УДК 004.457 DOI: 10.24151/1561-5405-2020-25-4-374-382

Разработка программного модуля отбора функций признаков на основе генетического алгоритма

Е.Н. Петров

Национальный исследовательский университет «МИЭТ», г. Москва, Россия

fiddenmar@gmail.com

Современные алгоритмы машинного обучения с учителем используют признаковое описание объектов для создания классифицирующих моделей. Такое описание может включать в себя большое количество признаков в зависимости от решаемой задачи. В работе проведен анализ проблемной ситуации в рамках предметной области, связанной с составлением признакового описания объектов библиографических данных. Предложен способ решения данной проблемы за счет применения генетического алгоритма. Сформулированы принципы разработки программного модуля в общем виде и даны детали реализации на языке программирования Python. В результате решается проблема перегрузки признакового представления малозначимыми признаками, обучение и переобучение ускоряется без потери качества классификации. Генетический алгоритм разработанного программного модуля в составе программного комплекса обработки библиографических данных может применяться для отбора наиболее заначимых признаков. В ходе вычислительного эксперимента получены следующие результаты: число используемых признаков уменьшилось с 26 до 15, качество классификации увеличилось на 3 % за счет отсева признаков, способствующих переобучению.

Ключевые слова: генетический алгоритм; обработка данных; разработка программного обеспечения; язык программирования Python

Для цитирования: Петров Е.Н. Разработка программного модуля отбора функций признаков на основе генетического алгоритма // Изв. вузов. Электроника. 2020. Т. 25. № 4. С. 374–382. DOI: 10.24151/1561-5405-2020-25-4-374-382

GA-Based Feature Selection Software Module Development

E.N. Petrov

National Research University of Electronic Technology, Moscow, Russia

fiddenmar@gmail.com

Abstract: The nowadays supervised machine learning algorithms use the feature description to classify objects. Such a description may include a great number of features provided the task demands it. In the work the genetic algorithm based feature selection as a part of the software complex of bibliographic data processing has been described. The analysis of the problem situation within the framework of the subject area, related to the feature description size of the bibliographic data objects, has been carried out. A method of solving the given problem due to the genetic algorithm feature selection has been proposed. The paper includes the general principles of the software model and the implementation details in the Python programming language. The problem of feature description and re-learning in bibliographic data processing has been solved, it has been shown that learning and re-learning accelerates without loss of the classification quality. The developed software for genetic algorithm feature selection can be applied within the framework of the software complex for bibliographic data processing.The following results have been obtained during the computational experiment: the number of features used decreased from 26 to 15, and the quality of classification increased by 3 % due to the elimination of features that contribute to retraining.

Keywords: genetic algorithm; data processing; software development; Python programming language

For citation: Petrov E.N. GA-based feature selection software module development. *Proc. Univ. Electronics*, 2020, vol. 25, no. 4, pp. 374–382. DOI: 10.24151/1561-5405-2020-25-4-374-382

Введение. В настоящее время обработка библиографических данных с целью учета научных трудов в высших учебных заведениях и других организациях осуществляется операторами вручную с помощью неавтоматизированных программных средств, что увеличивает временные и трудовые затраты. При этом качество выполненной работы остается низким. В работах [1, 2] для решения данной задачи разработана методика классификации библиографических данных с помощью условно-случайных полей и проведена верификация этой методики. В настоящей работе разрабатывается дополнительный модуль отбора функций признаков на основе генетического алгоритма и приводится его программная реализация.

Постановка задачи. Современные методы машинного обучения можно условно разделить на обучение с учителем и без учителя [3]. Обучение с учителем позволяет создавать статистические модели классификации на основе прецедентов, записываемых в виде комбинации «стимул – реакция». В рамках разработки и программной реализации

методики классификации с помощью условно-случайных полей в качестве «стимула» используется признаковое описание объекта, а в качестве «реакции» – соответствующий ему класс.

Признаковое описание – в общем случае вектор, составленный из значений фиксированного набора признаков на данном объекте. Признаки могут иметь различные типы, причем необязательно числовые, и в разработанной методике имеют бинарный вид («1» свидетельствует о наличии признака у объекта, «0» – о его отсутствии). Размер вектора, включающего в себя признаковое описание, в зависимости от типа и сложности задачи может исчисляться десятками, сотнями или тысячами элементов. В то же время, хотя время обучения будет расти пропорционально числу признаков, их индивидуальная и синергетическая значимость могут быть минимальными, а значит, их исключение из признакового описания позволит ускорить процесс обучения с минимальными потерями качества классификации в рамках некоторой погрешности, особенно для переобучающихся систем.

Таким образом, задача отбора наиболее значимых признаков и составление признакового описания, являющегося компромиссом между эффективностью и временем обучения, актуальна. Разработка программного модуля отбора функций признаков на основе генетического алгоритма в составе программного комплекса обработки библиографических данных – частное решение этой задачи.

Генетический алгоритм. В основе решения поставленной задачи лежит ее формализация в виде, пригодном для использования генетического алгоритма.

Генетический алгоритм – это эвристический алгоритм поиска, применяемый для решения задач оптимизации и моделирования путем последовательного подбора, комбинирования и вариации искомых параметров с использованием механизмов, напоминающих биологическую эволюцию. Данный алгоритм является разновидностью эволюционных вычислений. Отличительная особенность генетического алгоритма от других заключается в использовании оператора скрещивания, осуществляющего операцию рекомбинации решений-кандидатов, роль которой аналогична роли скрещивания в живой природе [4].

Задача формализуется так, чтобы ее решение могло быть закодировано в виде вектора (генотипа) генов, где каждый ген может быть битом, числом или другим объектом. В контексте решения поставленной задачи таким генотипом служит набор генов-признаков в виде последовательности битов, где «1» свидетельствует о наличии признака у объекта, а «0» – о его отсутствии. Некоторым, обычно случайным, образом создается множество генотипов начальной популяции. Они оцениваются с использованием функции приспособленности [4], в результате чего с каждым генотипом ассоциируется определенное значение (приспособленность), которое определяет, насколько хорошо описываемый им фенотип решает поставленную задачу. В качестве такой функции приспособленности выбрана величина, обратная количеству элементов в признаковом описании. Таким образом, у особей с меньшим числом признаков значение функции приспособленности будет больше. Также введено дополнительное ограничение: макрозначение F-меры (невзвешенное среднее значение по всем классам) не может быть ниже определенного значения, зависящего от этого параметра у модели с полным набором функций признаков.

Создание первоначальной популяции осуществляется следующим образом: для каждой особи определяется случайное число признаков, которые могут иметь значение «1», после чего случайным образом определяются гены, которым это значение присваивается. Несмотря на то что получаемое таким образом первое поколение оказывается неконкурентоспособным, генетический алгоритм достаточно быстро приводит его в жизнеспособную популяцию.

На этапе отбора каждая особь сначала проверяется на соответствие дополнительному ограничению и удаляется из популяции, если ему не соответствует. Затем, если число оставшихся особей превышает половину популяции, часть особей удаляется с помощью турнирной селекции, т.е. выбираются две случайные особи, из них остается та, у которой приспособленность выше. Затем к популяции применяются скрещивание и мутация, и на их основе генерируется следующее поколение.

Скрещивание особей в рамках решения задачи проводится с помощью фенотипного аутбридинга – подхода, при котором первый родитель выбирается случайно, а второй выбирается таким, чтобы имел наименьшую похожесть на первого родителя, измеряемую в зависимости от значения функции приспособленности [5]. Функцией похожести в данном случае является модуль разности приспособленностей особей. В качестве родителей выбираются все особи текущей популяции вне зависимости от удовлетворения ограничения на минимальное макрозначение *F*-меры. Такой подход к генерации следующего поколения позволяет сохранить разнообразие особей от поколения к поколению и предотвратить появление доминантного генотипа.

Создание новой особи реализовано следующим образом. За основу берется дизъюнкция генотипов родителей, после чего из числа признаков, имеющих значение «1», сохраняется случайная выборка признаков в количестве, равном среднему числу используемых родительских признаков, а остальные зануляются. В результате становится возможным сохранение существующих взаимодействий признаков и появление новых связей в последующих поколениях при постепенном уменьшении общего количества используемых признаков. К полученной популяции применяются случайные мутации, позволяющие избежать замыканий генотипов в локальном максимуме функции приспособленности. Для каждой особи вероятность появления мутации составляет 5 % для каждого признака и выражается в инвертировании бита использования признака.

Рассмотренные действия повторяются итеративно, моделируя эволюционный процесс, продолжающийся до тех пор, пока не будет найдено субоптимальное решение – минимальное количество признаков, дающих макрозначение *F*-меры не менее заданного значения.

Разработка программного модуля. Логика отбора признаков с помощью генетического алгоритма вынесена в отдельный программный модуль в составе программного комплекса преобразования библиографических данных (рис.1).

Разработанный модуль взаимодействует с модулем обучения, оценки и визуализации, формируя итоговое признаковое описание классов для подготовки и использования модели классификации на основе представленного набора признаков. Это же признаковое описание использует модуль классификации и преобразования, запрашивая его у модуля обучения, оценки и визуализации.

При работе модуля на вход ему подаются обучающая и тестовая выборки библиографических данных и признаковое описание их классов. Результатом работы программного модуля является признаковое описание меньшего или того же размера, полученное в ходе отбора вычисляемых признаков с помощью генетического алгоритма и имеющее макрозначение *F*-меры, полученное на тестовой выборке, не менее определенного значения (рис.2.).

По умолчанию входные и выходные параметры передаются в рамках программного комплекса. Однако для большей гибкости и удобства использования разработанного функционала программный модуль также имеет самостоятельный режим работы с возможностью передачи и получения данных через интерфейс командной строки. Схема алгоритма отбора признаков библиографических данных представлена на рис.3.

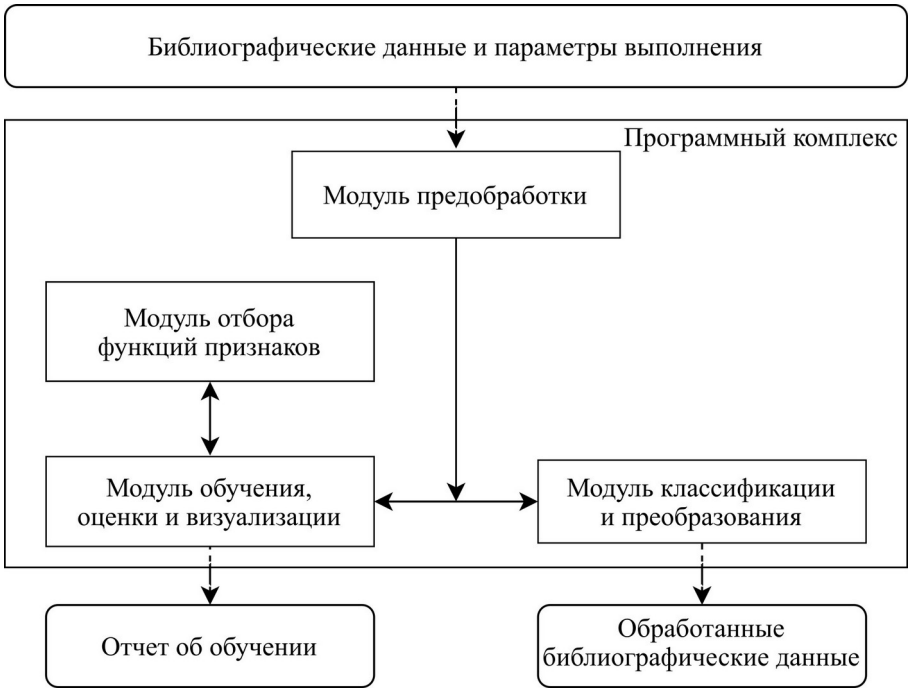

Рис.1. Структурная схема программного комплекса преобразования библиографических данных
Fig.1. Bibliographic data conversion software structure

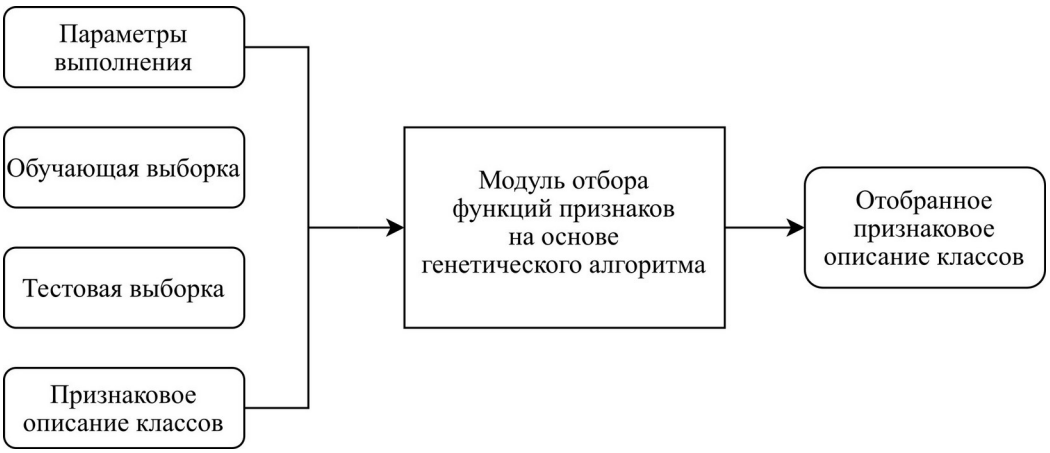

Рис.2. Схема работы программного модуля отбора функций признаков на основе генетического алгоритма
Fig.2. GA-based feature selection module scheme

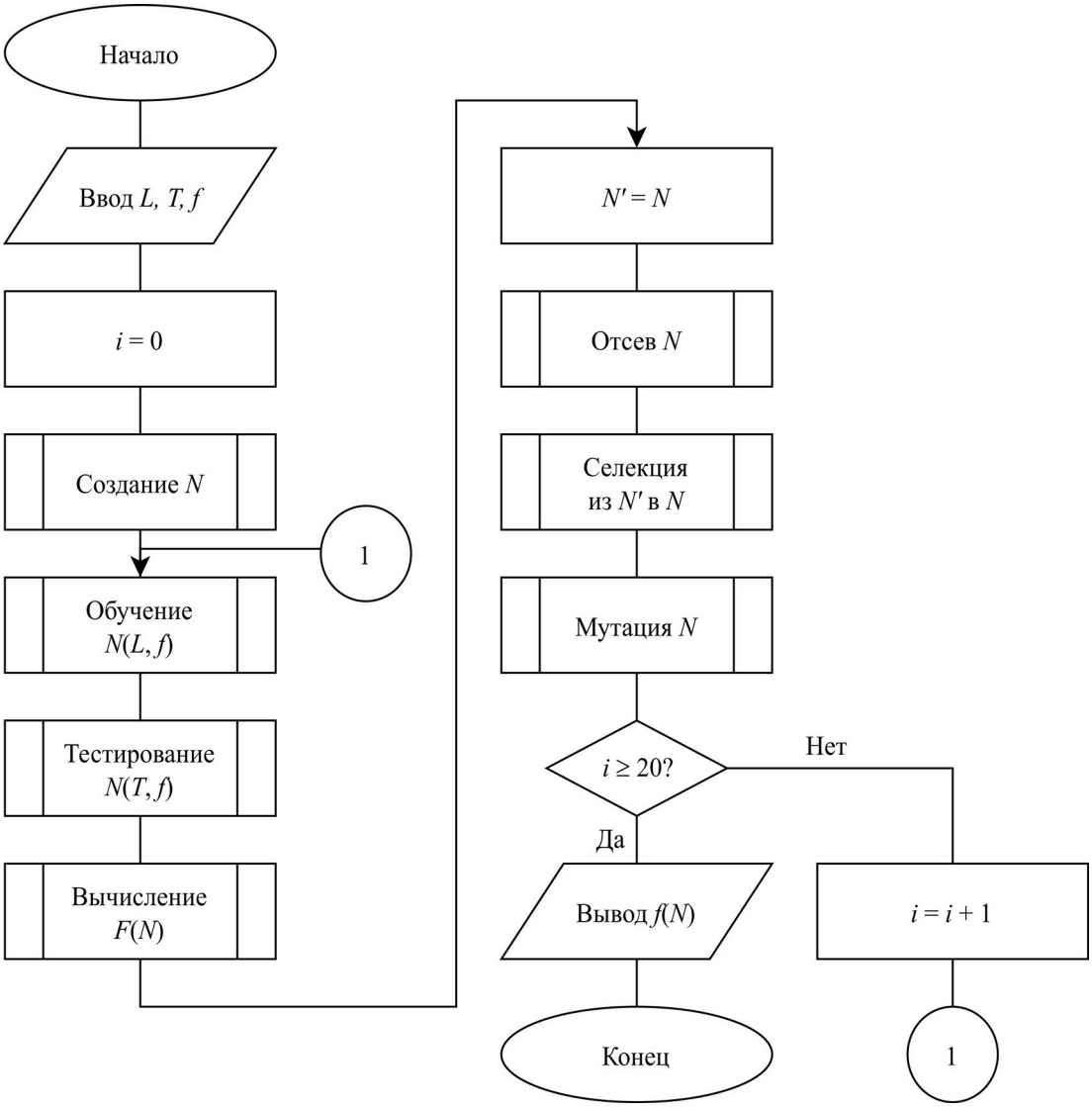

Рис.3. Схема алгоритма программного модуля отбора признаков на основе генетического алгоритма
Fig.3. GA-based feature selection module algorithm

В начале работы поступающие на вход обучающая выборка L и тестовая выборка T фиксируются на все время работы модуля. Также фиксируется исходный набор функций признаков f. Создается начальная популяция N, для каждой особи в популяции проводятся обучение и тестирование модели классификации библиографических данных на основе условно-случайных полей. Затем для каждой особи вычисляются макрозначение F-меры и функция приспособленности, после чего в новую популяцию переносятся особи, имеющие макрозначение F-меры не менее определенного значения. Если таких особей оказалось больше половины популяции, среди них проводится турнирная селекция. После этого остальные места в новой популяции занимают особи, полученные с помощью фенотипного аутбридинга старой популяции под воздействием мутации с вероятностью возникновения m %. Для новой популяции повторяется вся последовательность операций, начиная с обучения модели и заканчивая формированием следующего поколения. Алгоритм продолжает свою работу до тех пор, пока не будет достигнуто k-е поколение, после чего формирует и выводит результирующее признаковое описание, определяемое по наибольшему значению функции приспособленности.

Рис.4. Файловая структура программного модуля
Fig.4. Software module file structure

При реализации алгоритма выбраны следующие параметры генетического алгоритма: 20 особей в популяции, 20 поколений, вероятность мутации 5 %. Итоговое отсевное значение *F*-меры выбрано равным значению *F*-меры модели, обученной на полном наборе признаков. Для сохранения перспективных генотипов для каждого поколения сформировано свое отсевное значение, допускающее отклонение от итогового пропорционально числу прошедших итераций. Для сокращения времени обучения моделей за счет библиотечного модуля joblib обученные модели сохраняются в подпапке models (рис.4). В качестве имени файла используется хеш-значение генотипа, полученное с помощью функции hash.

Таким образом, становится возможным восстановление уже обученной модели из памяти в том случае, если особь с таким генотипом уже фигурировала в одном из поколений. С учетом того, что на каждом шаге эволюционного процесса до половины особей переносится в новое поколение, при дополнительном пересечении экономия вычислений может превышать 50 %. Генотип в данном случае представляет собой бинарную последовательность, поэтому для выбора имени можно использовать результат преобразования этого генотипа к целому числу:

model_dump_filename = int("".join(str(gen) for gen in genotype), 2)

Однако современные операционные системы поддерживают имена файлов размером до 255 символов, что соответствует преобразованию генотипа размера 844. Использование хеш-значения для генерации имени обусловлено независимостью его длины от размера генотипа и, как следствие, большей универсальностью. При этом общий размер популяции и число поколений делают возникновение коллизий маловероятным.

Результаты и их обсуждение. В ходе вычислительного эксперимента в качестве входных данных использовалась тестовая выборка размеченных библиографических записей, составленных на основе научных трудов сотрудников Института СПИНТех МИЭТ за последние пять лет на русском и английском языках общим объемом 2519 элементов. Входные данные делились на две части в соотношении 80 : 20 для обучения и проверки соответственно [6]. На этих данных проводились обучение и тестирование эталонной модели, использующей все признаки, а также генетической модели, использующей отобранные с помощью разработанного программного модуля признаки. *F*-мера эталонной модели составила 0,934. Развитие *F*-меры генетической модели представлено в табл.1.

Таким образом, качество классификации, выражающееся в макрозначении *F*-меры, увеличилось на 3 %. В то же время количество используемых признаков в признаковом описании сократилось на 40 % (с 26 до 15). Среди сокращенных признаков выделены следующие группы: признак начала предложения; признаки символьного состава для предыдущего и следующего слов; признаки регистра для предыдущего и следующего слов; признаки пунктуационных знаков для текущего и следующего слов. Наибольшее

число признаков сокращено в описании следующего слова, наименьшее – текущего. Сравнение поклассовых значений для *F*-меры представлено в табл.2. Качество классификации возросло для всех классов, кроме специальных разделителей.

Таблица 1

Развитие *F*-меры генетической модели

Table 1

Evolution of genetic model's *F*-score

Поколение	Наибольшие значения *F*-меры в поколении	Поколение	Наибольшие значения *F*-меры в поколении
1	0,896; 0,890; 0,889	11	0,940; 0,918; 0,912
2	0,896; 0,890; 0,889	12	0,940; 0,932; 0,918
3	0,896; 0,890; 0,889	13	0,943; 0,940; 0,932
4	0,896; 0,890; 0,889	14	0,943; 0,940; 0,932
5	0,896; 0,891; 0,890	15	0,943; 0,940; 0,932
6	0,896; 0,891; 0,890	16	0,968; 0,943; 0,940
7	0,897; 0,896; 0,895	17	0,968; 0,943; 0,940
8	0,897; 0,896; 0,895	18	0,968; 0,967; 0,946
9	0,940; 0,901; 0,897	19	0,968; 0,967; 0,946
10	0,940; 0,912; 0,901	20	0,968; 0,967; 0,946

Таблица 2

Поклассовое сравнение значений *F*-меры

Table 2

F-score comparison class by class

Класс библиографических данных	Эталонная модель	Генетическая модель
Авторы	0,975	1,000
Название статьи	0,995	1,000
Название журнала	0,907	0,958
Место издания	0,750	1,000
Страницы	0,929	0,929
Издательство	0,545	1,000
Номер издания	0,811	0,955
Год издания	0,865	0,882
Разделители	0,973	0,960

Заключение. Разработанный программный модуль отбора функций признаков на основе генетического алгоритма позволяет решать задачу отбора наиболее значимых признаков. Исключение малозначимых признаков в процессе отбора упрощает итоговую модель классификации, в результате чего снижается риск переобучения и повышается качество классификации. В ходе вычислительного эксперимента получены следующие результаты: число используемых признаков уменьшилось с 26 до 15, качество классификации увеличилось на 3 % за счет отсева признаков, способствующих переобучению.

Таким образом, проблема избыточности признакового описания в задаче классификации библиографических данных в рамках работы над программным комплексом преобразования библиографических данных решена.

Литература

1. ***Петров Е.Н.*** Исследование и разработка методики и алгоритма классификации библиографических данных с помощью условно-случайных полей // Итоги диссертационных исследований. Т. 2. Материалы X Всероссийского конкурса молодых ученых. М.: РАН, 2018. С. 91–98.

2. ***Петров Е.Н., Черников Б.В., Борисова Е.А.*** Верификация методики классификации библиографических данных на основе условно-случайных полей. Ч.1 // Современные наукоемкие технологии. 2019. № 11. С. 113–118.

3. ***Berry M.W., Mohamed A.Z., Yap B.W.*** Supervised and unsupervised learning for data science. Springer International Publishing, 2019. 187 p.

4. ***Вороновский Г.К., Махотило К.В., Петрашев С.Н., Сергеев С.А.*** Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности. Х.: Основа, 1997. 112 с.

5. An improved gene expression programming based on niche technology of outbreeding fusion / ***C.X. Wang, J.J. Zhang, S.L. Wu et al*** // Informatica. 2017. Vol. 41. P. 25–30.

6. ***Wang J., Li Z., Huang W., Xiao K.*** Character information extraction based on CRFsuite // 2016 International Conference on Advanced Electronic Science and Technology (AEST 2016). Atlantis Press, 2016. P. 147–154.

Поступила в редакцию 15.04.2020 г.; после доработки 20.05.2020 г.; принята к публикации 16.06.2020 г.

Петров Евгений Николаевич – аспирант Института системной и программной инженерии и информационных технологий Национального исследовательского университета «МИЭТ» (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, 1), fiddenmar@gmail.com

References

1. Petrov E.N. Research and development of CRF-based bibliographic data classification method. *Results of Thesis Researches*. *In Proceedings of X Russian Young Researches Competiotion.* Moscow, 2018, vol. 2, pp. 91–98. (in Russian).

2. Petrov E.N., Chernikov B.V., Petrov E.N., Borisova E.A. Software implementation and verification of CRF-based bibliographic data classification method. *Modern high technologies*, 2020, no. 11, part 1, pp. 113–118. (in Russian).

3. Berry M.W., Mohamed A.Z., Yap B.W. *Supervised and unsupervised learning for data science*. Springer International Publishing, 2019. 187 p.

4. Voronovsky G.K., Mahotilo K.V., Petrashev S.N., Sergeev S.A. et al. *Genetic algorithms, artificial neural networks and virtual reality problems*. X., Osnova Publ., 1997. 112 p. (in Russian).

5. Wang C.X., Zhang J.J., Wu S.L., Zhang F., Tromp J.G. An improved gene expression programming based on niche technology of outbreeding fusion. *Informatica,* 2017, vol. 41, pp. 25–30.

6. Wang J., Z. Li, W. Huang, K. Xiao Character information extraction based on CRFsuite. *2016 International Conference on Advanced Electronic Science and Technology (AEST 2016)*. Atlantis, 2016.

Received 15.04.2020; Revised 20.05.2020; Accepted 16.06.2020.

Information about the author:

Evgeny N. Petrov – PhD student of Institute of System and Software Engineering and Computer Science, National Research University of Electronic Technology (Russia, 124498, Moscow, Zelenograd, Shokin sq., 1), fiddenmar@gmail.com