УДК 004.02:004.032.26

Р.М. Алыгулиев
Институт информационных технологий НАН Азербайджана, г. Баку
a.ramiz@science.az

Метод кластеризации коллекции документов и алгоритм для оценки оптимального числа классов

Формализована задача оптимального разбиения коллекций документов. Критерием качества (оптимальность) разбиения является максимизация меры подобия между документами в кластерах. Предлагаемый подход сведен к задаче линейного целочисленного программирования (ЛЦП) с бинарными переменными. Приведен генетический алгоритм решения задачи ЛЦП.

Введение

Кластеризация текстовых документов в настоящее время является одной из самых важных и динамично развивающихся областей информационных технологий. Это обусловлено рядом причин, первая и важнейшая из которых – необходимость обработки значительных объемов текстовой информации Internet. Кластеризация как фундаментальный метод используется во многих областях, таких, как data mining [1], [2], информационного поиска [3-5], обнаружения тематики [6] и т.д. В области информационного поиска для улучшения эффективности широко применяется кластеризация документов. Кластеризация, т.е. разбиение совокупности документов на кластеры сходных по контенту, является одним из методов подготовки документов к поиску. Если прямой поиск не приведет к успеху, то кластеризованное множество документов принимается как результат поиска. Кластеризацию иногда называют предварительным этапом обработки в задачах классификации и резюмировании текстовых документов [7-10].

При кластеризации текстовых документов применяются разные методы и алгоритмы [11-13]. В настоящей работе для кластеризации совокупности текстовых документов предлагается метод, который математически выражается в виде задачи линейного целочисленного программирования с бинарными переменными. Задача линейного целочисленного программирования решается с помощью генетического алгоритма.

1. Сведение задачи кластеризации документов к линейному целочисленному программированию

Предположим, что задана совокупность документов $\mathrm{D} = (D_1, D_2, ..., D_n)$. Пусть m обозначает общее число слов в совокупности документов $\mathrm{D} = (D_1, D_2, ..., D_n)$, m_i – общее число слов в документе D_i, а m_{ij} – число появлений слова j в документе D_i. Тогда частота появления слова j в документе D_i будет определяться согласно формуле:

$$f_{ij} = \frac{m_{ij}}{m},\ i = 1,...,n;\ j = 1,...,m.$$

В общей постановке проблема кластеризации объектов (документов) заключается в том, чтобы всю анализируемую совокупность объектов, статистически представленную в виде матриц, где каждая строка соответствует одному объекту, разбить на сравнительно небольшое число (заранее известное или нет) однородных, в определенном смысле, групп или классов (называемые «кластеры») [1], [2], [14-16].

Для формализации этой проблемы удобно интерпретировать анализируемые объекты (документы) в качестве точек $D_i = (w_{i1},...,w_{im})$ в соответствующем признаковом пространстве, где w_{ij} означает вес слова j в документе D_i. Если исходные данные представлены в форме матрицы, то эти точки являются непосредственным геометрическим изображением многомерных наблюдений в m-мерном пространстве. Естественно предположить, что близость двух или нескольких точек в этом пространстве означает тематическую близость соответствующих объектов (документов), их однородность. Тогда проблема кластеризации состоит в разбиении анализируемой совокупности D точек – документов – на сравнительно небольшое число q классов $\mathrm{C} = (C_1, C_2,..., C_q)$ таким образом, чтобы документы, принадлежащие одному классу, были подобны друг другу. При этом предполагается, что для любого $k = 1,..., q$ $C_k \neq \varnothing$, $C_k \cap C_p = \varnothing$ при $k \neq p$ $(k, p = 1,..., q)$ и, следовательно, будет иметь место $\bigcup_{k=1}^{q} C_q = \mathrm{D}$. Полученные в результате разбиения классы часто называют кластерами (таксонами, образами), а методы их нахождения соответственно кластер-анализом, численной таксономией, распознаванием образов с самообучением.

В задаче кластеризации наиболее трудным и наименее формализованным является момент, связанный с определением понятия однородности объектов.

В общем случае понятие однородности объектов определяется заданием правила определения метрики, характеризующей либо расстояние между объектами из исследуемой совокупности, либо степень близости (сходства) тех же объектов.

Выбор метрики (или меры близости) между объектами, каждый из которых представлен значениями характеризующего его многомерного признака, от которого решающим образом зависит окончательный вариант разбиения объектов на классы, при любом используемом для этого алгоритме разбиения.

В каждой конкретной задаче этот выбор должен производиться по-своему, в зависимости от главных целей исследования.

Прежде чем перейти к выбору метрики, определим вес слова w_{ij}. Слово j имеет и локальный, и глобальный вес в документе D_i. Общий вес слова j в документе D_i равняется произведению его локального w_{ij}^{local} и глобального w_{ij}^{global} весов [17]:

$$w_{ij} = w_{ij}^{local} \cdot w_{ij}^{global}, \quad i = 1,..., n; \; j = 1,..., m, \tag{1}$$

где локальный вес определяется формулой:

$$w_{ij}^{local} = \log\left(1 + \frac{m_{ij}}{m_i}\right), \tag{2}$$

а глобальный вес слова j в документе D_i определяется таким образом:

$$w_{ij}^{global} = 1 - E_j. \tag{3}$$

Через E_j обозначена нормализованная энтропия слова j, которая вычисляется:

$$E_j = -\frac{1}{\log(n)} \sum_{i=1}^{n} \frac{f_{ij}}{m_i} \log\left(\frac{f_{ij}}{m_i}\right). \quad (4)$$

В последней формуле во избежание смещения, вызванного длиной (количество слов) документа, функция f_{ij} нормализована относительно длины m_i документа D_i.

Если учесть формулы (2) – (4) в (1), то получится окончательное выражение для вычисления веса слова j в документе D_i:

$$w_{ij} = \left(1 + \frac{1}{\log(n)} \sum_{i=1}^{n} \frac{f_{ij}}{m_i} \log\left(\frac{f_{ij}}{m_i}\right)\right) \log\left(1 + \frac{m_{ij}}{m_i}\right).$$

В задаче кластеризации одним из центральных вопросов является выбор метрики, определяющий смысловую близость двух документов. Результаты работы [18] показывают, что при определении смысловой близости документов целесообразно выбрать метрику косинуса. Согласно этой метрике мера подобия документов D_i и D_l вычисляется по формуле:

$$sim_{il} = sim(D_i, D_l) = \cos(D_i, D_l) = \frac{\sum_{j=1}^{m} w_{ij} w_{lj}}{\sqrt{\sum_{j=1}^{m} w_{ij}^2} \sqrt{\sum_{j=1}^{m} w_{lj}^2}}, \quad i, l = 1, 2, ..., n. \quad (5)$$

Математическая постановка задачи кластеризации требует формализацию понятия «качество разбиения». С этой целью в рассмотрение вводится понятие критерия (функционала) качества разбиения, который задает способ сопоставления с каждым возможным разбиением $\mathrm{C} = (C_1, C_2, ..., C_q)$ заданной совокупности документов $\mathrm{D} = (D_1, D_2, ..., D_n)$ на классы некоторого числа q, оценивающего степень оптимальности данного разбиения. Тогда задача поиска лучшего разбиения сводится к решению оптимизационной задачи.

В практике выбор функционала качества разбиения обычно осуществляется произвольно, опирается скорее на эмпирические и профессионально-интуитивные соображения, чем на какую-либо точную формализованную схему. Однако ряд распространенных в практике функционалов качества удается постфактум обосновать и осмыслить в рамках строгих математических моделей.

Выбор того или иного вида функционала качества в какой-то мере определяет класс разбиения, в котором следует искать оптимальное разбиение.

Из определения (5) следует, что функционал качества разбиения должен выбираться таким образом, чтобы он на каждом кластере обеспечивал максимизации суммарной меры подобия между документами. Следовательно, на каждом кластере C_k необходимо обеспечить максимизацию суммы:

$$\sum_{D_i, D_l \in C_k} sim_{il} \to \max, \quad (6)$$

где $k = 1,..., q$; $i, l = 1,..., n$.

Пусть x_{ilk} переменные, равные единице, если документы D_i и D_l относятся к кластеру C_k , и равные нулю, в противном случае:

$$x_{ilk} = \begin{cases} 1, & \text{если } (D_i, D_l) \in C_k \\ 0, & \text{в противном случае} \end{cases}.$$

Переменные x_{ilk} симметричны относительно индексов i и l , $x_{ilk} = x_{lik}$.

С учетом последнего обозначения формула (6) записывается в таком виде:

$$\sum_{k=1}^{q}\sum_{i=1}^{n}\sum_{l=1}^{n} sim_{il} x_{ilk} \to \max . \qquad (7)$$

Поскольку переменные x_{ilk} симметричны относительно индексов i и l, то формулу (7) можно переписать так:

$$\sum_{k=1}^{q}\sum_{i=1}^{n-1}\sum_{l=i+1}^{n} sim_{il} x_{ilk} \to \max . \qquad (8)$$

Такая формулировка позволяет сократить число операций на количество $\frac{n(n+1)}{2}$.

Из предположения $C_k \text{ I } C_p = \varnothing,\ k \neq p$ вытекает, что переменные x_{ilk} должны подчиняться следующим ограничениям:

$$\sum_{k=1}^{q} x_{ilk} \le 1 \quad \text{для любой пары } (i, l), \qquad (9)$$

где $i = 1,..., n-1$, $l = i+1,..., n$.

С другой стороны, каждый кластер должен содержать хотя бы один документ и не должен содержать все документы. Следовательно, должно иметь место:

$$1 \le \sum_{i=1}^{n} x_{iik} < n , \quad \text{для любого } k = 1,..., q . \qquad (10)$$

Наконец, согласно определению, для любой тройки (i, l, k):

$$x_{ilk} \in \{0, 1\}, \qquad (11)$$

где $i = 1,..., n-1,\ \ l = i+1,..., n,\ \ k = 1,..., q$.

Таким образом, проблема кластеризации документов сведена к задаче линейного целочисленного программирования с бинарными переменными (8) – (11).

2. Индекс оценки и определение количества кластеров

В задачах кластеризации наиболее важным индикатором структуры является количество кластеров [9], [11], [19]. Большинство методов кластеризации предполагает, что число кластеров заранее задается пользователем. Число кластеров непосредственно связано со сложностью структуры кластера. Для оценки структуры кластера (результата

кластеризации) вводится мера, называемая индексом оценки. В литературе [1], [9], [11], [12], [20], [21] предложено достаточное количество методов для оценки кластера. Чтобы оценить результат кластеризации предлагается следующий индекс:

$$V_R(q) = \sum_{k=1}^{q} n_k \, \overline{sim}(C_k),$$

где n_k – число точек (документов) в кластере C_k, ясно, что $\sum_{k=1}^{q} n_k = n$, а $\overline{sim}(C_k)$ – средняя мера подобия документов в кластере C_k, которая определяется так:

$$\overline{sim}(C_k) = \frac{2}{n_k(n_k - 1)} \sum_{\substack{D_i, D_l \in C_k \\ i \neq l}} sim_{il}.$$

При $n_k = 1$ принимается, что $\overline{sim}(C_k) = 1$.

С учетом последнего выражения формула для оценки результата кластеризации $V_R(q)$ будет принимать следующий вид:

$$V_R(q) = \sum_{k=1}^{q} \frac{2}{(n_k - 1)} \sum_{\substack{D_i, D_l \in C_k \\ i \neq l}} sim_{il} = \sum_{k=1}^{q} \frac{2}{(n_k - 1)} \sum_{i=1}^{n-1} \sum_{l=i+1}^{n} sim_{il} x_{ilk}. \qquad (12)$$

Нетрудно показать, что $V_R(q)$ является монотонно неубывающей функцией относительно параметра q, т.е. $V_R(q+1) \geq V_R(q)$. Пусть ε – заданная точность. Если для некоторого числа q будет выполняться условие

$$\frac{V_R(q+1) - V_R(q)}{V_R(q+1)} < \varepsilon,$$

то оно принимается за оптимальное количество кластеров.

Задача (8) – (11) относится к задачам комбинаторной оптимизации. Проблема комбинаторной оптимизации – одна из важнейших проблем современной вычислительной и прикладной математики. Среди различных методов решения задач оптимизации большое распространение получили поисковые методы оптимизации, особенно методы генетического поиска. Данный факт объясняется тем, что создаваемые генетические алгоритмы универсальны, просты для реализации и оказались достаточно эффективными для решения задач различного характера [11], [22-25]. Исходя из этого соображения, в следующем разделе предложен генетический алгоритм для решения задачи (8) – (11).

3. Генетический алгоритм решения задачи (8) – (11)

Генетический алгоритм относится к адаптивным методам поиска, который в последнее время часто используется для решения многих задач оптимизации. Генетический алгоритм использует принципы естественного отбора для нахождения решения задачи, который состоит из следующих этапов [23], [24]: 1) кодирование решения в виде хромосомы; 2) создание начальной популяции; 3) вычисление функций полезности (fitness – значение) для хромосом популяции (оценивание); 4) выбор хромосом из текущей популяции (селекция); 5) скрещивание; 6) мутация; 7) критерий завершения.

Кодирование решения. Хромосома кодируется в виде строки $X = [k_1, k_2, ..., k_n]$, где длина n равна числу документов. Значения генов k_i означают номера кластеров, которые принимают значения из множества $k_i \in \{1, 2, ..., q\}$, $i = 1, ..., n$, а позиция генов соответствует номерам документов [11], [25]. Решение задачи декодируется таким образом: если $k_i = k_l$, то $x_{ilk_i} = 1$, и если $k_i \neq k_l$, то $x_{ilk_i} = 0$, $x_{ilk_l} = 0$.

Создание, оценка разновидности и очистка начальной популяции. Случайно инициализируется определенное количество хромосом (начальная популяция) $\mathbf{P} = [X_s]$, $s = 1, ..., S$, S – размер (число хромосом) начальной популяции.

Как известно, скорость сходимости генетических алгоритмов и оптимальность найденного решения непосредственно связаны с разновидностью начальной популяции.

Пусть S_{ik} означает количество появлений числа k в позиции i, тогда $\omega_{ik} = \frac{S_{ik}}{S}$ будет означать частоту появления числа k в позиции i, $i = 1, ..., n$, $k = 1, ..., q$. Тогда разновидность популяции будет определяться формулой:

$$DIVER_1(\mathbf{P}) = -\frac{1}{\log(n)} \sum_{i=1}^{n} \sum_{k=1}^{q} \omega_{ik} \log(\omega_{ik}), \tag{13}$$

где полагается, что $0 \log 0 = 0$.

Разновидность популяции по-другому можно определить с помощью следующей формулы:

$$DIVER_2(\mathbf{P}) = \frac{1}{S(S-1)} \sum_{s=1}^{S-1} \sum_{t=s+1}^{S} H(X_s, X_t), \tag{14}$$

где величина $H(X_s, X_t)$ означает расстояние Хемминга между хромосомами X_s и X_t:

$$H(X_s, X_t) = \sum_{i=1}^{n} I_i .$$

Если в хромосомах X_s и X_t значения генов i-ой позиции совпадают, то величина I_i принимает значение, равное нулю, в противном случае она принимает значение, равное единице. Следовательно, $H(X_s, X_t) \in [0, n]$.

Пусть $\mathsf{N}_s(k)$ означает множество локусов в хромосоме X_s, значения генов которых равны числу k, $s = 1, ..., S$, $k = 1, ..., q$. Например, для хромосомы $X_s = [1, 2, 1, 3, 4, 1, 2, 4]$ $\mathsf{N}_s(1) = \{1, 3, 6\}$, $\mathsf{N}_s(2) = \{2, 7\}$, $\mathsf{N}_s(3) = \{4\}$, $\mathsf{N}_s(1) = \{5, 8\}$. Тогда отклонение (расстояние) между хромосомами X_s и X_t вычисляется таким образом:

$$d(X_s, X_t) = \sum_{k=1}^{q} \sum_{N_s \in \mathsf{N}_s(k)} \sum_{N_t \in \mathsf{N}_t(k)} |N_s - N_t| .$$

С учетом последней формулы для определения степени разновидности популяции вводится следующая мера:

$$Diver_3(\mathrm{P}) = \frac{1}{S(S-1)} \sum_{s=1}^{S-1} \sum_{t=s+1}^{S} d(X_s, X_t). \tag{15}$$

Поскольку начальная популяция инициализируется случайно, то нет гарантии, что в ней не будут появляться недопустимые хромосомы. Хромосомы, нарушающие одно из условий задачи (8) – (11), называются недопустимыми. Недопустимые хромосомы влияют на эффективность генетических алгоритмов. Следовательно, есть необходимость очищения начальной популяции от недопустимых хромосом. Согласно формуле (13) если для любого i $\omega_{ik} = 0$, то это будет означать, что все хромосомы в популяции недопустимы. Недопустимые хромосомы можно определить и с помощью величины n_{ks}, которая равна количеству появления числа k в хромосоме X_s. Если для некоторого числа k $n_{ks} = 0$, то s-я хромосома недопустима.

Отметим, что для повышения эффективности генетических алгоритмов применяются и другие подходы: метод штрафной функции [11], выбор подходящих операторов скрещивания и мутации [25].

Вычисление функции пригодности. Хромосомы оцениваются с использованием функции пригодности, в результате каждой из них присваивается определенное значение (fitness-значение). Fitness-значение определяет вероятность выживания хромосомы. В качестве функции пригодности берется целевая функция

$$F(x) = \sum_{k=1}^{q} \sum_{i=1}^{n-1} \sum_{l=i+1}^{n} sim_{il} x_{ilk} \quad \text{задачи (8) – (11).}$$

С использованием полученных fitness-значений выбираются хромосомы (селекция), допущенные к скрещиванию.

Селекция. Цель оператора селекции – выбрать из популяции хромосому для выполнения скрещивания, в результате которого из начальной популяции создается новая популяция, где присутствие той или иной хромосомы определяется ее fitness-значением. Используется селекция с линейным упорядочиванием. Хромосомы упорядочиваются в порядке возрастания относительно их fitness-значений, где наибольший ранг S присваивается лучшему решению, а ранг 1 – худшему. Вероятность выбора каждой хромосомы осуществляется согласно формуле [26]:

$$P_s = \frac{1}{S}\left(P^- + (P^+ - P^-)\frac{s-1}{S-1} \right), \; s = 1,..., S,$$

где $\frac{P^-}{S}$ – вероятность выбора худшего решения, а $\frac{P^+}{S}$ – вероятность выбора лучшего решения в популяции. Налагаются условия, что $0 \le P^- < 1$ и $P^+ = 2 - P^-$. При такой селекции хромосома, имеющая лучший ранг, будет иметь наибольший шанс на выживание.

Скрещивание. В работе [25] было показано, что при таком кодировании результат оператора скрещивания PMX (Partially Mapped Crossover) не нарушает структуры хромосомы. Другими словами, если хромосома допустима, то после воздействия оператора PMX не появляется недопустимая хромосома.

Мутация. Оператор мутации помогает выйти из локального экстремума. Пусть выбрана хромосома X_{s*}. Значение гена, соответствующее максимальному значению $n_{k_{\max} s*}$, $k_{\max} = \max\limits_{k} (n_{ks*})$, заменяется значением гена, соответствующим минимальному значению $n_{k_{\min} s*}$, $k_{\min} = \min\limits_{k} (n_{ks*})$. Такому условию могут удовлетворять

несколько чисел. В этом случае одно из этих чисел выбирается произвольно. Потом вычисляется fitness-значение новой хромосомы, и сравнивается со старым fitness-значением. Если новое fitness-значение является лучше, чем старое fitness-значение, то старая хромосома заменяется новой хромосомой, и если наоборот, то в популяции сохраняется старая хромосома.

Критерий завершения. Если за $r*$ итераций разница между двумя лучшими fitness значениями не превосходит заданный порог δ, $\left|F_{r+r*}^{best} - F_r^{best}\right| < \delta$, то процесс эволюции прекращается, где F_r^{best} – лучшее fitness-значение, найденное за r итераций.

Заключение

Настоящая работа посвящена проблеме кластеризации текстовых документов, где предложена новая математическая формулировка. Такая формулировка гарантирует однородность – смысловую близость документов в кластерах – и позволяет существенно сократить количества операций и ограничений. Предложенный подход описывается в виде задачи линейного целочисленного программирования. В задачах кластеризации основная трудность заключается в оценке результата кластеризации и определении количества кластеров. Приводится индекс оценки результата кластеризации, оценивающий степень гомогенности кластеров. Несмотря на то, что для кластеризации данных были предложены различные методы и алгоритмы, главной трудностью в задачах кластеризации остается определение оптимального количества кластеров. В работе на основе введенного индекса оценки (12) предлагается итерационный алгоритм определения количества кластеров.

Чтобы обеспечить практичность предлагаемого подхода, описывается генетический алгоритм для решения задачи (8) – (11). Эффективность работы генетических алгоритмов непосредственно зависит от разновидности популяции. С этой целью в предлагаемой работе для оценки степени разновидности популяции приводятся три меры (13), (14) и (15).

Литература

1. Grabmeier J., Rudolph A. Techniques of cluster algorithms in data mining // Data Mining and Knowledge Discovery. – October 2002. – Vol. 6, № 4. – P. 303-360.
2. Xu R., Wunsch D. Survey of clustering algorithms // IEEE Transactions on Neural Networks. – May 2005. – Vol. 16, № 3. – P. 645-678.
3. Hammouda K.M., Kamel M.S. Efficient phrase-based document indexing for web document clustering // IEEE Transactions on Knowledge and Data Engineering. – October 2004. – Vol. 16, № 10. – P. 1279-1296.
4. Runkler T.A., Bezdek J.C. Web mining with relational clustering // International Journal of Approximate Reasoning. – February 2003. – Vol. 32, №. 2-3. – P. 217-236.
5. Rodrigues E.M., Sacks L. A scalable hierarchical fuzzy clustering algorithm for text mining // Proceedings of the 5th International Conf. on Recent Advances in Soft Computing. – Nottingham (United Kingdom). – December 16-18, 2004. – P. 269-274.
6. Allan J. (ed.) Topic detection and tracking: event-based information organization // Kluwer Academic Publishers. – 2002. – 280 p.
7. Jin H., Wong M.-L., Leung K.-S. Scalable model-based clustering for large databases based on data summarization // IEEE Transactions on Pattern Analysis and Machine Intelligence. – November 2005. – Vol. 27, №. 11. – P. 1710-1719.
8. Hu P., He T., Ji D., Wang M. A study of Chinese text summarization using adaptive clustering of paragraphs // Proc. of the 4th International Conf. on Computer and Information Technology (CIT'04). – Wuhan (China). – September 14-16, 2004. – P. 1159-1164.

9. Алгулиев Р.М., Алыгулиев Р.М., Багиров А.М. Глобальная оптимизация в резюмировании текстовых документов //Автоматика и вычислительная техника. – 2005. – Vol. 39, № 6. – С. 52-59.
10. Radev D., Otterbacher J., Winkel A., Blair-Goldensohn S. NewsInEssence: summarizing online news topics // Communications of the ACM. – October 2005. – Vol. 48, № 10. – P. 95-98.
11. Алгулиев Р.М., Алыгулиев Р.М. Быстрый генетический алгоритм решения задачи кластеризации текстовых документов // Искусственный интеллект. – 2005. – № 3. – С. 698-707.
12. Khan M.S., Khor S.W. Web document clustering using a hybrid neural network // Applied Soft Computing. – September 2004. – Vol. 4, № 4. – P. 423-432.
13. Desai M., Spink A. An algorithm to cluster documents based on relevance // Information Processing and Management. – September 2005. – Vol. 41, № 5. – P. 1035-1049.
14. Chen Y., Bi J. Clustering by maximizing sum-of-squared separation distance // Proc. of the 5th SIAM International Conf. on Data Mining.– California (USA). – April 21-23, 2005. – P. 1-12.
15. Li X., Ye N. A supervised clustering and classification algorithm for mining data with mixed variables // IEEE Transactions on Systems, Man, and Cybernetics – Part A. – March 2006. – Vol. 36, № 2. – P. 396-406.
16. Li T. A unified view on clustering binary data // Machine Learning. – March 2006. – Vol. 62, № 3. – P. 199-215.
17. Yeh J-Y., Ke H-R., Yang W-P., Meng I.-H. Text summarization using a trainable summarizer and latent semantic analysis // Information Processing and Management. – 2005. – Vol. 41, № 1. – P. 75-95.
18. Alguliev R.M., Aliguliyev R.M. Effective summarization method of text documents // Proc. of the 2005 IEEE/WIC/ACM International Conf. on Web Intelligence (WI'05). – France. – September 19-22, 2005. – Compiegne University of Technology. – P. 264-271.
19. Salvador S., Chan P. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms // Proc. of the 16th IEEE International Conf. on Tools with Artificial Intelligence (ICTAI 2004). – Boca Raton (USA) – November 15-17, 2004. – P. 576-584.
20. Halkidi M., Batistakis Y., Vazirgiannis M. Cluster validity methods: part I // ACM SIGMOD Record. – June 2002. – Vol. 31, № 2. – P. 40-45.
21. Halkidi M., Batistakis Y., Vazirgiannis M. Cluster validity methods: part II // ACM SIGMOD Record. – September 2002. – Vol. 31, № 3. – P. 19-27.
22. Laszlo M., Mukherjee S. A Genetic algorithm using hyper-quadtrees for low-dimensional k-means clustering // IEEE Transactions on Pattern Analysis and Machine Intelligence. – April 2006. – Vol. 28, № 4. – P. 533-543.
23. Michalewicz Z. Genetic algorithms + data structures = evolution programs. – Berlin: Springer-Verlag, 1996. – 387 p.
24. Kureichik V.M. Genetic algorithms: state of the art, problems and perspectives // Journal of Computer and Systems Sciences International. – 1999. – Vol, 38. № 1. – P. 137-152.
25. Алгулиев Р.М., Алыгулиев Р.М. Генетический подход к оптимальному назначению заданий в распределенной системе // Искусственный интеллект. – 2004. – № 4. – С. 79-88.
26. Кисляков А.В. Генетические алгоритмы: математический анализ некоторых схем репродукции // Информационные технологии. – 2000. – № 12. – С. 9-14.

Р.М. Алигулієв

Метод кластеризації колекції документів та алгоритм для оцінки оптимального числа класів

Формалізована задача оптимального розбиття колекцій документів. Критерієм якості (оптимальність) розбиття є максимізація міри подібності між документами у кластерах. Пропонований підхід зведений до задачі лінійного цілочислового програмування (ЛЦП) з бінарними змінами. Наведений генетичний алгоритм рішення задачі ЛЦП.

R.M. Aliguliyev

A Clustering method of Document Collections and Algorithm for Estimation the Optimal Number of Clusters

The optimum partitioning problem of document collections is formalized. Criterion of quality (optimality) of partitioning is maximization of a similarity measure between documents within cluster. The offered approach is reduced to a linear integer programming (LIP) problem with binary variables. The genetic algorithm for solving of the LIP problem is resulted.

Статья поступила в редакцию 14.06.2006.