Расчёт параметров и информационных характеристик дискретных ИС

Выполнил: Горощеня Владислав Сергеевич, 3 курс 4 группа 1 подгруппа

2024

**Ход работы**

Алфавит – общее число знаков или символов, используемых для генерации или передачи сообщений. Его информационной характеристикой, т. е. источником сообщений на основе этого алфавита, является энтропия. Она отражает какое количество информации приходится в среднем на один символ. Данный термин был применён к техническим системам К. Шенноном и Р. Хартли.

По Шеннону энтропию высчитывают по следующей формуле:

Для вычисления энтропии сначала необходимо определить количество вхождений каждого из символов в строку. Программная реализация представлена в листинге 1.

Листинг 1. Реализация вычисления вхождений символов в строку.

|  |
| --- |
| map<char, int> CountSymbolsInText(string str) {  map<char, int>symbolsDictionary;  for (int i = 0; i < str.size(); i++) {  if (!symbolsDictionary.count(str[i])) {  symbolsDictionary.insert(make\_pair(str[i], 1));  }  else {  symbolsDictionary[str[i]] += 1;  }  }  return symbolsDictionary;  } |

Затем следует переход от подсчёта вхождений к вероятности их появления, что можно осуществить с помощью формулы Шеннона, приведённой выше. Её программная реализация представлена в листинге 2.

Листинг 2. Функция вычисления энтропии Шеннона

|  |
| --- |
| double ShannonEntropy(string str) {  map<char, int>symbolica = CountSymbolsInText(str);  double entropy = 0.0;  for (auto iter = symbolica.begin(); iter!=symbolica.end(); ++iter) {  double temp = (double)(iter->second) / str.size();  entropy -= temp\*log2(temp);  }  return entropy;  } |

В качестве данных для вычисления энтропии использовались алфавиты кириллический русский и латинский английский, на основе которых были сформированы текстовые документы, использующиеся для предоставления данных приложению.

Текстовый документ на русском представлен на рисунке 1.

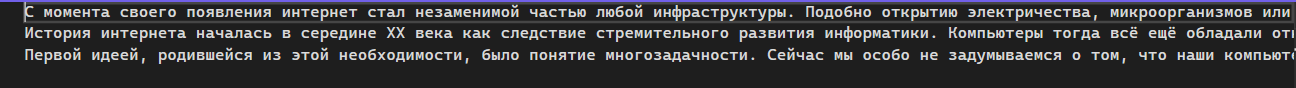


Рисунок 1 – Текст на русском языке

Текстовый документ на английском представлен на рисунке 2.

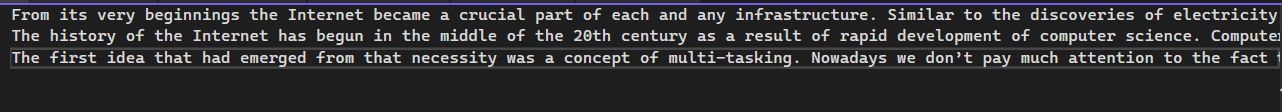


Рисунок 2 – Текст на английском языке

Также использовался бинарный алфавит, состоящий из 0 и 1 и записанный в текстовый документ.

На основе определения частоты встречаемости символов в данных текстах, можно построить гистограммы, отражающие эти показатели. Это представлено на рисунке 3.

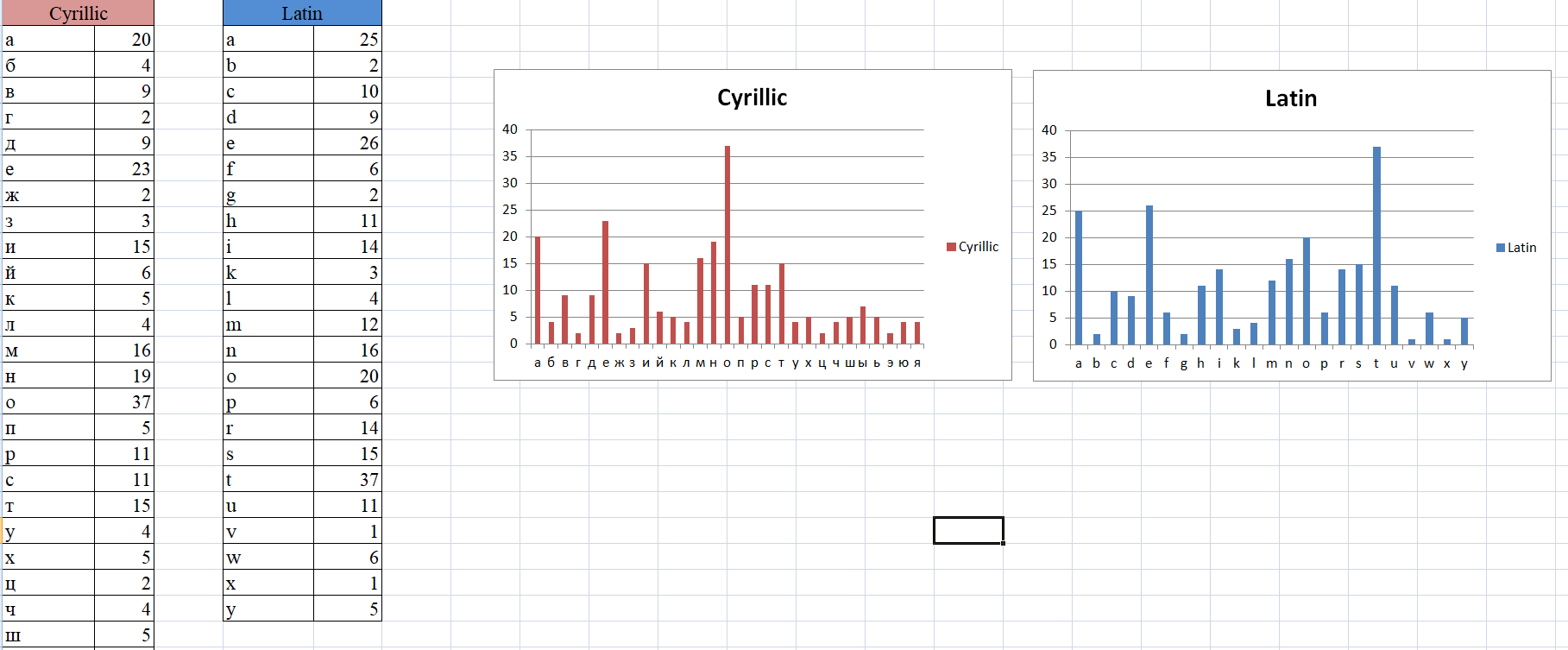


Рисунок 3 – Графики, отражающие частоту появления символов в алфавитах

Из этих данных можно получить вероятности появления каждого из символов . Подставляя их в формулу Шеннона, получаем результаты, приведённые на рисунке 4.

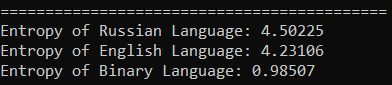


Рисунок 4 – Результаты подсчёта энтропии

При известной энтропии алфавита можно посчитать количество информации по следующей формуле:

Т.е. количество информации есть произведение энтропии и длины сообщения. Программная реализация представлена в листинге 3.

Листинг 3. Реализация подсчёта количества информации.

|  |
| --- |
| double InfoAmount(std::string alphabet, string str) {  return IsBinaryAlphabet(alphabet)?str.size(): ShannonEntropy(alphabet) \* str.size();  } |

Если алфавит является бинарным, то за количество информации принимается длина строки, ибо энтропия принимается за 1. Иначе энтропия домножается на количество символов в сообщении, чьё количество информации необходимо вычислить. Результаты представлены на рисунке 5.

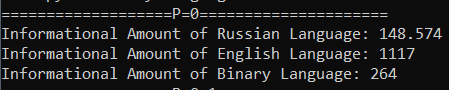


Рисунок 5 – Вывод количества информации

Важно подчеркнуть, что количество информации для последнего случая, включаещего в себя ФИО в кодировке ASCII, равняется количеству символов исходного сообщения: 264.

Возможны случаи ошибочной передачи сообщения, поэтому для вычисления количества информации используется формула эффективной энтропии:

где - условная энтропия:

.

Программная реализация функции вычисления эффективной энтропии представлена в листинге 5.

Листинг 5. Реализация функции вычисления эффективной энтропии.

|  |
| --- |
| double EffectiveEntropy(string str, double p) {  double q = 1 - p;  if (IsBinaryAlphabet(str) && (p == 0 || q == 0)) {  return 1;  }  else if (!IsBinaryAlphabet(str) && p == 1) {  return 0;  }  return 1 -( - p \* log2(p) - q \* log2(q));  } |

Как уже упоминалось, в бинарном алфавите эффективная энтропия равна длине строки, а при вероятности ошибки 0 или 1, и количеству информации соответственно. В небинарных, при вероятности ошибки 1, эффективная энтропия и количество информации равно 0.

Таким образом, функция подсчёта количества информации при наличии ошибок имеет вид, представленный в листинге 6.

Листинг 6. Функция подсчёта количества информации при наличии ошибок.

|  |
| --- |
| double InfoAmount(std::string alphabet, string str, double p) {  return ShannonEntropy(alphabet) \* str.size() \* EffectiveEntropy(alphabet, p);  } |

Результаты вычисления количества информации представлены на рисунке 6.

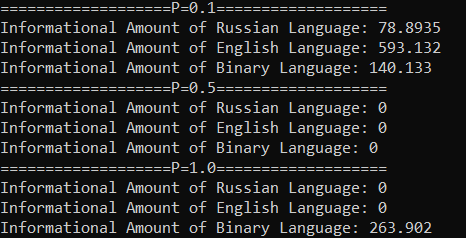


Рисунок 6 – Результаты вычисления количества информации при наличии ошибок

Как можно заметить, при вероятности ошибки =1, количество информации в бинарном алфавите практически равняется количеству информации при =0, что объясняется заменой всех битов сообщения на обратные, а в небинарном при =1 такое невозможно, ибо мощность алфавита больше 2, и каждый символ заменяется на другой произвольный из этого алфавита, поэтому и количество информации в таком случае равняется 0.