Комплексирование информации с использованием N-грамм при распознавании символьных последовательностей на видеопоследовательностях

Каратеев С.Л. Костромов Н.А. Бекетова И.В. Визильтер Ю.В.

ФГУП Государственный Научно Исследовательский Институт авиационных систем

г. Москва

Докладчик: Костромов Н.А.

ИОИ-2014 Греция, о. Крит

Предпосылки задачи

- Большое количество задач по считыванию символьных последовательностей (номера, текст)
- Регистрация и учет различных объектов
- Получение символьной информации с изображений

Проблемы при считывании символьной информации

- Изменяющиеся условия съемки
  - Движение объектов
  - Колебания камеры
  - Изменение освещенности, шумы различной природы
- Причины возможных ошибок
  - Неидеальность алгоритмов распознавания
  - Проблема предобработки и сегментации изображений

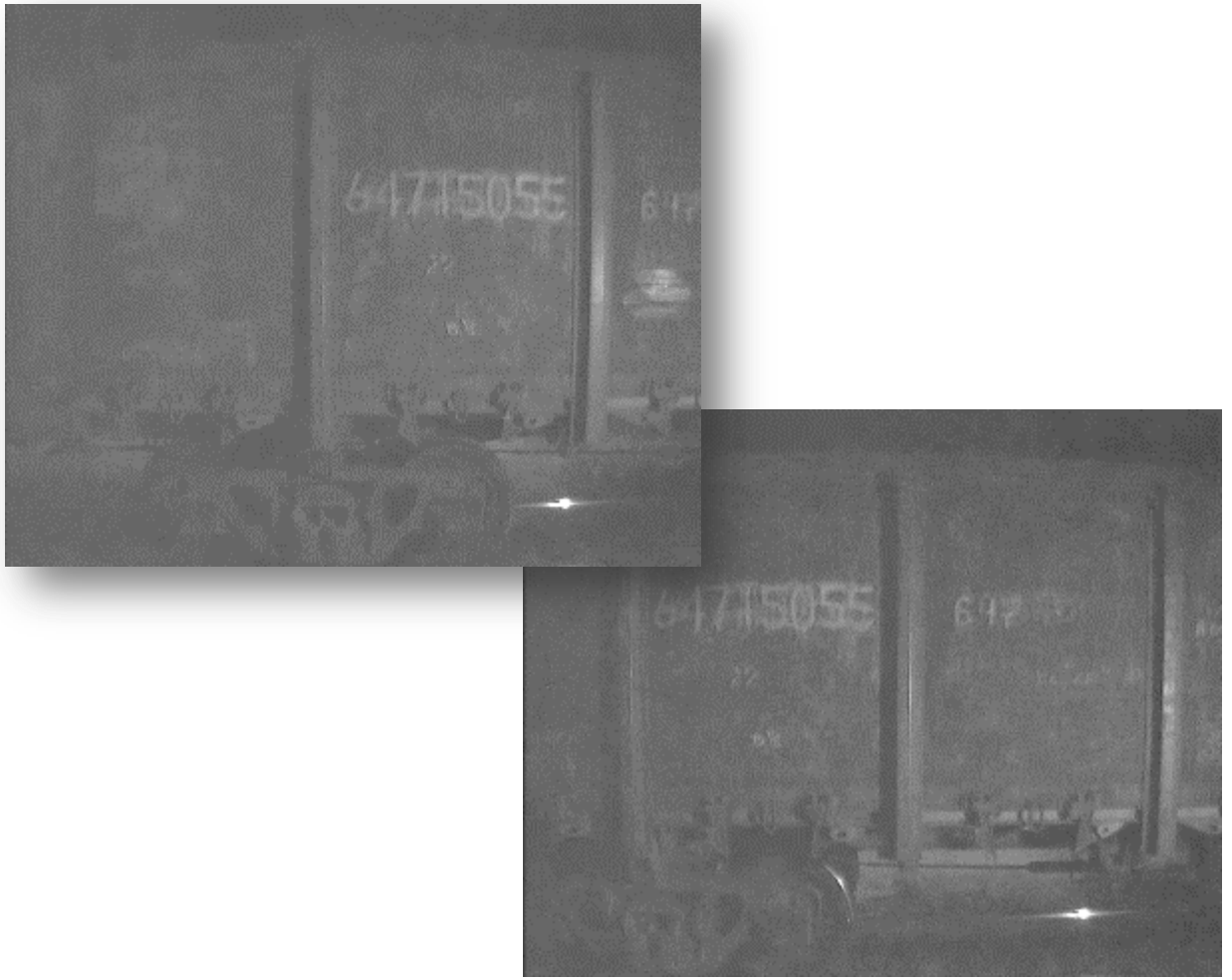
Традиционные методы решения и проблемы

- Методы
  - Коды с коррекцией ошибок
  - Контрольная сумма
  - Поиск по словарю
- Проблемы
  - Смешанность информации
  - Отсутствие контрольный суммы
  - Минимальная избыточность символьных последовательностей

Хорошо бы иметь метод, который позволит восстанавливать символьные последовательности за счет многократных и независимых считываний

Пример считывания номера 64715055 данных:

- 61715
- *6471511
- 54**5051
- 645055
- 64*50551
- 55
- 645055
- 745055
- 505
- 615055
- 65055
- 715155
- 7150553
- 671515
- 6415013
- 71505
- 55
- *4715055
- 5155
- 74**5055
- 7552
- 74**0550
- 51105
- 71*50551
- 7**07**1
- 7*50550*
- 67
- *5*1055*
- 715*1*1*
- 7055
- 51553
- 5
- 6
- 77
- 6*71771*
- 57
- 79464
- 3**7**1*



- Предлагаемое решение.

  На вход алгоритма подается множество независимо считанных символьных последовательностей. За счет применения статистических методов обработки входной информации совместно с адаптивным поиском достигается восстановление исходной последовательности символов.

- Плюсы метода:
  - Не требует априорных знаний о считываемой информации
  - Работает в реальном времени
  - Работает с любыми типами символьных последовательностей и не только
- Минусы метода:
  - В отдельных случаях требуется большое количество считываний символьной информации, зависящих от длины последовательности
  - Достоверность полученного результата имеет только косвенную оценку

Применяемые методы:

- N-граммы[1]
  - Последовательность из n элементов.
- Расстояние Левенштейна[2]
  - Редакционное расстояние или дистанция редактирования между двумя строками — это минимальное количество операций вставки одного символа, удаления одного символа и замены одного символа на другой, необходимых для превращения одной строки в другую.
  - Расстояние Левенштейна и его обобщения активно применяется: поиск в словарях, для исправления ошибок в слове, для сравнения текстовых файлов, в биоинформатике для сравнения генов, хромосом и белков.
- Методы статистического анализа
  - Оценка вероятности нахождения каждого символа в соответствующей позиции
- Генетический алгоритм
  - Поиск оптимальной гипотезы решения h, соответствующей исходной не зашумлённой последовательности.

[1]В. Ю. Гудков, Е. Ф. Гудкова N-граммы в лингвистике // Вестник Челябинского государственного университета. 2011. № 24 (239).

[2]Левенштейн В.И. Двоичные коды с исправлением выпадений, вставок и замещений символов. // Доклады Академий Наук СССР. Том 163, № 4, 1965.

Общий алгоритм работы:

1. Подготовка списка считанных последовательностей символов
2. Создание массивов N-грамм
3. Подсчет статистик по позициям символов
4. Использование генетического алгоритма для поиска лучшей гипотезы, соответствующей последовательности распознаваемой на реальном объекте
5. Выбранная гипотеза принимается как окончательное решение

Детали реализации ГА

- Генами являются:
  - N-граммы считанной последовательности
  - Считанные символьные последовательности
- Генерация первого поколения:
  - Генерация происходит из исходных данных при помощи генерации N-грамм и выравниванием их по длине требуемой последовательности
  - Из полученного пула генерируется гены путем применения оператора кроссовера и добавления новых генов из пула
- ГА завершает работу:
  - По достижению заданного количества генераций поколений
  - Лучшее решение не изменяется на протяжении заданного количества поколений

Функция оценки полезности гена

Для каждой гипотезы h вычисляется 2 функционала:

- Функционал основанный на расстоянии Левенштейна

Лучшими гипотезами являются гипотезы с наименьшими значениями функционала:

$R_h = \sum_m DL(h, N_m)$, где *DL*(_,_) – расстояние Левенштейна, а N_m – N-граммы полученные из переданных в алгоритм набора строк.

- Функционал основанный на статистической оценке

Наилучшими гипотезами являются гипотезы с максимальной вероятностью существования, равной

$P = \prod_i P(S_i)$, где *P(Si)* - оценка вероятности появления символа *Si* на i позиции

$G_h = \sum_i H(S_i)$ - функционал вероятности существования гипотезы *h.*

Пример данных реального считывания номера 64715055:

- 61715
- *6471511
- 54**5051
- 645055
- 64*50551
- 55
- 645055
- 745055
- 505
- 615055
- 65055
- 715155
- 7150553
- 671515
- 6415013
- 71505
- 55
- *4715055
- 5155
- 74**5055
- 7552
- 74**0550
- 51105
- 71*50551
- 7**07**1
- 7*50550*
- 67
- *5*1055*
- 715*1*1*
- 7055
- 51553
- 5
- 6
- 77
- 6*71771*
- 57
- 79464
- 3**7**1*

Результат: 64715055

Модельные примеры для простых символьных последовательностей:

• Абракадабра • Абрф0адарца • Абракрдабр3а • Араадабра • бркадмабра • бр5ь>ак8!а(@р@а • Абра4кадабра • Абтралкьиара • Абакадйбрнпа • Аыйбржакадлазра • Бракжадабра	• mankind • mnkind • mankind • qankeinwd • makeinnd • ysnqjind • myankind • musninr • manayna • mnkmnd • mankifd • makinud • mavsiid • aurind • adkyind • apnsinad • mvnkind • manrkcknd • mangkind • mvanpxkid	• 11112222 • 111222 • 114012822 • 11223262 • 11521222 • 18612222 • 31131122272 • 1111222 • 1101222 • 111222 • 808111622212
Результат: Абракадабра	*Результат:* mankind	*Результат:* 11112222

Работа с составными последовательностями

- Сегментация строк в множество $W=\{w_{ij}\}$ подстрок, где j – номер подстроки в строке i, разбивается на множество массивов $W=\{W_j\}$ состоящие из групп подстрок $W_j=\{w_{ij}\}$. Для каждой подстроки из множества подстрок, определен функционал близости подстроки w_{ij} к массиву W_k, $k \neq j$:

$$R_{ik} = \sum_m DL(w_{ij}, w_{mk})$$

- Для каждой полученной группы применяется алгоритм описанный ранее. Если длина под последовательности неизвестна, то ищется наилучший вариант с разной длинной

Пример для составной строки:

- Кто понял жизнь, тот не спешит
- то понячл жизн, тот н сыешит
- Кто 3по[нялжитнь, тмт не@1кпешиат
- Кто понял гжьизнг, от н спьеит
- !)то п(онг жзннн3 т>о не_сфсп4еши6
- Ктиж :онял жизэь,тот нм ыпешит
- Ко оял сж*изньщф, 1от не пешит
- тхоё понял жизэь тот не сп2ешзт
- К2 п0онял жизнь, тот не сф&с^:ит
- Кзтоопнял жинь*ттнв ъсдеши
- Кто понял9 жвзино тот {не пешит

- Кто
- то
- Кто
- Кто
- !)то
- Ктиж
- Ко
- тхоё
- К2
- Кто

- понял
- понячл
- 3по[нялжитнь
- понял
- п(онг
- :онял
- оял
- понял
- п0онял
- Кзтоопнял
- понял9

- жизнь,
- жизн,
- гжьизнг,
- жзннн3
- жизэь,
- сж*изньщф,
- жизэь
- жизнь,
- жинь
- жвзино

Результат: Кто понял жизнь, тот не спешит

Результаты на модельных данных

Вероятность ошибки на символ	Длина строки	Длина выборки 10	Длина выборки 20	Длина выборки 30
0.3	10	0.875	0.985	0.99
0.2	15	0.875	0.98	0.99
0.15	20	0.86	0.975	0.99

Заключение

- Создан алгоритм не требует априорных знаний о считываемой информации, преимуществом данного алгоритма является его независимость от содержимого строк и алфавита. Для коррекции ошибок в словах текстовых строк не требуются словари. Алгоритм находит правильную гипотезу на 40-50 % точнее, чем при простом использовании статистики для поиска гипотезы.
- Дальнейшие направление работы – искусственное зашумление изображений специальным шумом и последующее применение алгоритма к распознанным последовательностям.

Литература:

- Ritika Mishra, Navjot Kaur A Survey of Spelling Error Detection and Correction Techniques// International Journal of Computer Trends and Technology- volume4 Issue3- 2013 (372)
- К.Шеннон Работы по теории информации и кибернетике // Издательство иностранной литературы, Москва1963 (275)
- Ю. В. Визильтер, И.В. Бекетова, С. Л. Каратеев, Н. А. Костромов, О. В. Выголов. Автоматическое распознавание железнодорожных номерных знаков на видеопоследовательностях // Вестник компьютерных и информационных технологий. 2014. №9. (3-9)
- В. Ю. Гудков, Е. Ф. Гудкова N-граммы в лингвистике // Вестник Челябинского государственного университета. 2011. № 24 (239).
- Ukkonen E. Approximate String Matching with q-Grams and maximal matches. // Theoretical Computer Science, vol. 92, № 1, 1992
- Бондаренко А.В., Визильтер Ю.В., Клышинский Э.С., Силаев Н.Ж., Максимов В.Ю., Мусаева Т.Н. Формальный метод нечеткого поиска персональной информации // Препринты ИПМ им. М.В.Келдыша. 2009. № 64. 25 с.
- Левенштейн В.И. Двоичные коды с исправлением выпадений, вставок и замещений символов. // Доклады Академий Наук СССР. Том 163, № 4, 1965.

Спасибо за внимание!