Московский физико-технический институт
(Государственный университет)

Факультет управления и прикладной математики
Кафедра «Интеллектуальные системы»

ДИПЛОМНАЯ РАБОТА СТУДЕНТКИ 974 ГРУППЫ

«Методы структурного обучения для построения прогностических моделей»

Выполнила:
студентка 4 курса 974 группы
Варфоломеева Анна Андреевна

Научный руководитель:
к.ф-м.н., н.с. ВЦ РАН
Стрижов Вадим Викторович

Москва, 2013

Содержание

Аннотация

В работе исследуется задача прогноза существенно нелинейной модели методами структурной регрессии. Предлагается алгоритм, определяющий структуру оптимальной параметрической модели заданной сложности. Приводится проверка работы метода на синтетических данных. Решается прикладная задача сегментации структурированных текстов: для каждого сегмента библиографической записи определяется его тип поля в формате BibTEX. Также для каждой записи определяется тип ее библиографического описания.

Ключевые слова: *структурное обучение, нелинейные модели, индуктивное порождение, разметка текстов, сегментирование.*

Введение

Актуальность темы. Во многих прикладных задачах требуется решить задачу восстановления модели функциональной зависимости, содержащейся в данных. Алгоритмы выбора моделей имеют значительную вычислительную сложность в связи с необходимостью перебора большого числа моделей.

Цель работы. Предложить метод прогнозирования структуры суперпозиции регрессионной модели, описывающей предъявленную выборку оптимальным образом, опираясь на известные прецеденты выбора модели.

Методы исследований. При построении алгоритма использованы методы регрессионного анализа, структурного обучения, кластеризации.

Научная новизна.

- Предложена новая постановка задачи прогнозирования структуры модели;
- предложен алгоритм нахождения оптимальной структуры суперпозиции функции;
- предложен метод поиска оптимального дерева в матрице вероятностей переходов;
- разработан алгоритм разметки библиографических списков.

Практическая ценность. Разработан программный модуль, который прогнозирует структуру модели зашумленных данных, оценивает вероятности наличия в модели элементарных функций, визуализирует результаты. Также разработана программа, осуществляющая разметку библиографических записей согласно структуре BibTEX, и определяющая тип записи в данной структуре.

Положения, выносимые на защиту:

- метод структурного обучения суперпозиции модели;
- метод поиска оптимального дерева суперпозиции;
- алгоритм разметки библиографических списков.

Обзор литературы. Одним из методов для решения задачи восстановления функциональной зависимости по набору исходных данных является символьная регрессия [1]. Джон Коза предложил реализацию этого метода с помощью аналога эволюционного алгоритма [2]. Иван Зелинка предложил дальнейшее развитие этой идеи [3], получившее название аналитического программирования.

Алгоритм построения математической модели в аналитическом программировании выглядит следующим образом: задан набор элементарных функций (например, степенная функция, $+$, sin, tan и др.), из которых можно строить различные формулы. Начальный набор формул строится либо произвольным образом, либо на базе некоторых предположений эксперта. Затем на каждом шаге производится оценка каждой из формул согласно некоторой функции качества. На базе этой оценки у части формул случайным образом заменяется одна элементарная функция на другую (например, sin на cos или $+$ на $\times$), а у некоторой другой части происходит взаимный попарный обмен подвыражениями. Данный подход может быть описан в терминах эволюционного алгоритма: каждый индивид является формулой, изображенной в свою очередь в виде дерева. Тогда набор формул, существующий в определенный момент, представляет собой одно поколение. При этом хромосомы представляются поддеревьями, и, в отличие от классического генетического алгоритма, могут быть различного размера (длины). Описанный выше обмен подвыражениями представляет собой в этом случае генетическое скрещивание, замена одной элементарной функции у некоторых деревьев — мутацию. При этом возникает ряд сложностей, связанных с областями определения и арностями элементарных функций, записанных в узлах дерева. Данный метод фактически является ненаправленным поиском и перебирает большое количество неподходящих деревьев до того момента, как приблизится к оптимуму. Модификация этого метода предложена в работах [4, 5], где с помощью проверки на наличие тождественных деревьев знаительно сокращался объем перебора.

Альтернативой аналитическому программированию можно считать подход обучения в глубину (Deep Learning) [6, 7]. Этот подход заключается в иерархическом представлении данных, в котором на нижнем уровне находятся сам набор данных, а на каждом уровне выше - более абстрактное его представление, которое представляет собой некую скрытую комбинацию из данных, указанных ниже. Так, например, при использовании данного метода в обработке изображений, набором данных является матрица яркости пикселей некоторого изображения, на следующем уровне — данные о выраженных геометрических закономерностях на изображении (отрезки, кривые, окружности), на более высоких уровнях иерархии — более сложные и абстрактные выявленные закономерности. В одном из основных алгоритмов, использующих данный подход, иерархия строится при помощи нейронной сети с несколькими скрытыми слоями [8]. В одном из основных методов обучения в глубину нейронная сеть обу-

чается, получая на вход и на выход одинаковый набор данных, после чего каждый из уровней сети представляется как информация о данных на определенном уровне абстракции.

В данной работе предлагается рассмотреть метод построения математической модели, основанный на прогнозировании структуры функциональной зависимости. Предполагается, что функциональная зависимость существенно нелинейна и, аналогично описанному выше, является суперпозицией элементарных функций. При этом делается ограничение на максимальную сложность модели. Дерево суперпозиции представляется в виде матрицы. В таком виде задача сводится к задаче структурного обучения, описанной, например, в [9, 10, 16]. Методы структурного обучения решают задачу нахождения структуры или зависимости, имеющейся внутри исходных данных. Метод широко применим для синтаксического разбора предложений [11], компьютерного зрения [12].

Методы, предложенные ранее для решения задачи сегментирования текстов, описаны в [13], где граф цитирований строится с помощью полученной разметки библиографического списка. В [14] описана постановка и решение задачи разметки адресной строки. Используется скрытая марковская цепь, недостатком которой является неточное описание структурных зависимостей внутри исходных данных. Автоматическая разметка библиографических записей представлена в [15], для обучения модели которой использовалась библиотека одного из стандартов оформления библиографических записей. В силу этого модель неустойчиво работает на других стандартах.

1 Постановка задачи

1.1 Описание данных и постановка задачи

Задан набор $\mathfrak{D} = \{(\mathbf{D}_k, f_k)\}$, состоящий из регрессионных выборок $\mathbf{D}$. Каждая пара $\mathbf{D}_k = (\mathbf{X}_k, \mathbf{y}_k)$ состоит из $(m \times n)$-матрицы $\mathbf{X}$ и $(m \times 1)$-вектора $\mathbf{y}$. Для каждой регрессионной выборки $\mathbf{D}_k$ известна модель f_k, оптимально приближающая данную выборку. Задано множество $\mathcal{G}$ порождающих функций. Для каждой функции $g : \mathbb{R} \times \ldots \times \mathbb{R} \to \mathbb{R}$ из набора $\mathcal{G}$ определены её арность $v = v(g)$, области определения и значений: $\mathrm{dom}(g), \mathrm{cod}(g)$. Известно множество $\mathcal{F}$ суперпозиций порождающих функций, при этом заданы правила индуктивного порождения функции $f \in \mathcal{F}$:

$$\mathcal{F} = \{f_s \mid \mathbf{f}_s : (\hat{\mathbf{w}}_k, \mathbf{X}) \mapsto \mathbf{y}, s \in \mathbb{N}\}.$$

Каждой выборке $\mathbf{D}$ требуется поставить в соответствие оптимальную модель $f : \mathbb{W} \times \mathbb{X} \to \mathbb{R}$ из порождаемого множества моделей $\mathcal{F} = \{f_s\}$, где $\mathbb{W}$ — пространство параметров, доставляющую минимум заданной функции ошибки, определяемой ниже.

Другими словами, для множества моделей $\mathcal{F}$ требуется найти такой индекс $\hat{s}$, что функция $f_{\hat{s}}$ среди всех $f \in \mathcal{F}$ доставляет минимум функции ошибки S при фиксированной регрессионной выборке $\mathbf{D}$:

$$\hat{s} = \arg\min_{s \in \mathbb{N}} S(f_s \mid \hat{\mathbf{w}}_k, \mathbf{D}_k), \tag{1}$$

где $\hat{\mathbf{w}}_k$ — оптимальный вектор параметров модели f_s для каждой $f \in \mathcal{F}$ при данной регрессионной выборке $\mathbf{D}$:

$$\hat{\mathbf{w}}_k = \arg\min_{\mathbf{w} \in \mathbb{W}} S(\hat{\mathbf{w}} \mid f_s, \mathbf{D}_k). \tag{2}$$

В качестве функции ошибки S используется сумма квадратов регрессионных остатков:

$$S(\mathbf{w}_k \mid f_s, \mathbf{D}_k) = \| \mathbf{y} - f(\mathbf{w}_k, \mathbf{X}) \|_2 \ . \tag{3}$$

1.2 Способ задания структуры регрессионной модели

Каждой суперпозиции f ставится в соответствие дерево Γ_f вида (рис.1), строящееся по следующим правилам:

- корнем дерева является специальный символ “ * ”, имеющий одну дочернюю вершину;
- в остальных вершинах V_i дерева Γ_f находятся соответствующие порождающие функции $g_{r(i)}$ из набора $\mathcal{G}$;
- число дочерних вершин V_j у некоторой вершины V_i равно арности соответствующей функции g_r: $v = v(g_r)$;
- область определения порождающей функции дочерней вершины V_j содержит область значений функции родительской вершины V_i: $\mathrm{dom}(g_{r(i)}) \supset \mathrm{cod}(g_{r(j)})$;
- в листьях дерева Γ_f находятся свободные переменные x_i.

Каждому дереву Γ_f ставится в соответствие бинарная матрица Z (табл.1) размера $(1 + l) \times (l + n)$, где l – число элементарных функций набора $\mathcal{G}$, n — число свободных переменных $\mathbf{x}_i$. Элементы матрицы Z отвечают за наличие ребра между двумя вершинами в дереве. При этом строки матрицы отвечают только за те вершины, которые могут быть родительскими: вершина дерева “ * ” и порождающие функции g_s. Столбцы матрицы отвечают за потенциальные дочерние вершины: порождающие функции g_r и свободные переменные x_i. Таким образом, матрица Z состоит из квадратного блока, отвечающего за набор порождающих функций, добавленной сверху строки, отвечающей за вершину дерева “ * ”, и n добавленных справа столбцов, отвечающих за свободные переменные x_i. На матрицу Z по построению накладываются следующие ограничения:

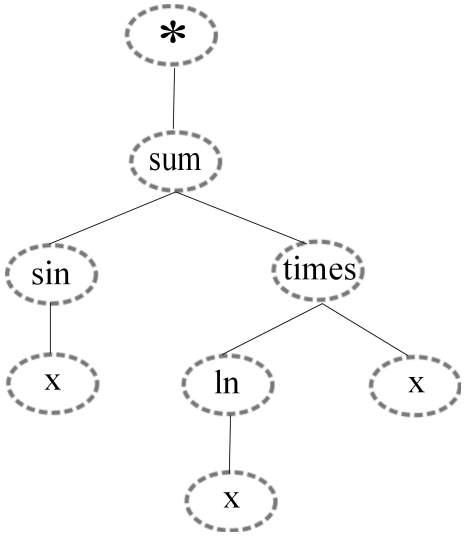

Рис. 1: Пример дерева суперпозиции $f = \sin(x) + (\ln x)x$.

	sum	times	ln	sin	x
$*$	1	0	0	0	0
sum	0	1	1	0	0
times	0	0	0	1	1
ln	0	0	0	0	1
sin	0	0	0	0	1

	sum	times	ln	sin	x
$*$	0.7	0.1	0.1	0.1	0.2
sum	0.2	0.7	0.8	0.1	0.2
times	0.1	0.3	0	0.8	0.8
ln	0.2	0.1	0.3	0.1	0.9
sin	0.1	0.2	0.1	0	0.8

Таблица 1: Матрица связей Z_f и матрица вероятностей связей P_f дерева Γ_f суперпозиции $f = \sin(x) + (\ln x)x$.

- в каждой строке i содержится либо количество единиц, равное арности $v = v(g_{r(i)})$ элементарной функции $g_{r(i)}$, отвечающей за i-ый столбец матрицы, либо ноль;
- в каждом столбце, отвечающем за порождающую функцию, может содержаться только одна единица;
- заполнение строк и столбцов проходит сверху–вниз и слева–направо, т.е. для записи очередного ребра в матрицу выбирается самый левый и верхний из "свободных" столбцов и строк, отвечающий тем же родительским и дочерним элементам.

Обозначим для удобства множество матриц, удовлетворяющих данным условиям как $\mathcal{M}$.

1.3 Уточнение постановки задачи

Поскольку по матрице из множества $\mathcal{M}$ можно однозначно восстановить суперпозицию функции, задача прогнозирования суперпозиции f сводится к поиску матрицы $\mathbf{Z}_f$ из множества $\mathcal{M}$, максимизирующей вероятность переходов в дереве суперпозиции:

$$\mathbf{Z}_f = \arg\max_{\mathbf{Z}\in\mathcal{M}} \sum_{i,j} P_{ij} \times Z_{ij}, \tag{4}$$

где матрица вероятностей переходов $\mathbf{P}$ определяется с помощью векторной логистической регрессии с функцией ошибки, соответствующей гипотезе порождения данных биномиальным распределением.

2 Решение задачи структурного обучения

2.1 Процедура прогнозирования структуры модели

Пусть с помощью векторной логистической регрессии найдена матрица вероятностей переходов P_f вида табл. 1. Ставится задача отыскания матрицы $\mathbf{Z}_f$ из допустимого множества матриц $\mathcal{M}$, удовлетворяющей условию 4. Для этого разобьем матрицу $\mathbf{P}_f$ на два блока. Блок $P'_{(1+l)\times l}$:

$$P'_{ij} = p(g_i \to g_j)$$

отвечает за вероятности переходов между порождающими функциями. Блок $P''_{(1+l)\times n}$:

$$P''_{ik} = p(g_i \to x_k)$$

содержит значения вероятностей перехода от порождающих функций к независимым переменным. Введем понятия открытой вершины. Назовем вершину i - *открытой*,

если она относится к порождающей функции, и при этом существует вершина, являющаяся для вершины i родительской, но у нее нет дочерних вершин:

$$(i \leq l)\&(\exists j : (j,i) = 1)\&(\nexists k : (i,k) = 1).$$

Также зададим значение K максимально допустимой сложности суперпозиции. Опишем процедуру построения оптимального дерева $\hat{\Gamma}_f$.

- На нулевом шаге процедуры объявляем вершину дерева открытой: $i = 1$.
- Пока количество единиц в матрице не превышает K, повторяем:
  1. выбираем максимальные вероятности переходов $c_j = \max\limits_{j=1,\ldots,l} P_{ij}$ для всех открытых вершин i;
  2. достраиваем матрицу из условия максимизации вероятности перехода: $j^* = \arg\max\limits_j c_j, \quad (i,j^*) = 1$;
  3. добавляем j^* к списку открытых вершин, если $(i,j^*) \in P'$;
- если количество единиц превышает K, ставим в соответствие всем открытым вершинам независимые переменные: $k^* = \arg\max\limits_k P''_{ik}, \quad (i,k^*) = 1$ для всех i-открытых.

Процедура может быть прервана, если множество открытых вершин пусто, но сложность суперпозиции еще не превысила заданную максимальную сложность K — в таком случае построенная оптимальная суперпозиция будет иметь меньшую сложность.

2.2 Вычислительный эксперимент на синтетических данных

Алгоритм был протестирован на выборке синтетических данных, полученных следующим образом. Экспертно задан набор порождающих функций $\mathcal{G}$, для каждой из которых известны арность функции $v = v(g)$, области определения и значений: $dom(g), cod(g)$. По набору $\mathcal{G}$ было построено конечное множество суперпозиций $\mathcal{F}$ — библиотека функций. Экспертно заданы значения независимых переменных $\mathbf{X}$ и вектор параметров модели $\mathbf{w}_s$. Значения зависимых переменных заданы как

$$\mathbf{y}_s = f_s(\mathbf{w}_s, \mathbf{X}) + \tau_f,$$

где τ_f — шумовая добавка, являющаяся случайной величиной из нормального распределения. На рис.2 изображен вид исходных моделей. В каждой строке i содержатся графики модели f_i, зашумленной независимо друг от друга 5 раз одним и тем же распределением. Таким образом, сгенерировано множество регрессионных выборок $\mathfrak{D} = \{(\mathbf{D}_s, f_s)\}$, где $\mathbf{D}_s = (\mathbf{X}, \mathbf{y})$.

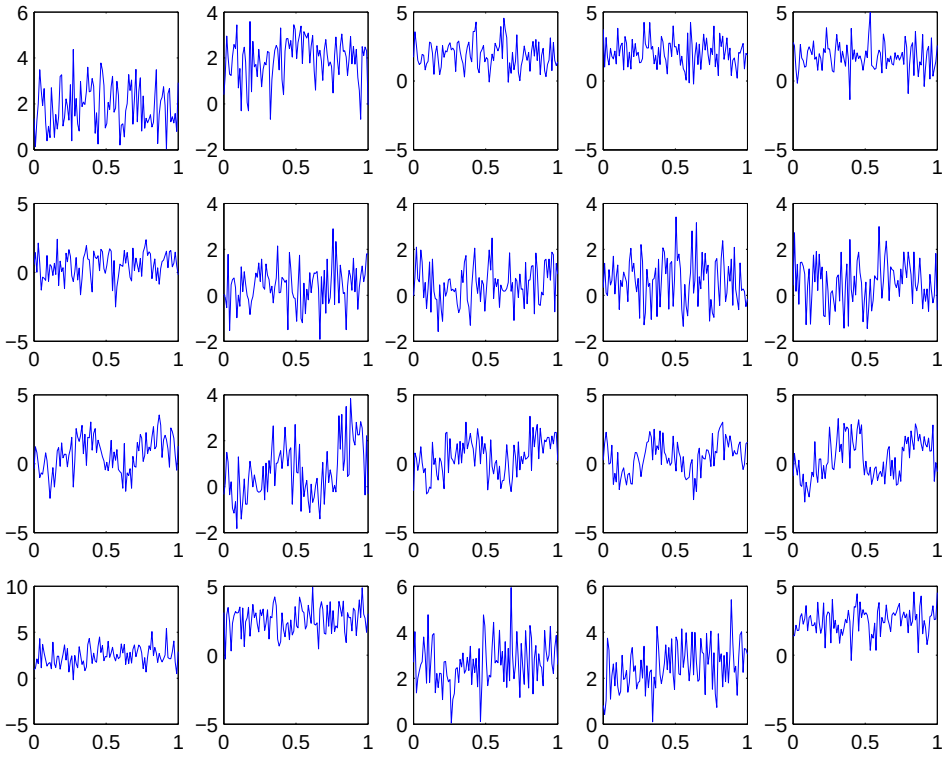

Рис. 2: Вид исходных данных.

Для обучения алгоритма векторной логистической регрессии была использована стандартная нейронная сеть в среде Matlab с двумя скрытыми слоями, имеющая на выходном слое сигмоидную функцию активации. На вход такой нейросети подается регрессионная выборка $\mathbf{D}_s = (\mathbf{X}, \mathbf{y})$, выходом алгоритма является матрица вероятностей $\mathbf{P}$. Далее при помощи указанной выше процедуры построения оптимального дерева $\hat{\Gamma}_f$, прогнозировалась искомая структура суперпозиции модели. Пример работы алгоритма изображен на рис.(3, 4, 5). Левая матрица соответствует исходной суперпозиции f_s, средняя матрица — построенной матрице вероятностей переходов $\mathbf{P}_f$, по которой, используя предложенную процедуру, находилась оптимальная прогнозируемая суперпозиция $\hat{f}_s$.

Для тестирования качества алгоритма использовался метод LOO(Leave-One-Out), по которому множество регрессионных выборок разбивается таким образом, что в обучении алгоритма использовались все выборки, за исключением одной: $\mathfrak{D} \setminus \{\mathbf{D}_k\}$. Контроль проводился на одной выборке $\mathbf{D}_k$, для которой по полученному алгоритму вычислялось оптимальное дерево суперпозиции $\hat{\Gamma}_k$, строилась модель $\hat{f}_k$, настраивались ее параметры $\hat{\mathbf{w}}_k$ и вычислялось значение ошибки

$$S(\hat{\mathbf{w}}_k, \hat{f}_s, f_s) = \| \mathbf{y} - f(\mathbf{w}_k, \mathbf{X}) \|_2.$$

Результаты прогнозирования по методу LOO представлены на рис.6, отражающем полученные вероятности переходов $\mathbf{P}$, и рис.7, отражающем оптимальные про-

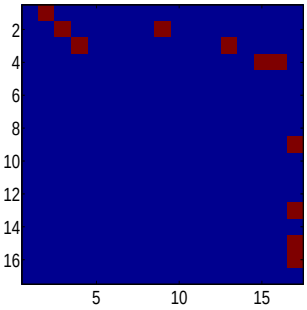

Рис. 3: Исходная матрица переходов $\mathbf{Z}_f$.

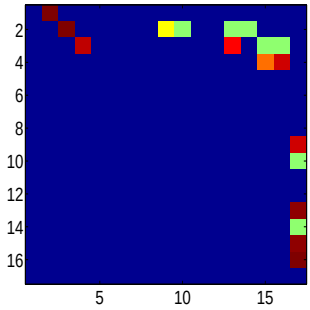

Рис. 4: Полученная матрица вероятности переходов $\mathbf{P}_f$.

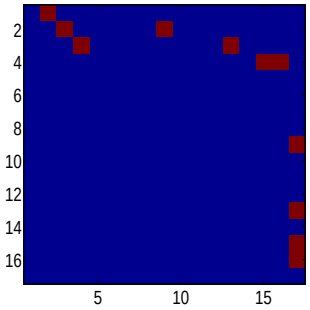

Рис. 5: Полученная матрица переходов $\hat{\mathbf{Z}}_f$.

гнозируемые матрицы переходов $\hat{\mathbf{Z}}$.

Из рис.7 видно, что не во всех строках получились одинаковые матрицы переходов, что отражает некую неустойчивость предложенного метода относительно вводимого шума τ. Качество работы алгоритма в зависимости от размера σ дисперсии нормальной случайной величины τ изображено на рис.8. Видно, что при увеличении шума качество прогнозирования падает, растет ошибка S. Немонотонность графика объясняется малыми для обучения размерами выборки. На рис.9 изображен график зависимости ошибки прогнозирования от возмущения $\delta\mathbf{w}$ вектора параметров $\mathbf{w}_k$ модели f_k контрольной выборки

$$\mathbf{D}_k = (\mathbf{X}_k, \mathbf{y}_k), \mathbf{y}_k = f(\mathbf{w}_k, \mathbf{X}_k) + \tau_f.$$

В данном случае также можно отметить увеличение ошибки S от размера возмущения $\delta\mathbf{w}$, но при этом качество прогнозирования остается достаточно хорошим при небольших значениях возмущения. Поиск метода, справляющегося с указанными ухудшениями качества прогнозирования, является задачей для последующего изучения.

3 Анализ прикладной задачи разметки библиографических списков

3.1 Проблема и предложение

Решается прикладная задача представления неформатированных библиографических записей в виде структуры в формате BibTEX [17] — стандарта управления коллекцией библиографических записей. Требуется разметить библиографические записи: определить соответствия между текстовыми сегментами и полями записей

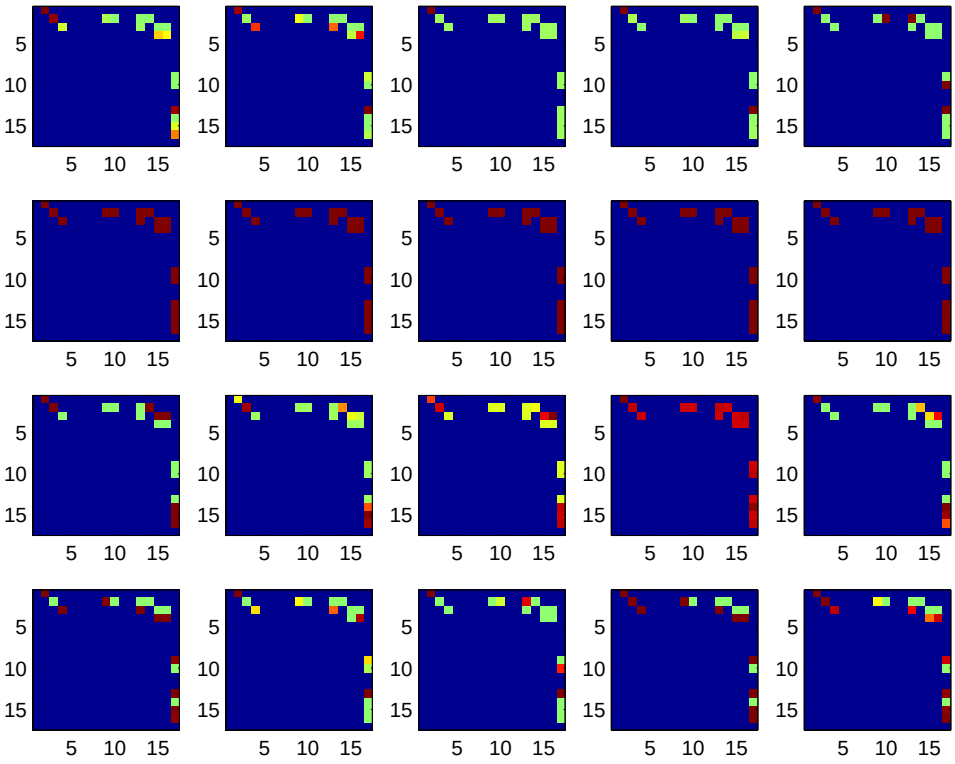

Рис. 6: Полученные матрицы вероятностей переходов $\mathbf{P}_f$.

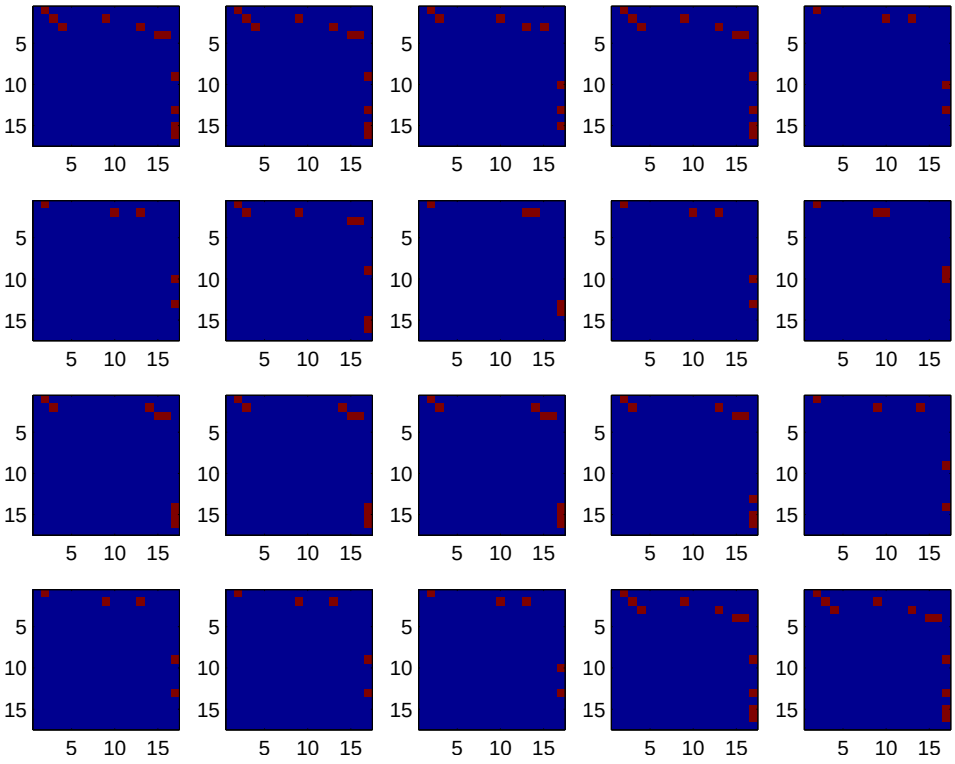

Рис. 7: Полученные матрицы переходов $\mathbf{Z}_f$.

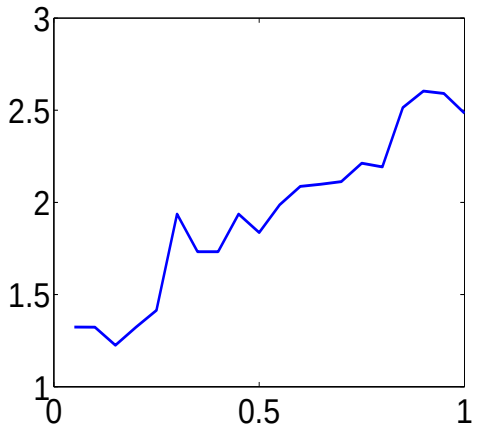

Рис. 8: Зависимость ошибки S от дисперсии шума σ.

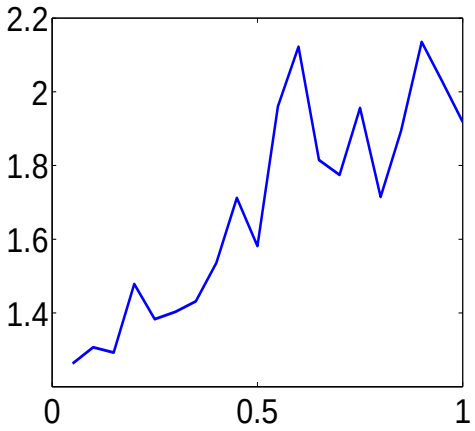

Рис. 9: Зависимость ошибки S от возмущения параметров модели $\delta\mathbf{w}$.

BibTEX, а также указать тип библиографической записи. Необходимость форматирования текстовой строки вызвана наличием различных стандартов (ГОСТ 7.82-2001, MLA и др.) определяющих различный порядок следования полей библиографической записи. Кроме того, запись может быть составлена с нарушением стандартов. Для поиска структуры текстовой строки предлагается использовать методы структурного обучения.

В табл.2 приведен пример неформатированной библиографической записи и полученные верные и неверные соотношения между сегментами записи и набором полей BibTEX. В работе решается задача выбора оптимального набора признаков модели. Для этого используется модификация ранее предложенного авторами [18, 19] алгоритма последовательного добавления и удаления признаков.

После построения соотношений между текстовыми сегментами и набором полей структуры BibTEX, требуется определить тип библиографической записи. Для этого определяется подмножество полей структуры BibTEX, присутствующих в библиографической записи. По полученному подмножеству строится новое признаковое описание библиографической записи. С помощью кластеризации k-means [20] определяется тип каждой записи.

Таким образом, решаются две прикладные подзадачи:

1. определение для каждого сегмента типа поля библиографической записи в структуре BibTEX.

2. определение типа библиографической записи.

Kwok T.-Y., Yeung D.-Y. Constructive Algorithms for Structure Learning in Feedforward Neural Networks for Regression Problems // IEEE Transactions on Neural Networks, 1997. Vol. 8. P. 630–645.		
	Верно	Неверно
Type	Article	Book
Author	Kwok T.-Y., Yeung D.-Y.	Constructive Algorithms for Structure Learning in Feedforward Neural Networks for Regression Problems
Title	Constructive Algorithms for Structure Learning in Feedforward Neural Networks for Regression Problems	IEEE Transactions on Neural Networks
Journal	IEEE Transactions on Neural Networks	—
Pages	630–645	1997
Volume	8	1997
Year	1997	630–645
Address	—	—
Publisher	—	—
Editor	—	Kwok T.-Y., Yeung D.-Y.
URL	—	—

Таблица 2: Пример работы алгоритма сегментирования

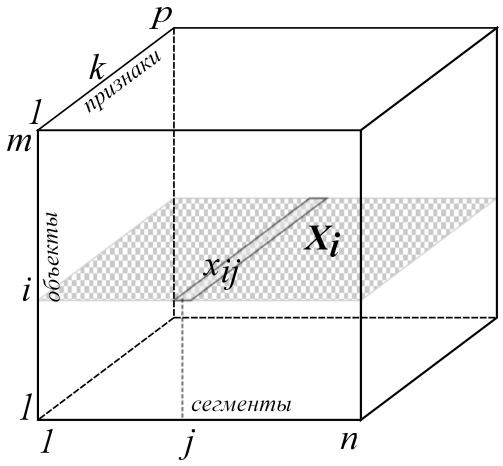

Рис. 10: Вид матрицы $\mathbf{X}$

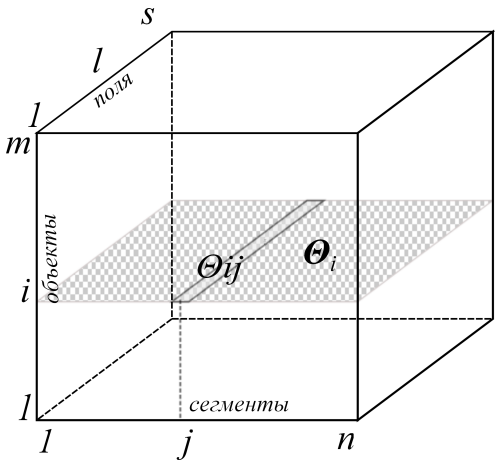

Рис. 11: Вид матрицы $\mathbf{Y}$, $\boldsymbol{\Theta}$

3.2 Формальная постановка задачи

Дан набор из m строк $\{t_1, t_2, \ldots, t_m\}$ – библиографических записей. Каждая запись t состоит из текстовых сегментов $t_i = \{t_i^1, t_i^2, \ldots, t_i^n\}$. Задан набор $\mathcal{G}$ порождающих функций $\mathcal{G} = \{g\}$, отображающих j-ый текстовый сегмент i-ой строки t_i^j в вектор-строку признаков $\mathbf{x}_{ij}$:

$$g : t_i^j \mapsto \mathbf{x}_{ij}.$$

Задана трехиндексная матрица $\mathbf{X}$ «объект – сегмент – признак» размера $m \times n \times p$, где m – число библиографических записей, n – число сегментов, p – число признаков. Каждому объекту, заданному двухиндексной матрицей $\mathbf{X}_i$, где i – номер объекта, поставлена в соответствие бинарная матрица ответов $\mathbf{Y}_i$ размера $n \times s$, где s – число типов полей структуры BibTEX. Элементы матрицы $\mathbf{Y}_i$ отвечают за принадлежность j-го сегмента к l-му типу поля библиографической записи:

$$Y_i(j,l) = \begin{cases} 1, \text{ если } \mathbf{x}_{ij} \text{ принадлежит к } l\text{-ому типу поля;} \\ 0, \text{ иначе,} \end{cases} \tag{5}$$

где $j \in \{1, 2, \ldots, n\}$ - индекс сегмента текстовой строки, $l \in \{1, 2, \ldots, s\}$ - индекс типа поля. Вводится двухиндексная матрица весовых параметров $\mathbf{W}$ размером $p \times s$, элементы которой w_{kl} отвечают за значимость k-ого признака для l-ого типа поля, $k = 1, \ldots, p$, $l = 1, \ldots, s$. При умножении матрицы $\mathbf{W}$ справа на вектор-строку признаков $\mathbf{x}_{ij}$ j-ого сегмента i-ого объекта получается вектор-строка $\boldsymbol{\theta}_{ij}$, элементы которой определяют оценку принадлежности данного сегмента к полям структуры BibTEX:

$$\underset{1\times p}{\mathbf{x}_{ij}} \underset{p\times s}{\mathbf{W}} = \underset{1\times s}{\boldsymbol{\theta}_{ij}}\,. \tag{6}$$

Тогда оптимальный прогнозируемый тип поля с индексом $\hat{l}$ для признакового

описания $\mathbf{x}_{ij}$ сегмента библиографической записи с фиксированным номером i определяется как индекс максимального элемента вектор-строки $\boldsymbol{\theta}_{ij}$:

$$\hat{l}_j = \arg \max_{l=1,2,\ldots,s} \boldsymbol{\theta}_{ij}(l).$$

Аналогично записывая строки $\boldsymbol{\theta}_{ij}$ для каждого вектора признаков $\mathbf{x}_{ij}$ с индексом j объекта $\mathbf{X}_i$, составляется матрица оценок $\boldsymbol{\Theta}_i$, значения которой определяют тип поля каждого сегмента объекта $\mathbf{X}_i$:

$$\hat{Y}_i(j,l) = \begin{cases} 1, \text{ если } l = \hat{l}_j; \\ 0, \text{ иначе.} \end{cases} \tag{7}$$

Требуется найти набор признаков $\mathcal{A}$ из множества $\mathcal{G}$ и веса $\mathbf{W}$ этих признаков, что расстояние $\text{Dist}(\hat{\mathbf{Y}}, \mathbf{Y})$между матрицей ответов $\mathbf{Y}$ и прогнозируемой матрицей $\hat{\mathbf{Y}}$ минимально:

$$\hat{\mathbf{W}} = \arg \min_{\mathbf{W},\mathcal{A}} \text{Dist}(\hat{\mathbf{Y}}, \mathbf{Y}) = \arg \min_{\mathbf{W},\mathcal{A}} \frac{1}{2} \sum_{i,j,k=1}^{m,n,p} |\hat{Y}_i(j,k) - Y_i(j,k)|. \tag{8}$$

Искомая матрица весов $\hat{\mathbf{W}}$ определяется минимумом аппроксимированного эмпирического риска Q (3.2) для случая логистической регрессии.

Введем обозначения, необходимые для определения функции эмпирического риска. Матрица весов признаков $\mathbf{W}$ разбивается на s независимых вектор-столбцов $\mathbf{w}_l$, соответствующих типу поля l, $\quad l \in \{1, 2, \ldots, s\}$, и для каждого столбца $\mathbf{w}_l$ векторизуется соответствующая ему часть матрицы $\mathbf{Y}$:

$$\underset{mn\times 1}{\mathbf{u}_l} = \begin{pmatrix} \mathbf{Y}_1(l) \\ \vdots \\ \mathbf{Y}_i(l) \\ \vdots \\ \mathbf{Y}_m(l) \end{pmatrix}, \text{ где } \mathbf{Y}_i(l) \text{ — } l\text{-ый столбец матрицы } \mathbf{Y}_i.$$

Матрица $\mathbf{X}$ записывается в двумерном виде:

$$\underset{mn\times p}{\mathbf{Z}} = \begin{pmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_i \\ \vdots \\ \mathbf{X}_m \end{pmatrix}, \text{ где } \mathbf{X}_i \text{ — матрица признаков } i\text{-го объекта.}$$

Обозначим $\mathbf{z}_i$ — i-ую строку матрицы $\mathbf{Z}$. Тогда функция эмпиричиского риска записывается в виде

$$Q = \sum_{i=1}^{m\times n} \sum_{l=1}^{s} \mathcal{L}(\mathbf{w}_l, \mathbf{z}_i, u_l(i), \mathcal{A}),$$

где функция потерь

$$\mathcal{L}(\mathbf{w}_l, \mathbf{z}_i, u_l(i), \mathcal{A}) = \log\bigl(1 + \exp\bigl(-\langle \mathbf{w}_l{}^{\mathcal{A}}, \mathbf{z}_i{}^{\mathcal{A}} \rangle u_l(i)\bigr)\bigr).$$

При этом $\mathbf{w}_l{}^{\mathcal{A}}$ и $\mathbf{z}_i{}^{\mathcal{A}}$ содержат только подмножество $\mathcal{A} \subset \mathcal{G}$ своих элементов индексов признаков:

$$\hat{\mathcal{A}} = \arg\min_{\mathcal{A} \subset \mathcal{G}} Q(\mathbf{W}, \mathbf{Z}, \mathbf{u}, \mathcal{A}).$$

3.3 Описание алгоритма выбора признаков

Требуется решить задачу поиска оптимального подмножества индексов признаков $\mathcal{A} \subset \mathcal{G}$ и оценки матрицы весов признаков $\mathbf{W}(\mathcal{A})$(8). Для этого используется следующая процедура последовательного добавления и удаления признаков:

Зададим начальное множество признаков $\mathcal{A}_0 = \varnothing$, параметр останова d и начальные значения функции эмпирического риска $\hat{Q} = Q(\varnothing)$ и номера итерации $t = 0$.

1. Пока мощность набора признаков $|\mathcal{A}_t|$ меньше общего числа признаков p
   (a) увеличиваем номер итерации $t = t + 1$;
   (b) находим оптимальный для добавления признак с индексом $\hat{g} = \arg\min_{g \in \mathcal{G} \setminus \mathcal{A}_{t-1}} Q(\mathcal{A}_{t-1} \cup \{\hat{g}\})$,
   и добавляем его к набору: $\mathcal{A}_t = \mathcal{A}_{t-1} \cup \{\hat{g}\}$;
   (c) если $Q(\mathcal{A}_t) < \hat{Q}$, то текущее минимальное значение эмпирического риска $\hat{Q} = Q$, номер оптимальной итерации $\hat{t} = t$;
   (d) если значение функционала не улучшалось на протяжении d шагов $t - \hat{t} \geq d$, то прервать цикл.
2. Пока мощность набора признаков $|\mathcal{A}_t|$ ненулевая
   (a) увеличиваем номер итерации $t = t + 1$;
   (b) находим оптимальный для удаления признак
   $\hat{g} = \arg\min_{g \in \mathcal{A}_{t-1}} Q(\mathcal{A}_{t-1} \setminus \{\hat{g}\})$,
   и удаляем его из набора: $\mathcal{A}_t = \mathcal{A}_{t-1} \setminus \{\hat{g}\}$;
   (c) если $Q(\mathcal{A}_t) < \hat{Q}$, то $\hat{Q} = Q$, $\hat{t} = t$;
   (d) если значение функционала не улучшалось на протяжении d шагов $t - \hat{t} \geq d$, то прервать цикл.
3. Повторять шаги 1. и 2. пока значения $Q(\mathcal{A}_{\hat{t}})$ убывают.

Алгоритм отбора признаков определяет их оптимальный набор $\hat{\mathcal{A}} = \mathcal{A}_{\hat{t}}$ с одновременным оцениванием матрицы весов $W(\mathcal{A}_{\hat{t}})$.

3.4 Определение типа библиографической записи

Матрица $\hat{\mathbf{Y}}_i$ (7) содержит полную информацию о типах полей l, содержащихся в i-ой библиографической записи. Для решения подзадачи об определении типа записи BibTEXсоставляется матрица $\mathbf{B}$ размера $m \times s$ по правилу

$$B(i,l) = \begin{cases} 1, \text{ если } l\text{-ая строка матрицы} \hat{\mathbf{Y}}_i \text{ ненулевая;} \\ 0, \text{ иначе.} \end{cases} \tag{9}$$

Таким образом, каждый элемент матрицы $B(i,l)$ определяет присутствие в i-ой библиографической записи l-ого типа поля BibTEX, а строка $\mathbf{b}_i$ является новым признаковым описанием объекта — библиографической записи. Поставим задачу разбиения объектов на фиксированное число кластеров r — типов записи в структуре BibTEX.

Обозначим $k(i)$ — номер кластера, к которому отнесена i-ая библиографическая запись и введем следующие две функции качества:
минимизацию среднего внутрикластерного расстония

$$F_0 = \frac{\sum_{i<j}[k(i)=k(j)]\rho(\mathbf{b}_i,\mathbf{b}_j)}{\sum_{i<j}[k(i)=k(j)]} \to \min,$$

и максимизацию среднего межкластерного расстояния

$$F_1 = \frac{\sum_{i<j}[k(i)\neq k(j)]\rho(\mathbf{b}_i,\mathbf{b}_j)}{\sum_{i<j}[k(i)\neq k(j)]} \to \max.$$

Задача кластеризации сводится к минимизации функции качества F :

$$F = \frac{F_0}{F_1} \to \min.$$

В качестве метрики используется диагонально взвешенная евклидова метрика:

$$\rho(\mathbf{x},\mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^T\Lambda^2(\mathbf{x}-\mathbf{y})}, \text{ где } \Lambda = \mathrm{diag}(\lambda). \tag{10}$$

Диагональная матрица $\Lambda \neq 1$ задает веса, соответствующие признакам описания библиографической записи. Ее значения определяются частотой вхождения поля в библиографические записи.

Кластеризация выполняется с помощью метода k-means. Метод состоит из двух основных шагов:

1. Для каждого элемента $\mathbf{b}_i$ находится ближайший к нему центр кластеров, к которому и относится данный элемент:

$$k(i) = \arg\min_{j=1,\ldots,r} \rho(\mathbf{b}_i,\mu_j).$$

2. Положение центра μ_j есть центр масс объектов, принадлежащих кластеру:

$$\boldsymbol{\mu}_j = \frac{\sum_{i=1}^{|D_j|} [k(i)=j]\mathbf{b}_i}{\sum_{i=1}^{|D_j|} [k(i)=j]}, \text{ где } |D_j| \text{ — мощность } j\text{-го кластера, } j \in \{1,\ldots,r\}.$$

Шаги алгоритма повторяются, пока кластеризация объектов меняется.

Начальное приближение центров кластеров μ_j, $j = 1,\ldots,r$ считается заданным. Результатом работы алгоритма служит вектор-столбец $\mathbf{k}$, элементы которого $k(i) = j$ отвечают за принадлежность i-ой библиографической записи к j-ому типу записи.

3.5 Вычислительный эксперимент

Алгоритм протестирован на выборке из 100 библиографических записей. Задано число используемых типов полей структуры BibTEX для данной выборки $s = 11$: автор, название работы, название источника, номера страниц, номер выпуска, номер тома, год, город, издательство, редакторы, ссылка на работу в интернете. Максимальное число текстовых сегментов $m = 9$. Для каждого сегмента генерировался столбец из $p = 18$ признаков: длина сегмента, порядковый номер, число различных знаков препинания (точки, запятые, тире, кавычки, скобки, двоеточия, точки с запятой), общее число сегментов объекта, число заглавных букв, число цифр, количество слов, наличие инициалов, наличие подряд идущих цифр (числа), наличие подряд идущих заглавных букв (аббревиатуры), общая длина всех сегментов в записи, общее исло слов в записи.

Таким образом формировалась матрица $\mathbf{X}$ размера $m \times n \times p$, где $m = 100$, $n = 9$, $p = 18$. Матрица ответов $\mathbf{Y}$ задана в виде (5). Множество объектов $(\mathbf{X}_i, \mathbf{Y}_i)$ разбито на обучающую и контрольную выборки.

На рис.12 показан набор признаков на каждой итерации алгоритма выбора признаков. Видно, что оптимальный набор признаков был найден за небольшое число итераций ($t = 10\ldots15$). Показано, что в данном случае некоторые признаки оказались неинформативны и были удалены сразу же после добавления в набор. На рис. 13 показано среднее число ошибок на обучающей выборке(сплошная линия) и контрольной выборке(пунктирная линия).

На рис. 14 показано количество совпадений экпертной сегментации с сегментацией, проведенной предложенным алгоритмом: красный цвет означает нахождение в данном квадрате максимального числа сегментов, синий — минимального. Нахождение сегмента на диагонали означает совпадение экспертной и полученной сегментации. Видно, что первые по порядку поля имеют лучшее качество сегментации. Последующие поля имеют значительно худшее качество сегментации. Разница возникает в силу того, что первые поля (автор, название и др.) чаще присутствуют в библиографических записях и выборка достаточно велика, чтобы получить адекват-

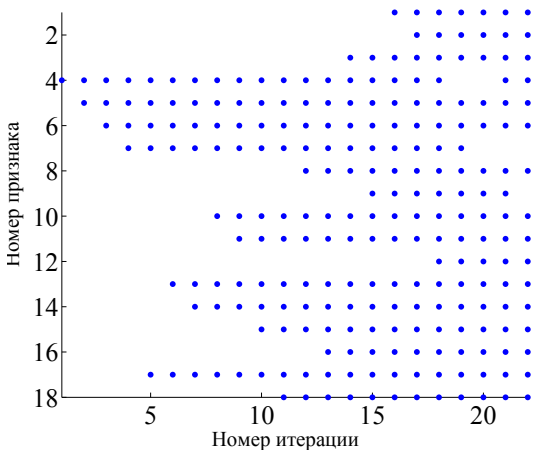

Рис. 12: Зависимость используемых признаков от номера итерации.

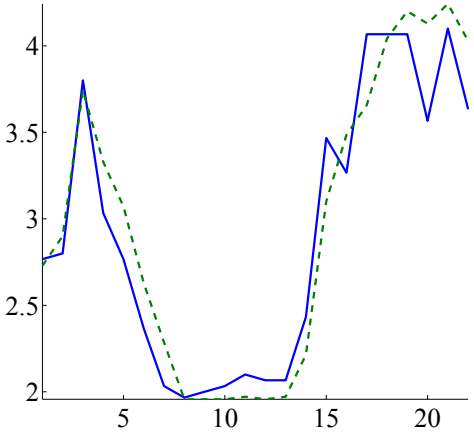

Рис. 13: Зависимость числа ошибок от номера итерации.

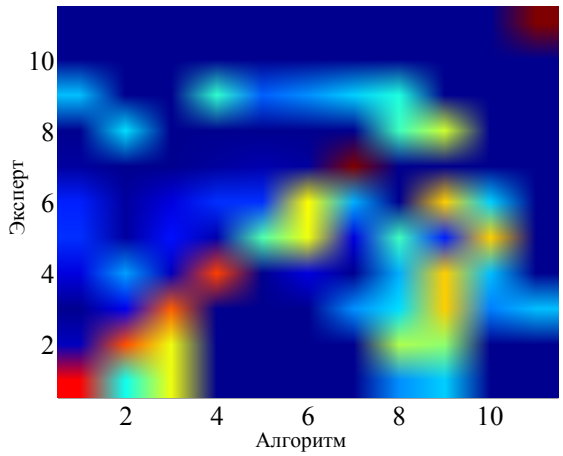

Рис. 14: Совпадение экспертной и полученной сегментации.

ную оценку вектора параметров логистической регрессии. Последние же поля присутствовали лишь в небольшом числе записей из выборки 100 записей (некоторые поля — меньше, чем в 10).

Библиографические записи были разделены на 6 заданных кластеров — типов библиографических записей: статья, книга, тезис конференции, диссертация, электронный источник, либо ни один из указанных типов. Параметры $\lambda_i(10)$метрики ρ заданы в зависимости от частоты вхождения i-го поля в библиографические записи. Чем чаще поле присутствует в библиографической записи, тем менее оно важно:

$$\lambda_{ii}^2 = \left(1 - \frac{\sum_{j=1}^{n} B(i,j)}{n}\right)^2, \quad 0 \le \lambda_i < 1, \quad i = 1, \ldots, s. \tag{11}$$

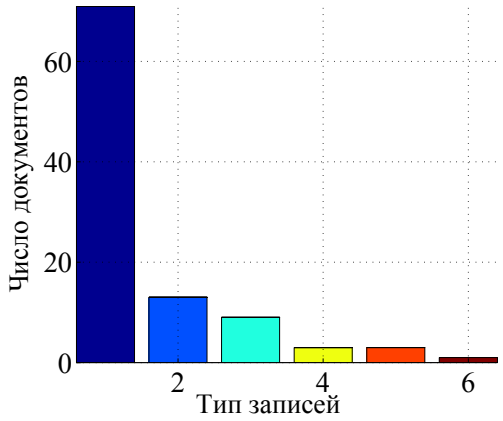

Рис. 15: Экспертная кластеризация записей.

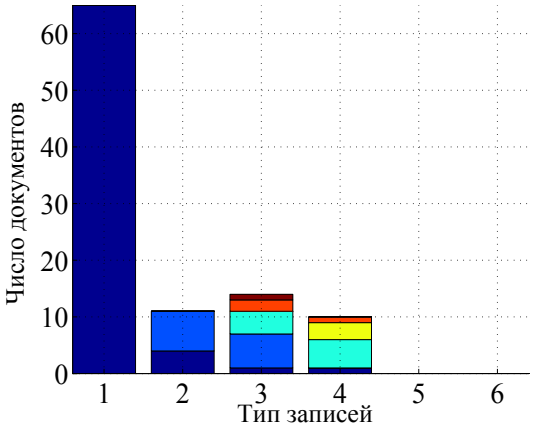

Рис. 16: Полученная кластеризация записей.

Например, поле “название” или “автор”, присутствует почти во всех типах записей, а “число страниц”, “название журнала” – для заметно меньшего числа. Начальное положение центра кластера μ_j задано теми векторами $\mathbf{b}_i = \mu_j$, которые соответствуют записям, содержащим редко встречающиеся типы полей, и при этом находящимся на удалении друг от друга больше заданной величины $\rho_{\min}$

Диаграмма 15 отражает количество записей, относящихся к каждому из типов BibTEXпо мнению эксперта. На диаграмме 16 представлен результат кластеризации.Типы записей, представленные малым числом примеров, не определились: как видно из рис.16, столбцы 5 и 6 имеют нулевые значения. На тех типах записей, которые были представлены в выборке бóльшим числом примеров, предложенный метод показал результат, слабо отличающийся от экспертной кластеризации.

Заключение

В работе ставится и решается задача прогнозирования суперпозиции модели исходных данных. Предложен алгоритм поиска оптимальной структуры модели. Качество предлагаемого метода проверено на выборке синтетических данных. Решена прикладная задача сегментирования структурированных текстов методами структурного обучения. Использован алгоритм последовательного выбора признаков, исследованы свойства алгоритма. Алгоритм проиллюстрирован выборкой из неформатированных библиографических записей, для которых каждому сегменту ставилось в соответствие поле в структуре BibTEX. Для каждой записи определялся её тип в данной структуре. Проведена оптимизация параметров алгоритма. Представлены результаты работы рассматриваемого метода сегментирования и кластеризации библиографических записей.

Публикации по теме

1. *Варфоломеева А.А.* Локальные методы прогнозирования с выбором метрики // Машинное обучение и анализ данных, 2012. Т.1, Вып. 3. Стр. 367–375.

2. *Варфоломеева А.А., Стрижов В.В.* Выбор признаков при разметке библиографических списков методами структурного обучения // НТВ СПбГУ, подано в редакцию.

Список литературы

[1] *Riccardo Poli, William B. Langdon, Nicholas F. McPhee* A Field Guide to Genetic Programming, 2008.

[2] *Koza, J. R.* Genetic programming. In Williams, J. G. and Kent, A. (editors) // Encyclopedia of Computer Science and Technology, 1998. Vol. 39. P.: 29-43.

[3] *Ivan Zelinka, Zuzana Oplatkova, and Lars Nolle* Analytic programming – symbolic regression by means of arbitrary evolutionary algorithms, August 2008.

[4] *Г.И. Рудой, В.В. Стрижов* Алгоритмы индуктивного порождения суперпозиций для аппроксимации измеряемых данных // Информатика и её применения, 2013. Vol: 1.

[5] *Г.И. Рудой, В.В. Стрижов* Упрощение суперпозиций элементарных функций при помощи преобразований графов по правилам // Интеллектуализация обработки информации. Доклады 9-й международной конференции, 2012. P. 140-143.

[6] *Yoshua Bengio* Learning Deep Architectures for AI // Foundations and Trends inMachine Learning, 2009. Vol: 2, No. 1.P.: 1–127.

[7] *Itamar Arel, Derek C. Rose, Thomas P. Karnowski* Deep Machine Learning—A New Frontier in Artificial Intelligence Research // IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE, November 2010. P. 13-19.

[8] *Yoshua Bengio, Aaron Courville, Pascal Vincent* Representation Learning: A Review and New Perspectives // Department of computer science and operations research, U. Montreal.

[9] *Martins A. F. T.* The Geometry of Constrained Structured Prediction: Applications to Inference and Learning of Natural Language Syntax. Carnegie Mellon University, 2012.

[10] *Jaakola T., Sontag D.* Learning Bayesian Network Structure using LP Relaxations // Proceedings of the 13th International Conference on Artifcial Intelligence and Statistics (AISTATS), 2010. Vol: 9. Issue: 1. P.: 358–365.

[11] *Jaakkola T.* Scaled structured prediction http://video.yandex.ru/users/ya-events/view/486/user-tag/научный%20семинар/

[12] *Lampert C. H.* Maximum Margin Multi-Label Structured Prediction. IST Austria (Institute of Science and Technology Austria), 2011.

[13] *Полежаев, В.* Задачи и методы автоматического построения графа цитирований по коллекции научных документов // Труды МФТИ, 2012. Vol. 4. Pp. 1-12.

[14] *Borkar V., Deshmukh K., Saravagi S.* Automatic segmentation of text into structured records. // Proceedings of the 2001 ACM SIGMOD international conference on management of data. New York: ACM, 2001. Vol. 30. No. 2. Pp. 175-186.

[15] Cititation Parser http://freecite.library.brown.edu/

[16] *Kwok T.-Y., Yeung D.-Y.* Constructive Algorithms for Structure Learning in Feedforward Neural Networks for Regression Problems // IEEE Transactions on Neural Networks, 1997. Vol. 8. P. 630–645.

[17] Библиографические записи в формате BibTeX // http://www.bibtex.org (20.12.2012).

[18] *Strijov V., Krymova E., Weber G.W.* Evidence optimization for consequently generated models // Mathematical and Computer Modelling, 2013. Vol: 57(1-2). P. 50-56.

[19] *Стрижов В.В., Крымова Е.А.* Выбор моделей в линейном регрессионном анализе // Информационные технологии, 2011. Вып. 10. Стр. 21–26.

[20] *Кузьмин А. А., Адуенко А. А., Стрижов В. В.* Выбор признаков и оптимизация метрики при кластеризации коллекции документов // Известия Тульского государственного университета, Естественные науки, 2012. Вып. 4, Стр. 119-131.

[21] *Bishop C.M.* Pattern Recognition and Machine Learning. LLC: Springer Science, 2006.