Декрементальный синтез вторичной структуры алгебраических байесовских сетей: статистическая оценка сложности[1]

Левенец Д.Г., студент кафедры информатики СПбГУ, стажер лаб. ТиМПИ СПИИРАН, daniel-levenets@yandex.ru
Зотов М.А., студент кафедры системного программирования СПбГУ, стажер лаб. ТиМПИ СПИИРАН, zotov1994@mail.ru

Аннотация

Рассмотрена задача декрементального синтеза вторичной структуры алгебраических байесовских сетей. Выполнен статистический анализ сложности декрементального алгоритма в сравнении с жадным алгоритмом. Статистические эксперименты визуализированы в виде графиков.

Введение

В теории машинного обучения и интеллектуальных системах, в качестве структуры, позволяющей хранить, обрабатывать и впоследствии отображать данные, часто рассматривают байесовские сети доверия (БСД). Такие сети, кроме некоторых данных о предметной области – фрагментах знаний (ФЗ), – могут хранить и точечные (скалярные) оценки вероятностей достоверности ФЗ. Другими словами, БСД формализуют данные о предметной области и позволяют сопоставлять их с вероятностью их выполнения. Одной из радновидностей БСД являются алгебраические байесовские сети (АБС) [5, 8, 9]. Такие сети отличаются от классических – БСД – тем, что могут принимать интервальные оценки вероятностей. Отметим, что ФЗ в совокупности с их оценками (в случае как и БСД, так и АБС) принято называть первичной структурой. В качестве вторичной структуры в АБС могут выступать так называемые графы смежности. Строгое определение будет дано далее, но сейчас отметим, что их особенность заключается в том, что между пересекающимися фрагментами знаний в графе обязательно должен присутствовать путь, который в графе представляется набором ребер. Таким образом, связи между ФЗ задают некоторые

[1]Статья содержит результаты исследований, частично поддержанных грантами РФФИ 15-01-09001-а, 14-01-00580

ограничения и вносят некоторые определенности в структуру графа смежности [6, 7]. Актуальной задачей является минимизация ребер в графе смежности и синтез минимального графа смежности по набору ФЗ или поддержание минимальности (здесь и далее – по числу ребер) в уже имеющейся структуре [5, 11]. В качестве алгоритмов синтеза минимального графа смежности были предложены прямой и жадный алгоритмы[5, 10]. В более поздних работах были предложены другие алгоритмы: инкрементальный[4], инкрементальный улучшенный[1] и, наконец, декрементальный. Указанные алгоритмы, в отличие от прямого и жадного, хорошо подходят в случаях, когда необходимо поддерживать связи в имеющейся структуре АБС в условиях постоянного изменения набора ФЗ (который, однако, меняется не весь и сразу, а постепенно и планомерно): алгоритмы хороши тем, что не перестраивают всю сеть заново, а "стараются"модернизировать имеющуюся, внося небольшие изменения. В работах[1, 2, 4] были проанализированы относительные временные сложности прямого и жадного, прямого и инкрементального, прямого и улучшенного инкрементального, жадного и инкрементального, жадного и инкрементального улучшенного алгоритмов соответственно. Результаты показали, что на некотором поддиапазоне мощностей графов (числа ФЗ) прямой алгоритм работает быстрее жадного, инкрементальный и улучшенный инкрементальный работают быстрее и прямого, и жадного. В отношении декрементального алгоритма был приведен только псевдокод, а также доказана корректность алгоритма. Целью настоящей работы является вычисление статистических относительных сложностей жадного и декрементального алгоритмов. Вычисление таких оценок позволило бы понять — в каких именно случаях и при каких характеристиках входных данных декрементальный алгоритм работает быстрее жадного, и в каких случаях медленнее.

Определения и обозначения

Воспользуемся системой терминов и обозначений, сложившихся в[5, 8, 9].

Пусть задан конечный алфавит символов A, а непустые множества символов (без повторов) — слова — рассматриваются как возможные значения нагрузок вершин графов (в основном, графов смежности) и их ребер. Пусть имеется набор вершин $V = \{v_1, v_2, \ldots, v_n\}$ и такие нагрузки $W = \{W_1, W_2, \ldots, W_n\}$, причем $\forall u \in V \;\; W_u$ является нагрузкой для вершины u. Контекст статьи позволяет для удобства отождествлять вершины их нагрузки, как соотносящиеся взаимооднозначно, поэтому мы станем использовать обозначения u и W_u взаимозаменяемо.

Определение 1. Назовем неориентированный граф $G = \langle V, E \rangle$ графом смежности, если он удовлетворяет следующим условиям:

1. $\forall u, v \in V : W_u \cap W_v \neq \emptyset \quad \exists$ путь P в графе G : $\forall s \in P \Rightarrow W_u \cap W_v \subseteq W_s$.
2. $\forall \{u, v\} \in E \Rightarrow W_u \cap W_v \neq \emptyset$.
3. $\nexists\ u, v \in V : W_u \subseteq W_v$, — нагрузка одной вершины графа не входит полностью в нагрузку любой другой вершины.

Граф смежности с минимальным и максимальным числом ребер мы будем называть *минимальным графом смежности* и *максимальным графом смежности* соответственно.

Определение 2. Пересечение нагрузок двух вершин $W_u \cap W_v$ будем называть *сепаратором*.

Описание вычислительных экспериментов

Подробное описание методики сравнения статистической сложности реализаций алгоритмов было приведено в статье[3], и в настоящей работе мы будем ей следовать, сравнивая реализации декрементального и жадного алгоритмов. Дополнительно отметим, что влияние на время работы алгоритма могут оказывать некоторые случайные факторы, например: текущее состояние ОС или загруженность процессора, на котором запущено приложение, вычисляющее данную оценку. Также время работы алгоритма зависит от набора данных, по которому выполняется алгоритм. И именно для минимизации влияния случайных факторов, а также для описания распределения отношений скорости работы алгоритмов на наборах данных, испльзуется упомянутая методика.

Перечисленные в предыдущем разделе алгоритмы были реализованы на языке программирования C#. Основная идея эксперимента для вычисления отдельного показателя относительной скорости работы алгоритмов заключалась в следующем.

1. Генерировался набор вершин V.

2. По набору вершин V синтезировался минимальный граф смежности G, который на шаге 3 будет выступать в качестве системы, из которой мы хотим удалить очередной ФЗ.

3. Из графа G с помощью декрементального алгоритма удалялась вершина v. Время выполнения алгоритма фиксировалось.

4. По множеству вершин $V \setminus \{v\}$, где v — удаляемая вершина, строился минимальный граф смежности с помощью жадного алгоритма. Время выполнения алгоритма также фиксировалось.

5. По результатам вычилялись отношения величин времени работы жадного и инкрементального алгоритмов.

Шаги 1–5 выполнялись для различных наборов вершин V; из результатов, полученных в сериях экспериментах, выделялись интересующие статистики.

Визуализация экспериментов

Результаты экспериментов, проведённых над жадным и декрементальным алгоритмами, были визуализированы в виде двухмерных графиков (Рис. 1–4), где на горизонтальной оси откладывалась мощность множества V, а по вертикальной оси откладывалась величина статистики.

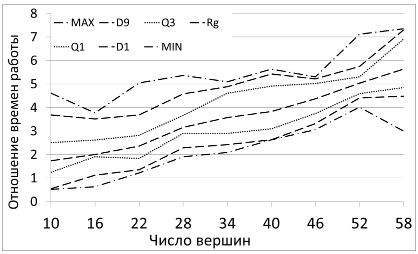

Рис. 1: 82% вершин — 2-4 порядков, 13% — 5-7, 5% — 8-10, 0% — 11-13

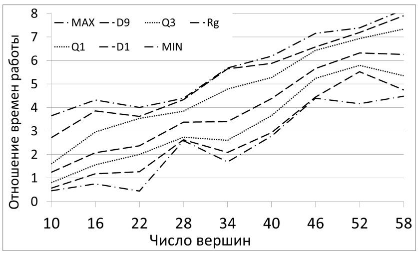

Рис. 2: 74% вершин — 2-4 порядков, 13% — 5-7, 8% — 8-10, 5% — 11-13

Анализ результатов вычислительных экспериментов позволяет сделать вывод, что декрементальный алгоритм работает существенно быстрее жадного алгоритма в поддиапазоне 22–58 вершин, а в поддиапазоне 10–19 имеет место переходная ситуация, в которой декрементальный алгоритм может работать медленнее жадного. Однако стоит

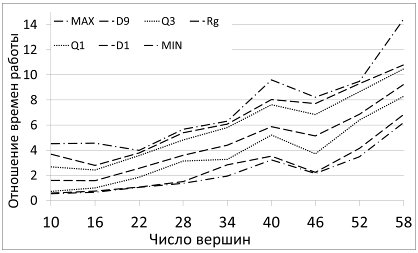

Рис. 3: 49% вершин — 2-4 порядков, 25% — 5-7, 15% — 8-10, 11% — 11-13

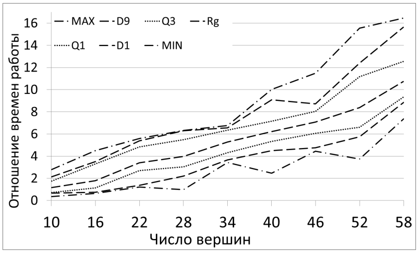

Рис. 4: 25% вершин — 2-4 порядков, 46% — 5-7, 15% — 8-10, 14% — 11-13

отметить, что время синтеза минимального графа смежности для 10–19 вершин незначительно.

Заключение

Была проведена серия экспериментов по сравнению скоростных показателей двух конкурирующих алгоритмов — жадного и декрементального. Результаты вычислительных экспериментов были визуализированы с целью наглядной демонстрации преимущества того или иного алгоритма, а также с целью упрощения анализа выходных данных вычислительных экспериментов. Результаты экспериментов демонстрируют, что декрементальный алгоритм имеет превосходство в скорости работы над жадным алгоритмом синтеза минимального графа смежности, за исключением некоторого отдельного поддиапазона вершин, в котором, однако, время синтеза является незначительным.

Литература

[1] Зотов М.А., Левенец Д.Г., Тулупьев А.Л., Золотин А.А. Синтез вторичной структуры алгебраических байесовских сетей: инкрементальный алгоритм и статистическая оценка его сложности // Научно-технический вестник информационных технологий, механики и оптики. 2016. Т. 16. № 1. С. 122–132.

[2] Зотов М.А., Тулупьев А.Л. Вторичная структура алгебраических байесовских сетей: статистическая оценка сложности прямого алгоритма синтеза // SCM'2015: XVIII Международная конференция

по мягким вычислениям и измерениям. Санкт-Петербург, 19-21 мая 2015 г. С. 158-162.

[3] Зотов М.А., Тулупьев А.Л. Синтез вторичной структуры алгебраических байесовских сетей: методика статистической оценки сложности и компаративный анализ прямого и жадного алгоритмов // Компьютерные инструменты в образовании, 2015. № 1. С. 3–16

[4] Левенец Д.Г., Зотов М.А., Тулупьев А.Л. Инкрементальный алгоритм синтеза минимального графа смежности // Компьютерные инструменты в образовании, 2015. № 6. С. 3–18.

[5] Опарин В.В., Тулупьев А.Л. Синтез графа смежности с минимальным числом ребер: формализация алгоритма и анализ его корректности // Труды СПИИРАН. — 2009. — № 11. — С. 142–157.

[6] Тулупьев А.Л. Алгебраические байесовские сети: глобальный логико-вероятностный вывод в деревьях смежности: Учеб. пособие. СПб.: СПбГУ; ООО Издательство «Анатолия», 2007. 40 с. (Сер. Элементы мягких вычислений).

[7] Тулупьев А. Л. Байесовские сети: логико-вероятностный вывод в циклах. СПб.: Изд-во С.-Петербургского ун-та, 2008. 140 с. (Элементы мягких вычислений.)

[8] Тулупьев А.Л., Николенко С.И., Сироткин А.В. Байесовские сети доверия: логико-вероятностный подход. — СПб.: Наука, 2006. — С. 607.

[9] Тулупьев А.Л., Сироткин А.В., Николенко С.И. Байесовские сети доверия: логико-вероятностный вывод в ациклических направленных графах. — СПб.: Издательство С.-Петербургского университета, 2009. — С. 400.

[10] Фильченков А.А. Синтез графов смежности в машинном обучении глобальных структур алгебраических байесовских сетей. Дисс... к-та физ.-мат. н. Самара, 2013. С. 339.(Самарск. гос. аэрокосм. ун-т им. ак. С. П. Королева (нац. исслед.))

[11] Фильченков А.А., Тулупьев А.Л. Структурный анализ систем минимальных графов смежности // Труды СПИИРАН. — 2009. — № 11. — С. 104–127.