Εργασία Ψηφιακών Τηλεπικοινωνιών

Κυριακόπουλος Αριστομένης¹

Πανεπιστήμιο Πατρών Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής up1054307@upnet.gr

Περίληψη Σκοπός της εργασίας είναι η δημιουργία συναρτήσεων για την κατανόηση της κωδικοποίησης Ηυφφμαν, της κωδικοποίησης Π $^{\circ}$ Μ και την μελέτη απόδοσης ομόδηνου ζωνοπερατού συστήματος Μ-ΦΣΚ. Παρουσιάζονται παρακάτω τα αποτελέσματα της προσπάθειας μου.

1 Ερώτημα 1 - Κωδικοποίηση Huffman

Η χωδιχοποίηση Huffman αποτελεί μια βέλτιστη χωδιχοποίηση διαχριτών πηγών και χρησιμοποιείται για την συμπίεση δεδομένων χωρίς χάσιμο πληροφορίας. Παραχάτω παρατίθεται οι διχές μου θεμελιώσεις των 3 βασιχών συναρτήσεων.

1.1 Κωδικοποίηση πηγής κώδικα

Huffman Dictionary Η συνάρτηση huffmanDict λαμβάνει ως είσοδο 2 διανύσματα, το 1 με τους αγγλικούς χαρακτήρες και το άλλο με την πιθανότητα του κάθε γράμματος από το αρχείο evxopt.txt. Αρχικά ταξινομούνται όλες οι πιθανότητες σε αύξουσα σειρά και λαμβάνονται οι δύο χαρακτήρες με την μικρότερη πιθανότητα να εμφανιστούν. Αυτοί εισέρχονται σε ένα κοινό κόμβο με το άθροισμα των πιθανοτήτων τους και εισέρχονται ξανά στα αντίστοιχα διανύσματά τους. Αυτό γίνεται μέχρι η πιθανότητες να φτάσει στο 1. Για το δοθέν κείμενο η έξοδος της συνάρτησης είναι η εξής:

```
Columns 1 through 8
                                        (1d)
 {'0110'}
            {'101111'}
                           {'10001'}
                                        {'10110'}
                                                     {'110'}
                                                                {'100100'}
                                                                              {'101110'}
Columns 9 through 16
                                              6111
             {'1111110011'}
                              {'111111000'}
                                              {'01000'}
                                                                                    {'00000'}
                                                                                                {'111110'}
 {'0101'}
                                                           {'10000'}
                                                                        {'0111'}
Columns 17 through 24
 {'1111110010'}
                  {'1010'}
                                          {'0001'}
                                                      {'010010'}
                                                                    {'100101'}
                                                                                  {'0100110'}
                                                                                                 {'0100111'}
                              {'1110'}
Columns 25 through 27
 {'1111111'} {'1111101'}
                              {'001'}
```

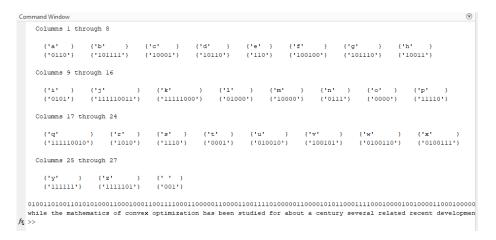
Eix 1: Huffman Dictionary Output.

Huffman Encoding Η συνάρτηση huffmanEnc λαμβάνει ως είσοδο το αρχείο που θέλουμε να κωδικοποιηθεί και το dictionary που δημιουργήθηκε από την συνάρτηση huffmanDict. Όσο το αρχείο εισόδου δεν είναι κενό, συγκρίνει έναν προς έναν τους χαρακτήρες με την αντίστοιχη κωδικοποίησή τους και δημιουργεί ένα διάνυσμα εξόδου με την ακολουθία από bit που το αναπαριστούν. Τα αρχικά ψηφία φαίνοντα παρακάτω και μπορούν να επαληθευτούν από το dictionary.

```
Command Window
  Columns 1 through 8
    {'a' } {'b' }
{'0110'} {'101111'}
                        {'c' }
{'10001'}
                                  {'10110'}
    {'0101'} {'111110011'}
                           {'11111000'}
                                       {'01000'}
                                                 {'10000'}
                                                           {'0111'}
                                                                    {'00000'}
                                                                             {'111110'}
   Columns 17 through 24
    {'1111110010'}
                                                                              {'0100111'}
                 {'1010'}
                          {'1110'}
                                    {'0001'}
                                             {'010010'}
                                                        {'100101'}
                                                                  {'0100110'}
   Columns 25 through 27
     ('1111111') ('1111101')
```

Eix 2: Huffman Encoding Output.

Huffman Decoding Η συνάρτηση huffmanDec λαμβάνει ως είσοδο το κωδικοποιημένο διάνυσμα όπως επίσης και το dictionary που έχει δημιοργηθεί. Διαβάζοντας έναν έναν τα bit εισόδου, δημιουργεί ένα προσωρινό λεξιλόγιο με σκοπό να ταυτοποιήσει την ακολουθία από bit που αντιστοιχεί στο κάθε γράμμα. Όταν το εντοπίσει, το γράμμα εισέρχεται στο vector εξόδου. Παρακάτω φαίνονται τα αποτελέσματα όλων των συναρτήσεων:



Eix 3: Huffman Decoding Output.

1.2 Αποδοτικότητα κώδικα

 Γ ια να υπολογίσουμε την αποδοτικότητα του κώδικα, πρέπει πρώτα να υπολογιστούν η εντροπία του κώδικα όπως επίσης και το μέσο μήκος κώδικα. Ο τύπος της εντροπίας είναι:

$$\sum iPx(i)*log(Px(i)) \tag{1}$$

και ο τύπος του μέσου μήκους κώδικα είναι:

$$\sum iPx(i) * Length(i)) \tag{2}$$

Υπολογίζοντας αυτά μπορώ να βρώ την αποδοτικότητα του κώδικα αφού αποτελεί απλώς μια διαίρεση μεταξύ εντροπίας και μήκους. Τα αποτελέσματα φαίνονται παρακάτω:

```
while the mathematics of convex optimization has been s
entropy =
4.1347

avgLenCode =
4.1630

efficiency =
0.9932
```

Eux 4: Huffman Efficiency

Όπως φαίνεται η αποδοτικότητα του κώδικα είναι στο 99% το οποίο είναι και ένα αποτέλεσμα που περιμέναμε αφού οι πιθανότητες που χρησιμοποιώ προέρχονται από το κείμενο που έχει κωδικοποιηθεί

1.3 Κωδικοποίηση με πιθανότητες συμβόλων αγγλικής γλώσσας

Χρησιμοποιώντας το αρχείο frequencies.txt για τις πιθανότητες των συμβόλων λαμβάνω το εξής αποτέλεσμα:

```
entropy =
     4.1679

avgLenCode =
     4.1875

efficiency =
     0.9953

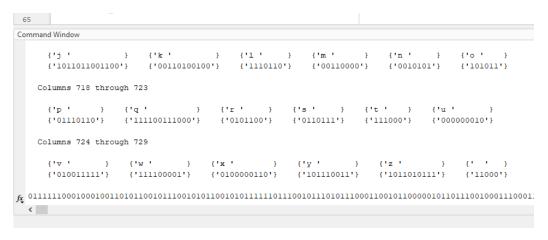
ft >>
```

Eix 5: Huffman Efficiency

Παρατηρώ ότι το μέσο μήχος χώδικα είναι λίγο μεγαλύτερο, το οποίο οφείλεται στο γεγονός ότι οι δοσμένες πιθανότητες είναι γενικευμένες για την αγγλική γλώσσα και δεν είναι προσαρμοσμένες πάνω στο δοθέν κείμενο.

1.4 Δεύτερη τάξη επέχταση πηγής Α

Αρχικά δημιουργώ την δεύτερη τάξη επέκταση πηγής Α, η οποία είναι όλοι οι συνδυασμοί γραμμάτων μεταξύ τους και υπολογίζω τις πιθανότητες κάθε συνόλου 2 χαρακτήρων πολλαπλασιάζοντας τις πιθανότητες του καθενός. Στο τέλος καταλήγω με 729 συνδυασμούς γραμμάτων (δηλαδή 262). Η τελική κωδικοποιηση φαίνεται παρακάτω μαζί με τα τελευταία ζεύγη του λεξιλογίου:



Eix 6: Huffman Second Order

Παρατηρώ ότι παρόλο που το μέσο μήχος χώδιχα έχει διπλασιαστεί, το οποίο είναι και ένα αποτέλεσμα που περιμέναμε λόγω της αύξησης των χαραχτήρων από 27 σε 729, η απόδοση της χωδιχοποίησης είναι μεγαλύτερη και η συμπίεση του χειμένου είναι μεγαλύτερη. Επίσης η εντροπία μια διαχριτής πηγής με την αντίστοιχη η τάξης επέχτασής της αχολουθεί τον τύπο:

$$H(\Phi n) = nH(\Phi) \tag{3}$$

Το οποίο επαληθεύεται με τα δεδομένα μας, αφού η εντροπία της δεύτερης τάξης επέχταση της πηγής A είναι διπλάσια.

1.5 Κωδικοποίηση εικόνας και χρήση δυαδικού συμμετρικού καναλιού

Αρχικά φορτώνω σε ένα struct το αρχείο cameraman.mat και υπολογίζω τις πιθανότητες των τιμών 0-255 που αποτελούν την κλίμακα του γκρι. Έπειτα δημιουργείται το dictionary για την εικόνα και κωδικοποιείται η πηγή. Για να βρούμε την

πιθανότητα p κάποιου bit να έχει μετατραπεί από 0 σε 1 ή από 1 σε 0 αρκεί να βρούμε την απόσταση hamming των δύο διανυσμάτων. Έπειτα, γνωρίζοντας την πιθανότητα μπορούμε να υπολογίσουμε την δυαδική εντροπία και τέλος η χωρητικότητα του καναλιού είναι η

$$C = 1 - H(p) \tag{4}$$

Τα αποτελέσματά μου φαίνονται παρακάτω:

```
95
96
97
98
H = -p*log2(p) - (1-p)*log2(1-p)
98
100
Command Window

p =

0.1799

H =

0.6798

C =

0.3202
```

Eux 7: Channel Capacity