атрибута, предлагается следующий критерий:

$$Gain(P) = Info(X) - Info_g(X),$$

где  $Info(X)$  - энтропия множества  $X$ .

$Info_g(X)$  находится по формуле:

$$Info_g(X) = \sum_{i=1}^n Info(X) \frac{|X_i|}{|X|},$$

где  $|X|$  - мощность множества  $X$ .

Множества  $X_1, X_2, \dots, X_m$  получены при разбиении исходного множества  $X$  по проверке  $P$ . Выбирается атрибут, дающий максимальное значение по критерию.

#### Библиографический список

1. Коршунов, Ю. М. Математические основы кибернетики. М. Энергоатомиздат, 1987
2. Breiman, L. Classification and Regression Trees. Wadsworth, Belmont, California, 1984
3. Ross Quinlan, J. C4.5: Programs for Machine learning. Morgan Kaufmann Publishers 1993.

П.А. Корнев, А.В. Шолохов

#### НЕЙРОСТРУКТУРНОЕ МОДЕЛИРОВАНИЕ СИСТЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ РЕШЕНИЯ СЕТЕВЫХ ПРОБЛЕМ

(Рязанский государственный радиотехнический университет)

С целью оперативного устранения большого числа быстро возникающих проблем в локальных вычислительных сетях успешно применяются не только возможности человеческого инженерного мышления, но и методы искусственного интеллекта [1,2] Для более эффективного использования современных возможностей машинного интеллекта необходимо использовать программный комплекс, автоматизирующий процесс генерации и реализации управляющих



решений. Рассмотрим алгоритмическое обеспечение такой программной системы, основанной на нейроструктурном моделировании.

Чтобы данный программный инструментарий корректно выполнял возложенные на него задачи по обнаружению сетевых конфликтов, его нейроструктурная модель (НСМ) должна быть синтезирована с учетом топологии локальной вычислительной сети (ЛВС), в которой он будет функционировать.

Для синтеза нейроструктурной модели системы принятия решений при ликвидации сетевых проблем (НСМ СПРЛСП) с учетом топологии ЛВС предлагается использовать следующий алгоритм (рис. 1).

Под параметрами сенсоров понимаются множества ARP-пакетов ЛВС, захваченные сенсорами в течение установочного времени  $T$ , на основе которых определяются топологии ЛВС и её подсетей, а также количество подсетей.

Нейроны скрытого слоя (слоев) с наилучшей функцией активации синтезируются по алгоритму, приведенному на рисунке 2, где:

- $C$  – общее количество сенсоров;
- $A$  – массив возможных функций активации (ФА);
- $B$  – общее количество возможных ФА в массиве  $A$ ;
- $f$  – функция активации нейронов текущего скрытого слоя.

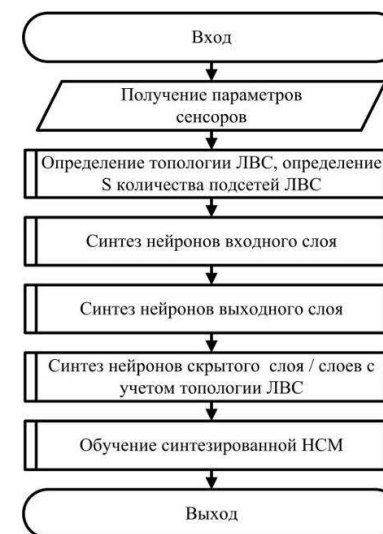


Рис. 1. Алгоритм синтеза нейроструктурной модели системы принятия решений при ликвидации сетевых проблем

Топология ЛВС и её подсетей учитываются при формировании обучающих выборок. По наименьшей среднеквадратической ошибке обучения  $E$  нейроструктурной модели с текущим скрытым слоем определяется наилучшая функция активации для его нейронов.