ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ ВОССТАНОВЛЕНИЯ ОТСУТСТВУЮЩИХ ДАННЫХ В СИМВОЛЬНЫХ ПОСЛЕДОВАТЕЛЬНОСТЯХ

М.Ю. Сенашова, А.Г. Рубцов
Институт вычислительного моделирования СО РАН
г. Красноярск

Введение. Задача восстановления отсутствующих данных актуальна как для фундаментальных, так и для прикладных областей науки. Существует достаточно много методов восстановления отсутствующих данных для данных представимых в виде таблицы чисел. Однако такие методы практически отсутствуют для данных, представляющих собой символьные последовательности. Теоретически любые данные можно представить в виде символьной последовательности.

Рассмотрим точную постановку задачи. Пусть имеется некоторый конечный алфавит Ω и пусть имеется некоторая последовательность символов из этого алфавита. Отсутствие части такой последовательности будем рассматривать как потерю данных, причем будем предполагать, что отсутствующая часть представляет собой связный диапазон. Ту часть символьной последовательности, в которой данные отсутствуют, будем в дальнейшем называть лакуной. Заполнять лакуну будем исходя из той информации, которая содержится в имеющихся в наличии частях последовательности. Эта информация – знание частот отдельных малых фрагментов, которые встречаются в последовательности. Строить заполнения будем с помощью копий этих малых фрагментов. Мы ищем не исходную отсутствующую последовательность, а заполнение, максимально похожее на имеющийся текст.

Словом длины q будем называть последовательность из q символов алфавита Ω. Опорным частотным словарем W толщины q будем называть список из всех слов этой длины, которые встречаются в исходном тексте, с указанием частот этих слов. Пополненным частотным словарем $\tilde{W}$ толщины q будем называть частотный словарь, который получается в результате построения заполнения. Левой (соответственно правой) опорой длины t $(0 \le t \le q)$ называется слово этой длины, которое располагается сразу слева (соответственно справа) от лакуны.

В общем случае, получается не одно заполнение лакуны, а несколько. Верхняя граница количества вариантов равна $|\Omega|^L$, где $|\Omega|$ – мощность алфавита, а L – длина лакуны. Из всех полученных заполнений лакуны нужно выбрать такое, которое максимально похоже на имеющиеся части последовательности. Данное условие выражается критерием минимума условной энтропии опорного частотного словаря относительно пополненного:

$$S = \sum_i f_i \, ln\left(\frac{f_i}{\tilde{f}_i}\right).$$

Здесь сумма берется по всем словам, встречающимся в полученном тексте, f_i – частота слова в опорном словаре, $\tilde{f}_i$ – частота слова в пополненном словаре.

Таким образом, необходимо получить заполнение, которое построено из слов опорного словаря и доставляет минимум условной энтропии.

Основными подходами к решению данной задачи являются имитационное моделирование кинетики химических реакций, а именно кинетическая машина Кирдина (КМК) и матричное представление частотного словаря. Данные подходы описаны в [1, 2].

КМК можно представить как химический реактор идеального смешения, в котором находятся “вещества”, реагирующие друг с другом по определенным правилам. Применительно к задаче восстановления данных “веществами” являются слова, составленные из алфавита Ω. В теории КМК правила реагирования могут быть трех типов: синтез, распад, замещение.

Для построения заполнения берется некоторое количество «затравок» (слов, совпадающих с левой опорой). На каждом шаге алгоритма для каждой затравки случайным образом выбирается слово из словаря, которое может вступить с ней в реакцию синтеза. Недостатком этого подхода является то, что данный алгоритм является вероятностным и

для построения всех возможных заполнений требует большого количества «затравок».

Следующим подходом к восстановлению данных является особое представление опорного частотного словаря. Словарь представляется в виде специальной матрицы, элементами которой являются строки символов. Далее после введения специальных операций умножения и сложения данная матрица возводится в степень $L+t+1$, где L – длина лакуны; t – длина опоры. После возведения в степень матрица содержит все возможные заполнения, которые можно построить из данного словаря.

Этот подход имеет преимущество перед использованием КМК в том, что строятся все возможные для данного словаря W заполнения, однако он требует больших ресурсов памяти и, в силу этого, не приемлем для заполнения лакун большой длины и больших словарей.

Оба эти подхода являются по своей сути переборными, а, следовательно, ресурсоемкими как по времени, так и по требуемой памяти. Причем оба эти подхода не гарантируют в силу особенностей конкретной последовательности получения оптимального в смысле условной энтропии заполнения. Поэтому возникает необходимость в таких методах и подходах, которые выбирали бы перспективные направления в пространстве поиска и отсекали неперспективные.

В роли таких алгоритмов могут быть использованы генетические алгоритмы, адаптированные к данной задаче. Преимущества данного подхода перед описанными выше, заключается в том, что генетические алгоритмы существенно сужают пространство поиска и с помощью генетических операторов рекомбинации, мутации и селекции выбирают перспективные области. К недостаткам данного подхода можно отнести тот факт, что генетические алгоритмы, являются стохастическими и не гарантируют в полной мере достижение необходимого результата. К тому же можно получить такие заполнения, в которых встречаются слова, не входящие в опорный словарь.

Генетический алгоритм. Генетический алгоритм (ГА) представляет собой метод оптимизации, основанный на концепциях естественного отбора и генетики. В этом подходе переменные, характеризующие решение, представлены в виде ген в хромосоме. ГА оперирует конечным множеством решений (популяцией) - генерирует новые решения как различные комбинации частей решений популяции, используя, такие операторы, как селекция, рекомбинация и мутация. Новые решения располагаются в популяции в соответствии со стратегией замещения.

Целью поиска является нахождение объекта с некоторыми свойствами. Поиск производится среди конечного множества потенциальных решений. Сначала определяются объекты этого множества O. Следующий шаг состоит в выборе некоторого представления объектов из пространства O. Представление определяется множеством S – пространством представлений. S выбирается с таким расчетом, что алгоритму поиска будет легче манипулировать членами S, чем O.

Пространство представлений всегда конечно. Отображение множества O в множество S называется представлением. Представление описывает связь между исследуемыми объектами, которые выступают в качестве потенциальных решений задачи поиска, и объектами, управлением и манипулированием, которыми занимается поисковый алгоритм.

На множестве объектов O должна быть определена функция цели $f(o)$, позволяющая сравнивать решения

$$f : O \to R,$$

такая, что для любых двух *o1*, *o2* из O, если *f(o1)>f(o2)*, то *o1* считается решением лучшим, чем *o2, R* – множество вещественных чисел.

Генетические операторы. Генетический алгоритм для поиска решения использует следующие операторы: селекцию, мутацию и рекомбинацию.

Генетический алгоритм применительно к задаче восстановления отсутствующих данных. Прежде чем построить поисковый алгоритм, нужно определиться с символьной моделью задачи: описать пространство потенциальных решений *O,* пространство представлений *S,* функцию кодирования *e* и декодирования e^{-1}; функцию цели *f*; генетические операторы.

Пространство решений. Пространство решений будет представлять собой строки заданной длины L (длина лакуны), составленной из символов алфавита Ω.

Представление решений. Пространство представлений будет в точности совпадать с пространством решений. Таким образом, здесь фенотип равен генотипу, функции кодирования и декодирования есть тождественные преобразования.

Мера пригодности. Задача поиска заключается в нахождении такой строки, которая доставляла бы минимум условной энтропии опорного частотного словаря относительно пополненного. В качестве критерия оценки индивида возьмем значение условной энтропии:

$$S = \sum_i f_i \, ln\left(\frac{f_i}{\tilde{f}_i}\right).$$

Не исключена возможность того, что мы получим заполнения, в которых будут слова, не встречающиеся в опорном словаре. Мы строим заполнения исходя из знаний опорного частотного словаря, поэтому мерой пригодности будет условная энтропия плюс некоторая штрафная функция, зависящая от количества "неизвестных" слов в заполнении:

$$M = S + Er(c) = \sum_i f_i \, ln\left(\frac{f_i}{\tilde{f}_i}\right) + Er(c).$$

Функция *Er(c)* – константа, значение которой равно количеству слов, которые есть в хромосомном наборе, но отсутствуют в опорном частотном словаре.

Оператор мутации. Пусть имеется некоторый конечный алфавит Ω, $a_i \in \Omega \ (i = 1,l)$ $a_i - i$-ый символ алфавита, а l – мощность алфавита (количество символов).

I = {abcdfs} – некоторый индивид. Мутация заключается в незначительном изменении генов индивида, а именно с вероятностью $1/(L-1)$ каждый символ меняется случайным образом на другой символ алфавита Ω.

Оператор скрещивания. Будем использовать следующие типы операторов: одноточечное скрещивание и двухточечное скрещивание.

Скрещивание заключается в выборе точки разрыва в хромосомах родителей и обмене соответствующих частей.

Оператор селекции. Задача селекции обеспечить селективное давление, которое продвигает популяцию "вперед". Будем использовать следующие типы операторов: пропорциональная селекция, турнирная селекция, ранговая селекция.

При пропорциональной селекции каждому индивиду назначается вероятность стать родителем прямо пропорционально пригодности индивида. Затем определенное количество раз реализуется случайная величина, с помощью которой отбираются родители.

При турнирной селекции из популяции случайным образом выбираются k индивидов. Сравнивается их пригодность и выбирается лучший индивид. Таким образом проводится турнир, k – параметр турнира.

При ранговой селекции индивидам присваиваются ранги в зависимости от величины пригодности. Индивиды сортируются по пригодности от наилучшего к наихудшему. Вероятности вычисляются в зависимости от рангов.

Результаты. Работа алгоритма была опробована на текстах различной степени сложности и длины.

Четырехбуквенный текст. В качестве тест–объекта брался генетический текст вируса пара-гриппа. Длина текста 15462 символов. Длина лакуны – 50 символов.

Толщина словаря 3. Восстановленное значение: taaaggaaagttaatg
agatgaaaggagatatttagagaatagaaggagg
Условная энтропия: 2.69418674928692E-06

Текст естественного языка. В качестве тест–объекта брался фрагмент текста Всемирной декларации прав человека на русском языке. Длина текста 7935 символов.

Толщина словаря 3. Восстановленное значение: аворавовос
Условная энтропия: 4.10311769746129E-05

Выводы. Применение генетических алгоритмов позволяет получать заполнения, близкие к оптимальным, и при этом не требует таких машинных ресурсов, каких требуют алгоритмы с использованием КМК и матричного представления частотного словаря.

Список литературы

1. Сенашова М.Ю., Рубцов А.Г., Садовский М.Г. Кинетическая машина Кирдина и задача восстановления утерянных данных // "Радіоелектроніка, Інформатика, Управління". – 2007. – № 1. – С. 87- 93.
2. Рубцов А.Г., Садовский М.Г., Сенашова М.Ю. Восстановление отсутствующих данных в символьных последовательностях. // Компьютерное моделирование и интеллектуальные системы: Сб. науч. трудов. – Запорожье: ЗНТУ, 2007. – С 206-212.
3. Рубцов А.Г., Садовский М.Г., Сенашова М.Ю. Оценка количества заполнений при восстановлении отсутствующих данных // Распределенные и кластерные вычисления. Избранные материалы Пятой школы-семинара. - Красноярск: Институт вычислительного моделирования СО РАН. – 2007. – С. 132-149.
4. Рубцов А.Г., Сенашова М.Ю., Садовский М.Г. Принцип максимального подобия в проблеме восстановления утерянных данных // Нейроинформатика и ее приложения: Материалы XIV Всероссийского семинара, 6-8 октября 2006 г. / под ред. А.Н. Горбаня, Е.М. Миркеса. Отв. за выпуск Г.М. Садовская, ИВМ СО РАН, Красноярск, 2006. – С. 88-90.