ЗАДАЧИ ДЛЯ СТУДЕНТОВ, ОСЕНЬ 2011 Г.

Сергей Николенко

1. Полная односторонняя функция на базе $(\mathbb{Z} \times \mathbb{Z})$-модулей – CS

Суть задачи. Есть работа [6], в которой строится tiling, моделирующий машины Тьюринга в контексте задачи принадлежности к подмодулям некоторого $(\mathbb{Z} \times \mathbb{Z})$-модуля (да, это на самом деле полугрупповая конструкция, конечно). Мне кажется, что там после очень простой модификации получается полная односторонняя функция. Я даже написал некоторый текст на эту тему и уже рассказал об этом на русско-финском семинаре RuFiDiM. Однако в тексте, конечно, дырок больше, чем не-дырок. Задача в том, чтобы провести и записать доказательство полностью – оно будет длинное и сложное (надо симулировать машины Тьюринга), но при этом оно почти в точности должно следовать образцу из [6].

Методы, чему можно научиться. Научится творчески владеть машинами Тьюринга. Методы – симуляция машин Тьюринга через tiling, как в [8, 11, 4]. Кроме того, по дороге невозможно будет не изучить хоть немножко алгебры.

Возможные перспективы. Сама эта работа, даже если будет доведена до конца, вряд ли будет очень интересной, дело техническое и довольно очевидное. Дальше нужно будет аккуратно посмотреть, есть ли аналогичные полные односторонние функции по соседству (в других похожих алгебраических конструкциях – начать можно с тех, которые прямо в [6] и упоминаются). Но есть некоторые шансы на дальнейшее интересное продолжение.

2. Байесовские рейтинг-системы – CS / SE

Суть задачи. Представьте, что вам нужно построить рейтинг-систему, в которой в отдельном матче участвуют несколько команд, причём их составы от матча к матчу постоянно меняются (т.е. рейтинговой единицей будет игрок, а не команда). Такая задача была решена в системе TrueSkill, разработанной в Microsoft Research [1, 3] и внедрённой в Xbox Live. Однако у TrueSkill есть ряд очевидных недостатков. В недавней работе [9] мы решили значительную часть этих проблем, придумав новую байесовскую рейтинг-модель. Теперь надо продолжать, расширять и реализовывать. Я сейчас вижу такой спектр задач:

(1) взять то, что у нас грязно написано в Maple, реализовать аккуратнее (в частности, написать вокруг него веб-сервис, чтобы можно было быстро и красиво смотреть на результаты);
(2) расширить базовую модель, добавив возможность учитывать результаты турниров в числах (сейчас только перестановка команд поступает на вход, но ведь иногда известны и конкретные результаты);
(3) думать над дальнейшими расширениями модели – другая дополнительная информация, нелинейные функции силы команды, комбинация турниров, о которых известны разные вещи. Идей много, надо доводить их до конца, реализовывать, сравнивать результаты экспериментов.

В этом проекте наверняка будет математическое содержание.

Методы, чему можно научиться. Методы – байесовский вывод, графические вероятностные модели, причём вы не просто узнаете, что это такое, а научитесь ими творчески владеть.

Возможные перспективы. Потенциальные теоретические результаты – новые вероятностные рейтинг-системы. Это можно публиковать в хороших местах. Первым практическим результатом должен быть сайт с рейтингом, посвящённый нашему dataset'у – я думаю, у хорошего такого сайта будет немало посетителей (это я SE-результат описываю). Более далёкий результат – если хорошо запрограммировать эту систему, можно сделать систему, которая позволяет другим людям строить на её основе свои собственные рейтинги. Это может быть интересный и достаточно популярный проект (и отличный SE-диплом – но это уже совсем далёкие мечты).

Prerequisites. Нужно владеть основами теории вероятностей.

3. Рекомендательные системы – SE / CS

Суть задачи. Рекомендательные системы – передний край (один из передних краёв :)) современного машинного обучения [5]. Есть масса разных методов для разных ситуаций. Суть проекта в следующем: существует dataset проекта, в котором пользователи рекомендуют друг другу интересные веб-страницы, разбитые на тематические категории. Их задача интересна тем, что постановка вопроса немножко отличается от классических рекомендаций (у них не ставят рейтинги в звёздочках и не выбирают из нескольких вариантов, а просто нажимают «Like»). План работ такой:

(1) реализовать или найти внешние реализации основных рекомендательных методов, запустить на dataset'е, сравнить, сделать удобную систему для проведения экспериментов;
(2) улучшить базовые рекомендательные системы с учётом специфики конкретно этой задачи.

Этот проект менее математический, чем предыдущий.

Методы, чему можно научиться. Методы – байесовский вывод, графические вероятностные модели, обучение с подкреплением, динамические модели, коллаборативная фильтрация.

Возможные перспективы. Немедленный практический выход тут очевиден – эта система уже реально работает, у неё уже довольно много пользователей, улучшать её – достойное дело. Студент в течение работы над проектом станет, надеюсь, полноценным исследователем в области машинного обучения, дальше можно будет продолжать работать над другими задачами. Если здесь появятся новые результаты, их, конечно, можно будет опубликовать.

Prerequisites. Нужно владеть основами теории вероятностей.

4. Контент-анализ музыкальных произведений – CS / SE

Суть задачи. Почему нам нравится одна музыка и не нравится другая? В противовес системам, использующим crowdsourcing (например, last.fm с их тэгами и коллаборативной фильтрацией — если вы слушаете X, а Вася слушает X и Y, вам порекомендуют Y) команда увлеченных музыковедов в США создала Pandora.com, главный девиз которого — рекомендовать музыку, основываясь только на её содержании, а не на внешних социальных факторах. Они верили, что можно разобрать песню, которая вам нравится, по косточкам, и на основе этого посоветовать другую. И у них есть своя аудитория: traffic rank в США по данным Alexa.com у Pandora выше, чем у last.fm — 61 против 427. Основа Pandora — кропотливая ручная работа экспертов, анализирующих каждую песню, добавляемую в базу. Конечно, хочется сделать то же самое автоматически, но существующие методы автоматического анализа музыки пока не позволяют этого. Мы попробуем создать часть такой системы, которая сравнивает песни по особенностям использования некоторых граней музыки — гармонии, ритма, возможно, еще мелодии и тембра. Мы будем пользоваться последними достижениями в области машинного обучения — системой приближенного вывода на графических вероятностных моделях Infer.NET, разрабатываемой в Microsoft Research. Этот проект будет вестись в тесном сотрудничестве с Игорем Балтийским; до этого Игорь в своей магистерской работе пробовал реализовать критерий для сравнения музыки по гармонии, отталкиваясь от простейших предположений; первые результаты показывают, что такая

реализация вполне возможна, но нужно, во-первых, использовать и другие грани музыки — хочется добавить ритм, поскольку ритм и гармония тесно связаны, — а во-вторых, разработать вероятностные модели, лучше согласующиеся с теорией музыки. В этом проекте у вас есть возможность познакомиться на практике с графическими вероятностными моделями — разрабатывать их, отталкиваясь от требований предметной области (музыки), реализовывать и запускать на ней интересные эксперименты.

Методы, чему научится человек. Методы – байесовский вывод, графические вероятностные модели, обработка сигналов.

Возможные перспективы. Это быстроразвивающаяся область, любые новые результаты здесь будут интересны сообществу. Кроме того, в случае успеха можно будет запустить реальный проект, основанный на разработанной системе.

Prerequisites. Нужно владеть основами теории вероятностей.

5. Модели Брэдли–Терри в протеомике – BIO

Суть задачи. В задачах секвенирования пептидов (как de novo, так и в гибридных методах spectral dictionaries) возникает проблема, которая в настоящее время решается не очень хорошо – проблема учёта высоты пиков в масс-спектрометрических спектрах [2]. Чаще всего просто используют бинарную модель – есть пик или нет его. Проблема там в том, что непосредственно высота пика почти не несёт никакой информации, но может быть важно, кто из них выше. Поэтому я предлагаю попробовать использовать одну из простых рейтинг-моделей – модель Брэдли–Терри – и выводить по имеющимся базам данных «рейтинг» пиков только из данных вида «один пик выше другого»; затем полученный рейтинг подавать одному из известных алгоритмов как весовые функции для пиков.

Методы, чему можно научиться. Методы – вероятностные модели, байесовский вывод, de novo peptide sequencing. Студент прочтёт много статей по секвенированию пептидов, научится обрабатывать данные в R (или Octave, как ему удобнее будет).

Возможные перспективы. Если результат окажется лучше имеющихся, то это однозначно хорошая публикация. Если нет – студент войдёт в тему исследований по протеомике, получит опыт реальной работы с масс-спектрометрическими базами.

6. Бикластеризация спектров – BIO

Суть задачи. Суть задачи связана с обработкой данных imaging mass-spectrometry (когда масс-спектрометрия используется для того, чтобы получить картинку, возможно, динамическую, некоторых культур, соединений и т.п.). Интересная новая задача – применить к таким данным алгоритмы бикластеризации, т.е. кластеризовать одновременно столбцы и строки матрицы спектров [7, 10]. Этого ещё никто не делал, поэтому любые положительные результаты в этом направлении будут интересны.

Методы, чему можно научиться. Методы – бикластеризация, масс-спектрометрия. Думаю, программирование будет в основном на Matlab/Octave.

Возможные перспективы. Специалисты утверждают, что это можно опубликовать, даже если никак не улучшать имеющиеся методы бикластеризации, а просто получить достаточно красивые картинки. Однако есть идеи и насчёт того, в каком направлении улучшать методы бикластеризации (точнее, как улучшать их основной параметр – функцию расстояния между спектрами).

Список литературы

[1] Dangauthier, P., Graepel, T., Minka, T., and Herbrich, R. Trueskill through time: Revisiting the history of chess. In *Advances in Neural Information Processing Systems 20* (Cambridge, MA, 2008), J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., MIT Press, pp. 337–344.

[2] Frank, A. M. A ranking-based scoring function for peptide-spectrum matches. *Journal of Proteome Research 8*, 5 (2009), 2241–2252.

[3] Graepel, T., Minka, T., and Herbrich, R. TrueSkill(tm): A Bayesian skill rating system. In *Advances in Neural Information Processing Systems 19* (Cambridge, MA, 2007), B. Schölkopf, J. Platt, and T. Hoffman, Eds., MIT Press, pp. 569–576.

[4] Kojevnikov, A. A., and Nikolenko, S. I. On complete one-way functions. *Problems of Information Transmission 45*, 2 (2009), 108–189.

[5] Koren, Y., and Bell, R. M. Advances in collaborative filtering. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer, 2011, pp. 145–186.

[6] Lohrey, M., and Steinberg, B. Tilings and submonoids of metabelian groups. *Theory of Computing Systems 48*, 2 (2011), 411–427.

[7] Madeira, S. C., and Oliveira, A. L. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics 1*, 1 (2004), 24–45.

[8] Nikolenko, S. I. *Provably Secure Constructions in Cryptography*. LAP Lamberts Academic Publishing, 2011.

[9] Nikolenko, S. I., and Sirotkin, A. V. A new bayesian rating system for team competitions. In *Proceedings of the* 28^{th} *International Conference on Machine Learning* (2011), pp. 601–608.

[10] Tanay, A., Sharan, R., and Shamir, R. Biclustering algorithms: A survey. In *Handbook of Computational Molecular Biology*, S. Aluru, Ed., Computer and Information Science Series. Chapman & Hall/CRC, 2005.

[11] Николенко, С. И. *Новые конструкции криптографических примитивов, основанные на полугруппах, группах и линейной алгебре*. PhD thesis, Санкт-Петербургское отделение Математического института им. В. А. Стеклова РАН, 2009.