



А.Г. Коробейников, И.Г. Сидоркина, С.Ю. Блинов, А.В. Лейман

АЛГОРИТМ КЛАССИФИКАЦИИ ИНФОРМАЦИИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ФИЛЬТРАЦИИ НЕЖЕЛАТЕЛЬНЫХ СООБЩЕНИЙ

Аннотация: В статье рассматривается задача фильтрации нежелательной информации или определения спамности документов на основе метода опорных векторов. Предложена модификация построения разделяющей гиперплоскости с использованием фейеровских отображений. В данном случае предложено произвести замену операции проектирования последовательностью таких отображений. Это позволяет работать с нестационарными данными, характерными для задач классификации документов.

Ключевые термины: Программное обеспечение, Классификация информации, спам, задачи сильной отделимости, гильбертово пространство, метод опорных векторов, гиперплоскость, фейеровские отображения, сходимость алгоритма, фильтрация

Введение

Информационные технологии породили все увеличивающийся поток разнородной информации. Основной задачей поисковых систем (поисковых машин) является предоставление качественных результатов, т.е. наиболее важных релевантных страниц. Для этого необходимо решать задачу классификации. Поэтому теория, методы и алгоритмы классификации информации являются бурно развивающимся научным направлением.

Одной из важнейших проблем, встающей практически перед каждым пользователем Интернет, является борьба со спамом, то есть задача фильтрации (классификации) поступающей информации.

В настоящее время разработан ряд технологий построения фильтров – сервисов для отсеивания нежелательной информации. Эти технологии можно разделить на настра-

иваемые вручную и автоматизированные. Настраиваемые вручную фильтры основываются на списках доступа и настраиваются непосредственно пользователем, который выбирает либо нежелательные адреса, при политике пропуска по «черному списку», либо разрешенные адреса, при политике пропуска по «белому списку». Однако ручные способы фильтрации нежелательных сообщений малоэффективны и требуют постоянного обновления списков доступа, создавая дополнительную нагрузку на пользователя. Кроме того, ручная категоризация неприменима, если необходимо классифицировать большой объем информации за ограниченное время.

Применение автоматизированных технологий фильтрации основано на использовании методов распознавания образов, искусственного интеллекта, применении математической статистики и т.д. [1, 2].

Одним из самых популярнейших на сегодняшний день является фильтр, основанный на байесовском подходе (наивный байесовский классификатор), в котором предполагается, что различные термы сообщения независимы друг от друга. Но для повышения эффективности таких фильтров необходимо учитывать семантические связи между термами, что требует привлечения методов семантического анализа, что существенно повышает нагрузку на систему и увеличивает время работы самого фильтра, при незначительном повышении эффективности фильтрации.

Таким образом, исходя из вышесказанного, существует потребность в разработке новых методов и алгоритмов классификации информации для решения задачи фильтрации нежелательных сообщений, что делает актуальной тему данной статьи.

В работе предложен алгоритм построения классификатора на базе метода построения разделяющей гиперплоскости в гильбертовом пространстве на основе фейеровского отображения [3,4].

Постановка задачи классификации

Формально постановка задачи классификации выглядит следующим образом.

Пусть дано конечное множество категорий (классов) $C = \{c_1, c_2, \dots, c_{|C|}\}$ и конечное множество документов $D = \{d_1, d_2, \dots, d_{|D|}\}$. Целевая функция (функционал, классификатор) $\Phi: D \times C \rightarrow \{-1, 1\}$, определяющая для каждой пары <документ, категория> соответствие их друг другу, не известна. Требуется найти классификатор Φ' , т.е. функцию, максимально близкую к функции Φ [4]. Если пересечение двух категорий пусто, то классификация бинарная, которая часто используется в фильтрации спама.

Если имеются образцы из каждой категории (объекты), про которые заранее известно, к какой категории они принадлежат, то такие задачи называют обучением с учителем, а из-

вестные данные называются обучающей выборкой. В предлагаемом методе используется именно этот подход.

Машинное обучение предполагает наличие обучающей и контрольной выборки, т.е. дана начальная коллекция документов $\Omega = \{d_1, d_2, \dots, d_{|\Omega|}\} \subset D$, где значения целевой функции Φ известны для $\forall (d_i, c_j) \in \Omega \times C$. Эта коллекция Ω разбивается на два непересекающихся множества: $\Omega = \Omega_1 \cup \Omega_2$ и $\Omega_1 \cap \Omega_2 = \emptyset$. (Ω_1 – спам, Ω_2 – не спам). Формирование этих подмножеств может происходить вручную, то есть эксперты, после работы с множеством документов мощности $|\Omega|$ говорят, что если документ d_i является спамом, то тогда он принадлежит множеству $d_i \in \Omega_1$. И наоборот, если документ не спам, то $d_i \in \Omega_2$.

Обозначим обучающее множество как $Q = \{(d_i, y_i)\}$, где $d_i \in \Omega$, $y_i \in C$. Классификатор Φ обучается индуктивно на основе выявленных характеристик документов [5].

Анализ метода построение классификатора

Построение классификатора будем проводить на основе метод опорных векторов (SVM – support vector machines) [6]. SVM – это набор схожих алгоритмов на основе обучения с учителем, применяющийся для анализа данных и распознавания образов в задачах классификации и регрессионном анализе. SVM является линейным классификатором. На основе обучающей выборки алгоритм помогает предсказать, в какую из двух заранее заданных категорий попадает элемент, подлежащий классификации.

Особым свойством SVM является непрерывное уменьшение эмпирической ошибки классификации и увеличение зазора. Поэтому этот метод также известен как метод классификатора с максимальным зазором. SMV также считается очень перспективным для

решения задач классификации (кластеризации) с использованием методов теории искусственного интеллекта.

Основная идея, используемая в SVM, это построение гиперплоскости или набора гиперплоскостей в пространстве более высокой размерности и максимизация расстояния между построенной гиперплоскостью и классами обучающей выборки. Но в настоящее время еще не разработаны общие методы построения таких гиперплоскостей (спрямляющих пространств или ядер классификатора), наиболее подходящих для конкретной задачи. Построение адекватного ядра является искусством и, как правило, опирается на априорные знания о предметной области. На практике «вполне разумные» ядра классификатора, выведенные из содержательных соображений, далеко не всегда оказываются положительно определёнными, что, естественно, не может не сказаться на качестве решения. Именно построению такой гиперплоскости, посвящена данная статья.

Разработка алгоритма решения задачи защиты от спама на базе SVM

В задаче классификации информации для задачи защиты от спама необходимо определять, является данный документ спамом, или нет. В предлагаемом подходе для решения этой задачи вводится система метрик. Пользователь разделяет свои письма и отмечает те, которые считает спамом, то есть строит обучающее множество Q . На основании этой информации строятся выпуклые оболочки в виде систем линейных неравенств. Первая система задает множество точек-документов, определяемых как спам. Вторая система – множество точек-документов, определяемых как не спам. Построив слой наибольшей толщины, разделяющий два многогранника, можно разработать

процедуру, позволяющую автоматически разделять (классифицировать) документы на «хорошие» и «плохие». Получив новый документ и прочитав его характеристики, получаем точку в рассматриваемом пространстве. Если данная точка попадает в «плохое» полупространство, мы делаем предположение, что это спам; если в «хорошее» – не спам. Если точка попадает внутрь слоя, письмо доставляется пользователю с пометкой «возможно, спам».

Для решения задачи сильной отделимости обычно применяют итерационный процесс, использующий операцию проектирования. Однако на практике применение этого метода существенно ограничено тем фактом, что далеко не всегда удастся построить конструктивную формулу для вычисления проекции точки на выпуклое множество. Поэтому целесообразно произвести замену операции проектирования последовательностью фейеровских отображений [7].

Кроме того, алгоритмы разделения многогранников на основе фейеровских отображений обладают тем преимуществом по сравнению с проективными и другими известными методами, что они применимы к нестационарным задачам, т.е. к задачам, в которых исходные данные могут меняться в процессе решения задачи. Такой нестационарной задачей являются, например, задача о спам-фильтре.

А теперь рассмотрим алгоритмы для решения задачи сильной отделимости.

Пусть даны два выпуклых непересекающихся многогранника $M \subset R^n$ и $N \subset R^n$, заданные системами линейных неравенств:

$$M = \{x | Ax \leq b\} \neq \emptyset; N = \{x | Bx \leq d\} \neq \emptyset. \quad (1)$$

Задача сильной отделимости заключается в нахождении слоя наибольшей толщины, разделяющего M и N . Эта задача эквивалентна задаче отыскания расстояния между M и N в смысле метрики:

$$\rho(M, N) = \min\{\|x - y\| \mid x \in M, y \in N\}. \quad (2)$$

Если $x_1 \in M$ и $y_1 \in N$ являются arg – точками задачи (2), то есть $\rho(M, N) = \|x_1 - y_1\|$, то слоем наибольшей толщины, разделяющим множества M и N является:

$$P = \{x \mid x \in P_1 \cap P_2\},$$

где $\langle \cdot, \cdot \rangle$ – скалярное произведение двух векторов

P_1 и P_2 – полупространства, задаваемые линейными неравенствами:

$$\langle x - x_1, x_1 - y_1 \rangle \leq 0 \text{ и } \langle y - y_1, x_1 - y_1 \rangle \geq 0.$$

Следовательно, задачу сильной отделимости можно сформулировать так:

$$\{x_1, y_1\} = \text{Arg min}\{\|x - y\| \mid x \in M, y \in N\}. \quad (3)$$

Задачу (3) можно решить при помощи известного алгоритма последовательного проектирования.

Алгоритм решения задачи сильной отделимости (\mathcal{H}).

Даны два выпуклых непересекающихся многогранника $M \subset R^n$ и $N \subset R^n$, заданные системами линейных неравенств (1).

Обозначим отображение (проектирование) точки на M через π_M , а на N – π_N . Зададим произвольное начальное приближение $w_0 \in R^n$. Выберем фиксированное положительное вещественное число ε . Тогда алгоритм решения задачи сильной отделимости будет состоять из следующих шести шагов.

Шаг 0. $k := 0$.

Шаг 1. $x_{k+1} := \pi_M(w_k)$.

Шаг 2. $y_{k+1} := \pi_N(w_k)$.

Шаг 3. $w_{k+1} := (x_{k+1} + y_{k+1})/2$.

Шаг 4. $k := k + 1$.

Шаг 5. Если $\min\{\|x_{k+1} - x_k\|, \|y_{k+1} - y_k\|\} \geq \varepsilon$, то перейти к Шагу 1.

Шаг 6. Конец.

Если множества M и N достаточно просты в смысле реализации операции проектирования точек на них, то представленный алгоритм \mathcal{H} может быть использован на практике. Но если M и N – произвольные многогранники, то \mathcal{H} не может быть признан эффективным, так как не известен универсальный конструктивный

метод построения проекции точки на многогранник. В этом случае решить задачу можно, если вместо операции проектирования использовать фейеровские отображения.

Дадим определение фейеровского отображения.

Пусть $\varphi : R^n \rightarrow R^n$. Отображение φ называется M -фейеровским, если выполняются следующие два условия:

$$\varphi(y) = y, \forall y \in M; \|\varphi(x) - y\| < \|x - y\|, \forall x \notin M, y \in M.$$

Класс M -фейеровских отображений обозначим через F_M .

Под фейеровским процессом, порождаемым однозначным M -фейеровским отображением $\varphi \in F_M$ при произвольном начальном приближении $x_0 \in R^n$, будем понимать последовательность $\{\phi^k(x_0)\}_{k=0}^{+\infty}$.

Известно [8], что в случае, когда однозначное M -фейеровское отображение φ является непрерывным, фейеровский процесс сходится к точке, принадлежащей множеству:

$$M: \{\phi^k(x_0)\}_{k=0}^{+\infty} \rightarrow \bar{x} \in M.$$

Это означает, что для любого вещественного $\varepsilon > 0$ существует целое положительное число K , такое, что для всех $k > K$ имеем .

Спроектируем M -фейеровское отображение, согласно [8]. Представим систему линейных неравенств, задающих многогранник M в виде:

$$M = \{x \mid Ax \leq b : l_j(x) = \langle a_j, x \rangle - b \leq 0, j=1, \dots, m\},$$

где $a_j, x^T 0$ для любого j . Зададим $l_j^{plus}(x)$ следующим образом:

$$l_j^{plus}(x) = \max\{l_j(x), 0\}, j = 1, \dots, m.$$

Тогда отображение вида:

$$\phi(x) = x - \sum_{j=1}^m (a_j)^2 \lambda_j \frac{l_j^{plus}(x)}{\|a_j\|^2}$$

будет M -фейеровским для любой системы положительных коэффициентов $\{a_j > 0\}$, $j = 1, \dots, m$, таких, что $\sum_{j=1}^m a_j = 1$ и коэффициентов релаксации $0 < l_j < 2$.

Аналогичным образом сконструируем N -фейеровское отображение y . Используя отображения φ и y можно построить алгоритм, решающий задачу сильной отделимости с использованием фейеровских отображений.

Определение. Пусть задано однозначное непрерывное отображение $\varphi \in \mathbf{F}_M$. Под φ – проектированием (псевдопроектированием) точки $x \in R^n$ на множество M будем понимать отображение $\pi_M^\varphi : R^n \rightarrow M$, задаваемое соотношением $\pi_M^\varphi(x) = \lim_{k \rightarrow \infty} \varphi^k(x)$. Точку $\pi_M^\varphi(x)$ будем называть псевдопроекцией точки x на множество M .

Предположим, что в контексте решения задачи (3) существуют два однозначных непрерывных фейеровских отображения $\varphi \in \mathbf{F}_M$ и $y \in \mathbf{F}_N$. Используя операции φ и y – проектирования, построим алгоритм \dot{A} , решающий задачу сильной отделимости с использованием фейеровских отображений.

Алгоритм \dot{A} . Зададим произвольное начальное приближение $w_0 \in R^n$. Выберем фиксированное положительное вещественное число ε . Тогда алгоритм решения задачи сильной отделимости с использованием фейеровских отображений будет состоять из следующих шести шагов.

Шаг 0. $k := 0$.

Шаг 1. $x_{k+1} := \pi_M^\varphi(w_k)$.

Шаг 2. $y_{k+1} := \pi_N^y(w_k)$.

Шаг 3. $w_{k+1} := (x_{k+1} + y_{k+1})/2$.

Шаг 4. $k := k + 1$.

Шаг 5. Если $\|w_{k+1} - w_k\| \geq \varepsilon$, то перейти к Шагу 1.

Шаг 6. Конец.

Основной вопрос при обосновании применимости алгоритма \dot{A} к задаче классификации информации заключается в доказательстве сходимости этого алгоритма к требуемому решению.

Обычно, для обоснования сходимости алгоритма \dot{A} необходимо более сильное свойство, чем просто фейеровость отображения φ , а именно локально – сильное фейеровское

отображение [9]. В этом случае сходимость алгоритма строго доказана [9].

Очевидно, что в алгоритме \dot{A} ресурсоемкими являются Шаг 1 и Шаг 2. На каждом из этих шагов реализуется *последовательный* фейеровский процесс, в результате которого получаем *псевдопроекцию* точки на многогранник. Многогранник, задаваемый системой линейных неравенств, всегда является выпуклым замкнутым множеством, то есть всегда существует решение.

А теперь рассмотрим алгоритм создания обучающей выборки.

Будем производить действия по следующему алгоритму:

1. Берем обучающее множество Ω и считываем его термы. Получаем множество термов:

$$T = \{t_1, t_2, \dots, t_p\}, p = 1, \dots, |T|, T = \cup d_i, i = 1, \dots, |D|.$$

2. Проводим лексикографическое упорядочивание множества, то есть преобразуем T в T_{base} т.е. $T \rightarrow T_{base}, |T_{base}| \leq |T|, L_{base} = |T_{base}|$.

3. На базе обучающего множества Ω формируется частотный словарь слов (термов) Datatable, в котором каждому $t_i \in T_{base}, i = 1, \dots, |L_{base}|$ соответствуют числа: ν_{1i} – частота встречаемости в Ω_1 и ν_{2i} – частота встречаемости в Ω_2 , $Datatable = \{t_1, \nu_{11}, \nu_{12}, t_2, \nu_{21}, \nu_{22}, \dots, t_r, \nu_{r1}, \nu_{r2}\}, r = L_{base}$.

4. Берем последовательно все документы $d_i \in \Omega_1, i = 1, \dots, |\Omega_1|$.

5. Считываем все термы из документа d_i : $TT = \{tt_1, \dots, tt_k\}, k = |d_i|$.

6. Проводим лексикографическое упорядочивание множества TT , то есть преобразуем TT в TT_{base} т.е. $TT \rightarrow TT_{base}, |TT_{base}| \leq |TT|, l_{base} = |TT_{base}|$.

7. На базе множеств T_{base}, TT_{base} , частотного словаря и вектора признаков формируем вектор:

$$x_i = \{\nu_{11}, \nu_{12}, \nu_{21}, \nu_{22}, \dots, \nu_{p1}, \nu_{p2}, sv_1, \dots, sv_n\},$$

$$\text{где } p = L_{base}, \text{ и } \nu_{ij} = 0, \text{ если } tt_i \in TT \cap T_{base} = \emptyset.$$

Обозначим множество векторов x_i через $X = \{x_i\}, i = 1, \dots, |\Omega_1|$. Получили множество, соответствующих спаму.

8. Повторяем шаги 4 – 7, но только для множества Ω_2 . Обозначим множество, полученное на шаге 8 через Y . Получили множество, соответствующее не спаму.

9. Строим по алгоритму \dot{A} разделяющую гиперплоскость между X и Y . Получили нормальный вектор к гиперплоскости w и параметр b (порог классификации).

10. Строим классифицирующую функцию Φ .

Окончательно алгоритм определения спамности документа выглядит так.

1. Берем документ $d_i \in D \setminus \Omega$, $i=1, \dots, |D \setminus \Omega|$.

2. Считываем все термы из документа d_i : $TT = \{tt_1, \dots, tt_k\}$, $k = |d_i|$.

3. Проводим лексикографическое упорядочивание множества TT , то есть преобразуем TT в $TT_{base} : TT \rightarrow TT_{base}$, $|TT_{base}| \leq |TT|$, $l_{base} = |TT_{base}|$.

4. Определяем множество признаков для данного документа. Если данный признак не определен, то присваиваем $sv_i = 0$.

5. На базе множеств T_{base} , TT_{base} , частотного словаря и множества признаков формируем вектор $x = \{v_{11}, v_{12}, v_{21}, v_{22}, \dots, v_{p1}, v_{p2}, sv_1, \dots, sv_n\}$,

где $p = l_{base}$, $v_{ij} = 0$, если $tt_i \in TT \cap T_{base} = \emptyset$.

6. Определяется спамность документа d_i при помощи классификатора:

$$\Phi(x) = \text{sign}(\langle w, x \rangle + b).$$

Заключение

Разработанный алгоритм определения спамности документа, или классификации, конечно же имеет свои недостатки. Попытки построить для решения задачи классификации один алгоритм, удовлетворяющих всех, заранее обречены на провал. Поэтому имеет смысл объединить несколько алгоритмов в композицию, в надежде на то, что погрешности различных алгоритмов взаимно компенсируются. Но в этом случае возникает масса вопросов. Например, при каких условиях качество композиции окажется лучше, чем у отдельных базовых алгоритмов? Как настраивать базовые

алгоритмы, учитывая, что они будут работать в составе композиции? Возможно ли приспособить для их настройки стандартные методы обучения? Как обойтись минимальным числом базовых алгоритмов?

Формально, эти вопросы можно записать следующим образом.

Пусть имеется задача обучения по прецедентам $\{X, Y, y^*, X', Y'\}$,

где X – множество объектов; Y – множество ответов; $y^* : X \rightarrow Y$ – отображение (неизвестная целевая зависимость); $X' = (x_p, \dots, x_p)$ – обучающая выборка; $Y' = (y^1, \dots, y^1)$ – вектор ответов на обучающих объектах, $y_i = y^*(x_i)$.

Требуется построить алгоритм $alg : X \rightarrow Y$, аппроксимирующий целевую зависимость y^* на всём множестве X .

Введем множество R , называемое пространством оценок [9]. Рассмотрим алгоритмы, имеющие вид суперпозиции:

$$alg(x) = \Pi(b(x)),$$

где $b : X \rightarrow R$ алгоритмический оператор, $\Pi : R \rightarrow Y$ – решающее правило.

Многие алгоритмы классификации имеют именно такую структуру: сначала вычисляются оценки принадлежности объекта категориям (классам), а затем решающее правило переводит эти оценки в номер класса. Значение оценки, как правило, характеризует степень уверенности классификации. В одних алгоритмах это вероятность принадлежности объекта данному классу, в других – расстояние от объекта до разделяющей поверхности. Возможны и другие интерпретации оценок. В статье был использован вариант:

$$Y = \{-1, +1\}, \Pi(z) = \text{sign}(z), alg(x) = \Phi(x) = \text{sign}(\langle w, x \rangle + b).$$

Так вот, разработкой теоретических положений построения

$$alg(x) = \Pi(b(x))$$

и будут посвящены дальнейшие научные исследования, которые позволят повысить эффективность фильтрации спама.

Библиография:

1. Сидоркина И.Г., Коробейников А.Г, Кудрин П.А. Алгоритм распознавания трехмерных изображений с высокой детализацией// Вестник МарГТУ, 2 (9), 2010 г., стр. 91-99.
2. Гришенцев А.Ю., Коробейников А.Г. Декомпозиция N-мерных цифровых сигналов по базису прямоугольных всплесков// Научно-технический вестник информационных технологий, механики и оптики. – 2012. – № 4 (80). – С. 75–79.
3. Халмош П., Гильбертово пространство в задачах, Перевод с английского И. Д. Новикова и Т. В. Соколовской; под ред. Р. А. Минлоса. — М.: Издательство «Мир», 1970. — 352 с.
4. Еремин И.И. Фейеровские методы сильной отделимости выпуклых полиэдральных множеств // Известия вузов. Сер. математика. -2006. -№ 12. -С. 33-43.
5. Лифшиц Ю. Классификация текстов. Алгоритмы для Интернета. 2005. URL: <http://yury.name/internet>.
6. R.-E. Fan, P.-H. Chen, C.-J. Lin. Working set selection using second order information for training SVM // Journal of Machine Learning Research, V. 6, 2005, pp. 1889–1918.
7. Еремин И.И., Мазуров В.Д. Нестационарные процессы математического программирования. -М.: Наука, 1979. -288 с.
8. Еремин И.И. Теория линейной оптимизации. -Екатеринбург: УрО РАН, 1999. -312 с.
9. Нурминский Е.А. Использование дополнительных малых воздействий в фейеровских моделях итеративных алгоритмов//Ж. вычисл. матем. и матем. физ. 2008. Т. 48. № 12. С. 2121–2128.
10. Журавлёв Ю. И. Об алгебраическом подходе к решению задач распознавания или классификации // Проблемы кибернетики. – 1978. – Т. 33. – С. 5–68. <http://www.ccas.ru/frc/papers/zhuravlev78prob33.pdf>.

References (transliteration):

1. Cidorkina I.G., Korobeynikov A.G, Kudrin P.A. Algoritm raspoznavaniya trekhmernykh izobrazheniy s vysokoy detalizatsiey// Vestnik MarGTU, 2 (9), 2010 g., str. 91-99.
2. Grishentsev A.Yu., Korobeynikov A.G. Dekompozitsiya N-mernykh tsifrovyykh signalov po bazisu pryamougol'nykh vspleskov//Nauchno-tekhnicheskii vestnik informatsionnykh tekhnologiy, mekhaniki i optiki. – 2012. – № 4 (80). – S. 75–79.
3. Khalmosh P., Gil'bertovo prostranstvo v zadachakh, Perevod s angliyskogo I. D. Novikova i T. V. Sokolovskoy; pod red. R. A. Minlosa. — M.: Izdatel'stvo «Mir», 1970. — 352 s.
4. Eremin I.I. Feyeroovskie metody sil'noy otdelimosti vypuklykh poliedral'nykh mnozhestv // Izvestiya vuzov. Ser. matematika. -2006. -№ 12. -С. 33-43.
5. Lifshits Yu. Klassifikatsiya tekstov. Algoritmy dlya Interneta. 2005. URL: <http://yury.name/internet>.
6. R.-E. Fan, P.-H. Chen, C.-J. Lin. Working set selection using second order information for training SVM // Journal of Machine Learning Research, V. 6, 2005, pp. 1889–1918.
7. Eremin I.I., Mazurov V.D. Nestatsionarnye protsessy matematicheskogo programirovaniya. -M.: Nauka, 1979. -288 s.
8. Eremin I.I. Teoriya lineynoy optimizatsii. -Ekaterinburg: UrO RAN, 1999. -312 s.
9. Nurminskiy E.A. Ispol'zovanie dopolnitel'nykh malykh vozdeystviy v feyeroovskikh modelyakh itera-tivnykh algoritmov//Zh. vychisl. matem. i matem. fiz. 2008. T. 48. № 12. S. 2121–2128.
10. Zhuravlev Yu. I. Ob algebraicheskom podkhode k resheniyu zadach raspoznavaniya ili klassifikatsii // Problemy kibernetiki. – 1978. – T. 33. – S. 5–68. <http://www.ccas.ru/frc/papers/zhuravlev78prob33.pdf>.