ОТЧЕТ О РЕШЕНИИ ЗАДАЧИ

JRS 2012 Data Mining Competition: Topical Classification of Biomedical Research Papers

Куракин Александр

(итоговый результат — 0.513)

Постановка задачи.

Рассмотрим множество некоторых текстов, вообще говоря, на биологическую тематику. Известно количество вхождений некоторых слов в эти тексты (даны номера слов) - признаки. Известно также, что тексты могут относится к одному и более классу (topic). Имея обучающую выборку, предсказать принадлежность текстов классам.

- Оценка эффективности происходит по F-мере.
- Классов-тем — 83.
- Признаков — ~25000.
- Обучающая выборка — ~10000 объектов.
- Признаки сильно разрежены.

Методики решения задачи.

Предпринимались попытки решить задачу тремя способами:

1) Метрические алгоритмы
2) Линейная классификация
3) Тестовые алгоритмы

Ввиду ограниченности времени я опишу лишь более удавшееся решение — линейное разделение.

Использовались две машины. На одной из них я реализовывал все вручную (Matlab 2011b), на втором — Octave 3.2 с установленным Shogun Toolkit.

Тестовые алгоритмы были отметены потому, что я не нашел там тесты. Перемножив x'x я получил ненулевые элементы лишь на диагонале.

С метрическими алгоритмами были проведены много экспериментов. Использовались:

1) Взвешенные соседи
2) Метод парзеновского окна
3) Метод потенциальных функций

Использовались (бездумно) метрики Shogun'a: MINKOWSKI, MANHATTAN, CANBERRA, CHEBYSHEW, GEODESIC, JENSEN, MANHATTANWORD, HAMMINGWORD, CANBERRAWORD, SPARSEEUCLIDIAN, EUCLIDIAN, CHISQUARE, TANIMOTO, COSINE, BRAYCURTIS, ATTENUATEDEUCLIDIAN. Затем, для оптимизации параметров, те метрики, которые я знал, я реализовывал самостоятельно (на первой машине).

В итоге я не смог преодолеть барьер **0.48**. Поэтому решил рассмотреть линейный алгоритм.

Линейная классификация. Итоговый алгоритм.

На эту мысль навел Петр, «признаков в 2.5 раз больше числа объектов ⇒ по отношению к одной тематике, обучающая выборка линейно разделима, очень уверенно разделима.» Однако, не скрываю, на выбор данного метода повлияло и то, что «получался» и «имею мало опыта». Очень хотел бы реализовать что-то сложнее, пусть и с меньшим результатом, но протестировать довольно быстро методы, в которых у меня нет опыта, не получалось — инструменты не справлялись с размерами признакового пространства, а уменьшить его не смог.

Итоговый алгоритм реализован на Octave и Shogun.

1) Разделение выборки на обучающую и тестовую.
2) Нормировка.
3) Классификация (с определенным порогом).
4) Учет зависимостей тем.
5) Урезание количества тем для каждого из текстов.

Разделение выборки на обучающую и тестовую.

Установлено, что чем больше обучающая выборка, тем лучше. Поэтому в итоге я разделял пропорции 19 к 1. Брались фиксированное подмножество — для выбора методов и подбора параметров, и случайные подмножества — для попытки локально улучшить тот или иной параметр.

Нормировка.

Для начала удалил нулевые признаки.

Далее испытывались следующие нормировки (по столбцам и по строкам):

1) Вычитание среднего арифметического
2) Вычитание медианы
3) Деление на максимальное значение
4) Деление на сумму

Вычитание среднего арифметического вызывало ошибки. Деление на сумму порождало пространство со слишком маленькими значениями признаков.

Нормировка по строкам — матрицы перестают быть разреженными. В итоге я использовал две нормировки:

1) Деление на максимальное значение столбца.
2) Вычитание медианы столбца и деление его на максимум.

Нормировка в итоге улучшала результат до 15% (в относительном измерении).

Классификация (с определенным порогом).

Использовался классификатор 'LIBLINEAR_L2R_LR' из набора Shogun. Путем итерационной максимизации результата было определено оптимальное значение нормализационной константы — в диапазоне 0.10 .. 0.14 . Порог отступа каждый раз оптимизировался отдельно.

Без нормировок результат был ~0.4.

Учет зависимостей тем.

Было известно, что некоторые темы используются «сцепленно». В обучающей выборке при большой вероятности при попадании в тему i текст попадал в тему j. Это было учтено, и при большой вероятности (порог оптимизировался каждый раз — обычно 0.6 .. 0.75) j-я тема отмечалась вместе с i-й.

Это увеличило результат на 1% (в абсолютном измерении).

Урезание количества тем для каждого из текстов.

После завершения описанных действий многие тексты имели слишком многом тем (до 30!), поэтому применялся и порог на количество тем. Экспериментом показано, что оптимально оставлять не более 5-6 тем.

Однако удивительно, но это улучшило результат менее, чем на 0.25%.

Повторение эксперимента.

Затем я брал случайные обучающие подвыборки и повторял эксперименты, чтобы «выскочил» результат немного побольше.

Итоговый результат.

Получилось **0.513**.

Советы новичкам и организационные выводы.

1) Не бойтесь :) «Реальная задача» - не приговор. И то, что в ней огромные данные — не повод для паники.
2) Пытайтесь любыми способами упростить задачу, свести к раннее изученным. В данном случае надо было преобразовать пространство признаков. Я постеснялся обозначить четко данный вопрос (неясность его для меня) перед преподавателем и группой. Группа решения не нашла, о себе вообще молчу :) А вопрос был в одной функции. Это было ошибкой.
3) Посмотрите сначала в ширину. Задача неоднозначная, даны лишь какие-то числа... Выбор метода — во многом эвристика. Пробуйте различные методы.
4) Не пытайтесь реализовывать методы, особенно, если вы решаете задачу в одиночку, на низком уровне. Посмотрите в сторону библиотек. А потом уже те методы, которые показывают хороший результат и которые можно тюнинговать/оптимизировать — реализуйте хоть на ассемблере.
5) Ну я в последний день не делал, хотя на коллективном участке. Скованно решал задачу и когда понимал, что что-то не выходит, меня сковывало еще больше. Особенно если я понимал, что проблема в размере данных. Но: не делайте в последний день! Задача интересная и прибыльная :)

Если повернуть время вспять...

1) Я бы не боялся реальной задачи.
2) Я бы попробовал больше методов.
3) Я бы пореже открывал вкладку «Обсуждение» на коллективном этапе. Я хотел достичь результата, но это смешно, когда мало опыта и страшно. Нужен был процесс (опыт), а не результат. А опыт получается собственноручно.
4) Я бы вытряс из Вас, Александр Геннадьевич, рассказ про SVD :-D.
5) Я бы работал в паре. Я всегда вижу этот путь продуктивнее, чем «одиночка» или «команда». Но не смог найти заинтересованного компаньона со сходныи уровнем опыта и организоционными предпочтениями.

Приложения

1) Код линейной классификации
2) Частичный код методов ближайших соседей
3) Параметры и результаты контролей-экспериментов, которые выполнялись итоговым алгоритмом с параметрами, выбранными случайно, но вблизи оптимальных значений.