УДК 025.4:004.421

Ю.А. Радионова

МЕТОД ПОСТРОЕНИЯ ОЦЕНОЧНОЙ ФУНКЦИИ, ОПРЕДЕЛЯЮЩЕЙ ЭФФЕКТИВНОСТЬ АЛГОРИТМОВ АВТОМАТИЧЕСКОЙ КЛАСТЕРИЗАЦИИ

***Радионова Юлия Александровна**, аспирант кафедры «Информационные системы» Ульяновского государственного технического университета, окончила механико-математический факультет Ульяновского государственного университета. Ведущий инженер-программист ФНПЦ ОАО «НПО «Марс». Сфера научных интересов — автоматизированные системы документооборота, организация хранилищ технической документации. E-mail: julia-owl@mail.ru*

Аннотация

В статье представлено описание метода построения оценочной функции, дающей представление об эффективности использования автоматических кластеризаторов для построения интеллектуального репозитария электронных документов.

Abstract

The article deals with description of criterion function method providing insight into efficiency of automatic clustering to create intelligent e-document repository.

Введение

Для организации интеллектуального репозитария электронной документации необходимо, прежде всего, создать систему разбиений документов по тематическим группам — классам. На основе материалов архива машинных носителей ФНПЦ ОАО «НПО «Марс» был разработан алгоритм экспертной классификации архива электронных документов, представляющий собой жесткую иерархию по видам, классам, разделам документации и тематике работ на предприятии.

При проведении экспертной классификации возникает ряд проблем: резкое возрастание затрат времени на процесс классификации при увеличении количества документов, сильное влияние на результат субъективного фактора.

Возникла необходимость исследования возможности применения автоматических методов классификации, позволяющих разбивать массив документации на однородные классы (кластеры) без участия эксперта.

В настоящее время существуют методы автоматической кластеризации массива документов, основанные на индексировании текстов документов и использовании нейронных сетей для кластеризации проиндексированных документов. Осуществление поиска в массиве документов, разбитом по тематическим группам — кластерам, значительно облегчает поиск необходимых документов. Но насколько приемлем подобный подход для технической документации проектной организации? Техническая и конструкторская документация имеет более жесткую структуру по сравнению с произвольными текстами и ряд особенностей, что может затруднить поиск документа только по индексированным текстам. Для индексирования текстовых документов разработано программное приложение «Индексатор», для последующей кластеризации — программные приложения, осуществляющие процесс кластеризации по индексированным текстам с применением нейронных сетей.

Для оценки эффективности алгоритмов автоматической кластеризации необходимо сделать вывод о том, насколько близко разбиение массива документации в результате кластеризации к разбиению этого же массива, полученному в результате экспертной классификации. Необходимо объединить в единую структуру разнородные данные различных кластеризаций и разработать алгоритм построения оценочной функции результатов.

1 Методы классификаций

1.1 Экспертная классификация электронной документации

Первоначальным, уникальным и основным реквизитом технического документа является его децимальный номер. Децимальный номер определяется по жесткой схеме, регламентируемой ГОСТ и системой обозначения тематики работ на предприятии. Исходя из этого, метод экспертной классификации был разработан на основе лексического анализа децимального номера документа [4].

Были выделены четыре типа классификации:

- по изделиям или тематике работ;
- по классам документации;
- по видам документации;
- по разделам документации.

Процесс классификации должен проходить автоматизированно с участием оператора - эксперта (для корректировки и управления классификацией). Для каждого типа классификации разработан справочник классов. В случае отсутствия необходимой информации в справочнике оператору предлагается ввести соответствующие обозначения, признаки и наименования.

Таким образом, получаем классификатор с базой знаний, накапливаемой в процессе классификации.

1.2 Кластеризация на основе нейронных сетей

Кластерный анализ является одним из методов анализа данных и представляет собой совокупность методов и процедур, разработанных для решения проблемы формирования однородных классов (кластеров) в произвольной предметной области. Если имеется выборка $X^k = \{x_1 \dots x_k\}$ и функция расстояния между объектами $\rho(x, x')$, то задача кластеризации состоит в разбиении выборки на подмножества, называемые кластерами, так, чтобы каждый кластер состоял из объектов, близких по метрике ρ, а объекты разных кластеров существенно отличались. Алгоритм кластеризации — это функция, которая любому объекту выборки ставит в соответствие метку кластера.

Для оценки эффективности автоматической кластеризации были выбраны два алгоритма — на основе сетей Кохонена и FCM-алгоритм нечеткой кластеризации, реализации которых разработаны аспирантами кафедры «Информационные системы» УлГТУ [2, 5]. Для предварительной индексации текстов используется алгоритм индексации [3].

1.3 Кластеризация методом Кохонена

Формирование системы классов — кластеризация — успешно решается нейронной сетью особого вида — ассоциативными картами Кохонена (Self-Organizing Map). Особенностью данного вида сети является процесс обучения нейросети без учителя (то есть без обучающей выборки) — так называемый алгоритм «победитель получает все». Обучение без учителя заключается в том, что образцы данных сортируются по классам с помощью меры похожести, в качестве которой выступает расстояние между векторами.

На каждом шаге обучения из исходного набора данных случайно выбирается один из векторов, а затем производится поиск наиболее похожего на него вектора коэффициентов нейронов. При этом выбирается нейрон-победитель, который наиболее похож на вектор входов. Под похожестью в данной задаче понимается расстояние между векторами, обычно вычисляемое в евклидовом пространстве.

Финалом работы алгоритма является набор векторов, каждый из которых указывает на центр гравитации кластера.

Для осуществления процесса кластеризации разработано программное приложение, реализующее алгоритм кластеризации Кохонена [5].

1.4 Алгоритм нечеткой кластеризации

Алгоритм нечеткой кластеризации — Fuzzy Classifier Means — называют FCM-алгоритмом. Целью FCM-алгоритма является автоматическая классификация множества объектов, которые задаются векторами признаков в пространстве признаков. Кластеры представляются нечеткими множествами, и границы между кластерами также являются нечеткими. FCM-алгоритм предполагает, что объекты принадлежат всем кластерам с определенной степенью вероятности, которая определяется расстоянием от объекта до соответствующих кластерных центров.

Для осуществления процесса кластеризации разработано программное приложение, реализующее FCM-алгоритм [2].

2 Математическая модель оценки качества кластеризации

Введем следующие обозначения:

$\overline{K}$ - разбиение массива документов, полученное в результате экспертной классификации;

$\hat{K}$ - разбиение того же массива документов, полученное в результате работы алгоритма автоматической кластеризации;

$\overline{K^i}$ - множество документов, входящих в i-й кластер согласно экспертному делению;

$i = \overline{1,n}$ - номер кластера, n — количество кластеров эксперта;

$\hat{K}^j$ - множество документов, входящих в j-й кластер согласно автоматическому разбиению;

$j = \overline{1,l}$ - номер кластера, l — количество кластеров автоматической системы.

Будем считать кластеризацию тем более качественной, чем ближе разбиение $\hat{K}$ к разбиению $\overline{K}$.

Устанавливаем пары $\left\langle \overline{K^i}, \hat{K}^j \right\rangle$ из расчета максимального совпадения элементов множеств $\overline{K^i}$ и $\hat{K}^j$.

Далее необходимо удалить одинаковые элементы из обоих множеств. В результате получаем:

$$\left\langle \overline{K_r^1}, \hat{K}_r^1 \right\rangle, .., \left\langle \overline{K}_r^{\max(n,l)}, \hat{K}_r^{\max(n,l)} \right\rangle,$$

где $\overline{K_r^i}$ и $\hat{K}_r^i$ - редуцированные множества документов экспертной и автоматической класте-

ризации, $i = \overline{1, \max(n,l)}$.

В результате можно получить целевую функцию, формализующую качество кластеризации, используя два критерия: отсутствие документов в кластере (то есть количество документов, которые должны быть в кластере, но отсутствуют в нем, - $\left|\overline{K_r^i}\right|$) и наличие «лишних» документов в кластере $\left|\hat{K}_r^i\right|$:

$$f_i = \alpha \cdot \left|\overline{K_r^i}\right| + (1-\alpha) \cdot \left|\hat{K}_r^i\right|,$$

где $\alpha = \overline{0,1}$ - коэффициент важности критерия;

$i = \overline{1, \max(n,l)}$ - номер кластера.

Для того, чтобы убрать зависимость значения целевой функции от количества кластеров в эксперименте, значение целевой функции нормируем:

$$\bar{f}_i = \frac{\alpha \cdot \left|\overline{K_r^i}\right| + (1-\alpha) \cdot \left|\hat{K}_r^i\right|}{\max\left(\left|\overline{K_r^i}\right|, \left|\hat{K}_r^i\right|\right)}.$$

3 Структуры данных классификации

Для сравнения результатов работы различных алгоритмов необходимо проанализировать полученные структуры баз данных и привести их к единой структуре.

3.1 Структура данных экспертной классификации

Экспертная классификация проводится по четырем признакам, наименования классов для которых накапливаются в процессе классификации и сохраняются в четырех таблицах-справочниках:

- clsDocuments (kod, abbreviatura, name) — виды документации;
- clsIzdelia (kod, dec_number, name) — изделия или тематика работ;
- clsClass_docum (kod, name) — классы документации;
- clsRazdel (kod, kod_zifra, name) — разделы документации.

Результаты экспертной интерактивной классификации сохраняются в виде двух таблиц:

- clsList_docum — список документов, в котором каждый документ определяется его десятичным номером (kod, dec_number);
- clsKlassifikator — классификатор документов, где каждая строка представляет собой наименование классификации (name_klassification), код класса (kod_klass), к которому принадлежит в данной классификации документ, и код документа (kod_docum).

3.2. Структура данных кластеризации Кохонена

Из таблиц базы данных, сформированных алгоритмом автоматической классификации Кохонена, для оценки эффективности интерес представляют следующие три:

- inalR (id, IRName) — список индексированных файлов;
- som_cluster (id, name) — список полученных в ходе эксперимента кластеров;
- som_cluster_res (res_id, cluster_id) — сопоставление списка документов (res_id) списку кластеров (cluster_id)[1].

При инициализации нового процесса кластеризации результирующие таблицы som_cluster и som_cluster_res очищаются, поэтому после каждого эксперимента их необходимо сохранять. Для разделения таблиц различных экспериментов используется номер текущего эксперимента, например, для эксперимента №001 таблицы имеют наименования som_cluster001 и som_cluster_res001.

3.3 Структура данных FCM-кластеризации

Для оценки результатов автоматической классификации FCM-методом необходимо сохранить пять таблиц базы данных:

- inalR — список индексированных файлов (id-идентификатор файла, IRName — наименование файла);
- fcm_experiment — список экспериментов (id — идентификатор эксперимента, description — краткое описание эксперимента);
- fcm_model — список моделей (id — идентификатор модели, experiment_id — идентификатор эксперимента из fcm_experiment);
- fcm_cluster — список полученных в ходе эксперимента кластеров (fcm_model_id — идентификатор модели из fcm_model, id — идентификатор кластера, name — наименование кластера);
- fcm_res_cluster — сопоставление списка документов списку кластеров (res_id — идентификатор документа, cluster_id — идентификатор кластера, grade — степень вероятности принадлежности документа кластеру).

Таблица inalR — результат индексирования массива электронных текстов — одна на все эксперименты (как Кохонена, так и FCM), в таблицах fcm_experiment, fcm_model, fcm_cluster, fcm_res_cluster накапливаются результаты всех экспериментов, проведенных FCM-методом. В поле description таблицы fcm_experiment при проведении экспериментов сохраняем порядковый номер эксперимента для его дальнейшей оценки.

---

1 В скобках указаны только те поля таблиц, которые необходимы для оценки эффективности алгоритма и подсчета значения целевой функции.

3.4 Преобразование данных в единую структуру

На первом этапе выполняется процесс «подгонки» структур таблиц экспертной классификации к структуре таблиц автоматической кластеризации для построения матриц соответствия.

На основе таблицы clsKlassifikator создается таблица resClusterExp.

```
select distinct name_klassification,kod_klass
into resClusterExp
from clsKlassifikator
```

В таблицу добавляется поле id, содержащее уникальный идентификатор записи.

Создается таблица соответствия полученного списка кластеров списку документов.

```
select clsList_docum.kod as kod_docum,
       resClusterExp.id as kod_cluster
into resClusterisatorExp
from clsList_docum,clsKlassifikator,
     resClusterExp
where (clsList_docum.kod=clsKlassifikator.
     kod_docum) and (clsKlassifikator.
     name_klassification=
     resClusterExp.name_klassification)
and (clsKlassifikator.kod_klass=
     resClusterExp.kod_klass)
```

Таблица inalR сохраняется как таблица resList_docum, в которую добавляется поле dec_number, заполняемое десятичным номером документа, выделенным из наименования файла и приведенным к стандартному «архивному» виду.

Так как списки документов для экспертной классификации и автоматической кластеризации формируются разными программными модулями, идентификационный код одного и того же документа в разных списках будет разным. Поэтому для сравнения результатов формируется также таблица resDocuments (kod_exp — код документа экспертной классификации, kod_avt — код этого же документа в автоматической кластеризации).

```
select kod as kod_exp,id as kod_avt
into resDocuments
from clsList_docum,resList_docum
where clsList_docum.dec_number=
     resList_docum.dec_number
```

3.4.1 Преобразование структуры данных экспериментов Кохонена

Таблицы som_clusterNNN и som_cluster_resNNN сохраняются под именами resClusterNNN и resClusterisatorNNN, где NNN номер текущего эксперимента.

```
1) create table resClusterNNN (id Integer,
name varchar(100))
2) insert into resClusterNNN (id,name) select
id, name from som_clusterNNN
3) create table resClusterisatorNNN (kod_
docum Integer, kod_cluster Integer)
4) insert into resClusterisatorNNN (kod_
docum, kod_cluster) select res_id, cluster_id
from som_cluster_resNNN
```

3.4.2 Преобразование структуры данных FCM-экспериментов

Особенностью данного алгоритма является то, что каждый документ принадлежит каждому кластеру с определенной степенью вероятности. В экспертной классификации каждый документ принадлежит только одному кластеру. Поэтому в FCM-кластеризации будем считать, что документ принадлежит только тому кластеру, которому он принадлежит с максимальной степенью вероятности.

Сначала формируем список кластеров, принадлежащих эксперименту NNN.

```
select id,name
into resFcmClusterNNN
from fcm_cluster
where fcm_model_id=
     (select id from fcm_model
     where experiment_id=
          (select id from fcm_experiment
           where description like 'NNN'))
```

Затем - список соответствия документов кластерам для данного эксперимента.

```
select *
into fcm_res_clusterNNN
from fcm_res_cluster
where (cluster_id in
     (select id from resFcmClusterNNN))
```

Далее для эксперимента NNN определяем список документов с максимальной степенью вероятности.

```
select res_id,max(grade) as max_grade
into max_grade_tmp
from fcm_res_clusterNNN
group by res_id
```

И на основе этого списка определяем список соответствия документов кластерам для эксперимента NNN.

```
select max_grade.res_id as kod_docum,
     fcm_res_clusterNNN.cluster_id as
     kod_cluster
into resFcmClusterisatorNNN
from max_grade
```

```
inner join fcm_res_clusterNNN
on max_grade.res_id =
    fcm_res_clusterNNN.res_id
and max_grade.max_grade =
    fcm_res_clusterNNN.grade
```

4 Алгоритм построения оценочной функции

Таким образом, получаем таблицы:

- общие для всех экспериментов:

resDocuments (kod_exp, kod_avt) — соответствие кодов документов экспертной классификации и автоматической кластеризации;

- экспертной классификации:

resClusterExp (id, name_klassification, kod_klass) — список классов;

resClusterisatorExp (kod_docum, kod_cluster) — соответствие документов классам;

- кластеризации Кохонена:

resClusterNNN (id, name) — список кластеров;

resClusterisatorNNN (kod_docum, kod_cluster) — соответствие документов кластерам;

- FCM-кластеризации:

resFcmClusterNNN (id, name) — список кластеров;

resFcmClusterisatorNNN (kod_docum, kod_cluster) — соответствие документов кластерам.

Далее формируются таблицы — матрицы соответствий (NNN — номер эксперимента):

Типы экспертной классификации	Кластеризация Кохонена (матрицы)	FCM-кластеризация (матрицы)
классы документации	resMatrixClass_documNNN	resFcmMatrixClass_documNNN
виды документации	resMatrixDocumentsNNN	resFcmMatrixDocumentsNNN
тематика работ	resMatrixIzdeliaNNN	resFcmMatrixIzdeliaNNN
разделы документации	resMatrixRazdelNNN	resFcmMatrixRazdelNNN

Структура матриц соответствия определяется следующими полями:

- kod_exp — код кластера экспертной классификации;
- kod_avt — код кластера автоматической классификации;
- count_eq — количество документов, принадлежащих обоим кластерам;
- exp_all — количество документов, принадлежащих кластеру экспертной классификации;
- avt_all — количество документов, принадлежащих кластеру автоматической классификации.

Каждая строка матрицы формируется для кластера экспертной кластеризации N (кластеры выбираются из resClusterExp для соответствующего типа классификации) и кластера автоматической кластеризации M (кластеры выбираются из resClusterNNN или resFcmClusterNNN). Количество одинаковых документов, попавших в класс N экспертной классификации и кластер M автоматической кластеризации, определяется как количество строк запроса.

```
select distinct
        resClusterisatorExp.kod_docum
from resClusterisatorExp
INNER JOIN resDocuments ON
resClusterisatorExp.kod_docum=
      resDocuments.kod_exp
CROSS JOIN resClusterisatorNNN
WHERE (resClusterisatorExp.kod_cluster=N)
AND (resDocuments.kod_avt IN
(select kod_docum from
                resClusterisatorNNN
where resClusterisatorNNN.kod_cluster=M))
```

Дополнительно в текущую строку матрицы добавляется общее количество документов, принадлежащих кластеру N экспертной кластеризации и кластеру M автоматической кластеризации.

```
select count(kod_docum) as count1
from resClusterisatorExp
where kod_cluster=N
select count(kod_docum) as count2
from resClusterisatorNNN
where kod_cluster=M
```

Текст sql-запросов приведен для случая кластеризации Кохонена. Для FCM-кластеризации таблица resClusterisatorNNN заменяется на resFcmClusterisatorNNN.

По сформированным матрицам соответствия вычисляется значение оценочной функции для каждого типа экспертной классификации и каждого эксперимента автоматической кластеризации:

```
select sum((0.5*(exp_all-count_eq)
+0.5*(avt_all-count_eq))/(exp_all*avt_all))
as res_function
from resMatrix<name_klassif><NNN>,
```

где name_klassif — тип экспертной классификации;

NNN — номер эксперимента кластеризации Кохонена;

```
select sum((0.5*(exp_all-count_eq)
+0.5*(avt_all-count_eq))/(exp_all*avt_all))
as res_function
```

from resFcmMatrix<name_klassif><NNN>, где name_klassif — тип экспертной классификации;

NNN — номер эксперимента FCM-кластеризации.

В данном случае коэффициент важности критерия выбран 0,5, а в качестве нормирующего коэффициента вместо значения, которое выбирается как максимум из количества документов экспертного класса и количества документов кластера автоматического алгоритма, берется произведение обоих количеств. Также для получения значения целевой функции для эксперимента в целом значения целевой функции для каждой пары класс — кластер суммируются.

5 План экспериментов по оценке алгоритмов автоматической кластеризации

Для исследования эффективности методов кластеризации с использованием нейронных сетей разработан план экспериментов, включающих классификацию методом архивариуса-эксперта и кластеризацию методами Кохонена и FCM.

Для осуществления первого этапа экспериментов в архиве электронной документации ФНПЦ ОАО «НПО «Марс» планируется подобрать небольшой комплект документации, содержащий документы преимущественно организационно-нормативного характера.

Комплект документации классифицируется архивариусом-экспертом по алгоритму, приведенному в данной работе, при этом должно происходить накапливание базы знаний по различным типам классификации.

Далее комплект документации индексируется с помощью разработанного программного приложения «Индексатор» [3]. Результаты индексирования должны быть сохранены в базе данных для последующей кластеризации.

Проиндексированные данные кластеризуются с помощью алгоритмов сетей Кохонена и FCM-метода с различными параметрами. Полученные результаты кластеризации сравниваются с результатами экспертной классификации с вычислением значений оценочной функции. Выделяются наборы параметров, значение оценочной функции для которых является наилучшим.

На следующем этапе экспериментов в архиве делается подборка большего количества документов, которая классифицируется экспертом с использованием базы знаний, накопленной при проведении классификации первого этапа.

Документация индексируется и кластеризуется с параметрами, дающими оптимальные значения оценочной функции на первом этапе экспериментов. Вычисляются значения оценочной функции.

По вычисленным значениям оценочной функции делается вывод об эффективности использования каждого алгоритма кластеризации для построения интеллектуального репозитария и наиболее приемлемых параметрах кластеризации.

Заключение

Таким образом, для оценки возможности применения алгоритмов автоматической кластеризации при организации поисковой среды архива технической документации выбраны алгоритмы кластеризации с помощью нейронных сетей.

Построена математическая модель функции, позволяющей оценить эффективность алгоритмов автоматической кластеризации.

Разработан метод сравнения данных, полученных в результате работы различных алгоритмов, построения матриц соответствия и вычисления значений оценочной функции.

Разработан план экспериментов для оценки эффективности применения данных алгоритмов для кластеризации массива электронных документов организационно-нормативного, конструкторского и программного содержания.

Для окончательной оценки эффективности алгоритмов автоматической кластеризации поставлена задача проведения экспериментов кластеризации с подбором параметров и создания программного приложения, обеспечивающего реализацию методов построения матриц соответствия и вычисления значений оценочной функции. Одной из функций приложения должна быть возможность варьирования коэффициента важности критерия и метода нормализации целевой функции.

СПИСОК ЛИТЕРАТУРЫ

1. Н.Г. Ярушкина. Основы теории нечетких и гибридных систем. — М.: Финансы и статистика, 2004. — 320 с.

2. А.А. Островский, Ю.А. Радионова. Кластеризация набора электронных информационных ресурсов // Автоматизация процессов управления. — 2008. - №1(11). — С. 101-104.

3. А.Г. Селяев. Взвешивание терминов в процессах индексирования электронных информационных ресурсов // Автоматизация процессов управления. — 2007. - №2(10). — С.93-96.

4. Ю.А. Радионова, В.Г. Тронин. Классификация технической документации на основе лексического анализа децимального номера // Автоматизация процессов управления. — 2008. - №3 (13). — С. 69-72.

5. Н.В. Корунова. Кластеризация документов проектного репозитария на основе нейронной сети Кохонена // Программные продукты и системы. — 2008. - № 4(84). — С. 60.