РОЗДІЛ IV. ІНФОРМАЦІЙНО-КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

УДК 004.052:004.275

В.В. Литвинов, д-р техн. наук
О.П. Мойсеенко, ассистент
Черниговский национальный технологический университет, г. Чернигов, Украина

SVM ПРИ КЛАССИФИКАЦИИ МУЛЬТИЯЗЫЧНЫХ ТЕКСТОВ

В.В. Литвинов, д-р техн. наук
О.П. Мойсеєнко, асистент
Чернігівський національний технологічний університет, м. Чернігів, Україна

SVM ПРИ КЛАСИФІКАЦІЇ РІЗНОМОВНИХ ТЕКСТІВ

Vitaliy Litvinov, Doctor of Technical Sciences
Oleg Moyseyenko, assistant
Chernigov National Technological University, Chernigov, Ukraine

SVM IN THE CLASSIFICATION OF MULTILINGUAL TEXTS

Описывается возможность модернизации базового алгоритма опорных векторов для решения задачи классификации коллекций разноязычных текстовых документов.

***Ключевые слова**: SVM, классификация, обработка текстовых документов.*

Описано можливість модернізації базового алгоритму опорних векторів для вирішення задачі класифікації колекцій різномовних текстових документів.

***Ключові слова:** SVM, класифікація, оброблення текстових документів.*

Describes the ability to upgrade the basic algorithm of support vectors to solve the problem of classification of collections of multilingual text documents.

***Key words**: SVM, classification, word processing documents.*

Введение. Метод опорных векторов SVM (Supporting Vector Machines) [1] относится к числу современных и успешных методов решения задач:

– классификации – отнесение к классам с заданными свойствами/параметрами;

– рубрицирования – сопоставление с иерархической системой классов;

– кластеризации – создание подмножеств близких тематически данных.

В настоящее время одной из проблем, возникающих при решении упомянутых задач с помощью SVM, является необходимость адаптации метода под конкретные цели, в нашем случае для автоматизации аналитической обработки динамичных коллекций разнородных, мультиязычных, естественно-языковых текстовых документов. Возникает необходимость изменения стандартного набора внутренних параметров алгоритма, т. е. таких параметров, которые задает сам пользователь и не изменяемых при обучении [1]. А также разработка новых программных механизмов и подходов для модернизации метода с целью достижения необходимых, высоких результатов в этой области.

Анализ. Используя SVM для работы с текстовыми данными, задачи которые он поставлен решать, обретают следующий вид:

1. Классификация мультиязычных документов возможна по следующим признакам:

– по наименованию;

– по языку документа;

– по наличию разноязычных включений (они могут передавать смысл содержимого более полно);

– по степени гласности (открытые и документы с ограниченным доступом);

– по юридической силе (подлинные и подложные);

– по срокам исполнения (срочные и несрочные);

– по происхождению (служебные и личные);

– по срокам хранения (временного и постоянного срока хранения);

– по степени обязательности (информационные и директивные – обязательные для исполнения).

2. Рубрикация – следующая ступень для выделения тематической составной документа (семантики). Для этого, нужно поделить текст на составные части с использованием заголовков и нумерации. Система рубрик включает заголовки частей, разделов, глав и параграфов, которые тоже нумеруются и в свою очередь подразделяются на абзацы. Под абзацем понимается отступ вправо в начале первой строки определенной части текста, а также та часть текста, которая находится между двумя такими отступами. В абзац объединяют предложения, связанные между собой по смыслу. Абзацы одного параграфа или главы также должны быть по смыслу связаны между собой и расположены в логической последовательности.

3. Кластеризация – выделение групп документов, имеющих схожую смысловую нагрузку (тематические группы).

Постановка задачи. Необходимость модернизации вызвана отсутствием готовых решений для задачи кластеризации динамичных коллекций мультиязычных текстовых документов, содержащими не только текстовые корпуса на различных языках, но и допускающими включения иностранных слов, что по сути является шумами, к которым обучаемые методы классификации, а SVM, в их числе, плохо устойчивы.

Решение. В данной статье описываются изменения в работе базового SVM метода для достижения поставленной задачи. В качестве стартовой комплектации был выбран некоммерческий проект с открытым программным кодом, созданный для научных исследований и экспериментов в работе метода опорных векторов. Именно в библиотеке SVMLight [2] алгоритм был наиболее полно и удачно реализован. Библиотека представлена на разных программных языках.

До сих пор не существует разработанных методов построения спрямляющих пространств или ядер, наиболее подходящих для конкретной задачи, так как само построение адекватного ядра является весьма сложной задачей [1].

Цель данной работы заключается в разработке комплекса методов, которые дополнят базовый алгоритм минимизацией реакции на шумы от разноязычных включений, дадут возможность, автоматически подбирать значения параметров границ решений, делая их более «плавным» для предотвращения образования дублирующих документов в разных классах или исключения целых классов документов, не подошедшим к поставленным критериям.

Так как метод направлен на работу с динамическими коллекциями, где количество классов, к которым может относится тот или иной документ, величина не постоянная, то необходимая мультикласовость достигается при решении очередности бинарных задач. Сначала один класс отделяется от остальных, потом второй, третий и т. д. После чего получим несколько SVM для каждого из классов. При появлении класса нового объекта (документа), эти SVM возвратят коэффициенты принадлежности, и класс такого объекта будет определен по максимальному значению этого коэффициента. В случае не определения принадлежности к уже существующим, а значит не возможности вхождения, класс сможет продолжить существовать самостоятельно.

В качестве объектов аналитического анализа будем использовать сжатое представление текста в виде вектора термов (признаков). Размерность пространства при построении вектора признаков равна числу различных термов, содержащихся в обучающей коллекции. Многие алгоритмы классификации очень чувствительны к времени вычисления, которое часто является функцией от длины вектора, представляющего документ, поэтому необходимо стараться уменьшить размерность пространства признаков.

Представить документ в виде вектора несложно. Сначала нужно сделать словарь коллекции – это список всех ключевых слов, которые в этой коллекции встречаются. Количество слов и есть размерностью векторов. После этого документ представляется в виде вектора, где i-тый элемент – это мера вхождения i-того слова словаря в документ. Это может быть + 1 или -1 в зависимости от того, входит ли слово в документ или нет, или количество вхождений. Или «доля» этого слова в документе. Вариантов много. В итоге получаются вектора большой размерности, где большая часть элементов – нули (рис.).

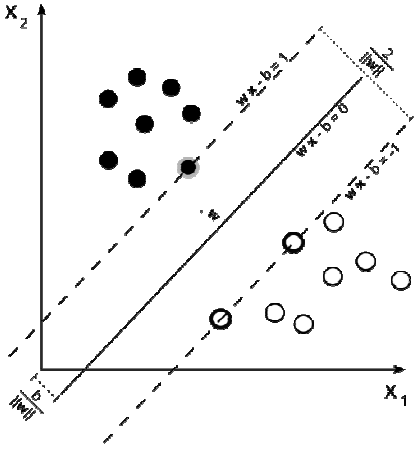

Рис. Два класса документов, которые представляется как точки (нулевой вектор)

Разумеется, классов будет значительно больше, и нам понадобится запускать алгоритм n раз. Точнее, у нас будет n классификаторов для каждой категории, умеющих определять попадает ли документ в эту категорию или нет.

Наиболее исследованным и распространенным решением при определении величин входящих в вектор термов является подход, основанный на предположении, что документы, принадлежащие одной рубрике, имеют близкие распределения относительных частот слов, входящих в текст.

Таким образом, при определении вектора термов для документа (размерность которого равна числу различных термов из всего массива) каждому слову из лексики коллекции ставится в соответствие координата в пространстве признаков. Она пропорциональна частоте слова в данном документе. Для определения этих координат, в случае весового представления текста, часто используют стандартную TfIDF функцию, которая определяется как [3]:

$$tfidf(t_i, d_j) = tf(t_i, d_j) \cdot idf(t_i), \quad (1)$$

где tf – частота терма ti в документе dj, idf – инверсная частота терма:

$$idf(t_i) = \log \frac{|D|}{df_i}. \quad (2)$$

В этой формуле |D| – количество всех документов в коллекции, dfi – количество документов, содержащих терм ti.

Кроме этого, каждый из документов коллекции, несомненно, обладает различной длинной, что вызывает необходимость нормализации частот полученных по первой формуле

$$w_{ij} = \frac{tfidf(t_i, d_j)}{\sqrt{\sum_{1}^{|T|} (tfidf(t_i, d_j))^2}}, \quad (3)$$

где вес i-го терма в j-тос документе wij рассчитывается, исходя из того, чтобы сумма квадратов весов каждого документа была равна 1.

Для тестирования классификатора можно использовать часть той же самой коллекции (только тогда не стоит использовать эту часть при обучении).

Для оценки работы классификаторов используются несколько метрик:

Accuracy = ((r – kr) + kw)/n Фактически – это точность.

Precision = kr/n

Recall = kr/r

Здесь kw – количество документов, которые классификатор неправильно отметил как не относящиеся к искомой категории;

kr – количество документов, которые классификаторов правильно отметил как относящиеся к искомой категории;

r – общее количество документов, относящихся к искомой категории, по мнению классификатора;

n – общее количество документов, относящихся к искомой категории.

Из этих метрик Precision и Recall являются самыми важными, так как они показывают насколько вообще наш классификатор точно или беспорядочно работает. Чем ближе обе метрике к единице — тем лучше.

Для «сглаживания» границ разделяющей плоскости вблизи их будем находить ближайшего соседа каждой точки. Расстояние между векторами измеряется по примеру евклидового расстояния:

$$D(x_i, x_j) = \sqrt{\sum_{p=1}^{n} (x_i^p - x_j^p)^2} \,. \qquad (4)$$

Расстояние точки к себе равно ∞. Сравнив расстояния и выбрав самый близкий нулевой вектор, делаем сравнение знаков векторов. В случае если они разные, то есть относятся к разным классам, такого соседа необходимо удалить. При больших объемах выборки, такие действия существенно не скажутся на результатах, при этом сильно помогут разграничить существующие классы документов, делая границы разделяющей плоскости более плавными. По знаку класса каждой выборки и его ближайшего соседа определятся новые границы решения.

Для уменьшения влияния разноязыковых шумов на качество классификации динамичных коллекций мультиязычных документов, возникает необходимость в разработке новых ядер, математических функций, для изменения положения в пространстве разделяемых признаков. Готовых решений в этой области автором обнаружено не было. На данный момент классификация документов происходит по отдельности для каждой языковой группы, что непременно ведет к появлению тематических дублей в каждом из классов документов. Причина в том, что для задач текстовой классификации широко используются стандартные наборы функций ядер, например, гауссовское ядро, которые не учитывают особенностей текстовых данных. Тем не менее, современные, глобальные поисковые системы, для решения задач классификации веб-страниц, начали использовать модификацию существующей ядерной функции SSK (String Subsequence Kernel), или же строкового ядра, которая до этого применялась только в задачах классификации протеинов. Это дает возможность говорить о том, что необходимость разработки новых ядер для поставленной задачи можно обойти путем модификации существующих ядер, того же SSK, только не для веб контента, а для не структурированных локальных текстовых документов, которые преобладают большим объемом данных и широкой тематической насыщенностью.

Для классификации пары мультиязычных текстовых документов можно представить их в виде последовательности символов, не учитывая языковые признаки. В таком случае

пространство признаков документа будет состоять из множества всех подстрок. Выбрав ключевые подстроки и пропустив их набор для каждого из двух сравниваемых документов, через блок перевода, сводимый к одному языку, например, русскому, и сравнив количество подстрок с общим переводом. Если их количество больше или равно значению некой весовой функции, то можно говорить о тематической схожести данных документов.

Выводы. Для подтверждения эффективности представленных способов модификации базового алгоритма опорных векторов в составе свободной библиотеке SVMLight, для решения поставленных задач, необходима окончательная программная реализация с последующим проведением цикла опытов на локальных коллекциях документов Reuters. Система классификации, рубрицирования и кластеризации динамичных коллекций мультиязычных текстовых документов находится в стадии разработки.

Список использованных источников

1. *Vapnik V.* Statistical learning theory. Wiley, New York, 1998 [Электронный ресурс]. – Режим доступа : http://books.google.com.ua.

2. *SVM-Light* Support Vector Machine [Электронный ресурс]. – Режим доступа : http://www.svmlight.joachims.org.

3. *Sebastiani F.* Machine Learning in Automated Text Categorization [Электронный ресурс]. – Режим доступа : http://books.google.com.ua.

4. *Reuters-21578,* DataSet (коллекция документов) [Электронный ресурс]. – Режим доступа : http://www.daviddlewis.com/resources/testcollections/reuters21578.

5. *Пескишева Т. А.* Современные системы и модули автоматической рубрикации текстовых документов [Электронный ресурс] / Т. А. Пескишева. – Режим доступа : http://www.vggu.ru.

УДК 621.317.3

А.И. Вервейко, канд. техн. наук
Черниговский национальный технологический университет, г. Чернигов, Украина

ВИРТУАЛЬНЫЙ ИЗМЕРИТЕЛЬ ФУНКЦИИ КРАТКОВРЕМЕННОЙ НЕСТАБИЛЬНОСТИ ЧАСТОТЫ

О.І. Вервейко, канд. техн. наук
Чернігівський національний технологічний університет, м. Чернігів, Україна

ВІРТУАЛЬНИЙ ВИМІРЮВАЧ ФУНКЦІЇ КОРОТКОЧАСНОЇ НЕСТАБІЛЬНОСТІ ЧАСТОТИ

Aleksandr Verveyko, PhD in Technical Sciences
Chernigov National Technological University, Chernigov, Ukraine

VIRTUAL METER OF THE FUNCTION OF SHORT-TERM INSTABILITY FREQUENCY

Получил дальнейшее развитие метод измерения функции кратковременной нестабильности частоты на базе преобразования период-временной интервал-код. Разработаны четыре варианта его реализации, получены аналитические соотношения для основных метрологических характеристик вариантов и проведен их сравнительный анализ. Реализованы автономный и виртуальный измерители, а также проведены экспериментальные исследования стандартных генераторов. Указаны особенности измерителей и пути их дальнейшего совершенствования.

Ключевые слова: *кратковременная нестабильность частоты, преобразователь период-временной интервал-код, автономный измеритель, виртуальный измеритель, САПР LabVIEW.*

Отримав подальший розвиток метод вимірювання функції короткочасної нестабільності частоти на базі перетворення період-часовий інтервал-код. Розроблено чотири варіанти його реалізації, отримано аналітичні співвідношення для основних метрологічних характеристик варіантів і проведено їх порівняльний аналіз. Реалізовано автономний і віртуальний вимірювачі, а також проведено експериментальні дослідження стандартних генераторів. Вказано особливості вимірювачів і шляхи їх подальшого вдосконалення.

Ключові слова: *Короткочасна нестабільність частоти, перетворювач період-часовий інтервал-код, автономний вимірювач, віртуальний вимірювач, САПР LabVIEW.*

The method for measuring of function short-term instability frequency got further development on the base of transformation period-temporal interval-code. Four variants of his realization are worked out, analytical correlations are got for