Πανεπιστήμιο Πατρών

Τμήμα Μηχ. Η/Υ & Πληροφορικής

Ψ Η Φ Ι Α Κ Ε Σ Τ Η Λ Ε Π Ι Κ Ο Ι Ν Ω Ν Ι Ε Σ

Εργαστηριακές Ασκήσεις 2021-2022 : 1ο σετ

*Φοιτητής: Τσιαμήτρος Κωνσταντίνος - ΑΜ 235913 – 9ο έτος*

# Ερώτημα 1 – Κωδικοποίηση Huffman

1. Υλοποιήθηκαν οι ζητούμενες συναρτήσεις.

Συγκεκριμένα, αναπτύχθηκαν οι εξής συναρτήσεις:

1. function [c] = my\_huffman\_dict(symbols, probs)

* κατασκευάζει και επιστρέφει μια κωδικοποίηση Huffman (c) (με βάση τον αλγόριθμο του βιβλίου των Proakis/Salehi στη σελίδα 686) για τα σύμβολα (symbols) και τις πιθανότητες εμφάνισής τους (probs)

1. function [code, avg\_len] = my\_huffman\_enco(tex, symbols, probs)

* κωδικοποιεί το κείμενο εισόδου (tex) σε Huffman κωδικοποίηση με τη βοήθεια της παραπάνω συνάρτησης. Υπολογίζει και επιστρέφει το κωδικοποιημένο κείμενο (code), καθώς και το μέσο μήκος κώδικα (avg\_len).

1. function [sig] = my\_huffman\_deco(code, symbols, probs)

* αποκωδικοποιεί το κωδικοποιημένο κείμενο εισόδου (code) και επιστρέφει το αρχικό κείμενο (sig).

1. Εκτιμήθηκαν οι πιθανότητες των συμβόλων της πηγής Α με βάση το κείμενο που δόθηκε.

Συγκεκριμένα, αναπτύχθηκε η συνάρτηση:

function [letters, freqs, tex] = calculate\_freqs()

η οποία επιστρέφει τα σύμβολα (letters), τις πιθανότητες εμφάνισης (freqs) καθώς και το κείμενο (tex – το οποίο διαβάζεται από ένα - hardcoded - αρχείο).

Η Εντροπία της (διακριτής) πηγής (χωρίς μνήμη - DMS) υπολογίζεται ως εξής:

Το μέσο μήκος κώδικα υπολογίζεται ως εξής:

Η αποδοτικότητα του κώδικα υπολογίζεται ως εξής:

Ο λόγος αυτός είναι πάντοτε μικρότερος της μονάδας, όπου n = 1 είναι το ιδανικό βέλτιστο.

Εκτελώντας το αρχείο erwtima1\_erwtisi2.m καλούνται οι παραπάνω συναρτήσεις, σειριακά, και παράγονται τα ακόλουθα μηνύματα:

Original text: while the mathematics of convex optimization has been studied…

Huffman Encoding: 0111111111111111111111011111111111101111011111111101001110111111111111010011…

Decoded signal: while the mathematics of convex optimization has been studied…

Source Entropy: 4.1347

Average Length of code: 13.963

Code Efficiency: 0.29612

Η παραπάνω έξοδος, επιβεβαιώνει την σωστή αποκωδικοποίηση του κειμένου. Επίσης, υπολογίστηκαν η Εντροπία της πηγής, το μέσο μήκος κώδικα καθώς και η αποδοτικότητα του κώδικα.

Παρατηρήσεις:

1. μια βελτίωση στον χρόνο εκτέλεσης της αποκωδικοποίησης του κειμένου θα ήταν να αξιοποιηθεί κάποιο suffix tree, όμως καθαρά για λόγους rapid development αποσιωπήθηκε, δεδομένου ότι οι διαδικασίες κωδικοποίησης-αποκωδικοποίησης απαιτούν χρόνο μικρότερο του ενός λεπτού
2. οι έξοδοι αποσιωπήθηκαν λόγω του μεγάλου μήκους τους
3. Τροποποιήθηκε η συνάρτηση

function [letters, freqs, tex, counts] = calculate\_freqs(f)

ώστε να δέχεται ένα Boolean όρισμα, και αν είναι True, τότε φορτώνει τις πιθανότητες εμφάνισης από το αρχείο frequencies.txt αντί να τις υπολογίζει με βάση το κείμενο.

Τρέχοντας το αρχείο erwtima1\_erwtisi3.m παράγεται η παρακάτω έξοδος:

Original text: while the mathematics of convex optimization has been studied…

Huffman Encoding: 0111111111111111111111101111111011111111011111111111011111111111111111111111111111101111111111111111111011111110111111111111111111111111111111…

Decoded signal: while the mathematics of convex optimization has been studied…

Source Entropy: 4.1679

Average Length of code: 13.963

Code Efficiency: 0.2985

Παρατηρώ, ότι αυξήθηκε λίγο η αποδοτικότητα του κώδικα, λόγω της αύξησης του λόγου της εντροπίας προς το μέσο μήκος κώδικα (αυξήθηκαν και οι δύο ποσότητες, όπως και ο λόγος τους συνεπώς, η εντροπία της πηγής αυξήθηκε πιο πολύ – σχετικά μιλώντας πάντα – από το μέσο μήκος του κώδικα, με βάση τις νέες πιθανότητες εμφάνισης των συμβόλων).

1. Θεωρώντας ότι τα σύμβολα της πηγής είναι ανεξάρτητα μεταξύ τους ανά δύο, τότε:

Συνεπώς, κατασκευάζουμε τον δυσδιάστατο πίνακα μεγέθους Ν (στο παράδειγμα για Ν=3 για λόγους χώρου):

|  |  |  |  |
| --- | --- | --- | --- |
|  | S0 | S1 | S2 |
| S0 | P(S0)\*P(S0) | P(S0)\*P(S1) | P(S2)\*P(S2) |
| S1 | P(S0)\*P(S1) | P(S1)\*P(S1) | P(S1)\*P(S2) |
| S2 | P(S0)\*P(S2) | P(S1)\*P(S2) | P(S2)\*P(S2) |

Ο παραπάνω πίνακας θα περιέχει τις πιθανότητες εμφάνισης των συμβόλων ανά δύο. Δηλαδή τις πιθανότητες εμφάνισης των συμβόλων μιας 2ης τάξης επέκτασης πηγής.

Οι μετρικές υπολογίζονται με τον ίδιο τρόπο, μόνο που τώρα, έχουμε δυάδες συμβόλων αντί για μεμονωμένα σύμβολα.

Εκτελώντας το αρχείο erwtima1\_erwtisi4.m παράγεται η παρακάτω έξοδος:

Source Entropy: 8.3383

Average Length of code: 364.9986

Code Efficiency: 0.022845

# Ερώτημα 2 – Κωδικοποίηση PCM

1. Μη ομοιόμορφος βαθμωτός κβαντιστής

Υλοποιήθηκε ο αλγόριθμος Lloyd-Max (η κλήση των υπεύθυνων συναρτήσεων γίνεται με τη βοήθεια του αρχείου erwtima1\_erwtisi1 που υπάρχει στον φάκελο ask2)

1. function [xq, MSE, SNR, D] = Lloyd\_Max(x, N, min\_value, max\_value, tol)

Η συνάρτηση αυτή έχει τη ζητούμενη μορφή της εκφώνησης, με μια διαφορά. Αντί να επιστρέφει τα centers σε ένα 2xΝ μητρώο, επιστρέφει δύο διανύσματα, xs και x\_hats όπου το διάνυσμα xs περιέχει τις τιμές του x άξονα και το x\_hats περιέχει τις τιμές του y άξονα των centers.

1. function [xq, D] = helper\_func(x, a\_i, x\_hat)

Λόγω της επαναληπτικής φύσεως του αλγορίθμου, υλοποιήθηκε αυτή η συνάρτηση, η οποία εκτελεί τους κατάλληλους υπολογισμούς για τον αλγόριθμο Lloyd-Max, σύμφωνα με τις οδηγίες του βιβλίου (Proakis/Salehi Κεφ. 7) και της εκφώνησης

1. function [] = my\_plot (x, xq, x\_hat, MSE, SNR)

Η συνάρτηση αυτή, κάνει plot τα input arguments με κατάλληλο τρόπο (οπτικοποίηση)

1. function [MSE, SQNR] = metrics(x, xq)

Η συνάρτηση αυτή υπολογίζει τις μετρικές MSE & SQNR και τις επιστρέφει στο κύριο πρόγραμμα.

1. function [xq, x\_hat, D] = helper\_func2(x, a\_i, min\_value, max\_value)

Η συνάρτηση αυτή εκτελεί τον επαναληπτικό αλγόριθμο Lloyd-Max και επιστρέφει τα αποτελέσματα στο κύριο πρόγραμμα

Για rng(2) οι παραπάνω συναρτήσεις παράγουν την παρακάτω έξοδο:







