Встраивание средств Data Mining в инфраструктуру виртуальной обсерватории♣

© Мамардашвили Н.А., Вовченко А.Е., Калиниченко Л.А.

ИПИ РАН
shviller@gmail.com, alexey_vov@ipi.ac.ru, leonidk@synth.ipi.ac.ru

Малков О.Ю.

ИНАСАН
malkov@inasan.ru

Патракова М.Е.

ВМиК МГУ
pm_box@inbox.ru

Аннотация

Методы извлечения знаний из данных (Data Mining) применяются в различных областях науки, в том числе в астрономии, как средства, помогающие получать новые знания, делать научные открытия. В данной работе обсуждается важность встраивания в состав виртуальных обсерваторий средств для решения астрономических задач методами Data Mining, рассматриваются существующие подходы, отдается предпочтение использованию ансамблей алгоритмов и предлагается соответствующая архитектура (Ensembled Weka) встраивания системы Weka в инфраструктуру виртуальной обсерватории.

1 Введение

Для решения проблемы интегрированного использования астрономических данных астрономическое сообщество разрабатывает новый подход к работе с результатами наблюдений, в основе которого находится концепция Виртуальной Обсерватории (ВО). Согласно этому подходу, разрабатываются специальные ИТ-методы и инструменты для интегрированного доступа к неоднородным распределенным архивам и каталогам астрономических данных, равно как и к вычислительным ресурсам и сервисам с целью решения задач над таким множеством неоднородных информационных ресурсов. При этом особенно важным является .извлечение знаний из данных, содержащихся в огромных, интегрированных распределенных репозиториях информации. Виртуальные обсерватории интенсивно разрабатываются в мире астрономическими институтами различных стран мира, усилия которых координируются Альянсом Международной Виртуальной Обсерватории (IVOA) и Комиссиями Международного Астрономического Союза (IAU). В России также ведутся работы по созданию Российской Виртуальной Обсерватории (РВО) [8] в рамках IVOA.

Весьма продвинутой инфраструктурной частью РВО является система AstroGrid [1], разработанная в Великобритании и установленная в Межведомственном Суперкомпьютерном Центре РАН и в Институте проблем информатики РАН в виде двух установок коллективного пользования для решения задач специалистами в области астрономии [10]. Пример использования системы AstroGrid РВО при решении астрономической задачи дан в [5].

Настоящий период развития науки характеризуется взрывоподобным процессом накопления информационных источников и сервисов обработки информации, число которых экспоненциально растет. Наиболее явно эта тенденция наблюдается именно в астрономии, где каждые полгода объем накопленных данных удваивается. При этом решение сложных задач, для которых требуется доступ ко многим информационным ресурсам, а также сервисам обработки данных, для простого пользователя становится непосильной задачей. Для решения такого рода проблем, наряду с другими, применяются методы и средства Data Mining, позволяющие реализовать автоматический анализ больших массивов информации и выявление в них ранее неизвестных закономерностей, предсказание неизвестных значений и определение важности известных значений.

Инфраструктура РВО постепенно расширяется новыми инструментами, которые должны обеспечивать решение разнообразных задач анализа данных. Важным является вопрос введения в инфраструктуру средств решения задач методами Data Mining. Поэтому одной из основных целей

Труды 9ой Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» - RCDL'2007, Переславль-Залесский, Россия, 2007.

данной работы является анализ подходов к встраиванию средств Data Mining в инфраструктуру РВО и разработка перспективных решений для предоставления астрономам готового инструмента Data Mining в среде РВО. Структура работы выглядит следующим образом. В начале дано краткое описание компонентов AstroGrid как среды, в которую осуществляется встраивание средств Data Mining. Далее дана общая характеристика методов и средств Data Mining, рассмотривается предлагаемое решение по встраиванию средств Data Mining в AstroGrid, приведен пример решения задачи классификации затменных двойных звезд в предложенной архитектуре.

2 Компоненты AstroGrid

Технически ВО - это открытая система стандартов и интероперабельных модулей, которые можно комбинировать для совместной работы разными способами. Разработчики инструментальных средств следуют стандартным интерфейсам, так чтобы их можно было включить (plug-in) в состав приложения, а центры данных предоставляют возможность доступа к своим данным стандартным образом, равно как и сервисы данных, уподобляемые запросам к базам данных. Основная функция AstroGrid заключается в предоставлении инфраструктурных программных средств поддержки ВО. Все программные средства, зарегистрированные в AstroGrid, выполняются либо в Центре коллективного пользования, либо как Java Web Start приложения, которые автоматически можно стартовать на машине пользователя.

Существуют четыре основных компонента AstroGrid РВО для пользователей, которые не являются разработчиками новых наборов данных или программных сервисов.

Клиент (AstroGrid Workbench [2]) обеспечивает средства взаимодействия с системой. Это инструмент, написанный на Java, который позволяет войти в систему, получить доступ к остальным компонентам, осуществлять мониторинг задач, и пр. С помощью клиента можно также реализовать переход к произвольным приложениям, разработанным третьей стороной (например, использование средств манипулирования изображениями Aladin [4]).

Центральную часть системы AstroGrid составляет Workflow – компонент спецификации потоков работ и управления ими. Он позволяет образовать композицию сложной последовательности задач (tasks). Задачи могут выполняться последовательно, совмещено во времени, циклически. Также ход выполнения задач может настраиваться посредством скриптов, написанных на языке Groovy [7]. Для спецификации потоков работ предоставляется графическое средство. Система разыскивает необходимые сервисы в реестре и передает им заданные параметры. Потоки работ представляются в XML на основе языка AstroGrid WF, производного от BPEL.

Следующим важным компонентом является реестр (Registry) астрономических ресурсов – не только каталогов и архивов изображений, сервисов, но и приложений, готовых к использованию, и др. К реестру можно задавать запросы, ресурсы можно просматривать для отыскания требуемых.

Наконец, четвертым компонентом AstroGrid является MySpace. Это некоторый вид пространства хранения промежуточных результатов, файлов, спецификаций потоков работ, запросов, позволяющий работать в группе и повторно использовать ранее полученные результаты и спецификации.

Для пользователей, которые намереваются разрабатывать новые ресурсы (наборы данных и программные сервисы), дополнительно предоставляются следующие компоненты:

- Средства конструирования приложений: Common Execution Architecture (CEA). В качестве приложения может выступать любой процесс обработки, который потребляет или продуцирует данные, что технически может включать:
  - приложение командной строки UNIX (UNIX command line application);
  - запрос к базе данных;
  - вызов Веб-сервиса;
  - Java-приложение.
- Доступ к наборам данных: Dataset Access (DSA)

DSA – это компонент, необходимый для включения наборов данных (баз данных) в систему AstroGrid. DSA предоставляет пользователям AstroGrid доступ к базе данных посредством запроса на ADQL [4] или посредством ConeSearch [4]. Также DSA предоставляет набор Веб-сервисов и, что наиболее важно, CEA-приложение, выполняющее запросы к базе данных.

3 Общая характеристика методов и средств data mining

Кратко Data Mining можно характеризовать как процесс автоматического выявления новых закономерностей и взаимосвязей в больших наборах данных для использования в науке и в других областях (например, в процессах принятия решений). Целью такого процесса является извлечение знаний - построение моделей. Грубо различают *прогнозирующие* модели, которые в явном виде содержат информацию для прогноза новых закономерностей, и *дескриптивные* модели, описывающие общие закономерности предметной области. К прогнозирующим обычно относят методы *классификации* (на основе характеристик объектов и набора заранее классифицированных случаев они позволяют создать модель определения классов, к которым такие объекты принадлежат), методы *регрессии* (эти методы позволяют прогнозировать новые значения непрерывных переменных на основе существующих; в

простейшем случае их основу составляют стандартные статистические методы (линейная регрессия)). К дескриптивным относятся методы *кластеризации* (создаваемые при этом модели нужны для разбиения данных на разные группы (кластеры) по критерию “похожести“ или “близости“, принимая во внимание “расстояние“ между различными объектами; в отличие от классификации, классы, по которым будет проводиться разбиение, заранее неизвестны), *ассоциативные методы* (соответствующие модели основаны на правилах типа “если…, то…” с использованием коэффициентов уверенности - если происходит событие *A*, то с вероятностью *p* происходит событие *B*). Для построения моделей разработаны различные алгоритмы, так что при создании модели одного типа может использоваться множество разнообразных алгоритмов (таких как, например, классификационные и регрессионные деревья решений, нейронные сети, генетические алгоритмы, методы поиска ассоциаций, методы дискриминантного анализа, байесовские сети, методы логистической регрессии и пр.).

Существует ряд инструментальных средств поддержки решения задач методами Data Mining. К ним относятся, например, Java Data Mining, Oracle Data Mining, MATLAB, Weka [9]. В настоящей работе выбрана система Weka – свободно распространяемая система, реализующая большое число разнообразных алгоритмов Data Mining. Этому выбору способствовало то, что авторы имели опыт применения Weka при решении задач в области астрономии [13]. В дальнейшем оказалось, что Weka планировалась в Великобритании для встраивания в систему AstroGrid как AstroWeka [3] - составная часть средств анализа данных ВО. Такое встраивание было реализовано в конце 2006 г. и доступно при помощи средств РВО, основанных на AstroGrid.

Weka позволяет решать такие типичные задачи Data Mining, как классификация, кластеризация, регрессия, построение ассоциативных правил и выделение ключевых атрибутов. Для этого имеется как API, дающий доступ к различным алгоритмам Data Mining, так и графический интерфейс, облегчающий подготовку данных, позволяющий манипулировать данными и визуализировать их. Также графический интерфейс позволяет применять один выбранный алгоритм (из доступных через API) к входным данным.

4 Анализ архитектурных решений для включения средств DM в AstroGrid и предлагаемое решение

В конце 2006 г. в инфраструктуре AstroGrid стал доступен инструмент Data Mining AstroWeka. AstroWeka представляет собой набор расширений графического интерфейса Weka, позволяющий работать с данными в MySpace, и обмениваться данными с другими приложениями в среде AstroGrid посредством интерфейса PLASTIC (PLatform for AStronomical Tool InterConnection). При этом AstroWeka является отдельным приложением, а не сервисом AstroGrid, поэтому возможности взаимодействия с системой AstroGrid ограничиваются способностью обмениваться данными с MySpace или другими приложениями, входящими в инфраструктуру AstroGrid. Как известно, основная цель инфраструктуры AstroGrid – предоставить астрономам возможности решения сложных научных задач, пользуясь спецификациями потоков работ. AstroWeka не предоставляет сервиса AstroGrid, который можно было бы использовать в потоках работ, что сильно ограничивает возможности применения средств Data Mining в инфраструктуре AstroGrid. Поэтому было принято решение создать сервис AstroGrid (CEA-приложение), предоставляющий средства Data Mining и основанный на Weka API. Реализация отдельных алгоритмов, входящих в Weka в виде CEA-приложений. была признана нецелесообразной в силу причин, приведённых в следующем разделе. Предпочтение было отдано ансамблям алгоритмов, что позволило ограничиться реализацией единственного CEA-приложения, с возможностью простой настройки этого приложения пользователем на решение новой задачи. Таким образом, предложенную реализацию можно использовать и как отдельное приложение, и как один из этапов потока работ, решающего сложную научную задачу.

5 Анализ использования ансамблей алгоритмов для решения задач Data Mining

5.1. Целесообразность применения ансамблей

В последние годы появились узкоспециализированные пакеты интеллектуального анализа данных. Для таких пакетов часто характерна ориентация на узкий круг практических задач, а их алгоритмической основой является какая-либо модель, использующая нейронную сеть, решающие деревья, ограниченный перебор, и т.п. Ясно, что подобные разработки существенно ограничены при практическом использовании. Во-первых, заложенные в них подходы не являются универсальными относительно размерностей задач, типа, сложности и структурированности данных, величины шума, противоречивости данных, и т.п. Во-вторых, созданные и «настроенные» на решение определенных задач, они могут оказаться совершенно бесполезными для других. Наконец, множество задач, представляющих интерес практическому пользователю, обычно шире возможностей отдельного подхода.

Таким образом, на настоящем уровне развития методов решения задач анализа данных, представляется предпочтительным путь создания программных средств, включающих разнообразные

существующие подходы. При этом повышаются шансы подбора из имеющихся алгоритмов такого алгоритма, который обеспечивал бы наиболее точное решение интересующих пользователя задач на новых данных. Разновидностью таких обобщенных средств автоматического решения задач распознавания и классификации являются ансамбли алгоритмов. Действительно, стандартной ситуацией является наличие нескольких альтернативных алгоритмов или решений, равнозначных для пользователя. Для выбора из них одного наиболее предпочтительного не хватает информации. Тогда естественной альтернативой выбору является создание на базе имеющихся алгоритмов или решений новых, более предпочтительных.

Существует несколько доводов в пользу использования ансамблей алгоритмов. Например, в [6] приводится три таких довода. Во-первых, доступные данные могут не содержать достаточно информации для выбора оптимального алгоритма из имеющихся. Даже после отбрасывания алгоритмов, которые дают большую, чем другие, ошибку на проверочной выборке, или же имеют большую сложность, может остаться достаточно много возможных алгоритмов, среди которых может и не содержаться лучший. Применение комбинации алгоритмов - один из способов решить такую проблему. Во-вторых, даже при наличии в данных достаточной информации задача нахождения оптимального алгоритма может быть слишком сложной. Например, задача построения дерева решений минимального размера, согласующегося с определенной обучающей выборкой, является NP-трудной. Поэтому обычно применяются различные эвристические методы, позволяющие найти субоптимальное решение. Таких субоптимальных решений может быть несколько. Использование ансамблей алгоритмов может рассматриваться как способ компенсации несовершенства алгоритмов поиска. Третий довод заключается в том, что вид разделяющих поверхностей доступных алгоритмов может не позволять выразить оптимальную разделяющую поверхность. Вместо этого доступные алгоритмы предоставляют несколько одинаково хороших способов аппроксимации такой поверхности. Например, разделяющие поверхности большинства типов деревьев решений состоят из гиперплоскостей, перпендикулярных одной из осей координат. Если оптимальная граница между классами имеет диагональный вид, то различные деревья решений будут выдавать различную аппроксимацию этой диагональной линии "ступенчатыми" ломанными. Комбинация этих различных ломанных, например, с помощью голосования позволит найти более точную аппроксимацию исходной диагональной прямой. Следует заметить, что в итоге получится также "ступенчатая" ломанная, что эквивалентно построению более сложного дерева решений. Тем не менее, если бы изначально был выбран алгоритм, требующий настройки такого большого количества параметров, то повысилась бы опасность получить в итоге переобученный алгоритм.

Представляется оптимальным вариант, когда решение производится ансамблем алгоритмов в два этапа [12]. Сначала задача решается независимо друг от друга всеми или частью из имеющихся алгоритмов. Далее, по полученным решениям вычисляется окончательное "коллективное" решение. Данный подход позволяет надеяться, что при синтезе коллективного решения ошибки отдельных алгоритмов будут компенсироваться правильными ответами других алгоритмов. Действительно, данная гипотеза практически подтверждается, и коллективные решения обычно оказываются наилучшими, или близкими к наилучшим по отдельным методам. Кроме того, данная двухэтапная схема позволяет эффективно решать задачи в автоматическом режиме, что делает доступным использование системы неквалифицированным пользователем.

Рассмотрим различные аспекты построения ансамблей алгоритмов. Более подробно эти аспекты рассмотрены в [11] и [12]. При этом необходимо отметить, что при поиске *ассоциативных правил* целью является нахождение частых зависимостей между объектами или событиями. Эти зависимости выражаются в виде правил вида *«если (условие) то (заключение)»*, и в силу этого плохо приспособлены для использования в ансамблях. Ассоциативные правила не будут рассматриваться в этой работе.

5.2 Принципы построения алгоритмических композиций

5.2.1 Усреднение

В этом случае корректирующая операция не получает информации о том, в какой области пространства находится объект, и работает только с ответами, выданными базовыми алгоритмами. Если базовые алгоритмы достаточно различны, то их погрешности компенсируются в результате усреднения. Причём усреднение следует понимать в обобщённом смысле, это не обязательно среднее арифметическое, и даже не обязательно линейная операция.

5.2.2 Специализация

Пространство объектов делится на области, в каждой из которых строится свой алгоритм, специализирующийся на объектах только этой области. Исходная задача разбивается на более простые подзадачи по принципу «разделяй и властвуй».

5.3 Получение результата

5.3.1 Для случаев классификации и регрессии

Наиболее часто используют следующие методы: обычное голосование (среднее арифметическое для регрессии), взвешенное голосование (среднее

арифметическое с учетом веса алгоритма для регрессии), иерархическая классификация (stacking). Суть *обычного голосования* заключается в том, что ответ каждого из классификаторов засчитывается как «голос» в пользу предсказанного им класса. В итоге выбирается класс с максимальным числом голосов. Различные методы *взвешенного голосования* отличаются тем, что голос каждого классификатора учитывается с определенным весом. Обычно этот вес вычисляется на основе оцененной точности классификатора. В силу выбранного метода построения ансамбля классификаторы имеют примерно одинаковую точность, то есть в нашем случае этот подход мало отличается от обычного голосования. Метод *stacking* основан на использовании полученного множества векторов ответов для обучения некоторого классификатора следующего уровня. Такой подход позволяет достичь высокой адаптивности, однако, при построении классификаторов верхнего уровня возможны те же проблемы, которые наблюдаются и у обычных классификаторов, например, проблема возможного переобучения.

5.3.2 Для случая кластеризации

Особенность кластеризации состоит в том, что метки, которые назначаются при кластеризации алгоритмами, весьма условны. Одно и тоже разбиение может выглядеть по разному даже при двух запусках одного алгоритма:

Объект	1	2	3	4	5	6	7	8	9
Запуск 1	1	1	1	2	2	2	2	3	3
Запуск 2	2	2	2	3	3	3	3	1	1

Для решения этой проблемы в [12] приведён следующий метод. Пусть имеется некоторый набор из N объектов, подлежащих кластеризации. Пусть в ансамбле участвует L алгоритмов и требуемое количество кластеров равно I. Для каждого алгоритма ансамбля построим разбиение данных на I кластеров. Для каждого такого разбиения построим матрицу $M^{(k)}$

$$M^{(k)} = \{m_{ij}^{(k)}\} \in \mathbf{R}_{N\times N}, k = 1,...,L$$

$$m_{ij}^{(k)} = \begin{cases} 1, & \text{если i-ый и j-ый объекты находятся в одном кластере} \\ 0, & \text{если i-ый и j-ый объекты находятся в разных кластерах} \end{cases}$$

Такая матрица показывает, какие объекты принадлежат одному кластеру, и по ней однозначно восстанавливается разбиение. Просуммируем эти L матриц:

$$S = \{s_{ij}\} \in \mathbf{R}_{N\times N}$$

$$s_{ij} = \sum_{k=1}^{L} m_{ij}^{(k)}$$

Зафикисируем индекс уверенности k. Построим новую матрицу

$$R = \{r_{ij}\} \in \mathbf{R}_{N\times N}$$

$$r_{ij} = \begin{cases} 1 & \text{если } s_{ij} \geq k \\ 0 & \text{иначе} \end{cases}$$

Восстановим по ней разбиение на кластеры. Это разбиение и будет искомым.

Заметим, что этот метод соответствует случаю простого голосования, в случае же взвешенного голосования матрицы $M^{(k)}$ суммируются с весами, приписанными соответствующим алгоритмам:

$$s_{ij} = \sum_{k=1}^{L} w_k m_{ij}^{(k)}$$

где w_i – вес *i*-того алгоритма.

5.3.3 Методы построения ансамблей:

- *манипулирование обучающей выборкой* (Bagging, Cross-validated committees, AdaBoost),
- *манипулирование набором признаков* (Manipulating the Input Features),
- *манипулирование целевыми ответами* (Manipulating the Output Targets),
- *использование рандомизации* (Injecting Randomness),
- *использование различных алгоритмов классификации*.

Все эти методы, кроме последнего, подразумевают, что используемые алгоритмы для построения различных алгоритмов ансамбля однородны.

Метод манипулирования обучающей выборкой заключается в использовании различных подмножеств имеющихся данных для обучения различных алгоритмов из ансамбля. Этот метод подразумевает, что для построения каждого отдельного классификатора используется лишь часть доступных данных.

Метод манипулирования набором признаков заключается в обучении различных классификаторов с использованием различных наборов атрибутов исследуемых объектов. Этот метод хорошо работает, если имеется много признаков и они сильно избыточны, причем информация разделена между ними достаточно равномерно.

Метод манипулирования целевыми ответами (в случае классификации) заключается в сведении задачи определения одного из большого количества классов к решению множества задач бинарной классификации.

Метод использования рандомизации состоит во внесении шума в данные, то есть изменении части значений. Этот метод также может применяться для восполнения неизвестных значений, что полезно в случаях, когда не все алгоритмы ансамбля могут работать в условиях недостающих данных.

Метод использования различных алгоритмов обучения подразумевает использование различных, часто разнородных алгоритмов (деревья решений,

функции, алгоритмы сравнения с шаблонами, байесовские классификаторы и.т.д.). Предполагается, что при использовании разнородных алгоритмов понижается корреляция ошибок, и результат оказывается более точным. В силу особенностей астрономических данных первые три метода представляются менее перспективными (так как с одной стороны, обучающая выборка зачастую невелика, и использование только её части может привести к заметному снижению точности, с другой стороны, множество атрибутов также мало, и могут присутствовать нетривиальные связи между различными атрибутами, наконец, структура классов объектов может сильно отличаться от двоичного дерева, возникающего в случае *метода манипулирования целевыми ответами)*, наилучшие же результаты ожидаются от комбинации последних двух (*метода использования рандомизации* и *использования различных алгоритмов обучения*).

6 Реализация встраивания средств Data Mining в инфраструктуру AstroGrid

Реализация встраивания средств Data Mining в инфраструктуру Astrogrid разбивается на две подзадачи: реализацию универсального средства Data Mining (ансамбль алгоритмов) и само встраивание в инфраструктуру AstroGrid, естественное для этой инфраструктуры.

Реализация ансамбля должна быть достаточно гибкой, чтобы при изменении задачи или её параметров (например, изменении атрибутов) не было необходимости менять реализацию. Представляется целесообразным иметь некоторый файл (в качестве формата был выбран XML), описывающий решаемую задачу. Таким образом, для решения новой задачи достаточно составить новое описание. Общая структура файла с описанием задачи включает следующее:

- Вид решаемой задачи (регрессия, классификация, кластеризация)
- Список алгоритмов, включаемых в состав ансамбля
- Список параметров задачи, таких, как желаемое количество кластеров в случае задачи кластеризации, целевой атрибут в задачах классификации и регрессии, иерархия классов в задаче классификации, порог индекса уверенности
- Список релевантных атрибутов (что позволяет исключить из рассмотрения атрибуты, не релевантные на данном шаге)
- Условие разделения данных на обучающую выборку и данные, подлежащие обработке (для случаев классификации и регрессии)
- Вид обобщающей функции (например, простое или взвешенное голосование).

На вход ансамблю подается каталог данных и файл с описанием задачи. Основываясь на описании задачи, программа преобразует данные, причём необходимо отметить, что действия по преобразованию могут иметь иерархическую структуру, определяемую иерархической природой данных (как например, в случае иерархической классификации). В случае классификации и регрессии происходит разделение данных на обучающую и классифицируемую выборки и обучение алгоритмов с использованием обучающей выборки.

Затем алгоритмы применяются к обрабатываемым данным. Поскольку не все алгоритмы Weka умеют работать с пропущенными значениями, то в случае отсутствия значения какого-либо атрибута объекта, оно заменяется для каждого алгоритма случайным образом одним из возможных значений этого атрибута. Это также позволяет оценить степень влияния этого атрибута при данных значениях известных атрибутов на результат.

Полученные результаты обрабатываются обобщающей функцией. Так, в случае классификации и простого голосования для каждого объекта определяется количество алгоритмов, приписавших ему данный класс, и после этого выбирается класс, получивший наибольшее число голосов. Число голосов запоминается как новый параметр - «*индекс уверенности*».

В общем случае после получения входных данных один или несколько раз выполняется следующая процедура: данные, подготовленные в соответствии с описанием задачи, передаются требуемым алгоритмам. Результаты работы алгоритмов передаются требуемой функции обобщения.

Результатом работы ансамбля является новая таблица, уже содержащая тот или иной результат в зависимости от типа задачи. Схема работы Ensembled Weka представлена на рис. 1.

Для того, чтобы пользователи могли работать с распределенными данными, а также использовать средства визуализации, необходимо некоторое промежуточное пространство для хранения данных. Таким пространством является MySpace.

Встраивание в AstroGrid системы Weka, ориентированное на применение ансамблей алгоритмов, называется далее Ensembled Weka. Ensembled Weka представляет собой CEA-приложение, принимающее в качестве входных параметров файл с описанием задачи и таблицу со входными данными. Результат работы ансамбля помещается в таблицу и возвращается в MySpace. Затем к ним возможны обращения со стороны различных приложений визуализации и анализа данных, доступных в среде AstroGrid, а также эти результаты могут быть использованы в качестве входных параметров других шагов сложных потоков работ, реализуя тем самым механизм связывания шагов в потоке работ.

Таким образом стандартный сценарий использования разработанных средств Data Mining в AstroGrid, выглядит следующим образом:

1. Подготовить данные с помощью AstroWeka, Topcat или любых других сторонних приложений.
2. Загрузить данные в MySpace с помощью клиента среды АстроГрид Workbench. В качестве альтернативы первым двум шагам может быть некоторый поток работ образующий входные данные.
3. Создать файл описания задачи
4. Загрузить файл описания задачи в MySpace.
5. Затем либо запустить приложение, выполняющее ансамбль алгоритмов (Ensembled Weka) либо написать поток работ с участием Ensembled Weka.

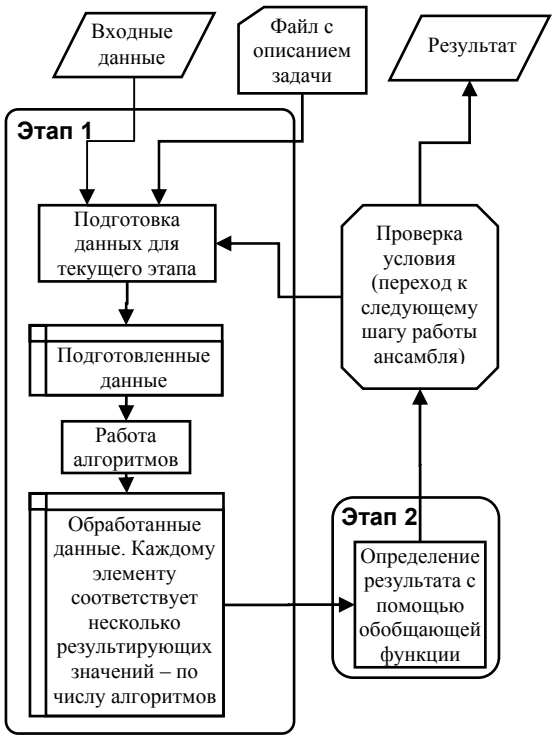

Рис. 1. Схема работы Ensembled Weka

Таким образом, пользователь AstroGrid может легко использовать разработанный сервис обработки данных средствами Data Mining, как решая простую задачу, так и решая сложную задачу в большом потоке работ. К моменту написания настоящей статьи Ensembled Weka поддерживает файлы описания задач, позволяющие определять разнообразные задачи классификации, но в ближайшее время планируется расширить их возможности для поддержки и других видов задача Data Mining – кластеризации и регрессии.

7 Пример решения задачи классификации затменных двойных в предложенной архитектуре

Разработанная реализация Ensembled Weka в составе AstroGrid была проверена на задаче классификации двойных затменно-переменных звезд, поставленной О.Ю.Малковым [13]. Классификация проводилась на новом, переработанном О.Ю.Малковым каталоге, учитывающем недостатки, выявленные при решении задачи классификации, описанном в [13].

7.1 Описание задачи

Затменно-переменные звезды – это системы двойных звезд, плоскость вращения которых образует малый угол с направлением на Землю. Такие звезды при вращении затмевают друг друга, что приводит к периодическому изменению их общей светимости. Часто компоненты такой звездной системы не могут наблюдаться независимо друг от друга, поэтому их изучение базируется на анализе изменения светимости.

Существует несколько каталогов затменно-переменных звезд, например: General Catalogue of Variable Stars (GCVS); A Finding List for Observers of Interacting Binary Systems, 5th Edition; Eclipsing variables in microlensing surveys. Данные из этих каталогов были собраны О.Ю.Малковым в один каталог, в котором сейчас есть информация о 6675 звездах. Из них определен класс у 1161 звезды.

7.2 Атрибуты каталога

Основная информация о звезде извлекается из так называемой кривой блеска – графика изменения светимости затменно-переменной системы во времени [13]. Этот график обычно имеет вид кривой с двумя минимумами, из которых более глубокий называется главным, а другой – вторичным. Первый из атрибутов, который можно встретить в каталогах, это морфологический тип кривой блеска. У кривых различных типов участки затмения и нормальной светимости переходят друг в друга с различной степенью плавности. Группа фотометрических атрибутов характеризует значения блеска в определенные моменты времени. К фотометрическим атрибутам относятся: блеск в максимуме, блеск в главном минимуме, глубина главного минимума, блеск во вторичном минимуме, глубина вторичного минимума, разность глубин минимумов, кроме того, к этой группе относится фотометрическая полоса, которая характеризует частоту, на которой проводились измерения предыдущих величин. Следующая группа атрибутов включает период полного затменного цикла, логарифм периода, а также указатель на переменность периода. Далее идет группа атрибутов фазы, характеризующих временные свойства затмений. Среди них: продолжительность главного затмения, продолжительность полной фазы в

главном минимуме, продолжительность вторичного затмения, продолжительность полной фазы во вторичном минимуме, а также период между минимумами. Следует отметить, что некоторые из этих атрибутов имеют смысл не для всех типов кривой блеска. Еще есть группа спектральных характеристик: спектральный класс компонента 1, класс светимости компонента 1, спектральный класс компонента 2 и класс светимости компонента 2. Набор этих атрибутов и образует ту информацию, которую можно извлечь из имеющегося каталога. Впрочем, для некоторых звезд значения некоторых из этих атрибутов неизвестны.

7.3 Классы двойных звезд

Для разделения двойных звезд на типы большое значение имеет понятие эквипотенциальной поверхности. Это такая область пространства, на которой потенциальная энергия тела единичной массы, связанная с вращением и гравитационным притяжением, постоянна. Для значений энергий, характерных для близких расстояний до центра одной из звезд, такие поверхности образуют независимые замкнутые фигуры, каждая из которых содержит внутри себя по звезде. При удалении от звезды, наступает момент, когда разные части эквипотенциальной поверхности касаются, а затем сливаются в одну. В зависимости от стадии развития системы, вещество двойных звезд может по-разному располагаться относительно такой минимальной общей эквипотенциальной поверхности. В зависимости от этого расположения и выделяют три основных класса двойных звезд. У звезд, принадлежащих к классу «разделенные» (Detached, D), вещество обоих компонентов локализовано внутри разных частей общей эквипотенциальной поверхности и никак не контактирует. У звезд из класса «полу-разделенные» (Semi-detached, SD) вещество одного из компонентов располагается компактно, как и у «разделенных», зато у второй вещество выходит за пределы общей эквипотенциальной поверхности и может перетекать на первый компонент. У системы класса «контактные» (Contact, C) вещество обоих компонентов достигает общей эквипотенциальной поверхности, образуя как бы одну звезду с двумя ядрами, но общей оболочкой. Существует также более подробное деление на классы. Полностью структура выглядит так:

C (Contact, контактные)
- CB (почти контактные системы)
  - CBF
  - CBV
- CE (контактные системы раннего спектрального типа)
- CG (гигантские C системы)
- CW (контактные системы позднего спектрального типа)
  - CWA
  - CWW

D (Detached, разделённые)
- D2S (разделенные симбиотические)
- DG (разделенные системы с гигантом или сверхгигантом)
- DM (разделенные системы, образованные звездами главной последовательности)
- DR (разделенные системы, содержащие два субгиганта)
- DW (разделенные системы с компонентом – белым карликом или предшественником белого карлика, и вторичным компонентом – маломассивной звездой)

S (Semi-detached, полу-разделённые)
- S2C (катаклизмические переменные системы)
- S2H (массивные рентгеновские)
- S2L (маломассивные рентгеновские)
- SA ("классические" Алголи)
- SC (холодные)
- SH (горячие)

7.4 Решение задачи

В первую очередь было необходимо подготовить данные для классификации, что было проделано с помощью средств, предоставляемых AstroWeka.

Был проведен анализ атрибутов, и были выбраны и удалены атрибуты, которые не имеют непосредственного отношения к определению класса звезды и, таким образом, не нужны для классификации (это абсолютное значение блеска в максимуме, блеска в главном минимуме и блеска во вторичном минимуме, а также фотометрическая полоса и указатель на переменность периода).

Также были удалены атрибуты, которые неизвестны для большинства звезд (это продолжительность полной фазы вторичного затмения, фаза между минимумами, показатель хромосферной активности и др.)

Далее был проведен анализ гистограммы атрибутов, показавший, что в случае, например, таких атрибутов, как показатель хромосферной активности (E8) и указатель на переменность периода (C4) звезды разных классов распределены примерно равномерно по всем значениям этого параметра. Это подтверждает правильность решения об исключении этого параметра.

С другой стороны уже видно, у каких атрибутов звезды разных классов распределены неравномерно, что говорит о том, что такие атрибуты будут оказывать решающее влияние на классификацию (например, морфологический тип кривой блеска).

Анализ проекций множества звезд на плоскости, образованные различными парами атрибутов, показал, что на значительной части этих проекций объекты разных классов распределены не полностью хаотически, а заметно отделение объектов различных классов. Тем не менее, на большинстве проекций облака различных классов довольно сильно пересекаются. Соответственно, в областях пересечения объекты разных классов перемешаны.

После этого было проанализировано поведение различных алгоритмов классификации на полученном наборе данных и выбраны те из них, которые показали себя наилучшим образом.

Точность оценивалась методом «cross-validation». Он заключается в том, что доступные данные делятся на несколько равных групп (в данном случае 10). Производится 10 сеансов обучения классификаторов. В качестве проверочной выборки используется одна из 10 групп, каждый раз разная. В качестве обучающей выборки – остальные группы. Полученный результат усредняется. Это самый объективный способ оценки точности индивидуального классификатора в системе Weka. Наиболее объективной же оценкой качества набора параметров будет, как уже говорилось, усреднения результата нескольких классификаторов, у которых индивидуальная точность хорошая, но не рекордная.

Сначала исследовалась точность классификаторов при использовании полного набора параметров. Здесь удалось достичь точности классификации около 88%. Затем по очереди удалялись один или два параметра, и исследовалась точность на таком сокращенном наборе. Выяснилось, что в большинстве случаев точность от этого меняется слабо или уменьшается. Это подтверждает гипотезу о том, что большинство хороших алгоритмов классификации самостоятельно справляются с выделением наиболее важных признаков при таком соотношении общего числа признаков и количества примеров в обучающей выборке. При удалении же морфологического типа и логарифма периода точность понижается особенно заметно и составляет около 75%. Это означает, что это ключевые параметры.

При отборе алгоритмов построения классификаторов для ансамбля учитывались оценка их точности на имеющихся данных, устойчивость точности при разных условиях обучения, а также оригинальность принципов работы алгоритма. В результате были отобраны

- **Байесовские классификаторы**
  - NaiveBayes
- **Классификаторы, представляемые в виде функций**
  - MultilayerPerceptron
  - Logistic
- **Классификаторы, основанные на сравнении с шаблонами**
  - KStar
- **Деревья решений**
  - J48
  - LMT
  - NBTree
  - RandomForest
- **Решающие правила**
  - PART
  - JRip

Результат классификации определяется с помощью простого голосования, по схеме, приведённой на рис. 2.

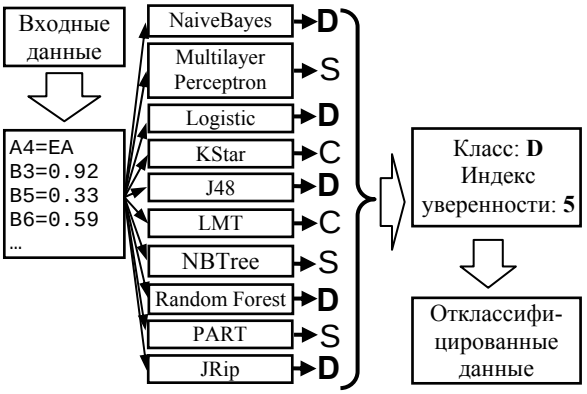

Рис. 2. Определение результата классификации с помощью простого голосования

После всех необходимых подготовок был сформирован XML-файл описания задачи, где были описаны атрибуты и иерархическая структура классификации:

```
<task_description>
<task type='classification'/>
<ensemble>
    <algorithm name='NaiveBayes'></algorithm>
    <algorithm
name='MultilayerPerceptron'></algorithm>
    <algorithm name='Logistic'></algorithm>
    <algorithm name='KStar'></algorithm>
    <algorithm name='J48'></algorithm>
    <algorithm name='LMT'></algorithm>
    <algorithm name='NBTree'></algorithm>
    <algorithm name='RandomForest'></algorithm>
    <algorithm name='PART'></algorithm>
    <algorithm name='JRip'></algorithm>
</ensemble>
<parameters>
    <parameter name='ConfidenceThreshold'
value='7'></parameter>
</parameters>
<classes>
  <attributes>
    number, morftype, B3_main_min_depth,
    B5_sec_min_depth, B6_depth_difference,
    C2_log_period, D1_main_eclipse,
    D3_main_phase, D4_sec_eclipse,
    E2_temp_class_1, E4_bright_class_1
  </attributes>
    <class name='D'>
    <class name='DM'> </class>
    <class name='DR'> </class>
    <class name='DG'> </class>
    <class name='D2S'> </class>
    <class name='DW'> </class>
  </class>
  <class name='S'>
    <class name='SA'> </class>
    <class name='SC'> </class>
    <class name='SH'> </class>
    <class name='S2C'> </class>
    <class name='S2H'> </class>
    <class name='S2L'> </class>
  </class>
  <class name='C'>
    <class name='CB'>
      <class name='CBF'> </class>
      <class name='CBV'> </class>
    </class>
```

Табл. 1. Точность ансамбля классификаторов при различных индексах уверенности для объектов классов C

Индекс уверенности	C	CB	CBF	CBV	CE	CG	CW	CWA	CWW	Всего объектов
10	100.00%	100.00%	-	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	131
9	100.00%	90.91%	100.00%	100.00%	100.00%	-	100.00%	100.00%	100.00%	77
8	100.00%	100.00%	100.00%	100.00%	100.00%	-	100.00%	41.67%	33.33%	64
7	100.00%	100.00%	-	-	100.00%	-	100.00%	36.36%	61.54%	65
6	97.67%	62.96%	0.00%	0.00%	-	50.00%	100.00%	52.94%	50.00%	94
5	92.31%	44.44%	0.00%	0.00%	0.00%	0.00%	66.67%	100.00%	0.00%	45
4	-	0.00%	-	-	0.00%	-	-	0.00%	-	5
Всего объектов	155	84	5	8	15	4	29	83	98	481

Табл. 2. Точность ансамбля классификаторов при различных индексах уверенности для объектов классов D

Индекс уверенности	D	D2S	DG	DM	DR	DW	Всего объектов
10	-	100.00%	100.00%	100.00%	100.00%	100.00%	106
9	-	100.00%	100.00%	88.46%	100.00%	100.00%	45
8	100.00%	-	100.00%	100.00%	100.00%	0.00%	13
7	-	100.00%	66.67%	100.00%	75.00%	-	10
6	0.00%	0.00%	33.33%	0.00%	60.00%	0.00%	28
5	-	-	20.00%	14.29%	100.00%	0.00%	14
4	-	-	-	0.00%	-	-	1
Всего объектов	2	5	21	153	22	14	217

Табл. 3. Точность ансамбля классификаторов при различных индексах уверенности для объектов классов S

Индекс уверенности	S	S2C	S2H	S2L	SA	SC	SH	Всего объектов
10	100.00%	-	-	-	95.79%	-	80.00%	103
9	100.00%	100.00%	-	-	98.68%	-	100.00%	250
8	85.71%	100.00%	-	-	100.00%	100.00%	100.00%	30
7	100.00%	100.00%	0.00%	-	0.00%	0.00%	-	18
6	100.00%	0.00%	-	-	6.67%	-	0.00%	27
5	50.00%	50.00%	-	-	0.00%	0.00%	0.00%	24
4	0.00%	0.00%	0.00%	0.00%	-	-	0.00%	11
Всего объектов	32	32	3	3	356	4	33	463

Рис. 3. Схема работы ансамбля классификаторов в задаче о классификации затменных двойных звёзд

```
    <class name='CE'> </class>
    <class name='CG'> </class>
    <class name='CW'>
      <class name='CWA'> </class>
      <class name='CWW'> </class>
    </class >
  </class >
</classes>
</task_description>
```

Этот файл, а также каталог объектов, были помещены в MySpace, и была проведена классификация. Схема работы Ensembled Weka для задачи о классфикации двойных затменных звёзд приведена на рис. 3.

В результате работы ансамбля было отклассифицировано 5514 звёзд, распределившихся по классам следующим образом:

C – 852
- CB – 89
  - CBF – 74
  - CBV – 149
- CE – 15
- CG – 1
- CW – 84
  - CWA – 427
  - CWW – 331

S – 547
- S2C – 3
- SA – 1902
- SC – 1
- SH – 13

D – 553
- DG – 41
- DM – 422
- DR – 10

В качестве порогового значения индекса уверенности было выбрано 7, и звёзды,

отклассифицированные с индексом уверенности, меньшим 7, получили неполную классификацию.

Результаты проверки точности ансамбля на звёздах с известным классом показаны в табл. 1-3.

Заключение

На основании проведённого анализа методов Data Mining и способов их интеграции в инфраструктуру Виртуальной Обсерватории было создано СЕА-приложение, получившее название Ensembled Weka и реализующее различные варианты построения ансамбля алгоритмов. Ensembled Weka решает задачу, определенную файлом описания задачи. При этом поддерживается возможность иерархической обработки данных.

СЕА-приложение Ensembled Weka, позволяющее решать задачи классификации, было размещено на сервере AstroGrid ИПИ РАН, информация о нём опубликована в реестрах AstroGrid, таким образом, Ensembled Weka доступно для использования.

К моменту написания статьи Ensembled Weka позволяет решать широкий набор задач классификации, естественным образом встраивая процесс решения в существующую инфраструктуру AstroGrid.

Описан пример решения задачи классификации двойных затменно-переменных звезд с помощью Ensembled Weka. Результаты этого решения показывают эффективность разработанного средства.

В дальнейшем планируется развитие средств Ensembled Weka для решения задач кластеризации и регрессии на основе ансамблей алгоритмов.

Литература

[1] AstroGrid Release 2007.1, http://software.astrogrid.org

[2] AstroGrid Workbench, http://www2.astrogrid.org/desktop

[3] AstroWeka - Data Mining in the Virtual Observatory, http://astroweka.sourceforge.net/

[4] Briukhov D.O., Kalinichenko L.A., Zakharov V.N., Panchuk V.E., Vitkovsky V.V., Zhelenkova O.P., Dluzhnevskaya O.B., Malkov O.Yu., Kovaleva D.A. Information Infrastructure of the Russian Virtual Observatory (RVO). Second Edition. IPI RAN, 2005, 173 p.

[5] Briukhov D., Kalinichenko L., Martynov D., Stupnikov S., Vovchenko A. (IPI RAS) Zhelenkova O. (SAO RAS). Distant galaxy discovery problem design in AstroGrid and Aladin at IPI RAS in Moscow, http://synthesis.ipi.ac.ru/synthesis/projects/astromedia/distgal.

[6] Tomas G. Dietterich. Machine Learning Research: Four Current Directions, AI Magazine, 1997, Vol. 18, no. 4, pp. 97–136.

[7] Groovy - An agile dynamic language for the Java Platform, http://groovy.codehaus.org/

[8] Kalinichenko L. A., Stupnikov S. A., Vovchenko A.E., Zakharov V. N., Zhelenkova O.P. Russian Virtual Observatory Community Centre for Scientific Problems Solving over Multiple Distributed Information Sources, The 8th Russian Conference on Digital Libraries RCDL2006, Suzdal, Russia, 2006

[9] Weka - Data Mining Software in Java, http://www.cs.waikato.ac.nz/ml/weka/

[10] АстроГрид Российской Виртуальной Обсерватории как Центр коллективного пользования для решения астрономических задач над распределенными репозиториями информации, накапливаемыми в мире, http://synthesis.ipi.ac.ru//synthesis/projects/astromedia/astroannounce

[11] Воронцов К. В. Оптимизационные методы линейной и монотонной коррекции в алгебраическом подходе к проблеме распознавания // ЖВМ и МФ. 2000.Т. 40, № 1.- С. 166–176.

[12] Ю.И.Журавлев, В.В.Рязанов, О.В.Сенько “Распознавание. Математические методы. Программная система. Практические применения.” Изд. Фазис, Москва, 2005.

[13] Л.А.Калиниченко, А.Е.Лебедев, О.Ю.Малков. Применение методов и средств извлечения знаний из данных в астрономии на примере классификации затменных звезд. Москва, ИПИ РАН, 2007.

Integration of Data Mining Tools in the Infrastructure of Virtual Observatory

Mamardashvili N.A., Patrakova M.E.,
Vovchenko A.E., Malkov O.Y.,Kalinichenko L.A.

Data Mining methods are used in different fields of science, including astronomy, as the means to help obtain new knowledge and make scientific discoveries. The importance of incorporation of means for the solution of astronomical problems by Data Mining methods into the virtual observatories is discussed. Existing approaches are examined, preferences are given to the application of the algorithmic ensembles, the respective architecture (Ensembled Weka) of incorporation of the Weka Data Mining system into the infrastructure of virtual observatory is proposed.

♣ Эти исследования были частично поддержаны грантами РФФИ 06-07-08072 и 06-07-89188, а также проектом 1-10 РАН программы “ Фундаментальные основы информационных технологий и систем ”