ISSN 1814-1196
Научный вестник НГТУ
том 67, № 2, 2017, с. 109–119

http://journals.nstu.ru/vestnik
Science Bulletin of the NSTU
Vol. 67, No. 2, 2017, pp. 109–119

ОБРАБОТКА ИНФОРМАЦИИ INFORMATION PROCESSING

УДК 681.518.2

Применение методов машинного обучения для построения рекомендательной системы отбора анкет абитуриентов*

В.В. МАЛЫШЕВ[1], С.С. СЛИВКИН[2], В.С. РУКАВИШНИКОВ[3], Е.В. БАЗАРКИН[4]

[1] *634034, РФ, г. Томск, ул. Усова, 4а, Национальный исследовательский Томский политехнический университет, ведущий инженер. E-mail: mvv@hw.tpu.ru*
[2] *634034, РФ, г. Томск, ул. Усова, 4а, Национальный исследовательский Томский политехнический университет, инженер. E-mail: ss@hw.tpu.ru*
[3] *634034, РФ, г. Томск, ул. Усова, 4а, Национальный исследовательский Томский политехнический университет, Центр подготовки и переподготовки специалистов нефтегазового дела, директор. E-mail: RukavishnikovVS@hw.tpu.ru*
[4] *634034, РФ, г. Томск, ул. Усова, 4а, Национальный исследовательский Томский политехнический университет, специалист. E-mail: bazarkin@me.com*

В работе рассматривается возможность применения методов машинного обучения для построения рекомендательной системы отбора анкет абитуриентов. Рекомендательная система на основании предсказаний, полученных от классификатора, выдает рекомендуемый статус для анкеты. Для построения математической модели классификатора используются сведения об анкетах предыдущих периодов, которые содержат данные об итоговом статусе каждой анкеты. Данные об анкетах необходимо специальным образом подготовить для использования в линейных математических моделях. Числовые признаки необходимо нормализовать, а категориальные – преобразовать в числа с помощью бинарного кодирования. Для оценки качества работы классификаторов их модель обучают на обучающей выборке, а качество предсказаний проверяют на тестовой выборке. Для исключения вероятности получения несбалансированной выборки (выборки, в которой объектов одного класса может быть значительно больше, чем другого) используется техника кросс-валидации – техника многостадийного разбиения данных. При каждом шаге данные разбиваются на части, затем происходит обучение модели на тренировочной выборке и валидация модели на тестовой выборке. Для уменьшения ошибки классификатора на каждом шаге происходит разбиение на разные части исходного набора данных. Проведено сравнение точности работы нескольких видов классификаторов и определен наиболее точный метод – случайный лес. Определен порядок имплементации выбранного классификатора в существующую автоматизированную систему сбора, обработки и учета анкетных данных. В результате проделанной работы определены методы классификации данных для построения модуля рекомендаций автоматизированной системы сбора, обработки и учета анкетных данных. Для повышения качества отбора слушателей в качестве направления дальнейшей исследовательской деятельности выбрана технология построения и внедрения чат-ботов для формирования психологического портрета абитуриента. Чат-боты – особый вид диа-

* *Статья получена 24 апреля 2017 г.*

логового взаимодействия на основе технологий искусственного интеллекта и распознавания естественного языка.

Ключевые слова: машинное обучение, классификатор, математическая модель, кросс-валидация, бинарное кодирование, Python

DOI: 10.17212/1814-1196-2017-2-109-119

ВВЕДЕНИЕ

Подготовка слушателей по магистерским программам – дорогостоящий процесс, особенно для учебных заведений, осуществляющих подготовку специалистов на основе свободного набора слушателей. Одним из таких учебных заведений является Центр подготовки и переподготовки специалистов нефтегазового дела Томского политехнического университета (http://hw.tpu.ru), который совместно с университетом Heriot-Watt University (Эдинбург) осуществляет подготовку по магистерским программам с последующим трудоустройством выпускников в российских и зарубежных нефтегазовых компаниях. Затраты на подготовку 83 % слушателей компенсируют заинтересованные компании-работодатели. Поэтому для коммерчески успешной работы Центра необходимо проводить предварительный отбор абитуриентов, способных успешно пройти обучение по магистерским программам. Процедура отбора проходит в несколько этапов и включает в себя прием анкет, тестирование по английскому языку и собеседование. Кроме основных формальных критериев (возраст, образование, гражданство) и тестирования по английскому языку определяющим для зачисления являются результаты собеседования, которые проводят сотрудники центра. Для автоматизации процесса набора была разработана автоматизированная система для сбора, обработки и учета анкетных данных абитуриентов, а также проверки знаний по английскому языку. В качестве апробации автоматизации процесса отбора слушателей инициативной группой сотрудников Центра было принято решение провести исследовательские работы для выявления возможности применения машинного обучения на различных этапах процесса набора слушателей.

1. МАШИННОЕ ОБУЧЕНИЕ

Машинное обучение (англ. machine learning) – обширный подраздел искусственного интеллекта, математическая дисциплина, использующая разделы математической статистики, численных методов оптимизации, теории вероятностей, дискретного анализа и извлекающая закономерности из данных. Машинное обучение уже нашло применение в следующих областях: в биоинформатике, в медицине (медицинская диагностика), в геологии и геофизике, в социологии, в экономике (кредитный скоринг, предсказание оттока клиентов, обнаружение мошенничества, биржевой технический анализ, биржевой надзор), в технике (техническая диагностика, робототехника, компьютерное зрение, распознавание речи), в офисной автоматизации (распознавание текста, обнаружение спама, категоризация документов, распознавание рукописного ввода). Сфера применений машинного обучения постоянно

расширяется. Повсеместная информатизация приводит к накоплению огромных объемов данных в науке, производстве, бизнесе, транспорте, здравоохранении. Возникающие при этом задачи прогнозирования, управления и принятия решений часто сводятся к обучению по прецедентам. Раньше, когда таких данных не было, эти задачи либо вообще не ставились, либо решались совершенно другими методами [1].

Системы с машинным обучением и технологиями искусственного интеллекта предполагают, что программа проводит обработку данных не по заданным правилам, а создает эти правила сами. Такие системы помогают лучше распознавать тексты, изображения и голосовые команды. Поэтому в сфере рекрутинга они позволяют, например, лучше сканировать резюме и описания вакансий, а также создавать программы-ассистенты, которые будут отвечать на распространенные вопросы и давать первоначальную информацию.

Задача классификации – это определение принадлежности объекта наблюдения к определенному классу. При наличии множества объектов, у которых уже известна принадлежность к классам, можно построить алгоритм, который сможет определять, к какому из классов будет принадлежать новый объект. В терминах машинного обучения объекты с известной принадлежностью к классам называются обучающей выборкой, а задача определения принадлежности нового объекта к классу – классификацией. Алгоритм, с помощью которого определяется принадлежность к классу, – классификатор.

Таким образом, применение машинного обучения для отбора анкет слушателей заключается в следующем: по ранее собранным анкетным данным построить алгоритм, который определяет для вновь поступившей анкеты будет ли она рекомендована для прохождения к следующему этапу отбора. Рассмотрим возможность построения такого алгоритма.

2. АНАЛИЗ И ПОДГОТОВКА ДАННЫХ

За время работы системы сбора, обработки и учета анкетных данных абитуриентов собралась обширная информационная база анкетных данных. Каждая запись представляет набор параметров анкеты абитуриента: статус анкеты (отклонена или принята), возраст, город проживания, вуз, средний балл диплома вуза и т. д.

Для построения моделей и анализа данных используется Python – высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время набор стандартных библиотек включает в себя большой объем полезных функций.

На первом этапе необходимо загрузить данные об анкетах из базы данных автоматизированной системы. При этом необходимо учитывать, что не все параметры анкеты (например, адрес, телефон, адрес электронной почты, название школы и т. д.) будут полезны для анализа. Наиболее важными являются данные со следующими признаками: статус анкеты (отклонена или принята), результаты теста, возраст, город проживания, вуз, средний балл диплома вуза и т. д. Загрузка данных осуществляется непосредственно из

СУБД SQL Server. Первоначальные данные представляют массив размером 1100 строк и 14 столбцов следующего вида.

Данные из базы данных автоматизированной системы

TestRes	TestTime	TestRes2	TestTime2	SCHOOL_Bal	EDU1_Bal	EDU2_Bal	EngExamRes	...	STATUS4	PERIODID
70.0	44.0		0.0	4.3	4.4			...		11.0
60.0	41.0		0.0	4.7	4.3	4.8		...		9.0
58.0	45.0		0.0	4.1	4.7	5		...	5	8.0
86.0	45.0		0.0	5.0	5.0	5		...	5	8.0

В машинном обучении признаки описывают объект в доступной и понятной для компьютера форме. Существует несколько классов, или типов признаков, имеющих свои особенности, и их нужно по-разному обрабатывать и по-разному учитывать в алгоритмах машинного обучения. Для этого необходимо разделить признаки на вещественные, т. е. те, которые представлены вещественными числами (средний балл по диплому, школьный средний балл, результат теста по английскому языку).

numeric_columns = ['TestRes', 'TestTime', 'SCHOOL_Bal', 'EDU1_Bal']

X_NumData=AbitData[numeric_columns].replace('', 0)

X_NumData=X_NumData.fillna(0)

X_NumData[['TestRes', 'TestTime', 'SCHOOL_Bal', 'EDU1_Bal']] = X_NumData[['TestRes', 'TestTime', 'SCHOOL_Bal', 'EDU1_Bal']].astype(float)

И категориальные признаки: пол, город, женат/холост, период, решение комиссии.

categorical_columns = ['DECISION', 'PERIODID', 'COURSE_ID', 'CITY_ID', 'SEX_ID', 'MARRIAGE_ID']

X_CatData = AbitData[categorical_columns].fillna('NA')

Особенность категориальных признаков состоит в том, что это элементы некоторого неупорядоченного множества, и нельзя говорить, что какое-то значение больше или меньше другого. Можно только сравнивать их на равенство. Но в линейных моделях нужно брать значение признака, умножать на вес, а потом складывать с другими числами, и эту операцию нельзя делать со значениями категориальных признаков. Для работы с категориальными признаками требуется их сначала преобразовать, чтобы их можно было использовать в линейных моделях. Один из наиболее популярных подходов к преобразованию категориальных признаков – бинарное кодирование, суть которого заключается в следующем. Пусть *j*-й признак – категориальный и принимает *n* возможных значений: *c*1; :::; *cn*: Пусть также *fj*(*x*) – значение этого признака на объекте *x*. Чтобы закодировать данный признак, вводится *n* новых бинарных признаков: *b*1(*x*);:::; *bn*(*x*); причем значение бинарного признака *bi* равно единице только в том случае, если на данном объекте *x* значение категориального признака *fj*(*x*) равно *ci*: *bi*(*x*) = [*fj*(*x*) = *ci*]. В результате

один категориальный признак заменяется *n* бинарными признаками. Средствами Python это делается следующим образом:

```
encoder = DV(sparse = False)
X_CatData_oh = encoder.fit_transform(X_CatData.T.to_dict().values())
print  X_CatData_oh, X_CatData_oh.shape
```

Также необходимо выделить целевой признак (принята или не принята анкета).

```
AbitData['PLCGO']=AbitData['STATUS4'].apply(lambda s: 1 if s in (1,2,3,4) else 0)
Y_Data=AbitData['PLCGO']
```

Из проведенного анализа вещественных признаков следует, что они принимают значения из разных диапазонов, а это может плохо отразиться на результатах при построении моделей (рис. 1).

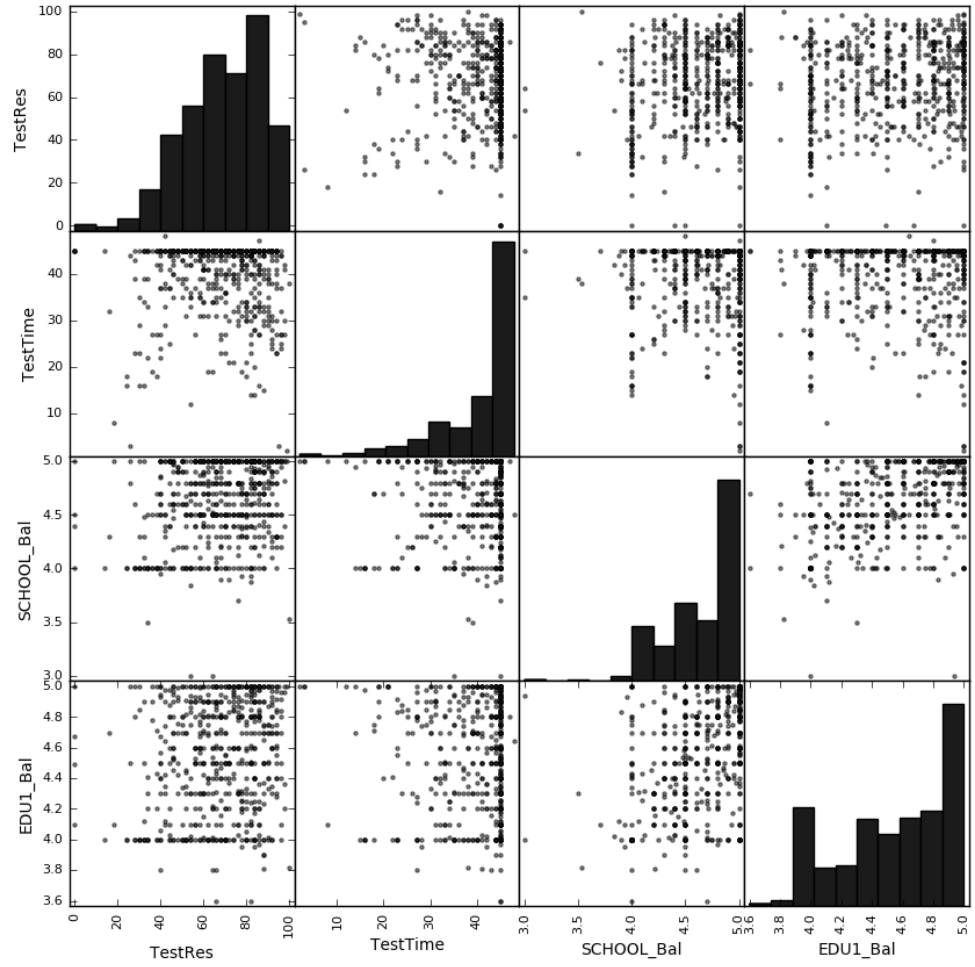

Рис. 1. Взаимное влияние параметров

Выходом из ситуации является нормировка – приведение признаков к единому масштабу:

```
scaler = StandardScaler()
scaler.fit(X_NumData,Y_Data)
```

```
X_Scaled = scaler.transform(X_NumData)
X_NumDataScaled = pd.DataFrame(X_Scaled, columns=numeric_columns)
X_NumDataScaled= X_NumData
```

Далее необходимо провести объединение категориальных и вещественных признаков в единый массив.

```
X_AllTrainData = np.hstack([X_NumTrain, X_train_cat_ohSY])
X_AllTestData = np.hstack([X_NumTest, X_test_cat_ohSY])
```

Когда данные для построения моделей классификации подготовлены, остается определить, каким образом необходимо оценивать точность работы моделей классификации – классификаторов. Для этого подготовленные данные разбиваются на две выборки – тренировочную и тестовую. После обучения классификатора на тренировочной выборке его оценивают на тестовой выборке. Разница в предсказаниях и будет являться критерием качества работы классификатора. Однако, при разбиении данных есть вероятность получить несбалансированные выборки (выборки, в которых объектов одного класса может быть значительно больше, чем другого). Поэтому для разбиения будем использовать кросс-валидацию. Кросс-валидация – это техника многостадийного разбиения данных. При каждом шаге данные разбиваются на части, затем происходит обучение модели на тренировочной выборке и валидация модели на тестовой выборке. Для уменьшения ошибки классификатора на каждом шаге происходит разбиение на разные части исходного набора данных.

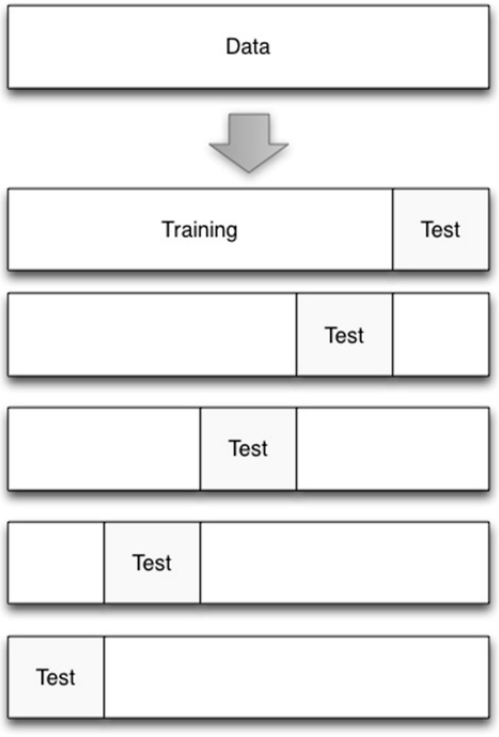

Рис. 2. Графическое пояснение метода кросс-валидации данных

Средствами Python это делается следующим образом:

(X_NumTrain, X_NumTest, Y_train, Y_test) = train_test_split(X_AllTrainData, X_AllTestData, test_size=0.3, random_state=0,stratify= X_AllTestData)

3. ПОСТРОЕНИЕ ОЦЕНКИ И ВЫБОР КЛАССИФИКАТОРОВ

Существует множество классификаторов, использующих разные алгоритмы классификации. У каждого классификатора есть свои преимущества и недостатки, потому главным критерием оценки классификатора является его точность. Построим ряд классификаторов и оценим точность их работы

Метод *K* ближайших соседей – простейший метрический классификатор, основанный на оценивании сходства объектов. Классифицируемый объект относится к тому классу, которому принадлежат ближайшие к нему *K* объектов обучающей выборки.

clf1 = KNeighborsClassifier(n_neighbors=3)
scores1 = cross_val_score(clf1, X_NumTrain, Y_train, cv=cv, n_jobs=-1)
точность классификатора: 0.6725425

Стохастический градиентный спуск – оптимизационный алгоритм, при котором для корректировки параметров модели используется градиент.

clf2 = SGDClassifier(random_state=1)
scores2 = cross_val_score(clf2, X_NumTrain, Y_train, cv=cv, n_jobs=-1)
точность классификатора: 0.708257

Логистическая регрессия – метод построения линейного классификатора, позволяющий оценивать апостериорные вероятности принадлежности объектов классам

clf3 = LogisticRegression(random_state=1)
scores3 = cross_val_score(clf3, X_NumTrain, Y_train, cv=cv, n_jobs=-1)
точность классификатора: 0.80323

Решающее дерево – алгоритм, воспроизводящий логические схемы, позволяющие получить окончательное решение о классификации объекта с помощью ответов на иерархически организованную систему вопросов. Причем вопрос, задаваемый на последующем иерархическом уровне, зависит от ответа, полученного на предыдущем уровне.

clf4 = DecisionTreeClassifier(random_state=1)
scores4 = cross_val_score(clf4, X_NumTrain, Y_train, cv=cv, n_jobs=-1)
Точность классификатора: 0.8257985

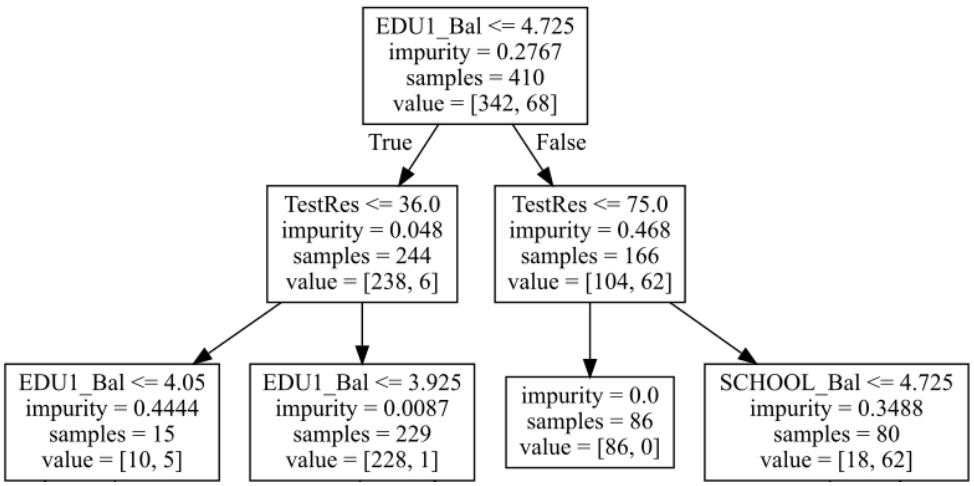

Рис. 3. Графическое представление решения методом решающего дерева

Случайный лес – алгоритм машинного обучения, заключающийся в использовании комитета (ансамбля) решающих деревьев:

clf5 = RandomForestClassifier(n_estimators = 50, max_depth = 10, random_state = 1)
scores5 = cross_val_score(clf5, X_NumTrain, Y_train, cv=cv, n_jobs=-1)
точность классификатора: 0.9624879

ВЫВОДЫ И ДАЛЬНЕЙШИЕ НАПРАВЛЕНИЯ ИССЛЕДОВАНИЙ

На основе проведенных исследований в качестве метода для построения системы предсказания для подбора анкет на магистерские программы рекомендуется использовать классификатор «Случайный лес». Для имплементации выбранного классификатора планируется разработать web-сервис, с которым будет взаимодействовать модуль рекомендации автоматизированной системы сбора, обработки и учета анкетных данных посредством передачи JSON данных.

СПИСОК ЛИТЕРАТУРЫ

1. *Воронцов К.В.* Машинное обучение: лекции [Электронный ресурс]. – URL: http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf (дата обращения: 07.06.2017).

2. *Koren Y., Bell R.M., Volinsky C.* Matrix factorization techniques for recommender systems // Computer. – 2009. – Vol. 42 (8). – P. 30–37.

3. Международное соревнование «Amazon.com – Employee Access Challenge» по анализу данных [Электронный ресурс]. – URL: https://www.kaggle.com/c/amazon-employee-access-challenge (дата обращения: 07.06.2017).

4. *Дьяконов А.Г.* Теория систем эквивалентностей для описания алгебраических замыканий обобщенной модели вычисления оценок // Журнал вычислительной математики и математической физики. – 2010. – Т. 50, № 2. – С. 388–400.

5. *Strang G.* Linear algebra and its applications. – 4th ed. – Belmont, CA: Thomson, Brooks/Cole, 2005.

6. *Martin C.D., Porter M.A.* The extraordinary SVD // American Mathematical Monthly. – 2012. – Vol. 119, N 10. – P. 838–851.

7. *Golub G.H., Van Loan C.F.* Matrix computations. – 3rd ed. – Baltimore, MD: Johns Hopkins University Press, 1996.

8. *Kolda T.G., Bader B.W.* Tensor decompositions and applications // SIAM Review. – 2009. – Vol. 51 (3). – P. 455–500.

9. LIBLINEAR – A Library for Large Linear Classification [Electronic resource]. – URL: http://www.csie.ntu.edu.tw/~cjlin/liblinear/ (accessed: 08.06.2017).

10. A dual coordinate descent method for large-scale linear SVM / C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S.S. Keerthi, S. Sundararajan // Proceedings of the 25th International Conference on Machine Learning: ICML 2008. – New York, NY: ACM, 2008. – P. 408–415.

11. *D'yakonov A.* A blending of simple algorithms for topical classification [Electronic resource] // Rough Sets and Current Trends in Computing: 8th International Conference, RSCTC 2012: Proceedings. – 2012. – Berlin; New York: Springer, 2012. – P. 432–438. – (Lecture Notes in Computer Science; vol. 7413). – URL: http://www.springerlink.com/content/73g4kl50m6112420 (accessed: 08.06.2017).

12. *Маннинг К.Д., Рагхаван П., Шютце Х.* Введение в информационный поиск. – М.: Вильямс, 2011. – 528 с.

13. *Журавлев Ю.И.* Об алгебраическом подходе к решению задач распознавания или классификации // Проблемы кибернетики. – М.: Наука, 1978. – Вып. 33. – С. 5–68.

14. *D'yakonov A.G.* Two recommendation algorithms based on deformed linear combinations // DCW-2011: ECML/PKDD Discovery Challenge 2011: proceedings of the ECML/PKDD Discovery Challenge Workshop. – [S. l.]: CEUR, 2011. – P. 21–28. – (CEUR workshop proceedings, vol. 770).

15. *Funk S.* Netflix update: try this at home [Electronic resource]. – URL: http://sifter.org/~simon/journal/20061211.html (accessed: 08.06.2017).

16. *Breiman L.* Random Forests // Machine Learning. – 2001. – Vol. 45 (1). – P. 5–32.

17. Библиотека scikit-learn для языка Python [Электронный ресурс]. – URL: https://github.com/scikit-learn/scikit-learn (дата обращения: 08.06.2017).

Малышев Виктор Владимирович, ведущий инженер Института природных ресурсов Национального исследовательского Томского политехнического университета. Основное направление научных исследований – методы машинного обучения. Имеет ряд публикаций. E-mail: mvv@hw.tpu.ru

Сливкин Станислав Сергеевич, инженер Института природных ресурсов Национального исследовательского Томского политехнического университета. Основное направление научных исследований – методы машинного обучения. Имеет ряд публикаций. E-mail: ss@hw.tpu.ru

Рукавишников Валерий Сергеевич, PhD(Geology), директор Центра подготовки и переподготовки специалистов нефтегазового дела Института природных ресурсов Национального исследовательского Томского политехнического университета. Основное направление научных исследований – 4D сейсмика. Имеет ряд публикаций. E-mail: RukavishnikovVS @hw.tpu.ru

Базаркин Евгений Васильевич, менеджер Центра подготовки и переподготовки специалистов нефтегазового дела Национального исследовательского Томского политехнического университета. E-mail: bazarkin@me.com

Application of machine learning methods for selecting master's program candidates[*]

V.V. MALYSHEV[1], *S.S. SLIVKIN*[2], *V.S. RUKAVISHNIKOV*[3], *E.V. BAZARKIN*[4]

[1] *National Research Tomsk Polytechnic University, 4a, Usov Street, Tomsk, 630034, Russian Federation, leading engineer. E-mail: mvv@hw.tpu.ru*
[2] *National Research Tomsk Polytechnic University, 4a, Usov Street, Tomsk, 630034, Russian Federation, engineer. E-mail: ss@hw.tpu.ru*
[3] *National Research Tomsk Polytechnic University, 4a, Usov Street, Tomsk, 630034, Russian Federation, PhD (Geol.). E-mail: RukavishnikovVS@hw.tpu.ru*
[4] *National Research Tomsk Polytechnic University, 4a, Usov Street, Tomsk, 630034, Russian Federation, engineer. E-mail: bazarkin@me.com*

The relevance of the research is caused by the importance of qualitative selection of students for master's programs. Today machine-learning methods are being actively implemented in many areas of science and technology. The aim of the research is the evaluation of the possibility of using machine learning at various stages of the selection process. A comparative analysis of classification methods is made and the order of implementation by the existing automated system for collecting, processing and accounting for personal data is determined. Research methods are aimed at constructing linear mathematical models. It is necessary to prepare initial data i.e. to normalize and convert categorical features. Categorical features are converted into numbers using binary coding. To assess the efficiency of classifiers, their model is taught on the training sample, and the quality of predictions is checked on the test sample. To exclude a chance of obtaining unbalanced samples, i.e. samples in which the number of objects of one class can be much larger than of the other, a cross-validation technique is used. At each step, the data is divided into parts, and then the model is trained on the training sample and validated on the test sample. To reduce the classifier error at each step the original data set is divided into several parts. A comparison of accuracy of several classifier types has been made and the most accurate method- a random forest- is determined. Further the order of implementing the chosen classifier in the existing automated system for collecting, processing and accounting for personal data is determined. Methods of data classification for building a module of recommendations for an automated system for collecting, processing and accounting for personal data are proposed. To improve the quality of candidate selection for further research activities, the technology of implementing chat bots has been chosen. Using this technology a psychological portrait of a candidate can be evaluated. Chat bots are a special kind of interactive interaction based on artificial intelligence technologies and natural language recognition.

Keywords: machine learning, classifier, mathematical model, cross-validation, binary coding, Python

DOI: 10.17212/1814-1196-2017-2-109-119

REFERENCES

1. Vorontsov K.V. *Mashinnoe obuchenie*: lektsii [Machine learning]. Available at: http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf (accessed 07.06.2017).

2. Koren Y., Bell R.M., Volinsky C. Matrix factorization techniques for recommender systems. *Computer*, 2009, vol. 42 (8), pp. 30–37.

3. International competition "Amazon.com – Employee Access Challenge" on data analysis. Available at: https://www.kaggle.com/c/amazon-employee-access-challenge (accessed 07.06.2017).

4. D'yakonov A.G. Teoriya sistem ekvivalentnostei dlya opisaniya algebraicheskikh zamykanii obobshchennoi modeli vychisleniya otsenok [The theory of equivalence systems for the description of

[*] *Received 24 April 2017.*

algebraic closures of a generalized model for computing estimates]. *Zhurnal vychislitel'noi matematiki i mate-maticheskoi fiziki – Computational Mathematics and Mathematical Physics*, 2010, vol. 50, no. 2. pp. 388–400. (In Russian).

5. Strang G. *Linear algebra and its applications*. 4th ed. Belmont, CA, Thomson, Brooks/Cole, 2005.

6. Martin C.D., Porter M.A. The extraordinary SVD. *American Mathematical Monthly*, 2012, vol. 119, no 10, pp. 838–851.

7. Golub G.H., Van Loan C.F. *Matrix computations*. 3rd ed. Baltimore, MD, Johns Hopkins University Press, 1996.

8. Kolda T.G., Bader B.W. Tensor decompositions and applications. *SIAM Review*, 2009, vol. 51 (3), pp. 455–500.

9. *LIBLINEAR – A Library for Large Linear Classification*. Available at: http://www.csie.ntu.edu.tw/~cjlin/liblinear/ (accessed 08.06.2017).

10. Hsieh C.-J., Chang K.-W., Lin C.-J., Keerthi S.S., Sundararajan S. A dual coordinate descent method for large-scale linear SVM. *Proceedings of the 25th International Conference on Machine Learning: ICML 2008*. New York, NY, ACM, 2008, pp. 408–415.

11. D'yakonov A. A blending of simple algorithms for topical classification. *Rough Sets and Current Trends in Computing: 8th International Conference, RSCTC 2012: Proceedings*. *Lecture Notes in Computer Science*, vol. 7413. Berlin, New York, Springer, 2012, pp. 432–438. Available at: http://www.springerlink.com/content/73g4kl50m6112420 (accessed 08.06.2017).

12. Manning C.D., Raghavan P., Schuütze H. *Introduction to information retrieval*. Cambridge, Cambridge University Press, 2008 (Russ. ed.: Manning K.D., Ragkhavan P., Shyuttse Kh. *Vvedenie v informatsionnyi poisk*. Moscow, Williams, 2011. 528 p.).

13. Zhuravlev Yu.I. Ob algebraicheskom podkhode k resheniyu zadach raspoznavaniya ili klassifikatsii [On the algebraic approach to solving the problems of recognition or classification]. *Problemy kibernetiki* [Problems of cybernetics]. Moscow, Nauka Publ., 1978, iss. 33, pp. 5–68.

14. D'yakonov A.G. Two recommendation algorithms based on deformed linear combinations. *DCW-2011: ECML/PKDD Discovery Challenge 2011: proceedings of the ECML/PKDD Discovery Challenge Workshop*. CEUR, 2011, pp. 21–28.

15. Funk S. *Netflix update: try this at home*. Available at: http://sifter.org/~simon/journal/20061211.html (accessed 08.06.2017).

16. Breiman L. *Random Forests*. *Machine Learning*, 2001, vol. 45 (1), pp. 5–32.

17. *Library scikit-learn for Python*. Available at: https://github.com/scikit-learn/scikit-learn (accessed 08.06.2017).