ห้องสมุดคณะวิศวกรรมศาสตร์
มหาวิทยาลัยนเรศวร

การศึกษาและจำลองการบีบอัดข้อมูลโดยอาศัยกระบวนการของฮัฟแมน

Study and Simulation of Data Compression using Huffman Encoding

นางสาววาสนา นากุ รหัส 47364070

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 1 7 ส.ค. 2553
เลขทะเบียน 15008815
เลขเรียกหนังสือ ปร. 94110 2550
มหาวิทยาลัยนเรศวร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2550

ใบรับรองโครงงานวิศวกรรม

หัวข้อดำเนินโครงงาน	การศึกษาและจำลองการบีบอัดข้อมูลโดยอาศัยกระบวนการของฮัฟแมน
ผู้ดำเนินโครงงาน	นางสาววาสนา นากุ รหัส 47364070
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.สุรเชษฐ์ กานต์ประชา
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2550

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

คณะกรรมการสอบโครงงานวิศวกรรม

....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.สุรเชษฐ์ กานต์ประชา)

....กรรมการ
(ดร.ชัยรัตน์ พินทอง)

....กรรมการ
(อาจารย์แสงชัย มังกรทอง)

หัวข้อดำเนินโครงงาน	การศึกษาและจำลองการบีบอัดข้อมูลโดยอาศัยกระบวนการของฮัฟแมน
ผู้ดำเนินโครงงาน	นางสาววาสนา นากุ รหัส 47364070
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.สุรเชษฐ์ กานต์ประชา
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2550

....

บทคัดย่อ

การสื่อสารข้อมูลมีความสำคัญในระบบสื่อสาร และได้ถูกพัฒนาให้ดีขึ้นเป็นลำดับ ระบบดิจิตอลได้นำมาใช้ในการส่งข้อมูลจากที่หนึ่งไปยังอีกที่หนึ่ง เช่น การสื่อสารข้อมูลระหว่างคอมพิวเตอร์ การรับข่าวสารจากอินเตอร์เน็ต และอื่น ๆ ข้อดีของการสื่อสารระบบดิจิตอล คือ สามารถที่จะแก้ไขบิตผิดพลาดที่ปลายทางได้ ซึ่งจะทำให้ผลของสัญญาณรบกวนลดลงหรือหมดไป โดยที่ในระบบอนาลอกไม่สามารถทำได้ และในการส่งข้อมูลทางดิจิตอลสามารถที่จะกำหนดขนาดของข้อมูลได้ โดยข้อมูลของดิจิตอลจะเป็นตัวกำหนดขนาดของแบนด์วิด กล่าวคือเมื่อข้อมูลมีจำนวนมากก็จะทำให้แบนด์วิดมีขนาดใหญ่ การลดขนาดของแบนด์วิดในขณะที่ปริมาณของข้อมูลที่ต้องการจะส่งยังคงเดิมเป็นสิ่งที่สำคัญในระบบสื่อสาร ซึ่งสามารถทำได้หลายวิธี การบีบอัดข้อมูลก็เป็นอีกวิธีหนึ่งที่นิยมนำมาใช้

โครงงานนี้เป็นการสร้างแบบจำลองการบีบอัดข้อมูลด้วยวิธีฮัฟแมนด้วยโปรแกรม MATLAB ซึ่งแสดงผลของโปรแกรมออกมาในรูปแบบ GUI (Graphic User Interface) โดยข้อมูลต่าง ๆ ที่แสดงออกมาได้แก่ จำนวนสัญลักษณ์ที่ใช้ในการบีบอัด ข้อมูลหรือไฟล์ข้อมูลที่จะทำการบีบอัด พจนานุกรม ความยาวของพจนานุกรม และอัตราการบีบอัดของข้อมูล ดังนั้นประสิทธิภาพในการบีบอัดข้อมูลและพารามิเตอร์ที่สำคัญได้ถูกแสดงในรูปของกราฟพร้อมทั้งได้อธิบายไว้ในโครงงานเล่มนี้

จากการศึกษาพบว่า สามารถสร้างโปรแกรมจำลองการบีบอัดของข้อมูลด้วยวิธีฮัฟแมน ซึ่งผลของอัตราในการบีบอัดและประสิทธิภาพของการบีบอัดของแต่ละชนิดมีความแตกต่างกัน คือถ้าจำนวนสัญลักษณ์ที่ใช้ในการบีบอัดมากขึ้น อัตราในการบีบอัดก็จะลดลงและมีค่าเข้าใกล้กับปริมาณข่าวสารเฉลี่ย ทำให้ประสิทธิภาพมีค่าเข้าใกล้หนึ่ง อย่างไรก็ตามก็ยังมีข้อเสีย นั่นก็คือ เมื่อจำนวนสัญลักษณ์ที่ใช้ในการบีบอัดมากขึ้น ขนาดของพจนานุกรมก็จะมีขนาดใหญ่ซึ่งก็จะส่งผลให้แบนด์วิดมีขนาดใหญ่อีกด้วย

Project Title	Study and Design about Data Compression by Huffman Encoding
Name	Miss Wassana Naku ID. 47364070
Project Advisor	Assistant Professor Surachet Kanprachar, Ph.D.
Major	Electrical Engineering
Department	Electrical and Computer Engineering
Academic Year	2007

....

ABSTRACT

Data communications is very important in communication system and has been developed in order to get a better service. To transmit data from one terminal to another, digital system is adopted; for example, to communicate between two computers, to get information from the Internet, and so on. The advantage of digital system is that the received error bits can be recovered at the receiving end; hence, the effect of the noise in the system is cancelled. This is not achieved in the analog system. The amount of the digital information is used to define the transmission bandwidth; that is, if the bit rate is high, the required transmission bandwidth becomes larger. To reduce the transmission bandwidth while keeping the complete information sent, some techniques have to be applied. Data compression is one of the techniques used.

The simulation of data compression using Huffman algorithm is studied in this project. The computer program is written by using MATLAB and the result of the program is displayed in the form of GUI (Graphic User Interface). The information available in GUI includes the number symbols per compression, to be compressed data or file, dictionary and its length, and code rate. Also, in the project, the efficiency of the compression and other important parameters are plotted and explained.

From the study, it is found that the written computer program can perform the data compression using Huffman algorithm. The code rate and efficiency of the compression for different types of data are achieved. If the number of symbols per compression is increased, the code rate is reduced and close to the entropy; that is, the efficiency approaches 1. However, there is one drawback of using large number of symbols per compression; that is, as the number of symbols per compression increases, the size of the dictionary becomes larger and can cause larger transmission bandwidth in sending the dictionary

กิตติกรรมประกาศ

โครงงานนี้ได้สำเร็จลุล่วงไปได้ด้วยดี ซึ่งได้รับความช่วยเหลือและสนับสนุนจากหลาย ๆ ท่านด้วยกัน ดังต่อไปนี้

ขอขอบคุณ ผศ.ดร.สุรเชษฐ์ กานต์ประชา อาจารย์ที่ปรึกษาโครงงานในการให้คำแนะนำความรู้ และคำปรึกษา ที่เป็นประโยชน์อย่างมากในการทำโครงงานเล่มนี้ ผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์เป็นอย่างดียิ่งและขอกราบขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ผู้ให้กำเนิด พร้อมกับส่งเสริมและให้กำลังใจแก่ผู้จัดทำในโครงงานเล่มนี้มาโดยตลอด

ขอบคุณเพื่อนทุกคนที่คอยให้ความช่วยเหลือ ให้กำลังใจและคำปรึกษาจนโครงงานเล่มนี้สำเร็จไปได้ด้วยดี

ประโยชน์และคุณค่าของโครงงานเล่มนี้ ผู้จัดทำขอมอบแด่ผู้มีพระคุณทุกท่านไว้ ณ โอกาสนี้

ผู้จัดทำโครงงาน

วาสนา นากุ

มหาวิทยาลัยนเรศวร

สารบัญ

หน้า

บทคัดย่อภาษาไทย .... ก
บทคัดย่อภาษาอังกฤษ .... ข
กิตติกรรมประกาศ .... ค
สารบัญ .... ง
สารบัญตาราง .... ฉ
สารบัญรูปภาพ .... ช

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงงาน 1
1.2 วัตถุประสงค์ของโครงงาน 1
1.3 ขอบเขตของโครงงาน 2
1.4 ขั้นตอนการดำเนินโครงงาน 2
1.5 ผลที่คาดว่าจะได้รับ 2
1.6 ตารางการดำเนินงาน 3
1.7 งบประมาณที่ต้องใช้ 3

บทที่ 2 หลักการ และทฤษฎีของระบบสื่อสาร

2.1 ทฤษฎีข่าวสาร 5
2.2 การนิยามปริมาณข่าวสารและเอนโทรปี 7
2.3 ช่องสัญญาณ 10
2.4 ข่าวสารร่วม 11
2.5 ความจุของช่องสัญญาณ 14
2.6 การเข้ารหัสแบบฮัฟแมน 18
2.7 การเข้ารหัสแบบเลมเปลซิฟ 22

บทที่ 3 การออกแบบโครงงาน และวิธีการดำเนินงาน

3.1 ศึกษาทฤษฎีที่เกี่ยวข้อง 24
3.2 ศึกษาการใช้โปรแกรม MATLAB 24

สารบัญ (ต่อ)

หน้า

3.3 การออกแบบการเขียนโปรแกรมเพื่อใช้ในการบีบอัดข้อมูล 25

3.4 การเขียนโปรแกรม 26

3.5 ทดสอบโปรแกรม 26

3.6 วิเคราะห์ผลการทดสอบโปรแกรม 26

บทที่ 4 ผลการดำเนินโครงงาน

4.1 โปรแกรมแสดงการบีบอัดข้อมูลโดยวิธีฮัฟแมนพร้อมกับแสดงผลออกมา
ในรูปของ Graphic User Interface (GUI) 28

4.2 ตัวอย่างในการทดลองโปรแกรมพร้อมกับตารางที่ได้จากการทดลอง 31

4.3 กราฟแสดงการเปรียบเทียบการบีบอัดข้อมูลของแต่ละสัญลักษณ์ในแต่ละไฟล์ 35

4.4 กราฟแสดงประสิทธิภาพการบีบอัดข้อมูลของแต่ละสัญลักษณ์ในแต่ละไฟล์ 47

บทที่ 5 สรุปผลการดำเนินงาน

5.1 ผลการดำเนินโครงงาน 55

5.2 ปัญหาที่พบขณะดำเนินโครงงาน 55

5.3 ข้อเสนอแนะ 56

เอกสารอ้างอิง 57

ภาคผนวก ก 58

ภาคผนวก ข 63

ประวัติผู้ดำเนินโครงงาน 77

สารบัญตาราง

ตารางที่		หน้า
2.1	แสดงการเข้ารหัสฮัฟแมน	19
2.2	แสดงการเข้ารหัสฮัฟแมนด้วยสัญลักษณ์ 2 ตัว	21
2.3	แสดงการเข้ารหัสแบบเลมเปลซิฟ	23
4.1	แสดงค่าต่าง ๆ ของข้อมูลที่ได้จากการทดลองการบีบอัดข้อมูลแบบฮัฟแมน	34
4.2	แสดงประสิทธิภาพของการบีบอัดข้อมูลแบบฮัฟแมน	36

มหาวิทยาลัยนเรศวร

สารบัญรูป

รูปที่		หน้า
2.1	รูปแบบของระบบสื่อสารที่เสนอโดยแชนนอน	6
2.2	รูปแบบของระบบสื่อสารข้อมูลที่เสนอโดยฟาโน	7
2.3	แสดงแผนภูมิการเข้ารหัสฮัฟแมน	19
3.1	แสดงแผนผังการทำงานของโปรแกรม	25
4.1	หน้าต่างที่ใช้ป้อนข้อมูลที่ต้องการจะบีบอัดและแสดงผลจากการทำงาน	29
4.2	แสดงไฟล์ชื่อ Entertainment	31
4.3	ตัวอย่างการบีบอัดข้อมูลแบบฮัฟแมนโดยใช้การบีบอัดทีละ 3 สัญลักษณ์ โดยการค้นหาไฟล์ชื่อ entertainment	32
4.4	การแสดงผลของโปรแกรมออกมาในรูปของไฟล์	33
4.5	แสดงการเปรียบเทียบระหว่างอัตราการบีบอัดของแต่ละสัญลักษณ์กับอัตราเฉลี่ยของข้อมูล Entertainment	38
4.6	การเปรียบเทียบของข้อมูล Entertainment ที่ได้หลังจากการบีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน	39
4.7	แสดงการเปรียบเทียบระหว่างอัตราการบีบอัดกับอัตราเฉลี่ยของข้อมูล News	40
4.8	การเปรียบเทียบของข้อมูล News ที่ได้หลังจากการบีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน	41
4.9	แสดงการเปรียบเทียบระหว่างอัตราการบีบอัดกับอัตราเฉลี่ยของข้อมูล Sportnews	42
4.10	การเปรียบเทียบของข้อมูล Sportnews ที่ได้หลังจากการบีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน	43
4.11	แสดงจำนวนของข้อมูลที่แตกต่างกันซึ่งมีการบีบอัดในจำนวน 1 สัญลักษณ์	45
4.12	แสดงจำนวนของข้อมูลที่แตกต่างกันซึ่งมีการบีบอัดในจำนวน 2 สัญลักษณ์	46
4.13	แสดงจำนวนของข้อมูลที่แตกต่างกันซึ่งมีการบีบอัดในจำนวน 3 สัญลักษณ์	47
4.14	แสดงประสิทธิภาพของการบีบอัดข้อมูลของข้อมูล Entertainment	48
4.15	แสดงประสิทธิภาพของการบีบอัดข้อมูลของข้อมูล News	49
4.16	แสดงประสิทธิภาพของการบีบอัดข้อมูลของข้อมูล Sportnews	50
4.17	แสดงประสิทธิภาพของข้อมูลที่แตกต่างกันเมื่อมีการบีบอัดด้วยจำนวน 1 สัญลักษณ์	51
4.18	แสดงประสิทธิภาพของข้อมูลที่แตกต่างกันเมื่อมีการบีบอัดด้วยจำนวน 2 สัญลักษณ์	52
4.19	แสดงประสิทธิภาพของข้อมูลที่แตกต่างกันเมื่อมีการบีบอัดด้วยจำนวน 3 สัญลักษณ์	53

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

จุดประสงค์หลักของการสื่อสาร คือการส่งข่าวสารจากแหล่งกำเนิดไปยังแหล่งรับข่าวสารให้ได้ปริมาณมาก รวดเร็ว และถูกต้องที่สุด แต่ประเด็นหนึ่งที่จะต้องถูกนำมาพิจารณาคือช่องกว้างทางความถี่ หรือแบนด์วิดที่ต้องใช้ในการส่งสัญญาณ เนื่องจากมีจำนวนจำกัดที่จะใช้ในการสื่อสาร ซึ่งถือว่าแบนด์วิดเป็นทรัพยากรสำคัญอย่างหนึ่งของระบบสื่อสาร ซึ่งในการทำการส่งข่าวสารนั้นต้องผ่านช่องสัญญาณการสื่อสาร ซึ่งในช่องสัญญาณนี้มักจะมีสัญญาณรบกวนเข้ามา ซึ่งทำให้ข่าวสารเกิดความผิดพลาด จึงทำให้ส่งผลกระทบต่อการรับข่าวสารหรือทำให้สมรรถนะและประสิทธิภาพในการรับส่งข่าวสารน้อย แล้วยังทำให้เกิดการสิ้นเปลืองแบนด์วิดอีก จึงทำให้ต้องมีการศึกษาค้นคว้าในเรื่องของทฤษฎีข่าวสารให้เข้าใจอย่างถ่องแท้ เพื่อที่จะนำไปศึกษาถึงการบีบอัดข้อมูลในแต่ละชนิด เนื่องจากว่าการเข้ารหัสแต่ละชนิดจะให้อัตราในการส่งและปริมาณของข่าวสารที่ต่างกัน ในการส่งข่าวสารแต่ละชนิดอาจจะมีข้อมูลหรือคำที่ซ้ำกัน ซึ่งสามารถที่จะบีบอัดข่าวสารพวกนี้ได้เพื่อให้มีขนาดเล็กลง เพื่อให้มีพื้นที่ในการส่งข่าวสารหรือข้อมูลอื่น ๆ ได้อีก ทำให้ไม่เปลืองแบนด์วิดหรือช่องสัญญาณ เมื่อทราบว่าแต่ละชนิดมีความสามารถหรือประสิทธิภาพในการบีบอัดได้มากหรือน้อย ก็จะทำให้สามารถเลือกวิธีเหล่านั้นได้ตามความเหมาะสม ดังนั้น ต้องศึกษาถึงการบีบอัดข้อมูลร่วมกับทฤษฎีข่าวสาร เพื่อที่จะได้ข่าวสารที่มีปริมาณมากและรวมถึงการประหยัดพื้นที่ของช่องสัญญาณอีกด้วย

1.2 วัตถุประสงค์ของโครงงาน

1. ศึกษาการบีบอัดข้อมูลแบบฮัฟแมน (Huffman coding)
2. สร้างโปรแกรมเพื่อจำลองการทำงานของการบีบอัดข้อมูล
3. เปรียบเทียบประสิทธิภาพของการบีบอัดข้อมูลแบบฮัฟแมน (Huffman coding) เมื่อเปลี่ยนแปลงจำนวนสัญลักษณ์ (symbols) ในการบีบอัด
4. เปรียบเทียบประสิทธิภาพการบีบอัดข้อมูลแบบฮัฟแมนกับปริมาณข่าวสาร (Entropy)

1.3 ขอบข่ายของการทำโครงงาน

1. สามารถบอกความแตกต่างของการบีบอัดข้อมูลของฮัฟแมน (Huffman coding)

2. สามารถที่จะเปรียบเทียบปริมาณข่าวสารที่ได้จากการบีบอัดข้อมูลของฮัฟแมน (Huffman coding) ในแต่ละสัญลักษณ์

3. สามารถบอกถึงประสิทธิภาพในการบีบอัดข้อมูลของฮัฟแมน (Huffman coding) ของแต่ละสัญลักษณ์

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาปัญหาที่เกี่ยวกับการเข้ารหัส
2. ศึกษาวิธีการและทฤษฎีที่จะนำไปใช้ปัญหาที่เกิดขึ้น คือ
   - ทฤษฎีข่าวสาร
   - การเข้ารหัสแบบฮัฟแมน (Huffman coding)
3. ศึกษาโปรแกรม MATLAB
4. ทดลองทฤษฎีที่ใช้ในการจำลองโดยใช้โปรแกรม MATLAB
5. เขียนโปรแกรมเพื่อจำลองการบีบอัดแบบฮัฟแมน
6. ทดลองการทำงานของโปรแกรมที่เขียน และทำการบีบอัดข้อมูลในลักษณะต่าง ๆ
7. การวิเคราะห์ประสิทธิภาพของการบีบอัดในลักษณะต่าง ๆ และเปรียบเทียบกับปริมาณข่าวสาร
8. สรุปผลการทดลองและทำโครงงานเป็นรูปเล่ม

1.5 ผลที่คาดว่าจะได้รับ

1. ทำให้เกิดความรู้และความเข้าใจในทฤษฎีที่ศึกษามากขึ้นเพื่อนำไปประยุกต์ใช้หรือนำไปศึกษาในด้านวิศวกรรมไฟฟ้าสื่อสารต่อไป

2. ทำให้เกิดความรู้และความสามารถในการใช้โปรแกรม MATLAB มากยิ่งขึ้น

3. สามารถที่จะเปรียบเทียบการบีบอัดข้อมูลในลักษณะต่างๆ ของการบีบอัดข้อมูลแบบฮัฟแมน

4. สามารถลดปริมาณช่องกว้างทางความถี่ที่ต้องใช้ในการส่งข้อมูล

1.6 ตารางการดำเนินงาน

ขั้นตอนการดำเนินงาน	พุทธศักราช 2550					พุทธศักราช 2551
	สิงหาคม	กันยายน	ตุลาคม	พฤศจิกายน	ธันวาคม	มกราคม
1. ศึกษาปัญหาที่เกี่ยวข้อง	⟷					
2. ศึกษาทฤษฎีต่าง ๆ ที่เกี่ยวข้อง	⟷	⟷				
3. ศึกษาโปรแกรม MATLAB		⟷	⟷			
4. เขียนโปรแกรมจำลองทฤษฎีที่ศึกษา			⟷	⟷		
5. ทดลองการทำงานและวิเคราะห์ผล					⟷	⟷
6. สรุปและทำรายงาน					⟷	⟷

1.7 งบประมาณที่ต้องใช้

1. ค่าเอกสาร	400 บาท
2. เข้ารูปเล่มของโครงงาน	500 บาท
3. ค่าใช้จ่ายอื่น ๆ	100 บาท
รวมเป็นเงินทั้งสิ้น	1,000 บาท

ต่อไปจะได้ศึกษาถึงทฤษฎีต่าง ๆ ที่เกี่ยวข้องกับการทำโครงงานเล่มนี้ ซึ่งจะศึกษาเกี่ยวกับทฤษฎีข่าวสาร การเข้ารหัสแบบฮัฟแมน (Huffman coding) ซึ่งถ้าไม่ได้ศึกษาหรือไม่มีความเข้าใจในทฤษฎี

เหล่านี้อย่างถ่องแท้ ก็ไม่สามารถที่จะทำการทดลองหรือทำตามวัตถุประสงค์ที่ต้องการได้ ซึ่งจะได้กล่าวต่อไปในบทที่ 2

บทที่ 2
หลักการและทฤษฎี

เมื่อกล่าวถึงการสื่อสารนั้น ก็จะหมายถึงการรับส่งข่าวสาร ซึ่งจะทราบว่าผู้ส่งจะส่งข้อมูลได้มากหรือน้อย ผู้รับจะรับข้อมูลได้มากน้อยเท่าไหร่นั้น จะต้องมีความรู้ความเข้าใจในทฤษฎีต่าง ๆ ในระบบสื่อสารก่อน ซึ่งในที่นี้ต้องการศึกษาถึงการบีบอัดข้อมูลถึงขีดความสามารถในการบีบอัด ดังนั้นก็จะต้องมีการศึกษาถึงทฤษฎีที่เกี่ยวข้องด้วยเช่นกัน ซึ่งจะกล่าวดังต่อไปนี้

2.1 ทฤษฎีข่าวสาร (Information Theory)[1, 2]

การส่งข่าวสารผ่านช่องสื่อสารเพื่อให้ถึงผู้รับที่ปลายทาง ในระบบสื่อสารนั้น การทราบขนาดหรือปริมาณของข่าวสารที่จะส่งเป็นประโยชน์มากเพราะจะช่วยให้สามารถเลือกช่องสัญญาณที่มีคุณสมบัติที่เหมาะสม เพื่อให้การรับและส่งข่าวสารมีประสิทธิภาพและถูกต้องที่สุด

ก่อนอื่นต้องทราบทฤษฎีพื้นฐานที่ใช้ในการอธิบายและวิเคราะห์ระบบสื่อสาร ที่กล่าวถึงการกำเนิดการส่งและรับข่าวสาร โดยทฤษฎีนี้ได้รับการพัฒนาจากกลุ่มนักวิจัย 2 กลุ่ม คือ

1. กลุ่มที่นำโดย โนเบิร์ต วีนเนอร์ (Nobert Wiener) ได้ทำการศึกษาการจัดการกับสัญญาณในสภาพที่มีสัญญาณรบกวนปนอยู่ ซึ่งก็คือการจัดการกับสัญญาณโดยใช้กระบวนการแรนดอม
2. กลุ่มที่นำโดย แชนนอน (C.E. Shannon) ทำการศึกษาในด้านของการเข้ารหัส

ซึ่งทฤษฎีข่าวสารบางครั้งถูกเรียกว่า ทฤษฎีสื่อสาร (Communication theory) โดยกล่าวถึงรูปแบบของระบบสื่อสาร การวัดปริมาณข่าวสาร การเข้ารหัสข่าวสาร ความจุของช่องสัญญาณ และการเข้ารหัสช่องสัญญาณ

รูปแบบของระบบสื่อสาร

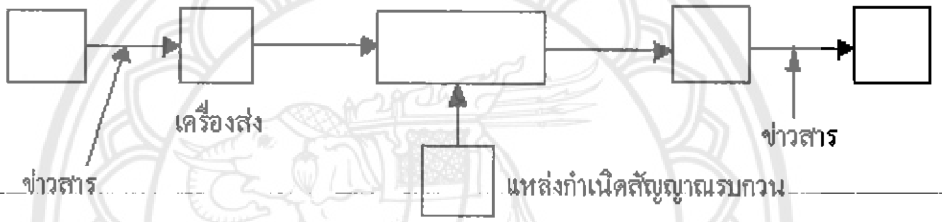
โดยทั่วไปแล้วการสื่อสาร คือ การส่งข่าวสารจากที่หนึ่งไปยังอีกที่หนึ่ง อย่างไรก็ตามการวิเคราะห์คุณสมบัติของระบบสื่อสารจะถูกต้องแม่นยำ แชนนอนได้เป็นผู้เสนอรูปแบบของระบบสื่อสารเป็นคนแรกในปี ค.ศ. 1948 โดยแบ่งส่วนประกอบต่าง ๆ ตามหน้าที่ และต่อมาในปี ค.ศ. 1961 ฟาโน (R.M. Fano) ได้ออกแบบระบบสื่อสารได้ละเอียดกว่าแชนนอน ซึ่งจะพูดถึงทั้งสองแบบดังนี้

1. รูปแบบของระบบสื่อสารที่เสนอโดยแชนนอน

แชนนอนได้แบ่งระบบการสื่อสารออกเป็นองค์ประกอบคือ แหล่งกำเนิดข่าวสาร (Information source) เครื่องส่งสัญญาณ (transmitter) ช่องสัญญาณ (channel) เครื่องรับสัญญาณ (receiver) และผู้รับข่าวสารปลายทาง (destination) แหล่งที่ให้ข่าวสารจะเป็นแหล่งที่กำเนิดข่าวสาร

(message) ข่าวสารจะเป็นตัวอักษรหรือเป็นคำ เป็นขบวนที่มีความหมาย ช่องสัญญาณเป็นตัวกลางที่นำสัญญาณจากเครื่องส่งไปยังเครื่องรับของสัญญาณ และในระหว่างที่มีการนำสัญญาณก็จะมีสัญญาณรบกวนเกิดขึ้น

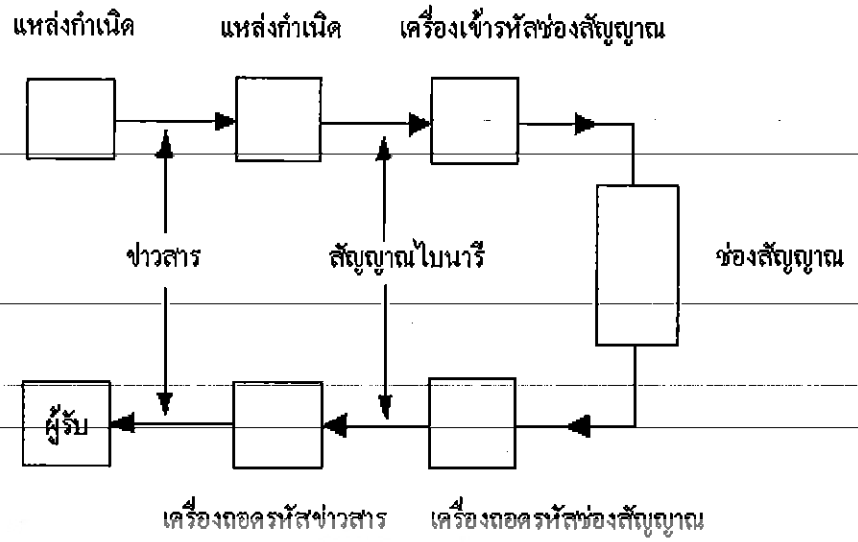
เครื่องส่งมีหน้าที่ในการแปลงข่าวสารให้อยู่ในสัญญาณที่เหมาะสมเพื่อ ที่จะส่งเข้าไปในช่องสัญญาณ เรียกว่า การเข้ารหัส (encoding or coding) สัญญาณรบกวนจะเข้ามาปนกับข่าวสารในช่องสัญญาณและก็ส่งไปยังเครื่องรับซึ่งก็มีสัญญาณรบกวนปนอยู่ เครื่องรับจะรับสัญญาณเข้ามา แล้วจะแปลงสัญญาณนั้นให้เป็นข่าวสารที่ไม่มีสัญญาณรบกวน ซึ่งเครื่องรับที่ดีนั้นจะต้องแปลงสัญญาณให้เป็นข่าวสารที่มีความผิดพลาดน้อยที่สุด ในการแปลงสัญญาณให้เป็นข่าวสารนั้น เรียกว่า การถอดรหัส (decoding) เพราะฉะนั้น การเข้ารหัสจะต้องทำให้สัญญาณมีการต่อต้านสัญญาณรบกวนสูงที่สุด แล้วในขณะเดียวกันการถอดรหัส ก็ต้องให้เกิดความผิดพลาดน้อยที่สุดด้วย



รูปที่ 2.1 รูปแบบของระบบสื่อสารที่เสนอโดยแชนนอน [1]

2. รูปแบบของระบบสื่อสารที่เสนอโดยฟาโน

รูปแบบของฟาโนเป็นรูปแบบของการสื่อสารข้อมูล โดยได้แยกส่วนที่เป็นเครื่องส่งในรูปแบบของแชนนอนเป็นเครื่องเข้ารหัสข่าวสาร (source encoder) และเครื่องเข้ารหัสช่องสัญญาณ (channel encoder) ในขณะเดียวกันก็แยกเครื่องรับ เป็นเครื่องถอดรหัสช่องสัญญาณ (channel decoder) และเครื่องถอดรหัสข่าวสาร (source decoder) เครื่องเข้ารหัสข่าวสารต้องทำการให้ข่าวสารออกมาในรูปของสัญญาณไบนารี จากนั้น เครื่องเข้ารหัสช่องสัญญาณก็จะจัดการกับสัญญาณไบนารีให้อยู่ในรูปสัญญาณที่เหมาะสมเพื่อส่งผ่านช่องสัญญาณ



รูปที่ 2.2 รูปแบบของระบบสื่อสารข้อมูลที่เสนอโดยฟาโน [1]

การแยกออกเป็น 2 ส่วนนี้ ทำให้เห็นว่าการเลือกวิธีการเข้ารหัสข่าวสารที่เหมาะสมกับความน่าจะเป็นของการเกิดข่าวสาร สามารถใช้ความยาวเฉลี่ยของสัญญาณไบนารีที่สั้นที่สุดในการส่งข่าวสารได้ นั่นคือ การบีบอัดข่าวสาร การเข้ารหัสข่าวสาร ซึ่งจะเลือกการส่งสัญญาณให้เหมาะสมกับช่องสัญญาณ เพื่อให้เครื่องถอดรหัสช่องสัญญาณปลายทางเกิดความผิดพลาดน้อยที่สุด

การนิยามปริมาณข่าวสารและเอนโทรปี[2]

ลักษณะทั่วไปของข่าวสารและการนิยามปริมาณข่าวสาร

ข่าวสารที่มนุษย์ใช้กันในทุกวันนี้มีอยู่มากมายหลายรูปแบบ เช่น ข่าวสารที่ใช้ในชีวิตประจำวัน ทางการเมือง สังคม และเศรษฐกิจ ก็ล้วนเป็นข่าวสารทั้งสิ้น อีกทั้งยังรวมไปถึงความรู้ที่ได้มาจากทฤษฎี และกฎเกณฑ์ทางสังคม ซึ่งก็เป็นข่าวสารอีกเช่นกัน แต่สิ่งที่เป็นจุดร่วมกัน คือ เมื่อมีการส่งข่าวสาร ผู้รับข่าวสารจะได้รับความรู้จากข่าวสารมากหรือน้อย จะมาพิจารณาตัวอย่าง เช่น "พรุ่งนี้ดวงอาทิตย์จะขึ้นทางทิศตะวันออก" และ "ในฤดูหนาวนี้หิมะจะตกที่อินเดีย" เมื่อเปรียบเทียบสองประโยคนี้มีความแตกต่างกันของปริมาณข่าวสาร นั่นคือ ในประโยคแรกทุกคนทราบอยู่แล้วว่าดวงอาทิตย์ต้องขึ้นทางทิศตะวันออกและเกิดขึ้นทุกวัน โอกาสหรือความน่าจะเป็นในการเกิดสูง จึงทำให้ไม่มีประโยชน์หรือไม่น่าสนใจ จึงถือว่ามีปริมาณข่าวสารน้อย เมื่อพิจารณาประโยคที่สอง จะเห็นว่าโอกาสที่จะเกิดขึ้นน้อยมาก และได้รับความสนใจมาก จึงทำให้ประโยคนี้มีปริมาณของข่าวสารมาก จะเห็นได้ว่าปริมาณของข่าวสารนั้น มีความสัมพันธ์กับโอกาส คือ ถ้าปริมาณของข่าวสารน้อยแสดงว่าโอกาสในการเกิดสูง แต่ถ้าปริมาณของข่าวสารมาก แสดงว่าโอกาสที่จะเกิดน้อย เมื่อพิจารณา

แหล่งกำเนิดข้อมูลที่มีการส่งข้อมูลออกทีละชุด โดยชุดที่ส่งออกมีรูปแบบได้จำกัด M แบบ ซึ่งอยู่ภายในเซต $\{x_1, x_2, ..., x_M\}$ โดยในการส่งข้อมูลแต่ละชุดมีความน่าจะเป็นเท่ากับ

$$P(X = x_k) = p_k \quad k = 1,2,...,M \tag{2.1}$$

ผลรวมความน่าจะเป็นในการส่งข้อมูลของแต่ละชุดต้องมีค่าเท่ากับ 1 เสมอ

$$\sum_{k=1}^{M} p_k = 1 \tag{2.2}$$

ด้วยเหตุนี้การนิยามปริมาณข่าวสารโดยทฤษฎีความน่าจะเป็นจึงมีความเหมาะสม แชนนอนได้เสนอให้ใช้ฟังก์ชันในรูปของลอกกาลิทึม ต่อไปนี้

$$I(x_k) \equiv \log_b\left(\frac{1}{p(x_k)}\right) = -\log_b p(x_k) \tag{2.3}$$

เมื่อ $p(x_k)$ เป็นความน่าจะเป็นที่จะเกิดข่าวสาร x_k (self-information)
$I(x_k)$ เป็นปริมาณข่าวสาร x_k
b เป็นฐานของลอกกาลิทึม

หน่วยของปริมาณข่าวสาร $I(x_k)$ นั้นจะขึ้นอยู่กับฐานของลอกกาลิทึมที่ใช้ คือถ้า b เป็นค่าเท่ากับ 2 หน่วยของปริมาณข่าวสารเป็น บิต (bit) แต่ถ้า b เป็นค่าเท่ากับ e หน่วยของปริมาณข่าวสาร คือ แน็ต (nat)

จากนิยามตามสมการที่ (2.3) พบว่าปริมาณข่าวสารมีคุณสมบัติต่อไปนี้

คุณสมบัติข้อที่ 1 $I(x_k) = 0$ สำหรับ $p_k = 1$ กล่าวว่าถ้าเหตุการณ์หนึ่งจะเกิดขึ้นอย่างแน่นอนแล้ว ปริมาณข่าวสารจะได้จากการเกิดของเหตุการณ์ดังกล่าวจะมีค่าเป็นศูนย์ คือ ไม่ได้ข่าวสารอะไรใหม่เลยจากการเกิดเหตุการณ์ดังกล่าว

คุณสมบัติข้อที่ 2 $I(x_k) \geq 0$ สำหรับ $0 \leq p_k \leq 1$ กล่าวว่าปริมาณข่าวสารที่ได้จากการเกิดของเหตุการณ์ใดเหตุการณ์หนึ่งจะมีค่ามากกว่าหรือเท่ากับศูนย์เสมอ จะไม่มีปริมาณข่าวสารใดที่เป็นลบ นั่นคือ การเกิดของเหตุการณ์ใหม่จะไม่ทำให้เกิดการสูญเสียของข่าวสารที่มีอยู่เดิม

คุณสมบัติข้อที่ 3 $I(x_k) > I(x_i)$ สำหรับ $p_k < p_i$ กล่าวว่าเหตุการณ์ใดมีโอกาสในการเกิดน้อยกว่าก็จะให้ปริมาณข่าวสารที่มากกว่าเสมอ

คุณสมบัติข้อที่ 4 $I(x_k x_i) = I(x_k) + I(x_i)$ ถ้า x_k และ x_i เป็นอิสระจากกันในเชิงสถิติ เป็นการแสดงให้เห็นว่าปริมาณข่าวสารของเหตุการณ์ 2 เรื่องที่ได้รับเข้ามาต่อเนื่องกันหรือเกิดพร้อมกันสามารถนำมาบวกรวมกันได้โดยตรง ทั้งนี้เหตุการณ์ทั้งสองต้องมีคุณสมบัติความเป็นอิสระจากกันเชิงสถิติ (statistically independent)

การนิยามเอนโทรปี[2]

โดยปกติแล้วจะทราบถึงปริมาณข่าวสารที่ได้จากแหล่งกำเนิดข้อมูลเฉลี่ยโดยรวมมากกว่าปริมาณข่าวสารของเหตุการณ์ที่แยกจากกัน ซึ่งปริมาณข่าวสารเฉลี่ยนี้ เรียกว่า **เอนโทรปี (entropy)** ซึ่งจะพิจารณาแหล่งกำเนิดข่าวสารที่มีจำนวนจำกัดเป็น M แบบ โดยความน่าจะเป็นแต่ละแบบเป็น p_1 ถึง p_M จะทำให้เอนโทรปีของข่าวสารเขียนได้ดังนี้

$$H(X) = \sum_{k=1}^{M} p_k I(x_k) = -\sum_{k=1}^{M} p_k \log_b \frac{1}{p_k} \qquad (2.4)$$

หน่วยของเอนโทรปีเป็น **บิตต่อสัญลักษณ์ (bit/symbol)**

คุณสมบัติของเอนโทรปีที่น่าสนใจอีกก็คือ

$$0 \le H(X) \le \log_b M \qquad (2.5)$$

จะเห็นว่าความสัมพันธ์นี้แสดงถึงขอบเขตค่าของเอนโทรปีทั้งค่าต่ำสุดและค่าสูงสุด **สำหรับขอบ**เขตต่ำสุดของค่าเอนโทรปีคือ จะต้องมีค่ามากกว่าหรือเท่ากับศูนย์เสมอ **ส่วนขอบเขตบนจะบอกว่าเอน**โทรปีจะมีค่าสูงสุดไม่เกิน $\log_2 M$ เกิดขึ้นได้ก็ต่อเมื่อสัญลักษณ์ทั้ง M **รูปแบบมีความน่าจะเป็นใน**การเกิดเท่ากับ $1/M$ เท่ากันหมด

เนื่องจากเป็นการประมาณของปริมาณข่าวสารที่ส่งออกจากแหล่งกำเนิดข่าว**สารแต่ละครั้งหรือแต่ละส**ัญลักษณ์ ถ้าพิจารณาแหล่งกำเนิดข่าวสารที่ให้กำเนิดสัญลักษณ์เป็น r สัญลักษณ์ต่อวินาที (symbols/sec) เพราะฉะนั้นจะได้ปริมาณข่าวสารเฉลี่ยต่อหนึ่งวินาทีหรือเป็นอัตราการกำเนิดข่าวสาร (information rate) เท่ากับ

$$R_s = rH(X) \qquad (2.6)$$

หน่วยของอัตราการกำเนิดข่าวสารนี้คือ **บิตต่อวินาที (bits/sec)**

ช่องสัญญาณ (Channel) [2, 4-6]

ช่องสัญญาณสื่อสาร (channel หรือ transmission channel) เป็นอุปกรณ์อย่างหนึ่งที่ทำการแปลงสัญญาณอินพุทให้เป็นสัญญาณเอาท์พุตหรือสื่อกลางในการนำส่งข่าวสารจากแหล่งกำเนิดไปที่จุดหมาย โดยปกติแล้วสัญญาณอินพุตและสัญญาณเอาท์พุตจะมีรูปร่างที่กำหนดไว้ก่อนแล้ว อย่างไรก็ตามการที่มีสัญญาณอินพุตเข้ามาค่าหนึ่งนั้น ก็จะทำให้สัญญาณเอาท์พุตออกมาได้หลายอย่าง ดังนั้นการแปลงสัญญาณของช่องสัญญาณนั้นก็จะเป็นลักษณะของกระบวนการทางสถิติที่สัญญาณเอาท์พุตจะเกิดขึ้นได้นั้นขึ้นอยู่กับสัญญาณอินพุตแบบมีเงื่อนไข นั่นคือ จะอยู่ในแบบความน่าจะเป็นแบบมีเงื่อนไข และจะถูกกำหนดในการเกิดของสัญญาณรบกวนที่มีอยู่ในช่องสัญญาณ ในกรณีสัญญาณอินพุตและสัญญาณเอาท์พุตเป็นสัญญาณแบบไม่ต่อเนื่อง เรียกว่า ช่องสัญญาณแบบไม่ต่อเนื่อง (discrete channel) และถ้าสัญญาณเป็นแบบต่อเนื่อง ก็จะเรียกว่า ช่องสัญญาณแบบต่อเนื่อง (continuous channel) และมีคุณสมบัติอีกอย่างหนึ่งของช่องสัญญาณ นั่นก็คือ คุณสมบัติแบบมีหน่วยความจำ (with memory) ซึ่งหมายถึง การเกิดของสัญญาณเอาท์พุตที่เวลานั้นจะขึ้นกับสัญญาณอินพุตเวลานั้นกับสัญญาณที่เกิดขึ้นก่อนหน้านั้นด้วย และแบบไม่มีหน่วยความจำ (memoryless) คือสัญญาณเอาท์พุตที่เกิดขึ้นที่เวลานั้นจะขึ้นอยู่กับสัญญาณอินพุตที่เวลานั้นเท่านั้น และยังมีคุณสมบัติอีกอย่างหนึ่ง นั่นก็คือ คุณสมบัติที่เป็นสเตชันนารี คือ ช่องสัญญาณที่มีความน่าจะเป็นแบบมีเงื่อนไขที่ไม่เปลี่ยนแปลงไปตามเวลา เพราะฉะนั้นเมื่อได้กล่าวถึงช่องสัญญาณที่ไม่มีหน่วยความจำนั้นก็จะรวมไปถึงช่องสัญญาณแบบสเตชันนารีและเป็นช่องสัญญาณแบบไม่ต่อเนื่องด้วย จะเรียกย่อว่า DMC (discrete memolyless channel)

ต่อไปจะพิจารณาแหล่งกำเนิดสัญญาณที่มีรูปแบบแตกต่างกัน M รูปแบบ $\{x_1, x_2, ..., x_M\}$ จะได้สัญญาณเอาท์พุตที่มีรูปแบบแตกต่างกัน N รูปแบบ $\{y_1, y_2, ..., y_N\}$ ส่วนของช่องสัญญาณโดยจะเขียนค่าความน่าจะเป็น $p(y_j \mid x_i)$ นั่นคือความน่าจะเป็นที่สัญญาณเอาท์พุตเป็น y_j เมื่อทราบว่า x_i เป็นอินพุตของช่องสัญญาณแล้ว หรือเท่ากับ $P(Y = y_j \mid X = x_i)$

โดยปกติแล้วการกำหนดความน่าจะเป็นสัญญาณ X แต่ละสัญลักษณ์ที่เข้าสู่ช่องสัญญาณ ทำให้ทราบ $p(x_i) = P(X = x_i)$ สำหรับ x_i ที่อยู่ในเซต $\{x_1, x_2, ..., x_M\}$ เมื่อทราบความน่าจะเป็นของตัวแปรแรนดอม X ต่อไปสัญญาณทางเอาท์พุตของช่องสัญญาณ เป็นตัวแปรแรนดอม Y โดยจะใช้หลักการของการกระจายความน่าจะเป็นแบบมาร์จินอล (marginal distribution probability) ดังต่อไปนี้

$$p(y_j) = P(Y = y_j)$$

$$= \sum_{i=1}^{M} P(Y = y_j \mid X = x_i) P(X = x_i) \tag{2.7}$$

นอกจากนี้สามารถที่จะหาการกระจายความน่าจะเป็นร่วม (joint probability distribution) ของตัวแปรแรนดอม X และ Y ต่อไปนี้

$$
\begin{aligned}
p(x_i \mid y_j) &= P(X = x_i, Y = y_j) \\
&= P(Y = y_j \mid X = x_i)P(X = x_i) \qquad (2.8) \\
&= p(y_j \mid x_i)p(x_i)
\end{aligned}
$$

ข่าวสารร่วม (Mutual information)[2, 4]

ที่ผ่านมาได้พูดถึงช่องสัญญาณ ซึ่งสัญญาณ X คือสัญญาณที่เข้าช่องสัญญาณ และสัญญาณ Y คือสัญญาณที่ออกช่องสัญญาณ ซึ่งสัญญาณ Y ที่ได้มาจากการเปลี่ยนแปลงของสัญญาณ X ที่มาจากการรบกวนจากช่องสัญญาณ ซึ่งในสัญญาณ X สามารถหาความไม่แน่นอนได้โดยใช้เอ็นโทรปี ดังนี้

$$H(X) = \sum_{i=1}^{M} p(x_i)\log_b\left(\frac{1}{p(x_i)}\right) \qquad (2.9)$$

โดยที่จะเปลี่ยนแปลงสมการของ $H(X)$ ใหม่เป็น

$$H(X) = \sum_{i=1}^{M} p(x_i)\log_b\left(\frac{1}{p(x_i)}\right)\sum_{j=1}^{N} p(y_j \mid x_i)$$

$$= \sum_{i=1}^{M}\sum_{j=1}^{N} p(y_j \mid x_i)p(x_i)\log_b\left(\frac{1}{p(x_i)}\right)$$

$$= \sum_{i=1}^{M}\sum_{j=1}^{N} p(x_i, y_j)\log_b\left(\frac{1}{p(x_i)}\right)$$

$$= \sum_{j=1}^{N}\sum_{i=1}^{M} p(x_i, y_j)\log_b\left(\frac{1}{p(x_i)}\right) \qquad (2.10)$$

ต่อไปสิ่งที่เกิดขึ้นกับความไม่แน่นอนของ X เมื่อได้รับสัญญาณ $Y = y_j$ โดยภาครับได้รับข่าวสารเพิ่ม และต้องไม่ลืมข่าวสารสัญญาณ Y ซึ่งก็คือข่าวสารของสัญญาณ X ที่ถูกรบกวนโดยช่องสัญญาณ การเปลี่ยนแปลงของความไม่แน่นอนของสัญญาณ X เมื่อเปรียบกับก่อนที่จะ

ได้รับสัญญาณ Y โดยที่จะเริ่มจากการหาค่าเอนโทรปีของตัวแปรแรนดอม X หลังจากที่ทราบตัวแปรแรนดอม $Y = y_j$ แล้ว ได้ดังนี้

$$H(X \mid Y = y_j) = \sum_{i=1}^{M} p(x_i \mid y_j) \log_b \left(\frac{1}{p(x_i \mid y_j)} \right) \tag{2.11}$$

สมการนี้จะมีค่าที่เปลี่ยนแปลงไปเนื่องจากค่า y_j เพราะฉะนั้น $H(H \mid Y = y_i)$ ซึ่งตัวแปรแรนดอมที่มีค่าเป็น $H(X \mid Y = y_1), H(X \mid Y = y_2), \ldots, H(X \mid Y = y_N)$ สามารถที่จะเป็นความน่าจะเป็นเท่ากับ $p(y_1), p(y_2), \ldots, p(y_N)$ ซึ่งสามารถหาค่าเฉลี่ยของตัวแปรแรนดอม Y ได้คือ

$$H(X \mid Y) = \sum_{j=1}^{N} H(X \mid Y = y_j) p(y_j)$$

$$= \sum_{j=1}^{N} \sum_{i=1}^{M} p(x_i \mid y_j) p(y_j) \log_b \left(\frac{1}{p(x_i \mid y_j)} \right)$$

$$= \sum_{j=1}^{N} \sum_{i=1}^{M} p(x_i, y_j) \log_b \left(\frac{1}{p(x_i \mid y_j)} \right) \tag{2.12}$$

$H(X \mid Y)$ คือ ความไม่แน่นอนของสัญญาณ X หลังจากที่ได้รับสัญญาณ Y ที่ทางออกของช่องสัญญาณ

$H(X)$ คือ ความไม่แน่นอนของสัญญาณ X ที่แหล่งกำเนิด

เมื่อพิจารณาแล้ว $H(X \mid Y)$ จะมีขนาดเล็กกว่า $H(X)$ เพราะว่าได้รับข่าวสารมากกว่าดังนั้นความไม่แน่นอนของ X หลังจากที่ได้รับข่าวสารเพิ่มจากช่องสัญญาณจะลดลง จึงเรียกว่า **ข่าวสารร่วม (mutual information หรือ transinformation)** จะมีค่าเท่ากับ

$$I(X;Y) = H(X) - H(X \mid Y) \tag{2.13}$$

จากสมการที่ (2.12) และ (2.13) จะได้ว่า

$$I(X;Y) = \sum_{j=1}^{N} \sum_{i=1}^{M} p(x_i, y_j) \log_b \left(\frac{1}{p(x_i)} \right) - \sum_{j=1}^{N} \sum_{i=1}^{M} p(x_i, y_j) \log_b \left(\frac{1}{p(x_i \mid y_j)} \right)$$

$$= \sum_{j=1}^{N}\sum_{i=1}^{M} p(x_i, y_j)\log_b\left(\frac{p(x_i \mid y_j)}{p(x_i)}\right) \tag{2.14}$$

คุณสมบัติของข่าวสารร่วม[2]

คุณสมบัติข้อที่ 1 $I(X;Y) = I(Y;X)$

คือ ข่าวสารร่วมต้องสมมาตร นั่นคือ $I(X;Y)$ ต้องเป็นองค์ประกอบความไม่แน่นอนของสัญญาณด้านเข้าหลังจากที่ได้รับข่าวสารทางด้านออกของช่องสัญญาณ และ $I(Y;X)$ ต้องเป็นองค์ประกอบความไม่แน่นอนของสัญญาณด้านออกเมื่อรู้ค่าของสัญญาณที่เข้าทางช่องสัญญาณ

คุณสมบัติข้อนี้อาศัยหลักเกณฑ์ความน่าจะเป็นแบบมีเงื่อนไขของเบย์

$$\frac{p(x_i, y_j)}{p(x_i)} = \frac{p(y_j, x_i)}{p(y_j)} \tag{2.15}$$

เมื่อนำไปแทนในสมการข่าวสารร่วม โดยสลับลำดับของผลบวกใหม่จะได้

$$I(X;Y) = \sum_{i=1}^{M}\sum_{j=1}^{N} p(x_i, y_j)\log_b\left(\frac{p(y_j \mid x_i)}{p(y_i)}\right)$$

$$= I(Y;X) \tag{2.16}$$

คุณสมบัติข้อที่ 2 $I(X;Y) \geq 0$ เสมอ

คือข่าวสารร่วม $I(X;Y)$ มีค่ามากกว่าหรือเท่ากับศูนย์เสมอ โดยอาศัยทฤษฎีของความน่าจะเป็น ดังนี้

$$p(x_i \mid y_j) = \frac{p(x_i, y_j)}{p(y_j)} \tag{2.17}$$

แล้วจะนำไปแทนลงในสมการของข่าวสารร่วม จะได้

$$I(X;Y) = \sum_{j=1}^{N}\sum_{i=1}^{M} p(x_i, y_j)\log_b\left(\frac{p(x_i, y_j)}{p(x_i)p(y_j)}\right) \tag{2.18}$$

จึงสรุปได้ว่า

$$I(X;Y) \geq 0 \text{ และ } I(X;Y) = 0 \tag{2.19}$$

ได้ก็ต่อเมื่อ $p(x_i, y_j) = p(x_i)p(y_j)$ สำหรับทุกค่าของ i และ j ก็คือ ไม่มีการเสียข่าวสารนั้น เนื่องจากการได้รับและพิจารณาสัญญาณด้านออกของช่องสัญญาณ และจะไม่ได้รับข่าวสารเพิ่มถ้าหากว่าสัญญาณทางด้านเข้าและทางด้านออกมีความเป็นอิสระในทางสถิติ

คุณสมบัติข้อที่ 3

$$I(X;Y) = H(Y) - H(Y \mid X) = H(X) - H(X \mid Y) \tag{2.20}$$

จะเห็นได้ว่าข่าวสารร่วมสามารถที่จะพิจารณาในด้านของเอนโทรปีของสัญญาณทางออก Y ดังนี้ $I(X;Y) = H(Y) - H(Y \mid X)$ โดยจะพิจารณา $I(X;Y) = H(Y) - H(X \mid Y)$ กับคุณสมบัติข้อที่ 1 ของข่าวสารร่วม

คุณสมบัติข้อที่ 4

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{2.21}$$

ข่าวสารร่วมของช่องสัญญาณมีความสัมพันธ์กับเอนโทรปีร่วมระหว่างสัญญาณเข้า X และสัญญาณออก Y โดยที่

$$H(X,Y) = \sum_{j=1}^{N}\sum_{i=1}^{M} p(x_i, y_j)\log_b\left(\frac{1}{p(x_i, y_j)}\right) \tag{2.22}$$

ความจุของช่องสัญญาณ[2, 4-6]

ช่องสัญญาณที่มีสัญญาณอินพุต X ที่มีรูปแบบจำกัด และสัญญาณเอาท์พุต Y มีรูปแบบจำกัดด้วยเช่นกัน โดยช่องสัญญาณมีความน่าจะเป็นของการเปลี่ยนของสัญญาณ เท่ากับ $p(y_j \mid x_i)$ จะให้ความจุของช่องสัญญาณ คือ C ซึ่งมีค่าเท่ากับปริมาณข่าวสารเฉลี่ยสูงสุดที่สามารถส่งผ่านช่องสัญญาณได้ ซึ่งก็คือเป็นการหาค่าสูงสุดของข่าวสารร่วม จะได้พิจารณาดังนี้

เมื่อ $I(X;Y)$ แสดงได้เป็น

$$I(X;Y) = \sum_{i=1}^{M}\sum_{j=1}^{N} p(x_i, y_j)\log_b\left(\frac{p(y_j \mid x_i)}{p(y_j)}\right) \tag{2.23}$$

สามารถแสดงค่าของ $p(x_i, y_j)$ ได้เป็น

$$p(x_i, y_j) = p(y_j \mid x_i)p(x_i) \tag{2.24}$$

และค่าของ $p(y_j)$ จะได้

$$p(y_j) = \sum_{i=1}^{M} p(y_j \mid x_i)p(x_i) \tag{2.25}$$

ดังนั้นถ้าแทนค่าของ $p(x_i, y_j)$ และ $p(y_j)$ ไปแล้ว จะได้ค่า $I(X;Y)$ ซึ่งจะขึ้นอยู่กับค่าของ $p(x_i)$ และ $p(y_j \mid x_i)$ เท่านั้น เพราะว่า เป็นค่าที่ไม่สามารถเปลี่ยนแปลงได้ของคุณสมบัติของช่องสัญญาณ ดังนั้นค่าของ $I(X;Y)$ จะขึ้นอยู่กับที่สัญญาณอินพุต จึงสามารถหาความจุของช่องสัญญาณได้ดังนี้

$$C = \max_{p(x_i)} I(X;Y) \tag{2.26}$$

มีหน่วยเป็น บิตต่อสัญลักษณ์ (bits/symbol)

การที่จะทำให้ $I(X;Y)$ มีค่ามากสุด จะต้องทำการเข้ารหัสแหล่งกำเนิดของ $p(x_i)$ ที่จะทำให้ $I(X;Y)$ มีค่ามากที่สุด

ความจุของช่องสัญญาณนอกจากจะสามารถส่งปริมาณข่าวสารต่อสัญลักษณ์แล้ว ยังมีความสามารถในการส่งปริมาณข่าวสารต่อหนึ่งวินาทีได้ด้วย ถ้าให้ s เป็นอัตราการส่งสัญลักษณ์สูงสุด จะได้ความจุของช่องสัญญาณได้ดังนี้

$$C_B = sC \tag{2.27}$$

หน่วยเป็น บิตต่อวินาที (bits/sec)

ตามสมการนี้จะเป็นอัตราข่าวสารสูงสุดที่จะส่งผ่านช่องสัญญาณไปได้

กรณีแหล่งกำเนิดข่าวสารแบบต่อเนื่องและช่องสัญญาณแบบต่อเนื่อง[2,4]

ที่ผ่านมาได้ทราบถึงแหล่งกำเนิดข่าวสารและช่องสัญญาณแบบไม่ต่อเนื่อง จริง ๆ แล้วจะมีความคล้ายคลึงกันกับแบบไม่ต่อเนื่อง

โดยก่อนหน้านั้นได้พิจารณาถึงแหล่งกำเนิดสัญญาณที่รูปแบบสัญลักษณ์จำกัดด้วยตัวแปรแรนดอมที่ไม่ต่อเนื่อง แต่ต่อไปนี้จะพิจารณาแหล่งกำเนิดสัญญาณที่ไม่จำกัดที่อยู่ในรูปตัวแปรแรนดอมแบบต่อเนื่อง (continuous random variable)

ให้ตัวแปรแรนดอมแบบต่อเนื่อง X มีฟังก์ชันการกระจายความน่าจะเป็น (pdf) เท่ากับ $f_X(x)$ จะใช้รูปแบบของเอนโทรปีตัวแปรแบบไม่ต่อเนื่อง จะได้ต่อไปนี้

$$h(X)=\int_{-\infty}^{\infty} f_X(x)\log_b\left(\frac{1}{f_X(x)}\right)dx \tag{2.28}$$

จะได้เอนโทรปีนี้ว่าเป็น **เอนโทรปีส่วนต่าง (differential entropy)** ที่จริงแล้วสมการนี้ไม่สอดคล้องกับตัวแปรแรนดอมแบบไม่ต่อเนื่องเลย ซึ่งจะพิจารณาตัวแปรแรนดอมที่ต่อเนื่องในรูปของตัวแปรแรนดอมที่ไม่ต่อเนื่องโดยใช้ลิมิต สมมติให้ $x_k = k\Delta x$ โดยที่ $k = 0,1,2,\ldots$ และ Δx เข้าใกล้ศูนย์ ตัวแปรแรนดอม X จะอยู่ในช่วง $[x_k, x_k+\Delta x]$ ด้วยความน่าจะเป็นเท่ากับ $f_X(x)\Delta x$ ดังนั้นตัวแปรแรนดอมในรูปแบบที่ไม่ต่อเนื่องจะได้ดังนี้

$$H(X)=\lim_{\Delta x\to 0}\sum_{k=-\infty}^{\infty} f_X(x_k)\Delta x\log_b\left(\frac{1}{f_k(x_k)\Delta x}\right)$$

$$=\lim_{\Delta x\to 0}\sum_{k=-\infty}^{\infty}\left[f_X(x_k)\log_b\left(\frac{1}{f_k(x_k)}\right)-f_X(x_k)\log_b(\Delta x)\right]\Delta x$$

$$=\int_{-\infty}^{\infty} f_X(x)\log_b\left(\frac{1}{f_k(x)}\right)dx-\lim_{\Delta x\to 0}\log_b(\Delta x)\int_{-\infty}^{\infty} f_X(x)dx$$

$$=h(X)-\lim_{\Delta x\to 0}\log_b(\Delta x) \tag{2.29}$$

จะสังเกตได้ว่า $h(X)$ จากสมการนี้คือค่าเอนโทรปีส่วนต่าง จะเห็นได้ว่า เอนโทรปีส่วนต่างมีค่าไม่เท่ากับเอนโทรปีของตัวแปรแรนดอมแบบไม่ต่อเนื่อง และจะมีพจน์เพิ่มขึ้นมาอีก ซึ่งจะมีค่าเป็นอนันต์ ดังนั้น เอนโทรปีแบบต่อเนื่อง $H(X)$ ของตัวแปรแรนดอมแบบต่อเนื่องก็มีค่าเป็นอนันต์ จะเรียกว่า **เอนโทรปีสัมบูรณ์ (absolute entropy)** ไม่สามารถที่จะใช้ $H(X)$ ในการเปรียบเทียบ

ระหว่าง 2 ตัวแปรที่ต่างกัน เนื่องจากมีค่าเป็นอนันต์ แต่ถ้าสนใจเฉพาะความแตกต่างของเอนโทรปี ก็จะเอาตัวแปรแรนดอมทั้ง 2 ตัวมาหักลบกัน ซึ่งค่าที่เป็นอนันต์จะหักล้างกันไปเหลือส่วนต่างของ $h(X)$ ดังนั้นการนิยามเอนโทรปีของตัวแปรแรนดอมแบบต่อเนื่องในส่วนต่างของเอนโทรปี $h(X)$ แทนเอนโทรปีสัมบูรณ์ $H(X)$ จึงจะสมเหตุสมผล

ข่าวสารร่วม สำหรับกรณีของตัวแปรแรนดอมแบบต่อเนื่อง ระหว่างตัวแปรแรนดอม X และ Y จะได้ดังนี้

$$I(X,Y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f_{X,Y}(x,y)\log_b\left(\frac{f_X(x\mid y)}{f_X(x)}\right) \tag{2.30}$$

โดยที่ $f_{X,Y}(x,y)$ เป็นฟังก์ชันความหนาแน่นความน่าจะเป็นร่วม (joint probability density function) ของตัวแปรแรนดอม X และ Y

ส่วน $f_X(x\mid y)$ คือฟังก์ชันความหนาแน่นความน่าจะเป็นแบบมีเงื่อนไข (conditional probability density function) ของตัวแปรแรนดอม X เมื่อทราบตัวแปรแรนดอม Y

สำหรับคุณสมบัติของข่าวสารร่วมแบบต่อเนื่องก็สามารถที่จะพิจารณาได้กับตัวแปรไม่ต่อเนื่องซึ่งก็คือ

คุณสมบัติข้อที่ 1 $I(X;Y)=I(Y;X)$

คุณสมบัติข้อที่ 2 $I(X,Y)\geq 0$ เสมอ

คุณสมบัติข้อที่ 3 $I(X;Y)=h(X)-h(X\mid Y)$

คุณสมบัติข้อที่ 4 $I(X;Y)=h(Y)-h(Y\mid X)$

โดยที่ $h(X)$ คือค่าเอนโทรปีส่วนต่างของ X

$h(Y)$ คือค่าเอนโทรปีส่วนต่างของ Y

สำหรับ $h(X\mid Y)$ คือค่าเอนโทรปีส่วนต่างแบบมีเงื่อนไข (conditional differential entropy) ของตัวแปรแรนดอม X เมื่อทราบตัวแปรแรนดอม Y จะได้ดังนี้

$$h(X\mid Y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f_{X,Y}(x,y)\log_b\left(\frac{1}{f_X(x\mid y)}\right)dxdy \tag{2.31}$$

และ $h(Y\mid X)$ คือค่าเอนโทรปีส่วนต่างแบบมีเงื่อนไข (conditional differential entropy) ของตัวแปรแรนดอม Y เมื่อทราบตัวแปรแรนดอม X จะได้ดังนี้

$$h(Y\mid X)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f_{X,Y}(x,y)\log_b\left(\frac{1}{f_X(y\mid x)}\right)dxdy \tag{2.32}$$

พารามิเตอร์อีกอย่างหนึ่งที่สำคัญคือ ความยาวเฉลี่ยของคำรหัสหรือจำนวนบิตโดยเฉลี่ยที่ต้องใช้ในการแทนสัญลักษณ์ของข่าวสาร หาได้จากความน่าจะเป็นของการเกิดสัญญาณ และจำนวนบิตที่ใช้แทนสัญลักษณ์แต่ละแบบ ให้แหล่งกำเนิดข่าวสารมีความเป็นไปได้ทั้งหมด M แบบภายในเซต $\{x_1, x_2, ..., x_M\}$ และความน่าจะเป็นของการเกิดสัญลักษณ์ เท่ากับ $\{p_1, p_2, ..., p_M\}$ แล้วจะได้จำนวนบิตเฉลี่ยหรือความยาวเฉลี่ยของคำรหัสของข่าวสารที่มีความยาวไม่จำกัด มีค่าเท่ากับ

$$\overline{R} = \sum_{k=1}^{M} p_k n_k \tag{2.33}$$

เมื่อ n_k คือความยาวของคำรหัสของแต่ละสัญลักษณ์

ถ้าในกรณีที่คำรหัสมีความยาวจำกัด จะได้จำนวนบิตเฉลี่ยหรือความยาวเฉลี่ย คือ

$$\overline{R} = \lceil \log_b M \rceil \tag{2.34}$$

สำหรับแหล่งกำเนิดข่าวสารที่มีค่าเอนโทรปีเท่ากับ $H(X)$ แล้วความยาวเฉลี่ยของคำรหัส คือ $\overline{R}$ จะต้องมีค่ามากกว่าหรือเท่ากับ $H(X)$ เสมอ และจะได้ประสิทธิภาพของการใช้จำนวนบิตในการแทนสัญลักษณ์ของคำรหัสได้เป็น

$$\eta = \frac{H(X)}{\overline{R}} \tag{2.35}$$

การเข้ารหัสแบบฮัฟแมน (Huffman coding)[2, 4]

เป็นการเข้ารหัสข่าวสารที่มีประสิทธิภาพและเหมาะสมที่สุดเนื่องจากให้ผลดีที่สุด คือ สามารถให้ชุดรหัสที่มีจำนวนบิตเฉลี่ยเข้าใกล้ค่าเอนโทรปีของแหล่งกำเนิดข่าวสารมากที่สุด และเป็นการเข้ารหัสที่มีความเกี่ยวข้องไปถึงความน่าจะเป็นของเหตุการณ์ของแต่ละสัญลักษณ์ และเป็นการเข้ารหัสที่มีความยาวของแหล่งกำเนิดข่าวสารที่เท่ากัน แต่จะได้ความยาวของคำรหัสที่ไม่เท่ากัน ซึ่งจะมีการวิธีการดังนี้

1. เรียงลำดับสัญลักษณ์ในแนวตั้ง โดยให้ความน่าจะเป็นของสัญลักษณ์ที่มีค่ามากที่สุดอยู่ด้านบนสุด
2. พิจารณาสัญลักษณ์ 2 ตัวล่างสุด แล้วจะกำหนดบิต 0 ให้กับสัญลักษณ์บน และบิต 1 ให้กับสัญลักษณ์ตัวล่าง แล้วรวมความน่าจะเป็นของทั้งสองตัว แล้วจะได้สัญลักษณ์ใหม่ที่เกิดจากผลรวมความน่าจะเป็นทั้งสอง

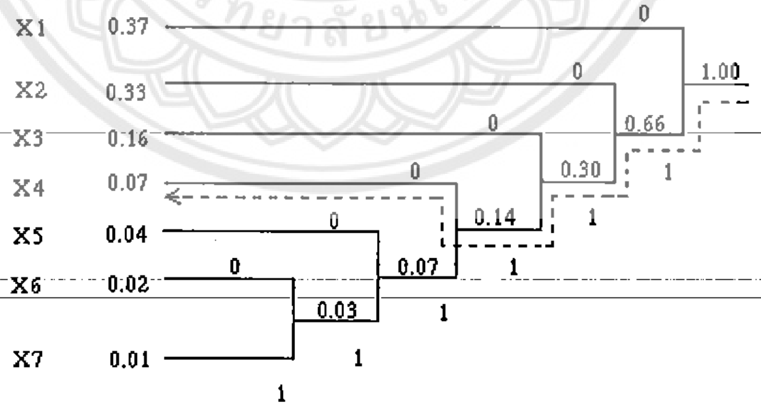
3. จะเห็นว่าจะมีจำนวนสัญลักษณ์ลดลงหนึ่งตัว แล้วก็ทำเช่นนี้ไปเรื่อยๆ จนเหลือสัญลักษณ์เพียง 2 ตัวสุดท้าย จึงจะสิ้นสุดการเข้ารหัส แล้วจะได้คำรหัสของแต่ละสัญลักษณ์ออกมาซึ่งจะเห็นว่า การเข้ารหัสจะเห็นเป็นโครงสร้างของต้นไม้

ตัวอย่างการเข้ารหัสฮัฟแมน[2]

สมมติว่ามีสัญลักษณ์อยู่ 7 ตัว คือ x_i, $i = 1,2,...,7$ และแต่ละตัวมีความน่าจะเป็นคือ $p_1 = 0.37, p_2 = 0.33, p_3 = 0.16, p_4 = 0.07, p_5 = 0.04, p_6 = 0.02$ และ $p_7 = 0.01$ โดยทำตามวิธีการ ซึ่งจะได้ตามลักษณะดังนี้

ตารางที่ 2.1 แสดงการเข้ารหัสฮัฟแมน

สัญลักษณ์	ความน่าจะเป็น	ปริมาณข่าวสาร	คำรหัส
x_1	0.37	1.4344	0
x_2	0.33	1.5995	10
x_3	0.16	2.6439	110
x_4	0.07	3.8365	1110
x_5	0.04	4.6439	11110
x_6	0.02	5.6439	111110
x_7	0.01	6.6439	111111



รูปที่ 2.3 แสดงแผนภูมิการเข้ารหัสฮัฟแมน

จากรูปจะเห็นว่าเมื่อทำการเข้ารหัสตามขั้นตอนแล้วจะได้โครงสร้างเป็นเหมือนต้นไม้ที่มีรากอยู่ทางขวามือสุดและบิตที่กำหนดให้สัญลักษณ์แต่ละตัวนั่นคือคำรหัสของแต่ละสัญลักษณ์ โดยจะเริ่มอ่านจากรากของต้นไม้ก่อนไปจนถึงปลายสุดทางด้านสัญลักษณ์แต่ละตัว จะสังเกตได้ว่าสัญลักษณ์ที่

อยู่ในลำดับล่างซึ่งมีความน่าจะเป็นในการเกิดน้อย จะมีคำรหัสที่มีขนาดยาวกว่าสัญลักษณ์ที่อยู่ในระดับบน ถ้าคำนวณหาค่าเอนโทรปีของแหล่งกำเนิดนี้จะได้

$$H(X) = -\sum_{k=1}^{7} p_k \log_2 p_k = 2.1152 \ bits$$

และความยาวเฉลี่ยต่อสัญลักษณ์จะคำนวณได้ดังนี้

$$\overline{R} = \sum_{k=1}^{7} n_k p_k$$

$$= 1(0.37) + 2(0.33) + 3(0.16) + 4(0.07) + 5(0.04) + 6(0.02) + 6(0.01)$$

$$= 2.1700 \ bits$$

เมื่อคำนวณหาประสิทธิภาพของการเข้ารหัสจะได้

$$\eta = \frac{H(X)}{\overline{R}} = \frac{2.1152}{2.1700} = 0.9747 = 97.47\%$$

จะเห็นว่าการเข้ารหัสในตัวอย่างมีประสิทธิภาพที่ใกล้เคียงกับทางทฤษฎีคือต่างกันไม่ถึง 2%

จากตัวอย่างนี้เป็นการเข้ารหัสแบบมีสัญลักษณ์เดียว ถ้าเกิดมีสัญลักษณ์ B สัญลักษณ์ ในกรณีนี้ก็จะได้ขอบเขตของการเข้ารหัส ดังนี้

$$BH(X) \le \overline{R}_B \le BH(X) + 1 \qquad (2.36)$$

โดยที่ เอนโทรปีของ B สัญลักษณ์ คือ $BH(X)$

และ $\overline{R}_B$ คือจำนวนเฉลี่ยของบิตต่อ B สัญลักษณ์

แล้วจะสามารถเขียนขอบเขตของการเข้ารหัสได้เป็น

$$H(X) \le \frac{\overline{R}_B}{B} < H(X) + \frac{1}{B} \qquad (2.37)$$

เมื่อ $\frac{\overline{R}_B}{B} \equiv \overline{R}$ คือจำนวนเฉลี่ยของบิตต่อสัญลักษณ์

ตัวอย่างของการเข้ารหัสแบบฮัฟแมนที่มีสัญลักษณ์ 2 ตัว

ตารางที่ 2.2 ตารางแสดงการเข้ารหัสฮัฟแมนที่สัญลักษณ์ 2 ตัว

สัญลักษณ์	ความน่าจะเป็น	ปริมาณข่าวสาร	คำรหัส
x_1x_1	0.1600	2.6439	10
x_1x_2	0.1400	2.8365	001
x_1x_3	0.1400	2.8365	010
x_2x_1	0.1225	3.0291	011
x_2x_2	0.1000	3.3219	111
x_2x_3	0.1000	3.3219	0000
x_3x_1	0.0875	3.5146	0001
x_3x_2	0.0875	3.5146	1100
x_3x_3	0.0625	4.0000	1101

การคำนวณหาเอนโทรปีดังนี้

$$2H(X) = -\sum_{k=1}^{9} p_k \log_2 p_k = 3.1177 \quad bits,$$

$$\Rightarrow H(X) = 1.5589 \quad bits.$$

และจำนวนบิตเฉลี่ยต่อสัญลักษณ์ได้ดังนี้

$$\overline{R}_B = \sum_{k=1}^{9} n_k p_k$$

$$= 2(0.1600) + 3(0.1400) + 3(0.1400) + 3(0.1225) + 3(0.1000) + 4(0.1000) + 4(0.0875) + 4(0.0875) + 4(0.0625)$$

$$= 3.1775 \quad bits/symbol \quad pair.$$

$$\Rightarrow \overline{R} = \frac{3.1775}{2} = 1.5888 \; bits/symbol.$$

และประสิทธิภาพของรหัสนี้คือ

$$\eta = \frac{1.5589}{1.5888} = 0.9812 = 98.12\%$$

การเข้ารหัสแบบ Lempel-Ziv[3,4]

จากข้างต้นได้ศึกษาถึงการเข้ารหัสแบบฮัฟแมนแล้วซึ่งจะขึ้นอยู่ความน่าจะเป็น แต่วิธีที่จะศึกษานี้ไม่ขึ้นอยู่กับความน่าจะเป็นหรือเป็นอิสระกับความน่าจะเป็นซึ่งจะวิเคราะห์พจนานุกรมโดยที่จะพิจารณาขบวนของคำ ซึ่งเมื่อแยกจากพจนานุกรมออกมาเป็นคำแล้วจะมีความยาวที่ไม่เท่ากัน แต่จะได้คำรหัสที่มีความยาวที่เท่ากัน ตามขั้นตอนต่อไปนี้

1. ระบุคำที่มีความยาวน้อยที่สุดก่อนของข้อมูล
2. แบ่งคำโดยที่แต่ละคำในข้อมูลหรือในพจนานุกรมจะต้องไม่เหมือนกัน
3. เพิ่มบิตเข้าไปเพื่อให้ได้คำใหม่ขึ้นมาโดยที่ความยาวของคำรหัสจะเท่ากันเท่ากัน

ตัวอย่างของการเข้ารหัสแบบ Lempel-Ziv

ถ้ามีรหัสอข้อมูลหนึ่งในพจนานุกรม คือ 101011011010101011 แล้วจะทำการแยกข้อมูลออกเป็นแต่ละคำ ซึ่งจะได้ 1,0,10,11,01,101,010,1011 สังเกตเห็นว่าแต่ละคำแตกต่างกันและมีความยาวไม่เท่ากัน อยู่ 8 คำ ดังนั้นจะเพิ่มบิตเข้าไป 3 บิต แล้วก็จะได้ 8 คำใหม่ออกมาที่มีความยาวเท่ากัน จะได้ (000,1), (000,0), (001,0), (001,1), (010,1), (011,1), (101,0), (110,1).
จะแสดงได้ดังตารางนี้

ตารางที่ 2.3 แสดงการเข้ารหัสแบบ Lempel-Ziv

พื้นที่ในพจนานุกรม	ขนาดคำในพจนานุกรม	คำรหัส
001	1	0001
010	0	0000
011	10	0010
100	11	0011
101	01	0101
110	101	0111
111	010	1010
-	1011	1101

ซึ่งจากตัวอย่างทั้งสองวิธีนี้จำนวนบิตเฉลี่ยต่อสัญลักษณ์ได้แล้ว ก็สามารถที่จะนำมาเปรียบเทียบกับค่าเอนโทรปีได้ เพื่อที่จะได้ทราบถึงประสิทธิภาพในการบีบอัดข้อมูล

แต่ในโครงงานเล่มนี้จะได้ศึกษาและจำลองการทำงานของการบีบอัดข้อมูลด้วยวิธีฮัฟแมนเพียงอย่างเดียว ดังจะได้แสดงในบทต่อไป

บทที่ 3

การออกแบบโครงงาน และวิธีการดำเนินงาน

ในบทที่ผ่านมาได้ศึกษาถึงหลักการและทฤษฎีที่เกี่ยวข้องกับทฤษฎีข่าวสารและการเข้ารหัสแบบฮัฟแมน (Huffman coding) ซึ่งในบทนี้จะกล่าวถึงโครงสร้างการทำงานและการออกแบบการเขียนโปรแกรมแบบจำลองการทำงานของการเข้ารหัสแบบฮัฟแมน โดยการใช้โปรแกรม MATLAB รวมไปถึงการการใช้ Graphic User Interfaces (GUI) เพื่อใช้ในการอธิบายและการแสดงค่าต่าง ๆ พร้อมกับการดำเนินงานของโครงงานนี้ โดยจะมีวิธีการดำเนินงานตามขั้นตอนดังต่อไปนี้

3.1 ศึกษาทฤษฎีที่เกี่ยวข้อง ได้แก่

3.1.1 ทฤษฎีข่าวสาร

3.1.2 ปริมาณข่าวสารและปริมาณข่าวสารเฉลี่ย (entropy)

3.1.3 การเข้ารหัสแบบฮัฟแมน

ซึ่งในขั้นตอนของการศึกษาทฤษฎีที่เกี่ยวข้องที่กล่าวมาข้างต้นได้อธิบายรายละเอียดไว้แล้วในบทที่ 2

3.2 ศึกษาการใช้โปรแกรม MATLAB[7, 8] ได้แก่

3.2.1 หลักการในการเขียนโปรแกรม MATLAB ซึ่งเป็นพื้นฐานและหลักการที่นำไปสู่การเขียนโปรแกรมที่ถูกต้อง

3.2.2 การใช้ฟังก์ชันที่เกี่ยวข้องเพื่อใช้ในการบีบอัดข้อมูลแบบฮัฟแมน ในขั้นตอนซึ่งเป็นกระบวนการทั้งหมดตั้งแต่ต้นจนจบได้มีการใช้ฟังก์ชันที่มีอยู่ในโปรแกรม MATLAB เพื่อให้ได้สิ่งที่ต้องการ

3.2.3 การพล็อตกราฟเพื่อใช้ในการเปรียบเทียบกันระหว่างข้อมูลแต่ละชนิด และใช้ในการเปรียบเทียบและวิเคราะห์ผล

3.2.4 การใช้ GUI (Graphic User Interface) ใช้ในขั้นตอนของการเขียนโปรแกรมการบีบอัดข้อมูลแบบฮัฟแมนเมื่อได้ผลหรือค่าต่าง ๆ ตามที่ต้องการแล้ว ค่าทั้งหมดจะแสดงออกมาในรูปแบบของ GUI ซึ่งได้มาจากการไปเรียกใช้ค่าที่ได้จากโปรแกรมการบีบอัดข้อมูลแบบฮัฟแมน

ปร.
ว494ก
2550

3.3 การออกแบบการเขียนโปรแกรมเพื่อใช้ในการบีบอัดข้อมูล

ในขั้นตอนนี้เป็นการออกแบบการเขียนโปรแกรมการบีบอัดข้อมูลโดยใช้วิธีการเข้ารหัสแบบฮัฟแมน ดังแสดงตามแผนผังข้างล่างนี้

รูปที่ 3.1 แสดงแผนผังการทำงานของโปรแกรม

จากรูปที่ 3.1 เป็นการอธิบายการทำงานของโปรแกรมตั้งแต่เริ่มแรกจนถึงขั้นตอนสุดท้าย โดยเริ่มจากเลือกจำนวนสัญลักษณ์ในการบีบอัด แล้วป้อนข้อมูลหรือไปค้นหาไฟล์ที่ต้องการบีบอัด จากนั้นโปรแกรมจะทำการบีบอัดข้อมูลโดยการใช้วิธีฮัฟแมน เมื่อโปรแกรมทำงานเสร็จแล้วก็จะแสดงเอาท์พุต พจนานุกรม อัตราในการบีบอัด และแสดงผลที่ได้จากการทำงานทั้งหมดของไฟล์หรือข้อมูลที่เลือกนั้นออกมา ก็ถือว่าจบขั้นตอนของการทำงานของโปรแกรม

3.4 การเขียนโปรแกรมเพื่อจำลองการทำงานตามโครงสร้าง

ในขั้นตอนของการเขียนโปรแกรมการทำงานตามโครงสร้างที่ออกแบบไว้ก่อนหน้าที่ ซึ่งจะแสดงไว้ในบทที่ 4

3.5 ทดสอบโปรแกรม

การทดสอบโปรแกรมโดยพิจารณา ขนาดของพจนานุกรม อัตราในการบีบอัดข้อมูล (coderate) ในแต่ละสัญลักษณ์ที่เลือกในการบีบอัด ซึ่งได้แสดงถึงความแตกต่างกัน และปริมาณข่าวสารเฉลี่ย (entropy) โดยอาศัยข้อมูลภาษาอังกฤษ 3 ชนิด ที่แตกต่างกันดังนี้

- Entertainment
- News
- Sportnews

จากข้อมูลเหล่านี้จะได้ศึกษาถึงความแตกต่างกันในการบีบอัดของแต่ละสัญลักษณ์ แล้วสิ่งที่ได้ก็คือเอาท์พุต อัตราในการบีบอัดข้อมูลของแต่ละสัญลักษณ์ ขนาดของพจนานุกรม ความยาวบิตที่ได้หลังจากการบีบอัด และเอาท์พุตไฟล์ ซึ่งจะได้อธิบายและวิเคราะห์การทดลองผลของโปรแกรมในบทที่ 4

3.6 วิเคราะห์ผลการทดสอบโปรแกรม

ในขั้นตอนการวิเคราะห์ผลการทดสอบโปรแกรมเพื่อจะนำไปสู่การสรุปผลการทดลองที่ถูกต้อง ซึ่งได้แสดงออกมาในรูปของตาราง และกราฟต่าง ๆ เพื่อแสดงการเปรียบเทียบการบีบอัดของแต่ละสัญลักษณ์ของข้อมูลที่แตกต่างกันและข้อมูลที่ส่งหลังถูกบีบอัด ซึ่งจะแสดงไว้ในบทที่ 4

ในบทต่อไปจะเป็นการออกแบบและเขียนโปรแกรมเพื่อจำลองการทำงานออกมาในรูปของ GUI โดยได้มีการทดลองใส่ค่าของจำนวนสัญลักษณ์ที่จะใช้ในการบีบอัดและเลือกไฟล์ที่ต้องการนำมา

ทดสอบการบีบอัดข้อมูล ซึ่งสิ่งที่ได้ก็จะเป็นค่าต่างๆที่เกี่ยวข้อง โดยจะมีการนำค่าต่างๆเหล่านั้นมาทำการวาดกราฟออกมาเพื่อใช้ในการวิเคราะห์ผลหรือเปรียบเทียบการบีบอัดในแต่ละกรณีต่อไป

บทที่ 4
ผลการดำเนินโครงงาน

ในบทนี้จะแสดงการออกแบบตัวโปรแกรมที่เกี่ยวกับการบีบอัดข้อมูลโดยวิธีฮัฟแมนพร้อมกับแสดงผลออกมาในรูปแบบของ Graphic User Interface (GUI) และจะทำการทดลองใส่จำนวนสัญลักษณ์และไฟล์ที่จะใช้ในการบีบอัด ซึ่งสามารถที่จะแสดงความแตกต่างของแต่ละสัญลักษณ์ในบีบอัด พร้อมกับเปรียบเทียบความแตกต่างเหล่านี้ออกมาในรูปของกราฟ เพื่อการวิเคราะห์ที่สะดวกและเข้าใจง่าย

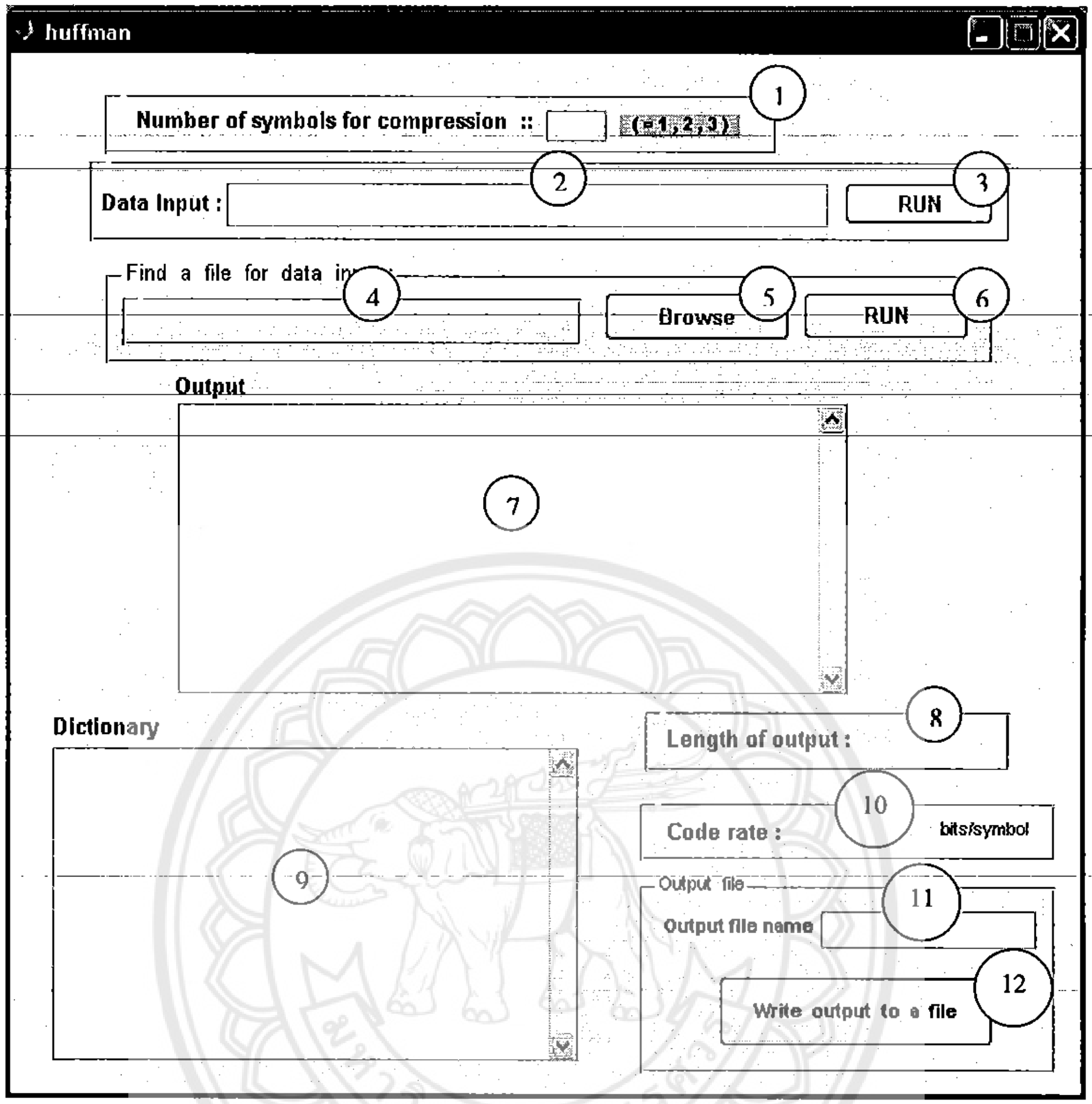
4.1 โปรแกรมแสดงการบีบอัดข้อมูลโดยวิธีฮัฟแมนพร้อมกับแสดงผลออกมาได้รูปของ Graphic User Interface (GUI)

4.1.1 โปรแกรมการบีบอัดข้อมูลโดยวิธีฮัฟแมน

ในขั้นตอนนี้จะเป็นขั้นตอนตามที่ได้ศึกษาทฤษฎีของฮัฟแมนตามที่ได้ศึกษามาแล้วในบทที่ 2 แล้วนำมาเขียนเป็นโปรแกรมซึ่งสิ่งที่ทำในโปรแกรม ได้แก่

- เป็นการเลือกว่าข้อมูลที่ต้องการบีบอัดเป็นไฟล์ที่ต้องไปค้นหาหรือว่าเป็นข้อมูลที่ต้องป้อนเข้าไป
- เลือกว่าจะบีบอัดข้อมูลครั้งละกี่สัญลักษณ์ ในที่นี้จะมีให้เลือกได้ 1-3 สัญลักษณ์
- หาคำรหัส (codeword) ของแต่ละสัญลักษณ์
- แสดงเอาท์พุตที่ได้จากการหาคำรหัสของแต่ละสัญลักษณ์ที่เลือกบีบอัด
- อัตราในการบีบอัดของข้อมูลที่เลือกบีบอัด
- ความยาวของจำนวนบิตที่ได้หลังจากการบีบอัด
- แสดงผลที่ได้ทั้งหมดของการบีบอัดของข้อมูลโดยวิธีฮัฟแมนออกมาในรูปของไฟล์ข้อมูล

จะได้ผลจากการทำงานของโปรแกรมพร้อมกับการแสดงค่าต่างๆ ออกมาให้เห็นในรูปของ GUI (Graphic User Interface) ดังแสดงตามรูปที่ 4.1



รูปที่ 4.1 หน้าต่างที่ใช้ป้อนข้อมูลที่ต้องการจะบีบอัดและแสดงผลจากการทำงานของการบีบอัดข้อมูล ซึ่งอยู่ในรูปแบบของ Graphic User Interface (GUI)

จากรูปที่ 4.1 เป็นการแสดงหน้าต่างที่ได้จากการเขียนโปรแกรมเพื่อสร้าง GUI ใน MATLAB เพื่อใช้ในการจำลองการบีบอัดข้อมูลแบบฮัฟแมน ซึ่งรายละเอียดของหน้าต่างที่ได้จะมีส่วนประกอบต่างๆทั้งในส่วนของการรับข้อมูลพื้นฐานที่ต้องการ และส่วนของการแสดงผลการทำงานของโปรแกรมหลังจากการบีบอัดข้อมูล ซึ่งจะได้แสดงรายละเอียดดังต่อไปนี้

(1) ช่องที่ใช้เลือกว่าต้องการบีบอัดข้อมูลกี่สัญลักษณ์ ซึ่งในที่นี้สามารถเลือกได้อยู่ 1-3 สัญลักษณ์ด้วยกัน

2 ช่องที่ใช้สำหรับป้อนข้อมูลลงไปในกรณีที่เลือกป้อนข้อมูล

3 ปุ่มที่สั่งให้โปรแกรมทำงานเพื่อบีบอัดข้อมูลที่ได้จากการป้อนข้อมูล

4 ช่องแสดงค่าที่ได้จากการไปค้นหาไฟล์เพื่อที่ต้องการนำมาบีบอัด

5 ปุ่มค้นหาไฟล์ที่ต้องการนำมาบีบอัด

6 ปุ่มที่สั่งให้โปรแกรมทำงานเพื่อบีบอัดข้อมูลที่ได้จากการค้นหาไฟล์

7 ช่องแสดงผลที่ได้จากการนำคำรหัส (codeword) ของแต่ละสัญลักษณ์มาเรียงกันตามข้อมูล

8 ช่องแสดงความยาวของคำรหัส (codeword) ทั้งหมด

9 ช่องแสดงผลที่ได้จากการรันโปรแกรม ซึ่งในช่องนี้จะแสดงสัญลักษณ์แต่ละตัวที่เลือกพร้อมกับความน่าจะเป็น และคำรหัส (codeword) ของข้อมูล

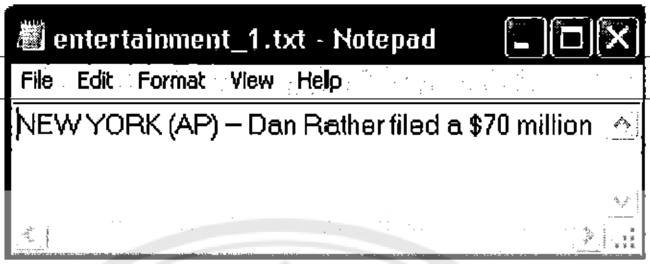
10 ช่องแสดงอัตราการบีบอัดของข้อมูลที่บีบอัด

11 ช่องที่ใช้สำหรับป้อนชื่อไฟล์ที่จะให้ผลจากการรันโปรแกรมออกมา ซึ่งชื่อไฟล์ต้องมีนามสกุลเป็น .txt

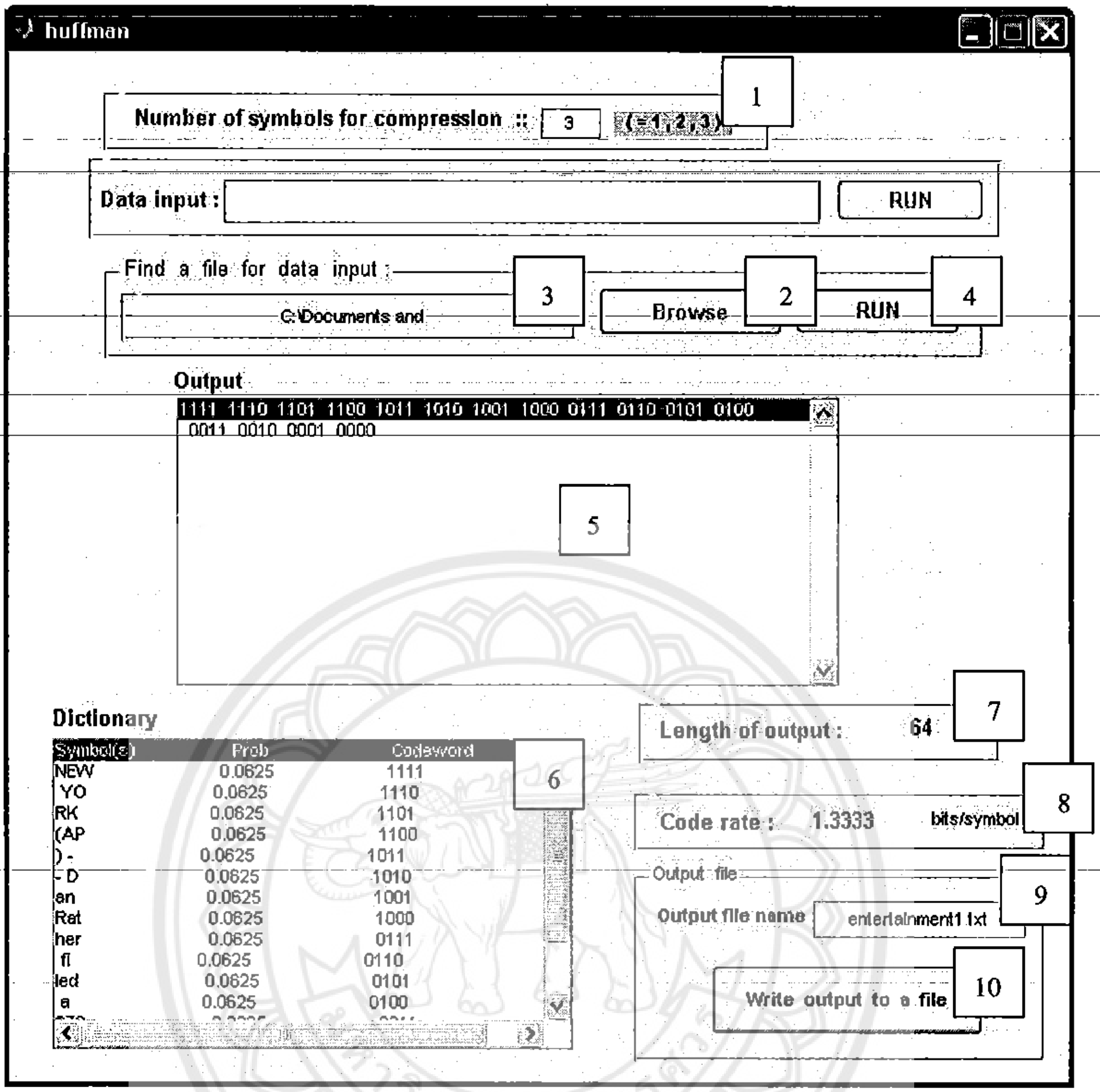
12 ปุ่มที่สั่งการให้โปรแกรมบันทึกข้อมูลจากการทำงานของโปรแกรมในรูปแบบของไฟล์ข้อมูล

4.2 ตัวอย่างในการทดลองโปรแกรมพร้อมกับตารางที่ได้จากการทดลอง

เพื่อให้เห็นภาพการทำงานของโปรแกรมได้ชัดเจนขึ้น ในขั้นต่อไปจะเป็นการทดลองโปรแกรมที่ได้อธิบายไว้ข้างต้นตามตัวอย่างดังต่อไปนี้ ซึ่งในกรณีนี้เป็นการป้อนข้อมูลที่ต้องการจะบีบอัดเป็นไฟล์ข้อมูลที่สร้างไว้ก่อนหน้านี้แล้ว โดยใช้ไฟล์ชื่อ entertainment_1.txt โดยมีข้อมูลในไฟล์ที่ต้องการบีบอัดแสดงในรูปที่ 4.2



รูปที่ 4.2 แสดงไฟล์ชื่อ Entertainment_1.txt ที่มีจำนวน 10 คำ



รูปที่ 4.3 ตัวอย่างการบีบอัดข้อมูลแบบฮัฟแมนโดยใช้การบีบอัดทีละ 3 สัญลักษณ์ โดยการค้นหาไฟล์ชื่อ entertainment_1.txt

จากรูปที่ 4.2 เมื่อป้อนชื่อไฟล์ในช่องที่ 9 แล้ว จากนั้นกดปุ่มที่ 10 ผลลัพธ์ทั้งหมดก็จะเขียนลงไปในไฟล์ที่ตั้งชื่อ ซึ่งจะเป็นไปตามรูปที่ 4.3

```
==================================================================
DATE : 09-Nov-2007
==================================================================
FILE NAME : C:\Documents and Settings\Mint_EE\Desktop\project\textFile\entertainment_1.txt
==================================================================
Number of symbols passed  :  3
==================================================================
DICTIONARY :
Symbol(s)                    Prob                Codeword
NEW                          0.0625              1111
 YO                          0.0625              1110
RK                           0.0625              1101
(AP                          0.0625              1100
) -                          0.0625              1011
- D                          0.0625              1010
an                           0.0625              1001
Rat                          0.0625              1000
her                          0.0625              0111
 fi                          0.0625              0110
led                          0.0625              0101
 a                           0.0625              0100
$70                          0.0625              0011
 mi                          0.0625              0010
lli                          0.0625              0001
on                           0.0625              0000
==================================================================
COMPRESSED  DATA :
1111  1110  1101  1100  1011  1010  1001  1000  0111  0110  0101  C100
  0011  0010  0001  0000
==================================================================
```

รูปที่ 4.4 การแสดงผลของโปรแกรมออกมาในรูปของไฟล์ (ที่ได้จากการกดปุ่มที่ 10 ตามรูปที่ 4.3)

จากการทดลองข้างต้นนี้ โดยใช้ข้อมูล Entertainment เป็นตัวอย่างในการทดลอง ในที่นี้เลือกจำนวนสัญลักษณ์ที่จะบีบอัดคือ 3 สัญลักษณ์ ในช่องที่ 1 แล้วไปค้นหาไฟล์ที่ต้องการบีบอัด โดยกดปุ่มที่ 2 เมื่อเลือกไฟล์ที่ต้องการได้แล้วโปรแกรมจะแสดง directory ของไฟล์ออกมาตามช่องที่ 3 ซึ่งในที่นี้ได้ใช้ข้อมูล Entertainment_1.txt ที่มีจำนวนคำอยู่ 10 คำ จากนั้นกดปุ่มที่ 4 เพื่อรันโปรแกรมจากไฟล์ที่เลือกและจำนวนสัญลักษณ์ที่บีบอัดข้อมูล เมื่อโปรแกรมรันเสร็จแล้วจะแสดงเอาท์พุตออกมาตามช่องที่ 5 ซึ่งเป็นการนำเอาคำรหัส (codeword) มาเรียงกันตามข้อมูลของไฟล์ที่เลือก โดยที่คำรหัสของแต่ละสัญลักษณ์พร้อมกับความน่าจะเป็นก็จะแสดงออกมาในช่องที่ 6 แล้วและในช่องที่ 7 จะแสดงความยาวของเอาท์พุตที่ได้จากช่องที่ 6 โดยนับเอาจำนวนบิตทั้งหมด มีค่าเท่ากับ 64 บิต และอัตราในการบีบอัดของข้อมูลก็คือ 1.3333 บิตต่อสัญลักษณ์แสดงในช่องที่ 8 สำหรับช่องที่ 9 เอาไว้ใส่ชื่อไฟล์ที่ต้องการเขียนค่าที่แสดงออกมาทางเป็นอีกไฟล์หนึ่งในตัวโปรแกรม MATLAB ในที่นี้ได้ใส่ไฟล์ที่ชื่อ entertainment1.txt จากนั้นกดปุ่มที่ 10 เพื่อให้ค่าที่ออกมาไปเขียนลงในไฟล์ที่ตั้งชื่อก่อนหน้านี้ ซึ่งก็จะได้ตามรูปที่ 4.4

จากตัวอย่างข้างต้นได้ทดลองใช้ข้อมูล entertainment ที่มีจำนวน 10 คำ และใช้ 3 สัญลักษณ์ในการบีบอัด เพียงไฟล์เดียว ซึ่งเมื่อทดลองใช้ข้อมูลประเภทเดียวกันที่มีจำนวนคำ 20 40 60 80 100 200 และ 400 คำ โดยใช้จำนวนสัญลักษณ์ในการบีบอัด 1 2 และ 3 สัญลักษณ์ในการบีบอัดจะได้อัตราในการบีบอัด จำนวนบิต และอัตราเฉลี่ยของข้อมูลที่แตกต่างกัน สำหรับข้อมูล News และ Sportnews ก็เช่นเดียวกัน ซึ่งผลจากการนำข้อมูลเหล่านี้ไปทำการบีบอัดได้แสดงให้เห็นในตารางที่ 4.1

ตารางที่ 4.1 การแสดงค่าต่าง ๆ ของข้อมูลที่ได้จากการทดลองการบีบอัดข้อมูลแบบฮัฟแมน

FileName	Number of words	Number of symbols	Number of bits	Number of symbols per compression						Entropy
				1		2		3		
				R	# bits	R	# bits	R	# bits	
Entertainment1	10	48	384	4.5319	218	2.3333	112	1.3333	64	1.3333
Entertainment2	20	101	808	3.9505	400	2.6078	264	1.6863	171	1.6759
Entertainment3	40	239	1912	4.4812	1072	3.2458	776	2.1	502	2.0733
Entertainment4	60	342	2736	4.693	1606	3.3684	1152	2.2222	760	2.2062
Entertainment5	83	509	4072	4.5634	2323	3.5603	1813	2.4717	1259	2.4450
Entertainment6	100	584	4672	4.5391	2651	3.5221	2057	2.4507	1432	2.4279
Entertainment7	208	1,294	10352	4.5418	5878	3.7462	4848	2.7724	3588	2.7517
Entertainment8	400	2,209	17672	4.4937	9927	3.7991	8393	2.9572	6533	2.9371

ตารางที่ 4.1(ต่อ) การแสดงค่าต่าง ๆ ของข้อมูลที่ได้จากการทดลองการบีบอัดข้อมูลแบบฮัฟแมน

Filename	Number of words	Number of symbols	Number of bits	Number of symbols per compression						Entropy
				1		2		3		
				R	#bits	R	#bits	R	#bits	
News1	10	60	480	4.0333	242	2.4333	146	1.4667	89	1.4407
News2	20	155	1240	4.6323	719	3.0641	475	1.9103	297	1.8869
News3	41	244	1952	4.2664	1042	3.1393	766	2.122	518	2.0940
News4	60	364	2912	4.3104	1569	3.3159	1207	2.2404	816	2.2347
News5	81	537	4296	4.3926	2359	3.4333	1844	2.4352	1308	2.4067
News6	102	596	4768	4.3833	2613	3.4583	2062	2.4817	1480	2.4555
News7	198	1,198	9584	4.4805	5368	3.709	4444	2.7264	3267	2.6611
News8	400	2,398	19184	4.4646	10707	3.7829	9072	2.9391	7048	2.9216
Sportnews1	10	49	392	3.9592	195	2.32	114	1.3725	68	1.3623
Sportnews2	21	107	856	4.6729	501	2.75	295	1.7407	187	1.7232
Sportnews3	40	221	1768	4.6154	1021	3.1532	697	2.0811	460	2.0605
Sportnews4	62	327	2616	4.5182	1478	3.2758	1072	2.2121	724	2.1937
Sportnews5	81	458	3664	4.4196	2025	3.3652	1542	2.342	1073	2.3200
Sportnews6	102	548	4384	4.4236	2425	3.4036	1866	2.3986	1315	2.3741
Sportnews7	202	1,177	9416	4.5755	5386	3.9606	4662	2.7181	3200	2.7024
Sportnews8	403	2,264	18112	4.5938	10401	3.8273	8666	2.978	6743	2.9236

ตาราง ที่ 4.2 แสดงประสิทธิภาพของข้อมูลที่ได้จากการทดลองการบีบอัดข้อมูลแบบฮัฟแมน

Filename	Number of words	Number of symbols	Efficiency for different number of symbols (%)		
			1	2	3
Entertainment1	10	48	29.42	57.14	100
Entertainment2	20	101	42.42	64.26	99.38
Entertainment3	40	239	42.27	63.88	98.73
Entertainment4	60	342	47.01	65.5	99.28
Entertainment5	83	509	53.58	68.67	98.92
Entertainment6	100	584	53.49	68.93	99.07
Entertainment7	208	1,294	60.59	73.45	99.25
Entertainment8	400	2,209	65.36	77.31	99.32
News1	10	60	35.72	59.21	98.23
News2	20	155	40.73	61.58	98.78
News3	41	244	49.08	66.70	98.68
News4	60	364	51.84	67.39	99.75
News5	81	537	54.79	70.10	98.83
News6	102	596	56.02	71	98.94
News7	198	1,198	59.39	71.75	97.60
News8	400	2,398	65.44	77.23	99.40
Sportnews1	10	49	34.41	58.72	99.26
Sportnews2	21	107	36.88	62.66	98.99
Sportnews3	40	221	44.64	65.35	99.01
Sportnews4	62	327	48.55	66.97	99.17
Sportnews5	81	458	52.49	68.94	99.06
Sportnews6	102	548	53.67	69.75	98.98
Sportnews7	202	1,177	49.35	68.23	99.42
Sportnews8	403	2,264	63.64	76.39	98.17

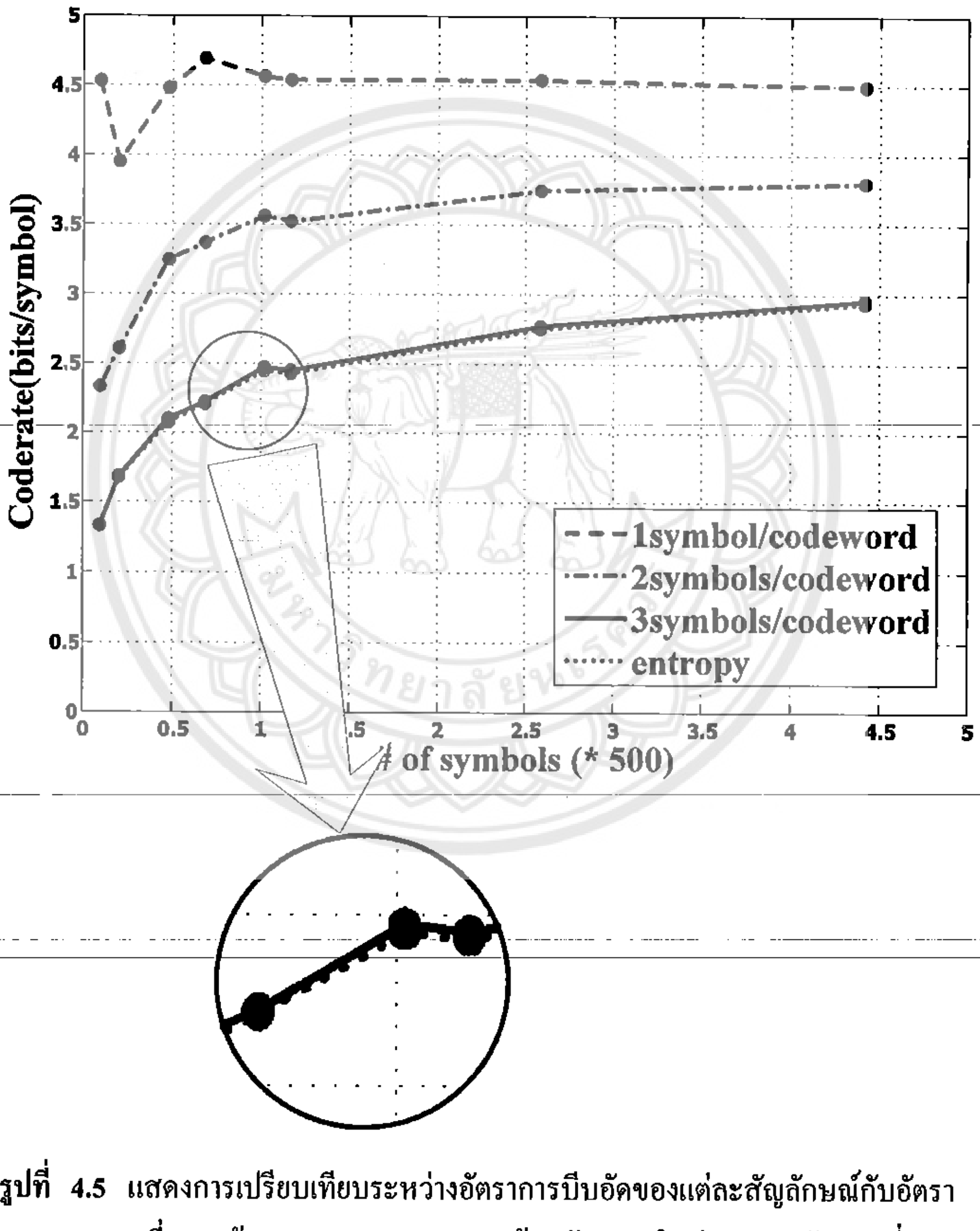
จากตารางที่ 4.1 ได้แสดงค่าต่าง ๆ ของข้อมูลภาษาอังกฤษทั้ง 3 ชนิดคือ Entertainment, News และ Sportnews โดยที่มีจำนวนคำที่แตกต่างกันได้แก่ 10 20 40 60 80 100 200 และ 400 คำ เมื่อได้ทำการบีบอัดข้อมูลเหล่านี้ด้วยจำนวนสัญลักษณ์ครั้งละ 1 2 และ 3 สัญลักษณ์แล้ว จะแสดงค่าออกมาก็คือ อัตราในการบีบอัดข้อมูล และจำนวนบิตที่ได้หลังจากการบีบอัด พร้อมกับแสดงอัตราเฉลี่ยของข้อมูลหรือเอนโทรปี (entropy) ดังที่แสดงในตาราง ยกตัวอย่างเช่น ข้อมูล Sportnews ที่มีจำนวนคำ 102 คำ จำนวนสัญลักษณ์ของคำ 548 สัญลักษณ์ และจำนวนบิตที่จำเป็นจะต้องใช้ในการส่งหากไม่มีการบีบอัดข้อมูลคือ 4,384 บิต (8 บิตต่อ 1 สัญลักษณ์) เมื่อทำการบีบอัดข้อมูลโดยวิธีฮัฟแมนโดยใช้จำนวนสัญลักษณ์ในการบีบอัด 1 สัญลักษณ์ จะได้อัตราในการบีบอัดข้อมูลเท่ากับ 4.4236 บิตต่อสัญลักษณ์ และจำนวนบิตที่จะต้องใช้ในการส่งมีค่าเท่ากับ 2,425 บิต เมื่อใช้ 2 สัญลักษณ์ในการบีบอัด จะได้อัตราในการบีบอัดข้อมูลเท่ากับ 3.4036 บิตต่อสัญลักษณ์ และจำนวนบิตที่จะต้องใช้ในการส่งมีค่าเท่ากับ 1,866 บิต และสุดท้ายเมื่อใช้ 3 สัญลักษณ์ในการบีบอัด จะได้อัตราในการบีบอัดข้อมูลเท่ากับ 2.3986 บิตต่อสัญลักษณ์ และจำนวนบิตที่จะต้องใช้ในการส่งมีค่าเท่ากับ 1,315 บิต พร้อมกับได้แสดงอัตราเฉลี่ยในการบีบอัดข้อมูลหรือเอนโทรปี (entropy) ที่มีค่าเท่ากับ 2.3741 บิตต่อสัญลักษณ์ ซึ่งจะสังเกตเห็นว่าเมื่อทำการบีบอัดข้อมูลโดยใช้จำนวนสัญลักษณ์ในการบีบอัดเพิ่มมากขึ้น อัตราในการบีบอัดข้อมูลก็จะมีค่าเข้าใกล้อัตราเฉลี่ยในการบีบอัดข้อมูลมากยิ่งขึ้น ดังตัวอย่างข้างต้นเมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 1 สัญลักษณ์จะมีอัตราในการบีบอัดเท่ากับ 4.4236 บิตต่อสัญลักษณ์ และเมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์มีอัตราในการบีบอัดเท่ากับ 3.4036 บิตต่อสัญลักษณ์ และเมื่อข้อมูลถูกบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ ก็จะมีอัตราในการบีบอัดเท่ากับ 2.3986 บิตต่อสัญลักษณ์ซึ่งมีค่าเข้าใกล้กับอัตราเฉลี่ยข้อมูลมากที่สุด นั่นก็คือ 2.3741 บิตต่อสัญลักษณ์

จากตารางที่ 4.2 ได้แสดงประสิทธิภาพในการบีบอัดข้อมูลของจำนวนสัญลักษณ์ที่ใช้ในการบีบอัด 1 2 และ 3 สัญลักษณ์ของข้อมูลทั้งสามชนิด คือ Entertainment News และ Sportnews ที่มีจำนวนคำที่แตกต่างกันเช่นเดียวกับตารางที่ 4.1 ซึ่งจากตัวอย่างข้างต้นเมื่อทำการบีบอัดข้อมูล Sportnews ที่มีจำนวนคำ 102 คำ เมื่อทำการบีบอัดด้วยจำนวนสัญลักษณ์ 1 สัญลักษณ์ จะมีประสิทธิภาพในการบีบอัด 53.67 เปอร์เซ็นต์ แต่เมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์ มีประสิทธิภาพในการบีบอัด 69.75 เปอร์เซ็นต์ และเมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ ก็จะมีประสิทธิภาพในการบีบอัด 98.98 เปอร์เซนต์

ข้อมูลอื่น ๆ ก็เช่นเดียวกันซึ่งได้ถูกแสดงไว้ในตารางแล้ว จากตารางเมื่อทราบค่าต่าง ๆ ที่ได้จากการทดลองของข้อมูลทั้ง 3 ชนิดแล้ว ก็นำมาวาดกราฟแสดงการเปรียบเทียบอัตราการบีบอัดข้อมูลแต่ละสัญลักษณ์กับอัตราเฉลี่ยข้อมูล และจำนวนสัญลักษณ์ที่ใช้ในการบีบอัด เพื่อให้เห็นความแตกต่างและประสิทธิภาพของการบีบอัดข้อมูลที่ชัดเจน ซึ่งจะแสดงในหัวข้อถัดไป

4.3 กราฟแสดงการเปรียบเทียบการบีบอัดของแต่ละสัญลักษณ์ในแต่ละไฟล์ ซึ่งค่าที่นำมาพล็อตกราฟนี้ได้จากตารางที่ 4.1

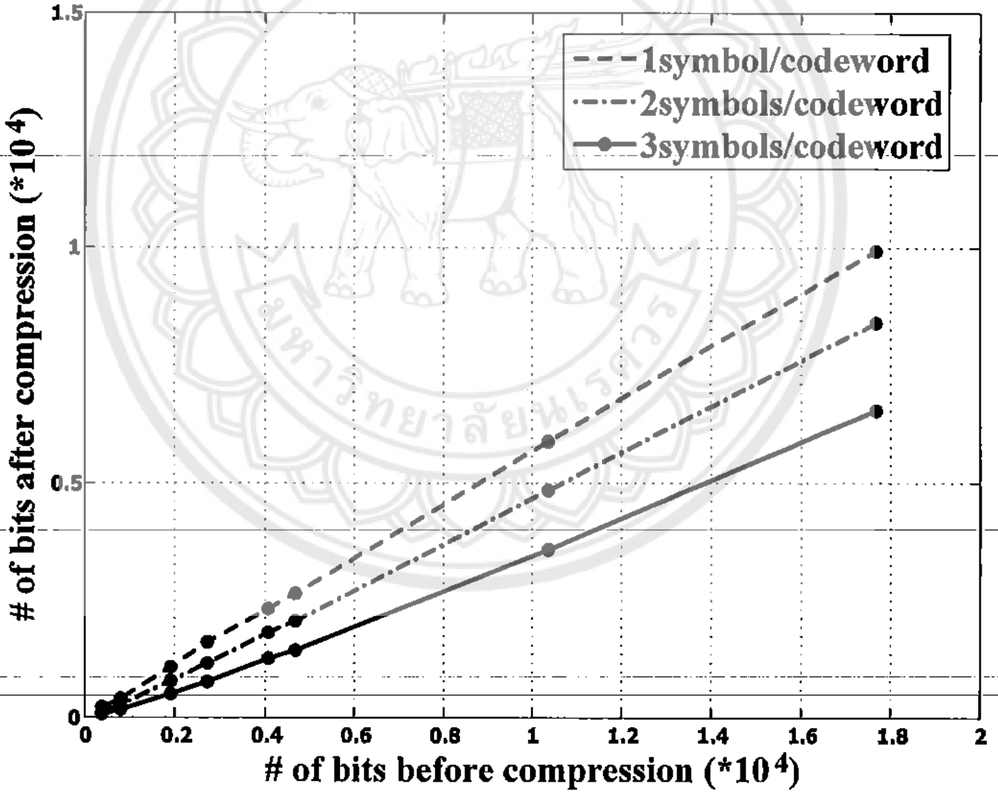
ต่อไปเป็นกราฟแสดงการเปรียบเทียบการบีบอัดของข้อมูล Entertainment ที่มีจำนวนคำที่แตกต่างกันได้แก่ 10 20 40 60 80 100 200 และ 400 คำตามลำดับ พร้อมกับจำนวนสัญลักษณ์ 1 2 และ 3 สัญลักษณ์ โดยเปรียบเทียบกันระหว่างอัตราในการบีบอัดซึ่งอยู่ในแกนตั้ง และจำนวนสัญลักษณ์ของแต่ละกลุ่มคำ ซึ่งอยู่ในแกนนอน พร้อมกับเปรียบเทียบกับอัตราเฉลี่ยของการบีบอัดหรือเอนโทรปี (entropy)



รูปที่ 4.5 แสดงการเปรียบเทียบระหว่างอัตราการบีบอัดของแต่ละสัญลักษณ์กับอัตราเฉลี่ยของข้อมูล Entertainment พร้อมกับขยายในส่วนของ อัตราเฉลี่ยของข้อมูลและอัตราการบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์

จากกราฟรูปที่ 4.5 จะเห็นว่าเมื่อบีบอัดข้อมูล entertainment ด้วยจำนวนสัญลักษณ์ที่แตกต่างกันพร้อมกับอัตราเฉลี่ยของข้อมูลแล้ว เส้นกราฟที่ได้จากการบีบอัดด้วยจำนวน 1 สัญลักษณ์มีระยะห่างจากเส้นกราฟของอัตราเฉลี่ยของข้อมูลมากที่สุด รองลงมาก็เป็นเส้นกราฟที่ได้จากการบีบอัดด้วยจำนวน 2 สัญลักษณ์ สำหรับเส้นกราฟที่ได้จากการบีบอัดด้วยจำนวน 3 สัญลักษณ์มีค่าใกล้เคียงกับเส้นกราฟของอัตราเฉลี่ยของข้อมูลมากที่สุด แสดงว่าการบีบอัดด้วยจำนวน 3 สัญลักษณ์มีประสิทธิภาพในการบีบอัดข้อมูลที่มีประสิทธิภาพดีกว่าการบีบอัดข้อมูลด้วยจำนวนสัญลักษณ์ 1 และ 2 สัญลักษณ์

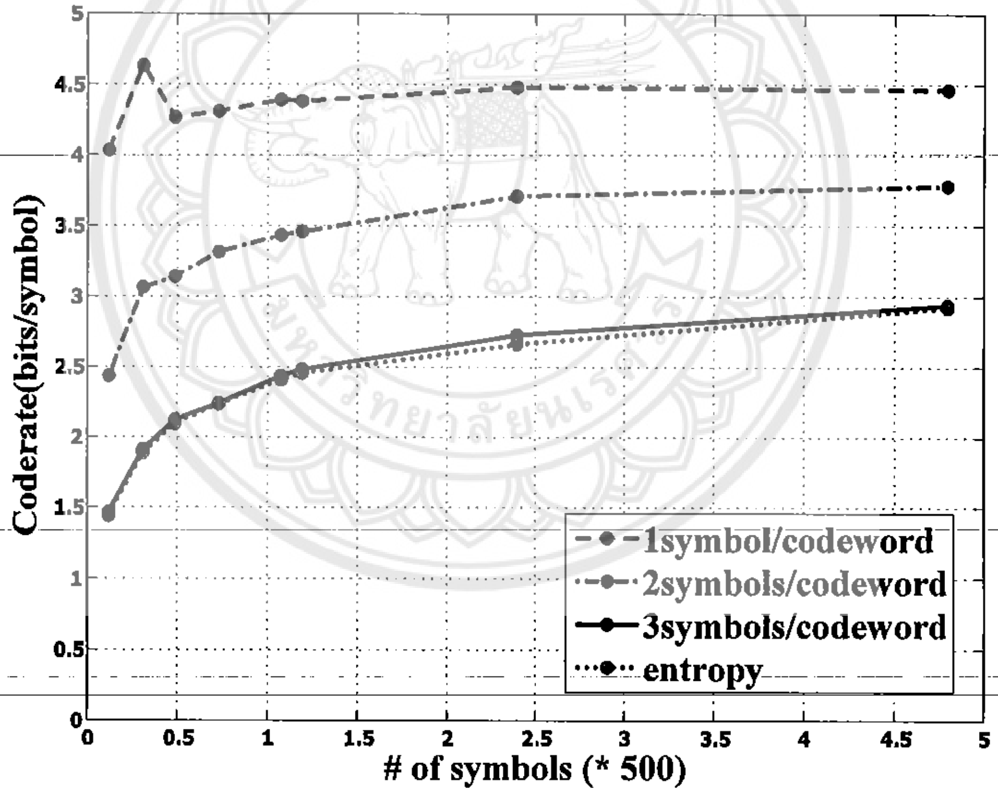
รูปกราฟที่ 4.6 แสดงการเปรียบเทียบการบีบอัดของข้อมูล Entertainment ที่มีจำนวนคำที่แตกต่างกัน ได้แก่ 10 20 40 60 80 100 200 และ 400 คำ ด้วยจำนวนสัญลักษณ์ 1 2 และ 3 สัญลักษณ์ตามลำดับ ซึ่งได้เปรียบเทียบกันระหว่างจำนวนบิตของข้อมูลก่อนที่จะทำการบีบอัดซึ่งอยู่ในแกนตั้ง กับจำนวนบิตของข้อมูลที่ได้หลังจากทำการบีบอัดซึ่งอยู่ในแกนนอน ได้แสดงไว้ดังกราฟข้างล่างนี้



รูปที่ 4.6 เป็นการเปรียบเทียบของข้อมูล Entertainment ที่ได้หลังจากการบีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน

จากรูปที่ 4.6 จะเห็นว่าจำนวนบิตของข้อมูลที่ได้หลังจากการบีบอัดมีจำนวนน้อยกว่าจำนวนบิตของข้อมูลก่อนที่จะถูกบีบอัด ซึ่งเมื่อนำจำนวนสัญลักษณ์ 1 2 และ 3 สัญลักษณ์มาเปรียบเทียบกันแล้ว ก็จะได้เส้นกราฟของจำนวน 1 สัญลักษณ์ที่มีความชันมากกว่าเส้นกราฟของจำนวน 2 สัญลักษณ์และจำนวน 3 สัญลักษณ์ จะสังเกตเห็นว่าการบีบอัดด้วยจำนวน 3 สัญลักษณ์มีความชันน้อยที่สุด นั่นก็หมายความว่าเส้นกราฟที่ถูกบีบอัดด้วยจำนวน 3 สัญลักษณ์ มีจำนวนบิตของข้อมูลทีได้หลังจากการบีบอัดน้อยที่สุด ดังนั้นจำนวนบิตที่ใช้ในการส่งข้อมูลก็มีจำนวนที่น้อยด้วยเช่นกัน

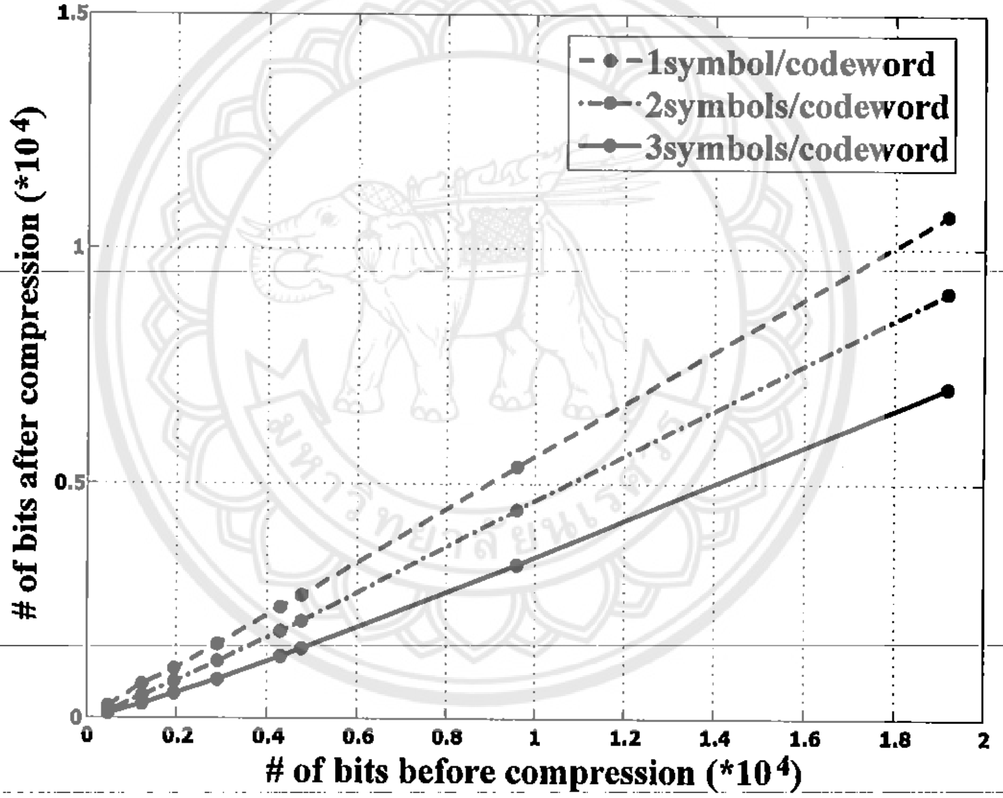
รูปกราฟที่ 4.7 แสดงการเปรียบเทียบการบีบอัดของข้อมูล News ที่มีจำนวนคำและจำนวนสัญลักษณ์ที่แตกต่างกัน เช่นเดียวกับรูปกราฟที่ 4.5 ที่มีแกนตั้ง เป็นอัตราการบีบอัดของข้อมูล และแกนนอน เป็นจำนวนสัญลักษณ์ของคำแต่ละกลุ่ม โดยที่เปรียบเทียบกันระหว่างเส้นกราฟของข้อมูลที่บีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกันกับเส้นกราฟของอัตราเฉลี่ยของข้อมูลหรือเอนโทรปี (entropy)



รูปที่ 4.7 แสดงการเปรียบเทียบระหว่างอัตราการบีบอัดกับอัตราเฉลี่ยของข้อมูล News

จากรูปที่ 4.7 เมื่อเปรียบเทียบการบีบอัดของข้อมูล News ด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน จะเห็นว่าเส้นกราฟของข้อมูลที่ถูกบีบอัดด้วยจำนวน 1 สัญลักษณ์มีระยะห่างจากเส้นกราฟของอัตราเฉลี่ยข่าวสารมากที่สุด รองลงมาก็เป็นเส้นกราฟของข้อมูลที่ถูกบีบอัดด้วยจำนวน 2 สัญลักษณ์ สำหรับเส้นกราฟของข้อมูลที่ถูกบีบอัดด้วยจำนวน 3 สัญลักษณ์จะมีค่าเข้าใกล้กับเส้นกราฟของอัตราเฉลี่ยข่าวสารมากที่สุดเช่นเดียวกับข้อมูล Entertainment

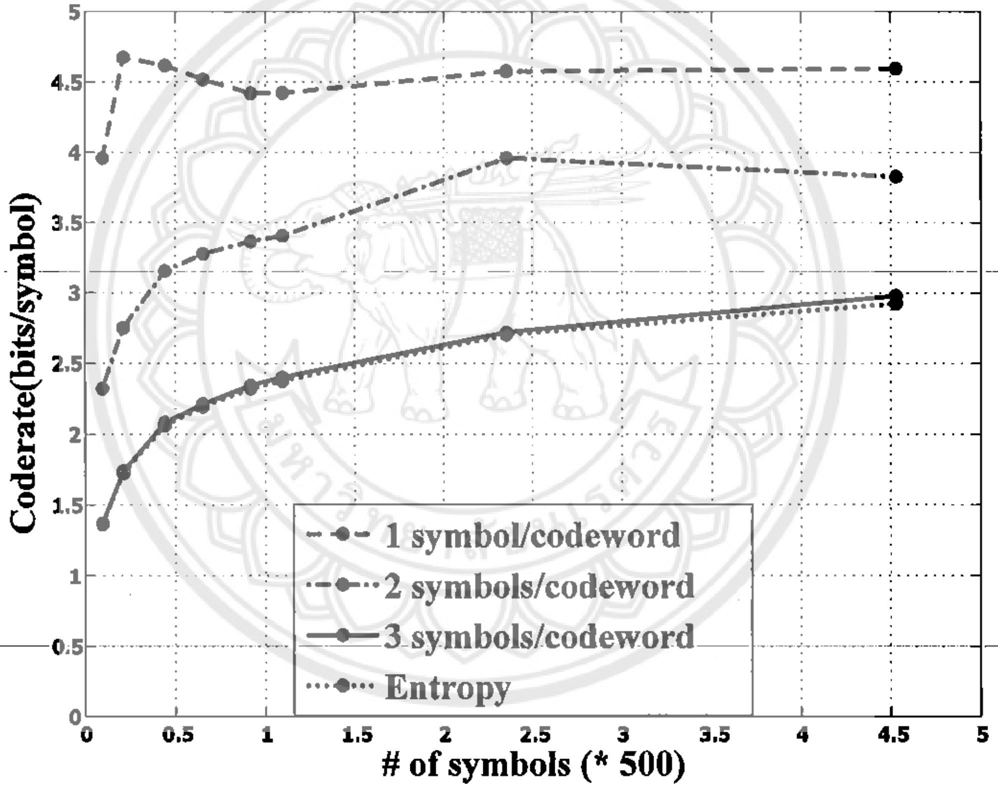
รูปกราฟที่ 4.8 แสดงการเปรียบเทียบการบีบอัดของข้อมูล News ด้วยจำนวนคำและจำนวนสัญลักษณ์ที่แตกต่างกันเช่นเดียวกับรูปที่ 4.6 ที่มีแกนตั้งเป็นจำนวนบิตข้อมูลก่อนที่จะทำการบีบอัด และแกนนอน เป็นจำนวนบิตข้อมูลที่ได้หลังจากการบีบอัด โดยเปรียบเทียบกันระหว่างเส้นกราฟของจำนวนสัญลักษณ์ที่แตกต่างกันทั้งสามเส้นดังแสดงตามรูปข้างล่างนี้



รูปที่ 4.8 เป็นการเปรียบเทียบของข้อมูล News ที่ได้หลังจากการบีบอัด
เมื่อบีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน

จากรูปที่ 4.8 เมื่อเปรียบเทียบจำนวนบิตหลังจากการบีบอัดของข้อมูล News ด้วยจำนวนสัญลักษณ์ทั้ง 3 สัญลักษณ์แล้ว จะเห็นได้ว่าเส้นกราฟที่แสดงการบีบอัดข้อมูลของการบีบอัดด้วยจำนวน 3 สัญลักษณ์มีความชันน้อยที่สุด ก็จะสามารถส่งจำนวนบิตข้อมูลได้มากที่สุดเช่นเดียวกับข้อมูล Entertainment

รูปกราฟที่ 4.9 แสดงการเปรียบเทียบการบีบอัดของข้อมูล Sportnews ที่มีจำนวนคำและจำนวนสัญลักษณ์ที่แตกต่างกัน เช่นเดียวกับรูปกราฟที่ 4.7 ที่มีแกนตั้ง เป็นอัตราการบีบอัดของข้อมูล และแกนนอน เป็นจำนวนสัญลักษณ์ของคำแต่ละกลุ่ม โดยที่เปรียบเทียบกันระหว่างเส้นกราฟของข้อมูลที่บีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกันกับเส้นกราฟของอัตราเฉลี่ยของข้อมูลหรือเอนโทรปี (entropy)

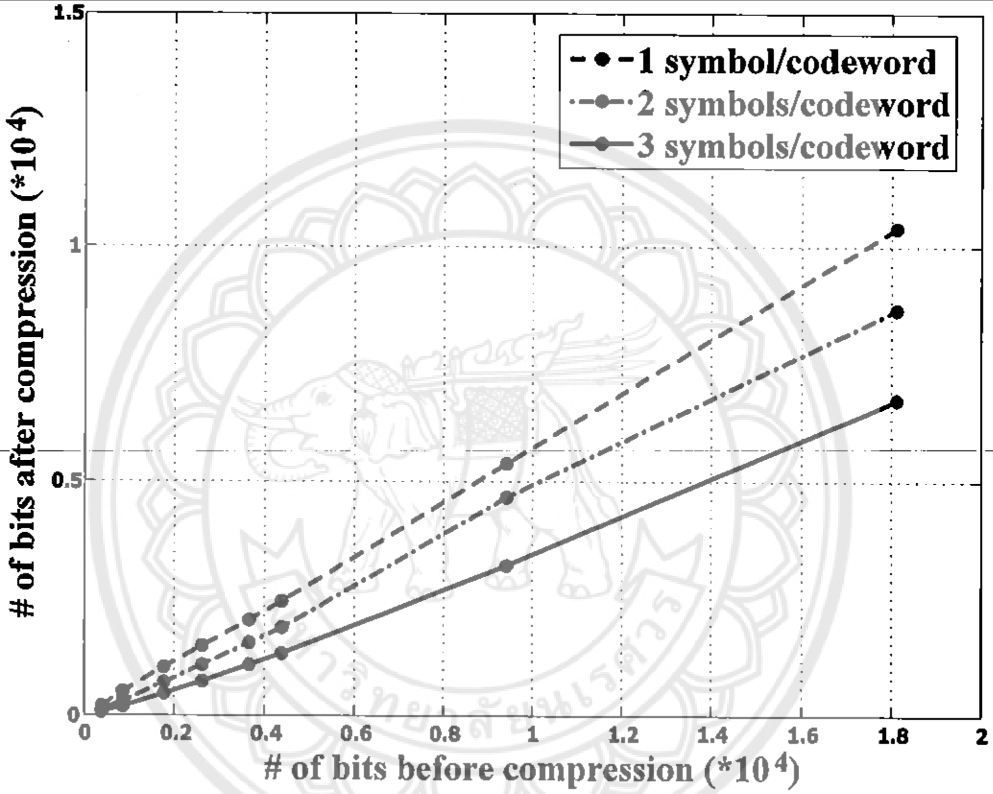


รูปที่ 4.9 แสดงการเปรียบเทียบระหว่างอัตราการบีบอัดกับอัตราเฉลี่ยของข้อมูล Sportnews

จากรูปที่ 4.9 เป็นการเปรียบเทียบการบีบอัดของข้อมูล Sportnews ด้วยจำนวนสัญลักษณ์ 1 2 และ 3 สัญลักษณ์กับอัตราเฉลี่ยของข้อมูลหรือเอนโทรปี (entropy) จะเห็นว่าเส้นกราฟของการบีบอัดข้อมูลด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ มีค่าใกล้เคียงกับอัตราเฉลี่ยของข้อมูลมากที่สุด

เช่นเดียวกับข้อมูล Entertainment นั่นก็หมายความว่ามีความสามารถในการบีบอัดข้อมูลได้มากกว่าการบีบอัดด้วยจำนวนสัญลักษณ์ 1 และ 2 สัญลักษณ์

รูปกราฟที่ 4.10 แสดงการเปรียบเทียบการบีบอัดของข้อมูล Sportnews ด้วยจำนวนคำและจำนวนสัญลักษณ์ที่แตกต่างกันเช่นเดียวกับรูปที่ 4.8 ที่มีแกนตั้ง เป็นจำนวนบิตข้อมูลก่อนที่จะทำการบีบอัด และแกนนอน เป็นจำนวนบิตข้อมูลที่ได้หลังจากการบีบอัด โดยเปรียบเทียบกันระหว่างเส้นกราฟของจำนวนสัญลักษณ์ที่แตกต่างกันทั้งสามเส้นดังแสดงตามรูปข้างล่างนี้



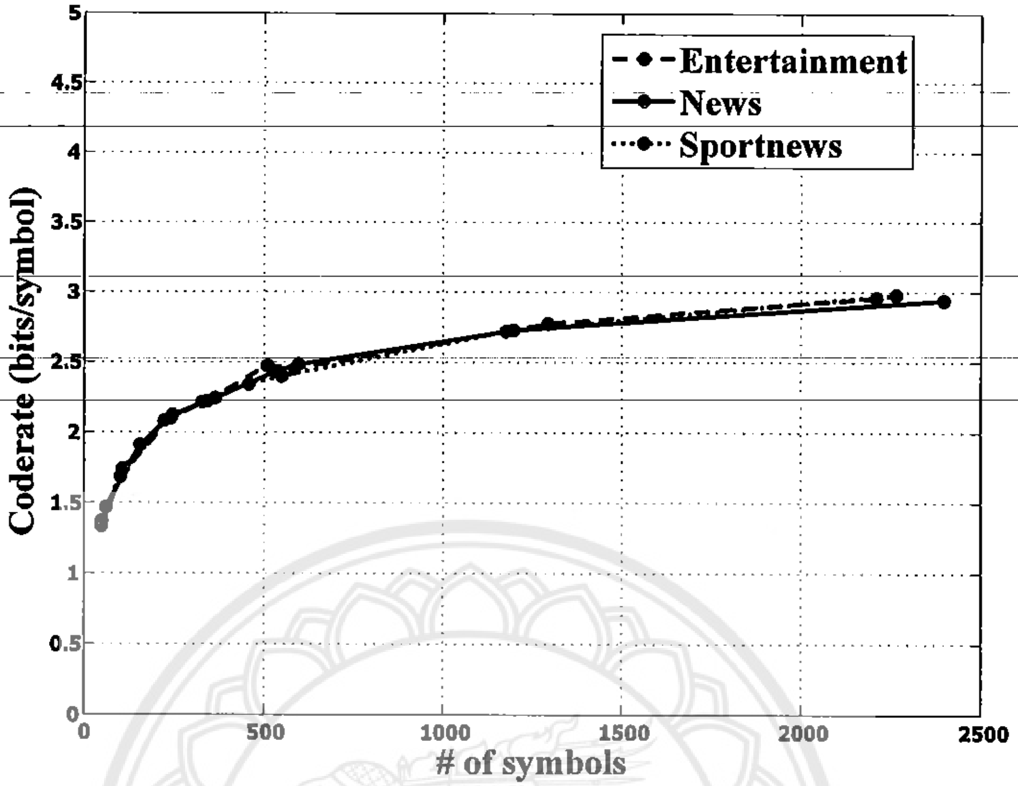
รูปที่ 4.10 เป็นการเปรียบเทียบของข้อมูล Sportnews ที่ได้หลังจากการบีบอัด เมื่อบีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน

จากรูปที่ 4.10 เมื่อเปรียบเทียบจำนวนบิตของข้อมูลที่ได้หลังจากการบีบอัดของข้อมูล Sportnews ด้วยจำนวนสัญลักษณ์ทั้ง 3 สัญลักษณ์แล้ว จะเห็นว่า เส้นกราฟของจำนวนสัญลักษณ์ 3 สัญลักษณ์มีความชันน้อยที่สุด ซึ่งก็หมายความว่ามีการส่งจำนวนบิตของข้อมูลที่ได้หลังจากการบีบอัดได้มากที่สุด เช่นเดียวกันกับข้อมูล Entertainment

จากกราฟทั้งหมดก่อนหน้านี้ได้แสดงการบีบอัดของข้อมูลทั้ง 3 ชนิด ซึ่งแต่ละชนิดก็บีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกันแล้ว ก็จะเห็นได้ว่าการบีบอัดของข้อมูลทั้ง 3 ชนิด โดยที่ถูกบีบอัดด้วยจำนวนสัญลักษณ์ที่ใช้ในการบีบอัด 3 สัญลักษณ์ มีค่าเข้าใกล้กับอัตราเฉลี่ยของข้อมูลมากกว่าการบีบอัดด้วยจำนวนสัญลักษณ์ 1 และ 2 สัญลักษณ์ อีกทั้งยังมีจำนวนบิตของข้อมูลที่ได้หลังจาการบีบอัดที่น้อยที่สุด ก็จะสามารถส่งข้อมูลไปได้มาก แต่ในขณะเดียวกันก็ทำให้ขนาดของพจนานุกรมของข้อมูลก็จะต้องใหญ่ จะเห็นได้จากการทดลองข้างต้นเมื่อทำการบีบอัดข้อมูล Sportnews ด้วยจำนวนคำ 102 คำ ซึ่งเมื่อทำการบีบอัดด้วยจำนวนสัญลักษณ์ 1 สัญลักษณ์ จะมีจำนวนของข้อมูลที่ต้องส่ง 332 อักขระ ถ้าทำการบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์ มีจำนวนของข้อมูลที่ต้องส่ง 1372 อักขระ และเมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ จะมีจำนวนของข้อมูลที่จะต้องส่ง 1569 อักขระ

ต่อไปจะเป็นการเปรียบเทียบกันของจำนวนสัญลักษณ์ที่เท่ากันในข้อมูลที่แตกต่างกัน ซึ่งจะได้แสดงให้เห็นในกราฟต่อไปนี้

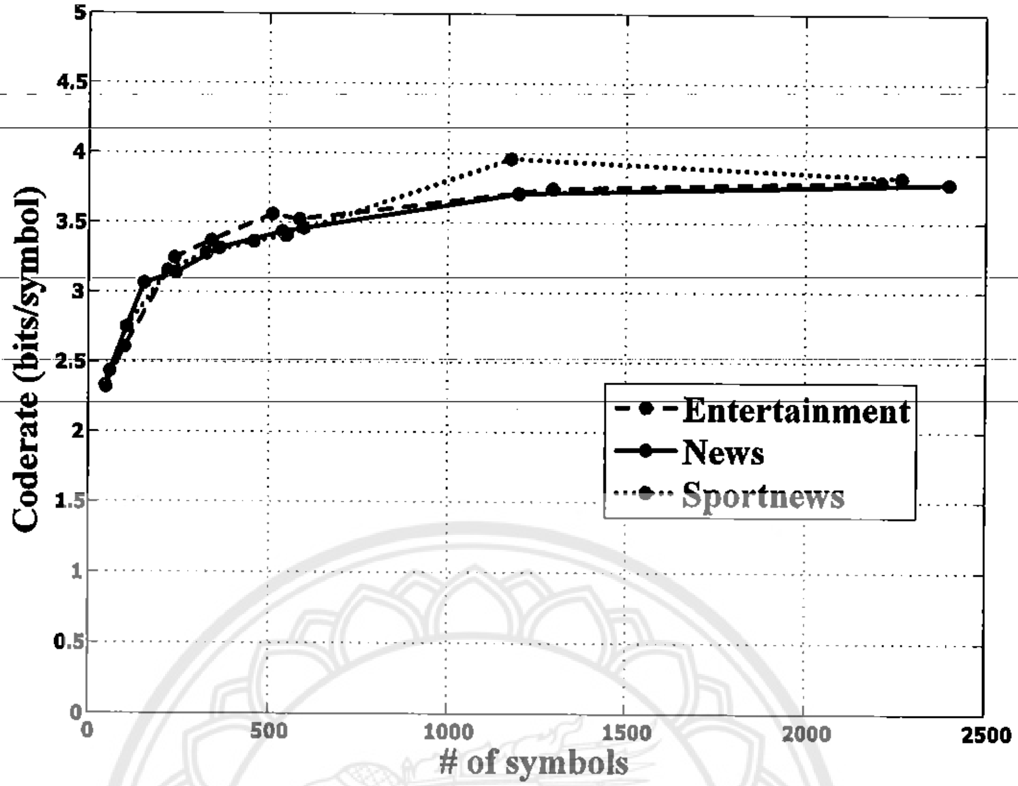
กราฟรูปที่ 4.11 แสดงการเปรียบเทียบกันของข้อมูลภาษาอังกฤษทั้งสามชนิด ได้แก่ Entertainment News และ Sportnews ที่มีจำนวนคำ 10 20 40 60 80 100 200 และ 400 คำ ตามลำดับ ด้วยจำนวนของสัญลักษณ์ที่ใช้ในการบีบอัด 1 สัญลักษณ์ โดยแกนตั้งแสดงอัตราในการบีบอัดข้อมูล และแกนนอนแสดงจำนวนสัญลักษณ์ของแต่ละกลุ่มคำ



รูปที่ 4.11 แสดงจำนวนของข้อมูลที่แตกต่างกันซึ่งมีการบีบอัดในจำนวน 1 สัญลักษณ์

จากรูปที่ 4.11 จะเห็นได้ว่า เมื่อนำจำนวนสัญลักษณ์ที่เท่ากันคือ 1 สัญลักษณ์มาเปรียบเทียบกันด้วยข้อมูลที่แตกต่างกัน จะสังเกตว่าข้อมูลทั้ง 3 ชนิด มีจำนวนของข้อมูลที่ถูกบีบอัดแล้วแตกต่างกันอย่างเห็นได้ชัดเจนอีกทั้งยังมีข้อมูลที่ยังมากด้วย

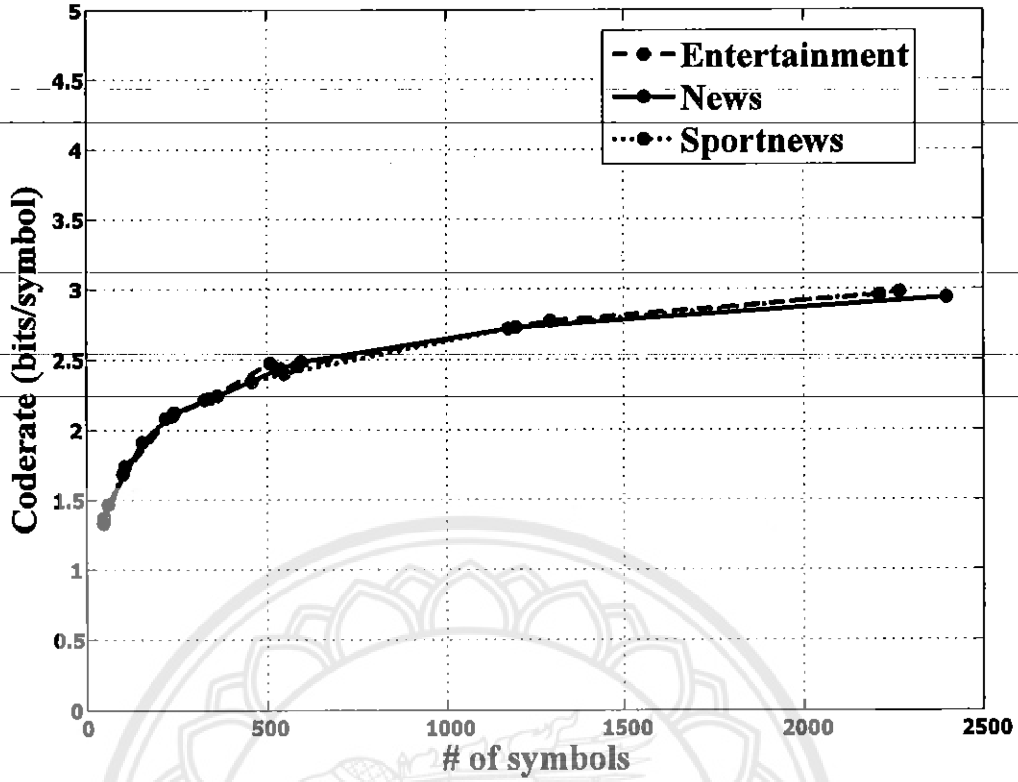
กราฟรูปที่ 4.12 แสดงการเปรียบเทียบกันของข้อมูลภาษาอังกฤษทั้งสามชนิดและมีจำนวนคำที่แตกต่างกันเช่นเดียวกันกับรูปที่ 4.11 ด้วยจำนวนของสัญลักษณ์ที่ใช้ในการบีบอัด 2 สัญลักษณ์ โดย แสดงอัตราในการบีบอัดข้อมูลในแนวแกนตั้ง และแสดงจำนวนสัญลักษณ์ของแต่ละกลุ่มคำในแนวแกนนอน



รูปที่ 4.12 แสดงจำนวนของข้อมูลที่แตกต่างกันเมื่อมีการบีบอัดในจำนวน 2 สัญลักษณ์

จากรูปที่ 4.12 เห็นว่าเมื่อบีบอัดข้อมูลในจำนวนสัญลักษณ์ 2 สัญลักษณ์ ของข้อมูลทั้ง 3 ชนิดแล้ว จำนวนข้อมูลที่ได้เริ่มจะมีค่าใกล้เคียงกันมากกว่ารูปที่ 4.11 และจำนวนของข้อมูลยังน้อยกว่าเมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์อีกด้วย

กราฟรูปที่ 4.13 แสดงการเปรียบเทียบกันของข้อมูลภาษาอังกฤษทั้งสามชนิดและมีจำนวนคำที่แตกต่างกันเช่นเดียวกันกับรูปที่ 4.12 ด้วยจำนวนของสัญลักษณ์ที่ใช้ในการบีบอัด 3 สัญลักษณ์ โดย แสดงอัตราในการบีบอัดข้อมูลในแนวแกนนอน และแสดงจำนวนสัญลักษณ์ของแต่ละกลุ่มคำในแนวแกนตั้ง



รูปที่ 4.13 แสดงจำนวนของข้อมูลที่แตกต่างกันซึ่งมีการบีบอัดในจำนวน 3 สัญลักษณ์

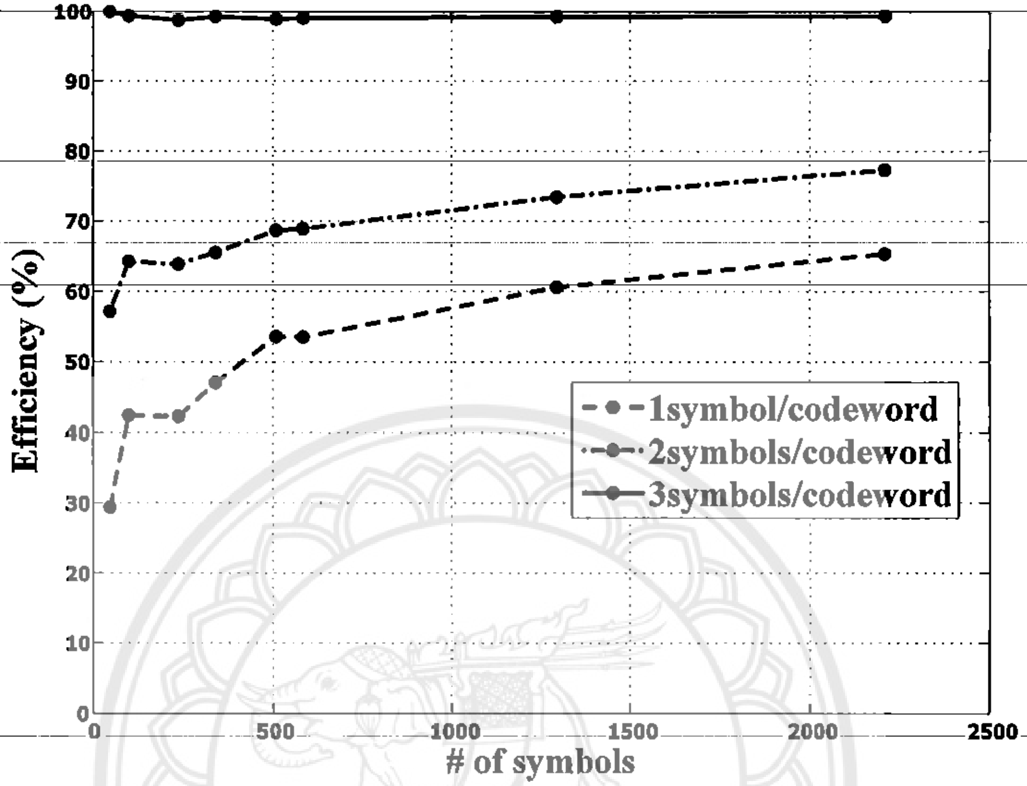
จากรูปที่ 4.13 จะเห็นว่าจำนวนของข้อมูลทั้ง 3 มีค่าใกล้เคียงกันมากที่สุดหลังจากบีบอัดข้อมูลแล้ว อีกทั้งยังมีจำนวนน้อยที่สุดด้วย

จะสังเกตได้ว่า เมื่อข้อมูลต่างชนิดกันถูกบีบอัดด้วยจำนวนสัญลักษณ์ที่เท่ากัน ก็จะมีจำนวนของข้อมูลที่ต่างกัน ดังแสดงในรูปที่ 4.11 4.12 และ 4.13 คือ เมื่อข้อมูลที่ถูกบีบอัดด้วยจำนวนสัญลักษณ์ 1 สัญลักษณ์ จะทำให้ข้อมูลมีจำนวนมากที่สุด และเมื่อถูกบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์ ก็ยังคงมีมากแต่ก็น้อยกว่าข้อมูลที่ถูกบีบอัดด้วยจำนวนสัญลักษณ์ 1 สัญลักษณ์ ในขณะเดียวกันเมื่อใช้จำนวนสัญลักษณ์ 3 สัญลักษณ์ในการบีบอัด ปรากฏว่าจำนวนของข้อมูลที่ได้มีค่าน้อยที่สุดอีกทั้งข้อมูลภาษาอังกฤษทั้ง 3 ชนิดยังมีค่าใกล้เคียงกันมากที่สุดอีกด้วย

4.4 กราฟแสดงการเปรียบเทียบประสิทธิภาพของการบีบอัดของแต่ละสัญลักษณ์ในแต่ละไฟล์ ซึ่งค่าที่นำมาพล็อตกราฟนี้ได้จากตารางที่ 4.2

ต่อไปเป็นกราฟแสดงการเปรียบเทียบประสิทธิภาพการบีบอัดของข้อมูล Entertainment ที่มีจำนวนคำที่แตกต่างกันได้แก่ 10 20 30 40 60 80 100 200 และ 400 คำตามลำดับ พร้อมกับจำนวน

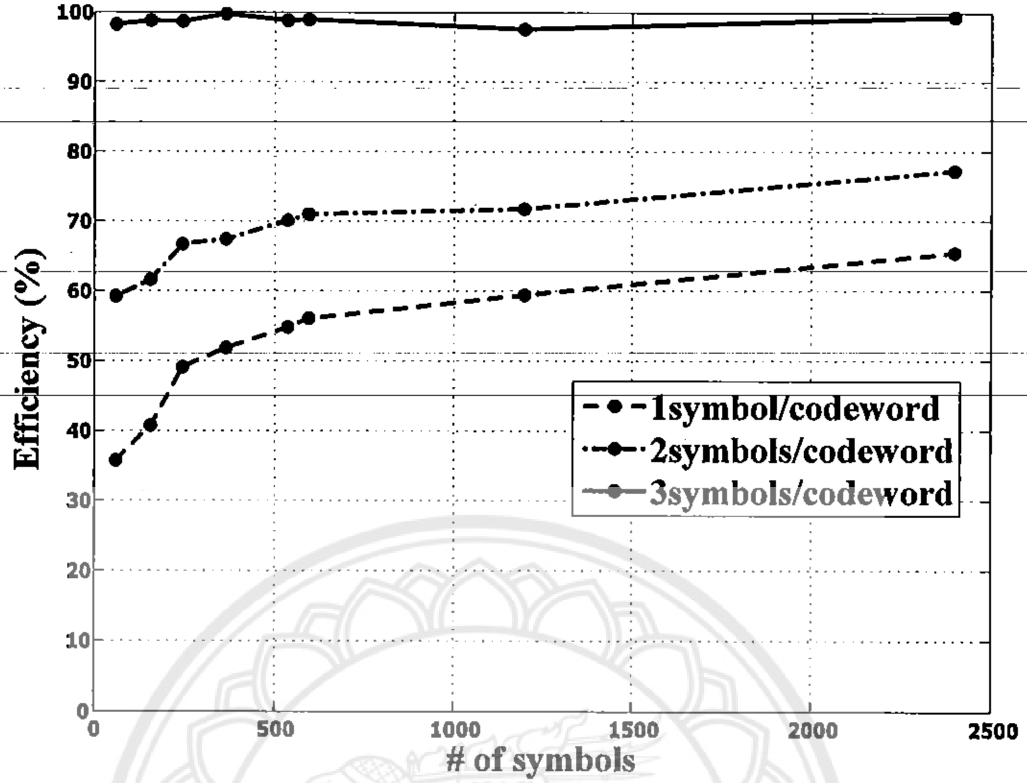
สัญลักษณ์ 1 2 และ 3 สัญลักษณ์ โดยเปรียบเทียบกันระหว่างประสิทธิภาพในการบีบอัดซึ่งอยู่ในแกนตั้ง และจำนวนสัญลักษณ์ของแต่ละกลุ่มคำ ซึ่งอยู่ในแกนนอน



รูปที่ 4.14 แสดงการเปรียบเทียบประสิทธิภาพการบีบอัดข้อมูลของแต่ละสัญลักษณ์ของข้อมูล Entertainment

จากกราฟรูปที่ 4.14 จะเห็นว่าเมื่อทำการบีบอัดข้อมูล Entertainment ด้วยจำนวนสัญลักษณ์ที่แตกต่างกันแล้วเส้นกราฟที่ได้จากการบีบอัดด้วยจำนวน 3 สัญลักษณ์ มีประสิทธิภาพในการบีบอัดมากที่สุด รองลงมาก็เป็นเส้นกราฟที่ได้จากการบีบอัดด้วยจำนวน 2 สัญลักษณ์ และจำนวน 1 สัญลักษณ์ ตามลำดับ

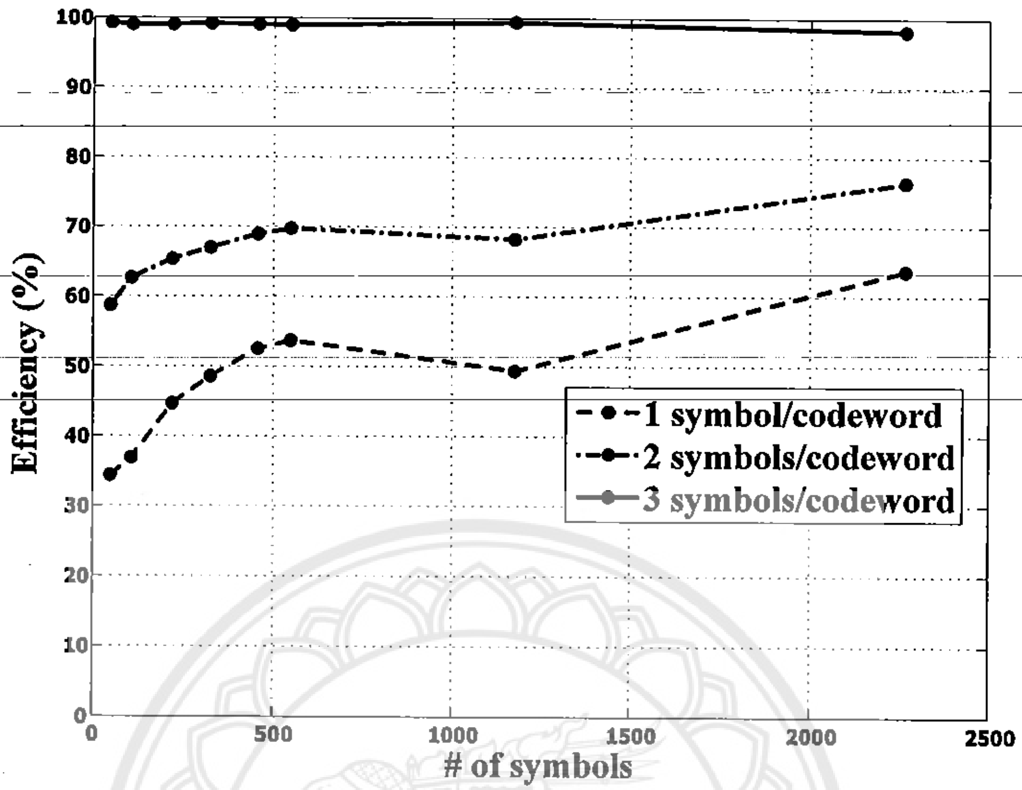
รูปกราฟที่ 4.15 แสดงการเปรียบเทียบประสิทธิภาพของการบีบอัดข้อมูลของข้อมูล News ที่มีจำนวนคำและจำนวนสัญลักษณ์ที่แตกต่างกันเช่นเดียวกับกราฟรูปที่ 4.14 ที่มีแกนตั้งเป็นประสิทธิภาพของการบีบอัดข้อมูล และแกนนอนเป็นจำนวนสัญลักษณ์ของแต่ละกลุ่มคำ



รูปที่ 4.15 แสดงการเปรียบเทียบประสิทธิภาพการบีบอัดข้อมูลของแต่ละสัญลักษณ์ของข้อมูล News

จากกราฟรูปที่ 4.15 จะเห็นว่าเมื่อทำการบีบอัดข้อมูล News ด้วยจำนวนสัญลักษณ์ที่แตกต่างกันแล้วเส้นกราฟที่ได้จากการบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์มีประสิทธิภาพสูงมากที่สุด รองลงมาเป็นการบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์ และ 1 สัญลักษณ์ ตามลำดับ

รูปกราฟที่ 4.16 แสดงการเปรียบเทียบประสิทธิภาพของการบีบอัดข้อมูลของข้อมูล Sportnews ที่มีจำนวนคำและจำนวนสัญลักษณ์ที่แตกต่างกันเช่นเดียวกับกราฟรูปที่ 4.15 ซึ่งมีแกนตั้งเป็นประสิทธิภาพของการบีบอัดข้อมูล และแกนนอนเป็นจำนวนสัญลักษณ์ของแต่ละกลุ่มคำ



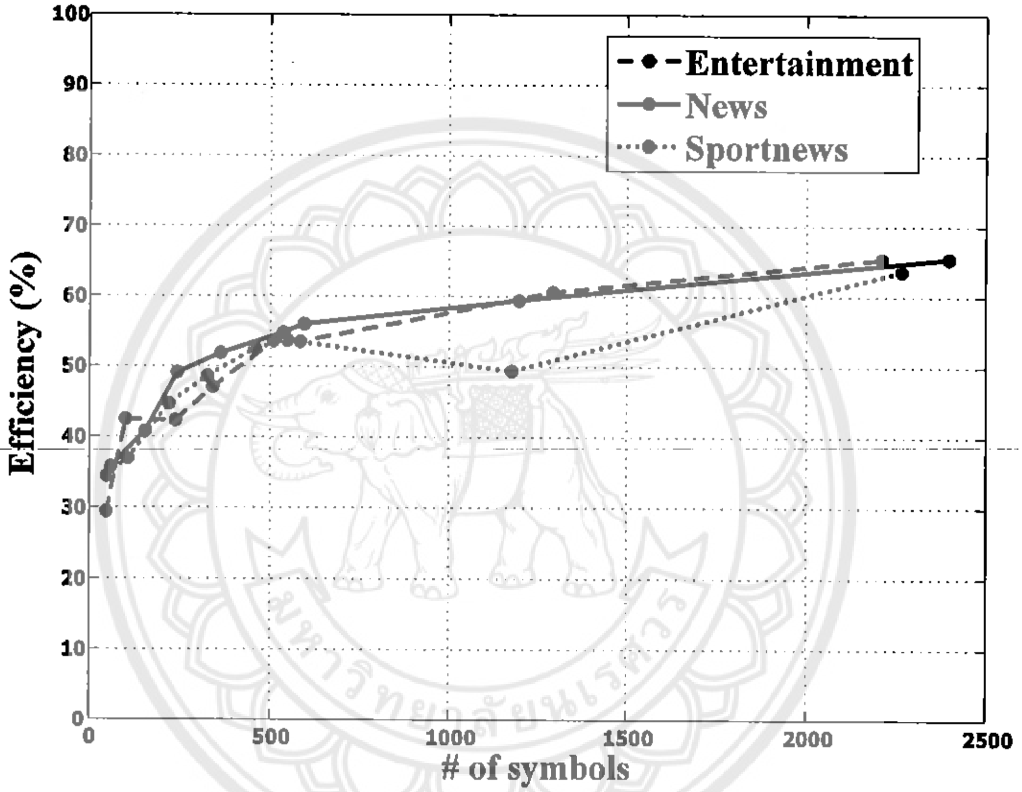
รูปที่ 4.16 แสดงการเปรียบเทียบประสิทธิภาพการบีบอัดข้อมูลของแต่ละสัญลักษณ์ของข้อมูล Sportnews

จากกราฟรูปที่ 4.16 จะเห็นได้ว่าเมื่อทำการบีบอัดข้อมูล Sportnews ด้วยจำนวนสัญลักษณ์ที่แตกต่างกันแล้วเส้นกราฟที่ได้จากการบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์มีประสิทธิภาพในการบีบอัดมากที่สุด เมื่อเทียบกับจำนวนสัญลักษณ์ 2 และ 1 สัญลักษณ์ ตามลำดับ

จะสังเกตได้ว่า เมื่อทำการบีบอัดข้อมูลด้วยจำนวนสัญลักษณ์ที่แตกต่างกันประสิทธิภาพในการบีบอัดก็จะแตกต่างกัน ดังแสดงในรูปที่ 4.14 4.15 และ 4.16 คือ เมื่อทำการบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ จะทำให้มีประสิทธิภาพในการบีบอัดข้อมูลมากที่สุด และเมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์ก็ยังคงมีประสิทธิภาพซึ่งน้อยกว่าเมื่อใช้ 3 สัญลักษณ์ในการบีบอัด ในขณะเดียวกันเมื่อใช้จำนวนสัญลักษณ์ 1 สัญลักษณ์ในการบีบอัดทำให้มีประสิทธิภาพในการบีบอัดน้อยที่สุด เพราะฉะนั้นการบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ มีประสิทธิภาพในการบีบอัดมากที่สุด

ต่อไปจะเป็นการเปรียบเทียบประสิทธิภาพในการบีบอัดข้อมูลของจำนวนสัญลักษณ์ที่เท่ากันในข้อมูลที่แตกต่างกัน ซึ่งจะได้แสดงให้เห็นในกราฟต่อไปนี้

รูปกราฟที่ 4.17 แสดงการเปรียบเทียบประสิทธิภาพในการบีบอัดของข้อมูลภาษาอังกฤษทั้งสามชนิด ได้แก่ Entertainment News และ Sportnews ที่มีจำนวนคำ 10 20 40 60 80 100 200 และ 400 คำตามลำดับ ซึ่งทำการบีบอัดด้วยจำนวนสัญลักษณ์ 1 สัญลักษณ์ โดยแกนตั้งเป็นประสิทธิภาพในการบีบอัด และแกนนอนเป็นจำนวนสัญลักษณ์ของกลุ่มคำ

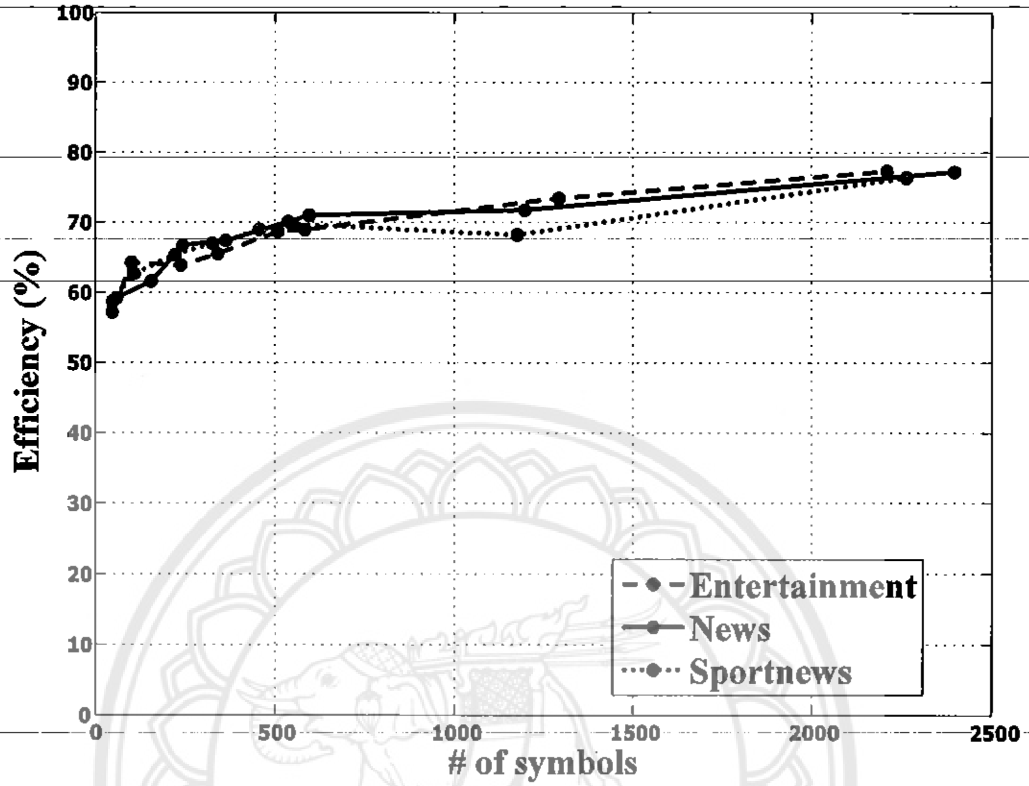


รูปที่ 4.17 แสดงประสิทธิภาพในการบีบอัดของข้อมูลที่แตกต่างกันโดยบีบอัดด้วยจำนวน 1 สัญลักษณ์

จากรูปที่ 4.17 เห็นได้ว่า เมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 1 สัญลักษณ์ มาเปรียบเทียบกันด้วยข้อมูลที่แตกต่างกัน สังเกตเห็นว่าข้อมูลทั้งสามชนิดมีประสิทธิภาพในการบีบอัดที่แตกต่างกันอีกทั้งยังมีประสิทธิภาพในการบีบอัดที่น้อยอีกด้วย

รูปกราฟที่ 4.18 แสดงการเปรียบเทียบประสิทธิภาพในการบีบอัดของข้อมูลทั้งสามชนิดและมีจำนวนคำที่แตกต่างกัน เช่นเดียวกันกับรูปที่ 4.17 ซึ่งทำการบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์

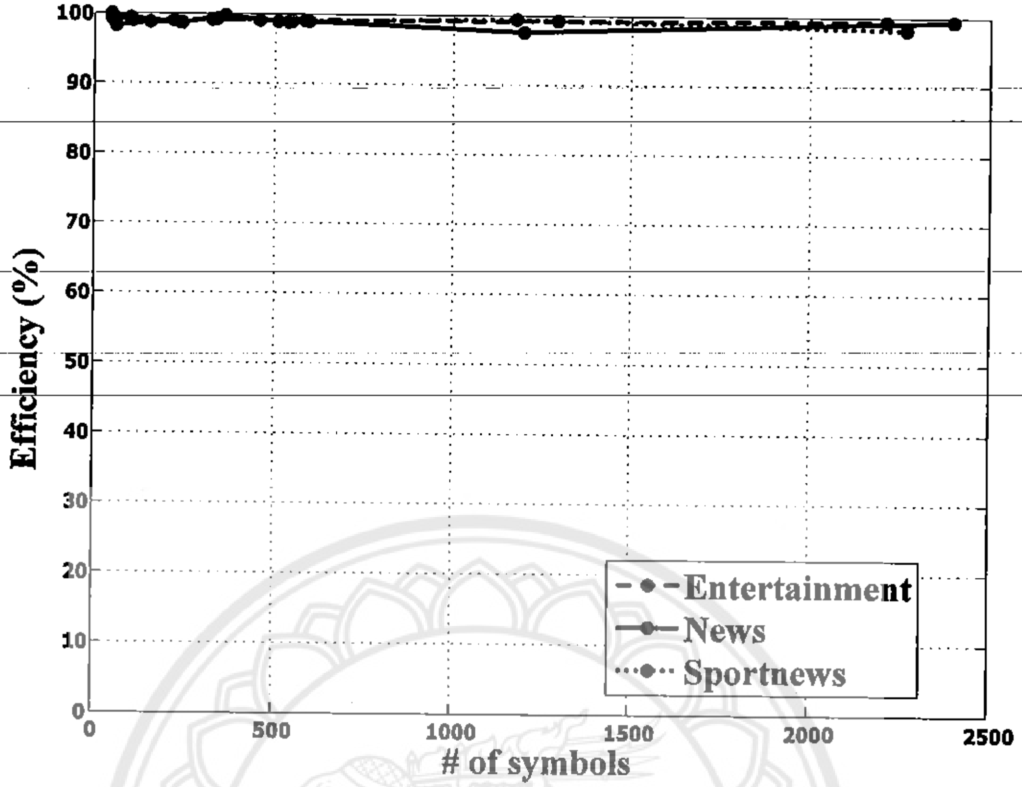
โดยแสดงประสิทธิภาพในการบีบอัดในแกนตั้งและแสดงจำนวนสัญลักษณ์ของแต่ละกลุ่มคำในแกนนอน



รูปที่ 4.18 แสดงประสิทธิภาพในการบีบอัดของข้อมูลที่แตกต่างกันโดยบีบอัดด้วยจำนวน 2 สัญลักษณ์

จากรูปที่ 4.18 เมื่อทำการบีบอัดข้อมูลทั้งสามชนิดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์แล้ว ประสิทธิภาพในการบีบอัดของข้อมูลยังแตกต่างกัน และยังมีประสิทธิภาพในการบีบอัดที่ยังน้อย

รูปกราฟที่ 4.19 แสดงการเปรียบเทียบประสิทธิภาพในการบีบอัดของข้อมูลทั้งสามชนิดและมีจำนวนคำที่แตกต่างกัน เช่นเดียวกันกับรูปที่ 4.18 ซึ่งทำการบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ โดยแสดงประสิทธิภาพในการบีบอัดในแกนตั้งและแสดงจำนวนสัญลักษณ์ของแต่ละกลุ่มคำในแกนนอน



รูปที่ 4.19 แสดงประสิทธิภาพในการบีบอัดของข้อมูลที่แตกต่างกันโดยบีบอัดด้วยจำนวน 3 สัญลักษณ์

จากรูปที่ 4.19 เมื่อทำการบีบอัดข้อมูลทั้งสามชนิดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ จะเห็นว่าข้อมูลทั้งหมดมีประสิทธิภาพในการบีบอัดที่ค่อนข้างจะใกล้เคียงกันอีกทั้งยังมีประสิทธิภาพในการบีบอัดมากที่สุด

จะสังเกตได้ว่า เมื่อทำการบีบอัดข้อมูลด้วยจำนวนสัญลักษณ์ที่เท่ากัน ในข้อมูลที่แตกต่างกัน ประสิทธิภาพในการบีบอัดก็จะแตกต่างกันด้วย ดังแสดงในรูปที่ 4.17 4.18 และ 4.19 คือ เมื่อทำการบีบอัดด้วยจำนวนสัญลักษณ์ สัญลักษณ์ จะทำให้มีประสิทธิภาพในการบีบอัดข้อมูลน้อยที่สุด เมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 2 สัญลักษณ์ก็ยังคงมีประสิทธิภาพซึ่งยังถือว่ามากกว่าเมื่อใช้ 2 สัญลักษณ์ในการบีบอัด เมื่อใช้จำนวนสัญลักษณ์ 3 สัญลักษณ์ในการบีบอัดทำให้มีประสิทธิภาพในการบีบอัดมากที่สุด ดังนั้นเมื่อบีบอัดด้วยจำนวนสัญลักษณ์ 3 สัญลักษณ์ จึงถือว่ามีประสิทธิภาพในการบีบอัดมากที่สุด

ดังนั้น จากผลการดำเนินงานของโครงการนี้ควรที่จะเลือกจำนวนสัญลักษณ์ที่ใช้ในการบีบอัดให้เหมาะสม เพราะว่าเมื่อข้อมูลถูกบีบอัดด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน จะทำให้จำนวนของ

ข้อมูลหรือบิตข้อมูลที่ได้ก็จะมีจำนวนต่างกัน แล้วยังส่งผลไปถึงขนาดของพจนานุกรมรวมถึงแบนด์วิดที่ถูกใช้งาน ซึ่งในที่นี้ถ้าหากเลือกจำนวนสัญลักษณ์ 3 สัญลักษณ์ในการบีบอัดข้อมูล ก็สามารถบีบอัดได้ดีกว่าจำนวนอื่นและบิตข้อมูลที่ได้ก็น้อย อีกทั้งยังมีประสิทธิภาพในการบีบอัดข้อมูลมากที่สุด จึงทำให้สามารถส่งข้อมูลได้ครั้งละมาก ๆ แต่ในขณะเดียวกันก็จะทำให้พจนานานุกรมมีขนาดใหญ่ไปด้วยเช่นกัน

ในบทที่ 4 นี้ ได้แสดงการบีบอัดข้อมูลโดยวิธีฮัฟแมนโดยใช้โปรแกรม MATLAB พร้อมกับใช้ GUI (Graphic User Interface) ในการแสดงผล แล้วทำการบีบอัดข้อมูลด้วยจำนวนสัญลักษณ์ที่แตกต่างกัน เพื่อนำมาเปรียบเทียบประสิทธิภาพในการบีบอัดของแต่ละสัญลักษณ์ เพื่อจะได้นำไปตัดสินใจในการเลือกใช้ให้เหมาะสม

ในบทต่อไปจะเป็นการสรุปผลการดำเนินงานของโครงงานนี้ พร้อมกับปัญหาที่พบในขณะที่ทำโครงงานและข้อเสนอแนะ เพื่อจะได้นำไปแก้ไขได้อย่างถูกต้องในโอกาสต่อไป

มหาวิทยาลัยนเรศวร

บทที่ 5

สรุปผลการดำเนินงาน

5.1 ผลการดำเนินโครงงาน

โครงงานนี้เป็นการจำลองการทำงานของการส่งข้อมูลในระบบสื่อสาร ด้วยการเข้ารหัสด้วยวิธีฮัฟแมนเพื่อบีบอัดข้อมูลที่จะใช้ในการส่ง ซึ่งจะส่งผลให้ข้อมูลที่จะสามารถส่งผ่านช่องสัญญาณมีปริมาณมากขึ้น โดยในการศึกษาจะอาศัยโปรแกรม MATLAB พร้อมกับแสดงการจำลองการทำงานออกมาในรูปของ GUI (Graphic User Interface) โดยทำการบีบอัดด้วยจำนวนสัญลักษณ์ 1 2 และ 3 สัญลักษณ์ ด้วยข้อมูลที่ใช้ในการศึกษาในครั้งนี้เป็นข้อมูลตัวอักษรภาษาอังกฤษ 3 ชนิดด้วยกัน ได้แก่ Entertainment News และ Sportnews ซึ่งแต่ละชนิดก็จะมีการศึกษาผลของจำนวนคำที่ใช้ในการบีบอัดแตกต่างกันไป นั่นคือ 10 20 40 80 100 200 และ 400 คำ โดยผลของการศึกษาได้ถูกแสดงโดยละเอียดตามตารางที่ 4.1 และมีการเปรียบเทียบประสิทธิภาพของการบีบอัดข้อมูลระหว่างจำนวนของสัญลักษณ์ที่ใช้กับอัตราเฉลี่ยของข้อมูลหรือเอนโทรปี (Entropy) และจำนวนบิตที่ได้หลังจากการบีบอัดออกมาในรูปของกราฟ เพื่อใช้ในการเลือกตัดสินใจว่าควรที่จะใช้จำนวนสัญลักษณ์ในการบีบอัดแต่ละครั้งเป็นจำนวนเท่าไรที่จะทำให้ประสิทธิภาพในการส่งข้อมูลดีที่สุด

จากการดำเนินโครงงานนี้สามารถสรุปผลได้ว่า จำนวนสัญลักษณ์ 3 สัญลักษณ์ที่ใช้ในการบีบอัดข้อมูลสามารถบีบอัดข้อมูลได้มากที่สุดเมื่อเปรียบเทียบกับอีก 2 กรณีที่เหลือ และมีอัตราการบีบอัดข้อมูลที่ใกล้เคียงกับอัตราเฉลี่ยของข้อมูล ซึ่งหากทำการบีบอัดโดยอาศัยจำนวนสัญลักษณ์ในการบีบอัดมากขึ้น ก็จะทำให้สามารถเพิ่มปริมาณข้อมูลข่าวสารในการสื่อสารไปยังปลายทางได้มากขึ้น แต่ข้อเสียของการบีบอัดโดยอาศัยจำนวนสัญลักษณ์ในการบีบอัดที่มีค่าสูง คือ ขนาดของพจนานุกรมจะมีขนาดใหญ่ขึ้น ซึ่งจะทำให้สิ้นเปลืองแบนด์วิดในการส่งตัวพจนานุกรมนี้ไปยังปลายทางเช่นกัน

5.2 ปัญหาที่พบขณะดำเนินงาน

5.2.1 การแสดงผลการบีบอัดข้อมูลนี้ได้แสดงออกมาในรูปของ GUI (Graphic User Interface) ดังนั้นเครื่องคอมพิวเตอร์ควรใช้โปรแกรม MATLAB เวอร์ชั่น 7 ขึ้นไป เพื่อที่ได้จะแสดงผลออกมาได้

5.2.1 ผู้ดำเนินโครงงานยังไม่ค่อยมีความเข้าใจอย่างถ่องแท้กับการใช้คำสั่งบางคำสั่งในโปรแกรม MATLAB ทำให้การดำเนินงานช้ากว่าระยะเวลาที่กำหนด

5.3 ข้อเสนอแนะ

5.3.1 โครงงานนี้สามารถนำไปปรับใช้เป็นสื่อการเรียนการสอนได้

5.3.2 เปรียบเทียบในตัวอักษรภาษาไทยหรือข้อมูลในลักษณะอื่น

5.3.3 ศึกษาขนาดของพจนานุกรมที่ใช้ในการส่งข้อมูล

เอกสารอ้างอิง

[1] บัณฑิต โรจน์อารยานนท์. **หลักการไฟฟ้าสื่อสาร.** กรุงเทพฯ: จุฬาลงกรณ์มหาวิทยาลัย, 2541.

[2] ลัญฉกร วุฒิสิทธิกุลกิจ. **หลักการไฟฟ้าสื่อสาร.** กรุงเทพฯ: จุฬาลงกรณ์มหาวิทยาลัย, 2546.

[3] Proakis, John G., and Masoud Salehi. **Communication Systems Engineering.** 2nd ed, New Jersey: Prentice-Hall, Inc., 2002.

[4] Bose Ranjan. **Information Theory, Coding and Cryptography.** Singapore: McGraw-Hill companies, 2003.

[5] Carlson, A., et al. Bruce. **Communication System.** 4th ed, New York: McGraw-Hill companies, Inc., 2002.

[6] Shanmugam, K. Sam. **Digital and Analog Communication systems.** Singapore: John Wiley & Sons, Inc., 1979.

[7] รศ.ดร.มนัส สังวงศิลป์ และวรรัตน์ ภัทรอมรกุล. **"คู่มือการใช้งาน MATLAB ฉบับสมบูรณ์".** พิมพ์ครั้งที่ 1: สำนักพิมพ์ อินโฟเพรส. 2543.

[8] ลัญฉกร วุฒิสิทธิกุลกิจ. **"MATLAB การประยุกต์ใช้งานทางวิศวกรรมไฟฟ้า".** พิมพ์ครั้งที่ 1: สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย. 2547.

ภาคผนวก ก

โปรแกรมการจำลองการบีบอัดข้อมูลด้วยวิธีฮัฟแมนโดยใช้โปรแกรม MATLAB

```
%%%HUFFMAN  CODING COMPRESSION%%%
function [Comp, outPut, Coderate, lengthOut]=ff(input, cha, choise)
%%%CHOOSE DATA FILE OR INSERT DATA%%%
if choise==1
   Data = input;
elseif choise==2
   fid = fopen(input, 'r');
   Data = fread(fid, 'uint8=>char');
   fclose(fid);
   Data = Data';
   Data(int32(Data)==13)=[];
End

%%%CHOOSE NUMBER OF SYMBOL FOR COMPREESION%%%
Dict=struct('text',{});
n = cha;

nMod = mod(length(Data),n);
if nMod~=0
   for i=1:1:(n-nMod)
      Data=[Data '$'];
   end
end
Data;
k=1;
for ii=1:n:length(Data)-(n-1)
   S=[Data(ii:ii+n-1)];
   isRe=false;
```

```
for jj=1:1:length(Dict)
  if Dict(jj).text==S
    counter(jj)=counter(jj)+1;
    isRe=true;
    break;
  end
end

if isRe==false
  Dict(k).text=S;
  counter(k)=1;
  k=k+1;
end
end

if nMod~=0
  Dict(end).text=[Dict(end).text(1:nMod)];
End

%%%%%%%%FIND CODEWORD EACH OF SYMBOL%%%%%%%%
prob=counter/sum(counter);
prob_dummy=prob;
P=struct('path', {});
P(1).path=[];

for n=1:1:length(prob_dummy)-1
pointStart=[1:1:length(prob_dummy)];

minProbFirst = min(prob_dummy);
[row1 colFirst] = find(prob_dummy==minProbFirst);
```

```
prob_dummy(colFirst(1))=2;

minProbSec = min(prob_dummy);
[row1 colSec] = find(prob_dummy==minProbSec);

colSec_dummy = colSec(1);
if colFirst(1)<colSec(1)
   colSec(1)=colSec(1)-1;
end

prob_dummy = prob_dummy([1:colFirst(1)-1 colFirst(1)+1:end]);
prob_dummy = prob_dummy([1:colSec(1)-1 colSec(1)+1:end]);

pointStart = pointStart([1:colFirst(1)-1 colFirst(1)+1:end]);
pointStart = pointStart([1:colSec(1)-1 colSec(1)+1:end]);

sumMin = minProbFirst+minProbSec;

prob_dummy = [prob_dummy sumMin];

pointStart = [pointStart colSec_dummy colFirst(1)];
pointEnd = [1:1:length(prob_dummy) length(prob_dummy)];

P_dummy = ones(length(prob_dummy)+1, length(prob_dummy))*(-1);

for ii=1:1:length(prob_dummy)+1

   if ii==length(prob_dummy)
      P_dummy(pointStart(ii), pointEnd(ii))=0;
   elseif ii==length(prob_dummy)+1
```

```
            P_dummy(pointStart(ii), pointEnd(ii))=1;
        else
            P_dummy(pointStart(ii), pointEnd(ii))=2;
        end
    end
        P(n).path=P_dummy;
    end

    C=struct('codeword', {});
    for ii=1:1:length(prob)
        r=ii;
        C(ii).codeword=[];
        for jj=1:1:length(prob)-1
            degree=P(jj).path;
            c=find(degree(r, :)>=0);
            if degree(r, c)~=2
                C(ii).codeword=[num2str(degree(r, c)) C(ii).codeword];
            end
            r = c;
        end
    end
    C(1:end).codeword;
    dictOut=struct2cell(Dict(:))';
    temp=char(dictOut);
    temp(int32(temp)==10)=char(96);
    dictOut=cellstr(temp);
    probOut=cellstr(num2str(prob'));
    codeOut=struct2cell(C(:))';

    outPut=cellstr(['Symbol(s)  '; 'Prob       ';'Codeword   '])';
    outPut=[outPut; [dictOut probOut codeOut]];
```

```
%%%FIND CODERATE OF EACH FILE%%%
for ii=1:1:length(prob)
   A(ii)=prob(ii)*length(C(ii).codeword);
end

R=sum(A)/length(Dict(1).text);
length(Dict);
r=num2str(R);
unit='     bits per characters';
Coderate=[r];% unit]

%%%DICTIONARY & DICTIONARY LENGTH%%%
n=cha;
AA=[];
Comp=[];

for ii=1:cha:length(Data)-(cha-1)
   for jj=1:1:length(Dict)-1
      if [Data(ii:ii+cha-1)]==Dict(jj).text
         Comp=[Comp C(jj).codeword ,blanks(2)];
         AA=[AA C(jj).codeword];
      end
   end
end
Comp=[Comp C(end).codeword];
AA=[AA C(end).codeword];
BB=length(AA);
CC=num2str(BB);
lengthOut=[CC];
```

ภาคผนวก ข

โปรแกรมการแสดงผลการจำลองการบีบอัดของข้อมูลด้วยวิธีฮัฟแมน โดยแสดงออกมาในรูปแบบของ Graphic User Interface (GUI)

```
%%%HUFFMAN CODING FOR GRAPHIC USER INTERFACE%%%
function varargout = huffman(varargin)
% HUFFMAN M-file for huffman.fig
%    HUFFMAN, by itself, creates a new HUFFMAN or raises the existing
%    singleton*.
%
%    H = HUFFMAN returns the handle to a new HUFFMAN or the handle to
%    the existing singleton*.
%
%    HUFFMAN('CALLBACK',hObject,eventData,handles,...) calls the local
%    function named CALLBACK in HUFFMAN.M with the given input arguments.
%
%    HUFFMAN('Property','Value',...) creates a new HUFFMAN or raises the
%    existing singleton*.  Starting from the left, property value pairs are
%    applied to the GUI before huffman_OpeningFunction gets called.  An
%    unrecognized property name or invalid value makes property application
%    stop.  All inputs are passed to huffman_OpeningFcn via varargin.
%
%    *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%    instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Copyright 2002-2003 The MathWorks, Inc.

% Edit the above text to modify the response to help huffman
```

```
% Last Modified by GUIDE v2.5 11-Jul-2007 12:55:08

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @huffman_OpeningFcn, ...
                   'gui_OutputFcn',  @huffman_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
  gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
   [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
   gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before huffman is made visible.
function huffman_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to huffman (see VARARGIN)

if length(varargin)~=0
```

```
  temp=varargin;
  temp(1);
  %temp(2)
end
handles.current_data=varargin;

% Choose default command line output for huffman
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes huffman wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = huffman_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

function nWindows_Callback(hObject, eventdata, handles)
% hObject    handle to nWindows (see GCBO)
```

```
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of nWindows as text
%        str2double(get(hObject,'String')) returns contents of nWindows as a double

% --- Executes during object creation, after setting all properties.
function nWindows_CreateFcn(hObject, eventdata, handles)
% hObject    handle to nWindows (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Executes on button press in browse.
function browse_Callback(hObject, eventdata, handles)
% hObject    handle to browse (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
[FileName,PathName] = uigetfile;
set(handles.filePath, 'String', [PathName FileName]);
huffman([PathName FileName]);

% --- Executes on selection change in showOutPut.
function showOutPut_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to showOutPut (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns showOutPut contents as
%cell array
%       contents{get(hObject,'Value')} returns selected item from
%       showOutPut

% --- Executes during object creation, after setting all properties.
function showOutPut_CreateFcn(hObject, eventdata, handles)
% hObject    handle to showOutPut (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
   set(hObject,'BackgroundColor','white');
else
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Executes on selection change in dictionary.
function dictionary_Callback(hObject, eventdata, handles)
% hObject    handle to dictionary (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hints: contents = get(hObject,'String') returns dictionary contents as cell array
%       contents{get(hObject,'Value')} returns selected item from
%       dictionary
```

```
if length(handles.current_data)~=0
   huffman(handles.current_data);
end


% --- Executes during object creation, after setting all properties.
function dictionary_CreateFcn(hObject, eventdata, handles)
% hObject    handle to dictionary (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
   set(hObject,'BackgroundColor','white');
else
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function dataInput_Callback(hObject, eventdata, handles)
% hObject    handle to dataInput (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of dataInput as text
%        str2double(get(hObject,'String')) returns contents of dataInput as a double


% --- Executes during object creation, after setting all properties.
function dataInput_CreateFcn(hObject, eventdata, handles)
% hObject    handle to dataInput (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
```

```
% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
   set(hObject,'BackgroundColor','white');
else
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Executes on button press in run.
function run_Callback(hObject, eventdata, handles)
% hObject    handle to run (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
temp=handles.current_data;
if length(temp)~=0
   mm=char(temp);
   fileName=mm(1,:);
   fileName=cellstr(fileName);
   fileName=char(fileName);
   n=get(handles.nWindows, 'String');
   n=str2num(n);
   if ismember(n, [1 2 3])
      fileName;
      [out, DPC, CR, LO]=ff(fileName, n, 2);
      out;

      %out=['12345678910111213123456789101'];%%
      charPerLine=70;
      jj=1;
      ii=charPerLine;
      while ii<length(out)
```

```
        finalOut(jj, 1)=cellstr(out(ii-charPerLine+1:ii));
        jj=jj+1;
        ii=ii+charPerLine;
    end
    finalOut(jj, 1)=cellstr(out(ii-charPerLine+1:end));

%   a=['100';'001'];%%
%   b=['234';'567'];%%
%   c=['567';'123'];%%
%   DPC=[cellstr(a) cellstr(b) cellstr(c)];%%

    nCol=3;
    dpc=char(DPC);
    nStep=length(dpc)/nCol;
    aSpace=[];
    for ii=1:1:nStep
        aSpace=[aSpace; '          '];
    end
    finaldpc=[];
    for k=1:nStep:length(dpc)
        finaldpc=[finaldpc dpc([k:k+nStep-1], :) aSpace];
    end

    set(handles.showOutPut, 'String', finalOut);
    set(handles.dictionary, 'String', finaldpc);
    set(handles.codeRate, 'String', CR);
    set(handles.lengthOutput, 'String', LO);

else
    set(handles.showOutPut, 'String', []);
    set(handles.dictionary, 'String', []);
```

```
        set(handles.codeRate, 'String', []);
        set(handles.lengthOutput, 'String', []);
    end

    send=...
    [cellstr(fileName);...

    cellstr('=====================================================================
    ===');...
        cellstr(['DATE : ' date]);...

    cellstr('=====================================================================
    ===');...
        cellstr(['FILE NAME : ' fileName ]);...

    cellstr('=====================================================================
    ===');...
        cellstr(['Number of symbols passed  :  ' num2str(n) ' ' ]);...

    cellstr('=====================================================================
    ===');...
        cellstr('DICTIONARY : ');...
        cellstr(finaldpc);...

    cellstr('=====================================================================
    ===');...
        cellstr('COMPRESSED  DATA : ');...
        cellstr(finalOut);...

    cellstr('=====================================================================
    ===');...
```

```
    cellstr(['Length of compressed data : ' LO ' ']);...

   cellstr('=================================================================
===');...
    cellstr(['CODERATE : ' CR '  bits per symbol']);...

   cellstr('=================================================================
===')];
    %send=send';

    huffman(char(send));
   end

   % --- Executes on button press in runDataInput.
   function runDataInput_Callback(hObject, eventdata, handles)
   % hObject    handle to runDataInput (see GCBO)
   % eventdata  reserved - to be defined in a future version of MATLAB
   % handles    structure with handles and user data (see GUIDATA)
   temp=handles.current_data;
   input=get(handles.dataInput, 'String');
   n=get(handles.nWindows, 'String');
   n=str2num(n);
   if ismember(n, [1 2 3]) & length(input)~=0
     [out, DPC, CR, LO]=ff(input, n, 1);
     %out=['12345678910111213123456789101'];%%
     charPerLine=70;
     jj=1;
     ii=charPerLine;
     while ii<length(out)
       finalOut(jj, 1)=cellstr(out(ii-charPerLine+1:ii));
       jj=jj+1;
```

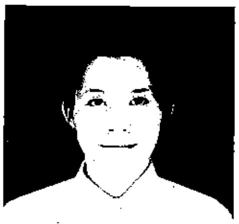
```
    ii=ii+charPerLine;
  end
  finalOut(jj, 1)=cellstr(out(ii-charPerLine+1:end));

  nCol=3;
  dpc=char(DPC);
  nStep=length(dpc)/nCol;
  aSpace=[];
  for ii=1:1:nStep
     aSpace=[aSpace; '                    '];
  end
  finaldpc=[];
  for k=1:nStep:length(dpc)
     finaldpc=[finaldpc dpc([k:k+nStep-1], :) aSpace];
  end

  set(handles.showOutPut, 'String', finalOut);
  set(handles.dictionary, 'String', finaldpc);
  set(handles.codeRate, 'String', CR);
  set(handles.lengthOutput, 'String', LO);

  send=...

[cellstr('==================================================================
====');...
     cellstr(['DATE : ' date])

cellstr('=================================================================
===');...
     cellstr(['DATA INPUT : ' input]);...
```

```
cellstr('====================================================================
===');...
     cellstr(['Number of symbols passed  :  ' num2str(n) ' ' ]);...

cellstr('====================================================================
===');...
     cellstr('DICTIONARY : ');...
     cellstr(finaldpc);...

cellstr('====================================================================
===');...
     cellstr('COMPRESSED  DATA : ');...
     cellstr(finalOut);...

cellstr('====================================================================
===');...
     cellstr(['Length of compressed data :  ' LO ' ']);...

cellstr('====================================================================
===');...
     cellstr(['CODERATE :  ' CR '   bits per symbol']);...

cellstr('====================================================================
===')];

    if length(temp)~=0
      mm=char(temp);
      fileName=mm(1,:);
      send=[cellstr(fileName); send];
    end
```

```
        huffman(char(send));

    else
        set(handles.showOutPut, 'String', []);
        set(handles.dictionary, 'String', []);
        set(handles.codeRate, 'String', []);
        set(handles.lengthOutput, 'String', []);
        %huffman(temp(1));
    end

% --- Executes on button press in writeFile.
function writeFile_Callback(hObject, eventdata, handles)
% hObject    handle to writeFile (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

temp=handles.current_data;
fileName=get(handles.writeFileName, 'String');
if length(fileName)~=0
    temp=char(temp);
    temp=temp(2:end, :);

    [sr sc]=size(temp);
    dummy=[];
    for ii=1:1:sr
        dummy=[dummy temp(ii,:) char(10)];
    end

    fid = fopen(fileName,'wb');
    fwrite(fid,dummy,'char');
    fclose(fid);
```

```
end

function writeFileName_Callback(hObject, eventdata, handles)
% hObject    handle to writeFileName (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of writeFileName as text
%        str2double(get(hObject,'String')) returns contents of outFileName as a double

% --- Executes during object creation, after setting all properties.
function writeFileName_CreateFcn(hObject, eventdata, handles)
% hObject    handle to writeFileName (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
  set(hObject,'BackgroundColor','white');
else
  set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
```

ประวัติผู้เขียน



ชื่อ นางสาววาสนา นาคุ

ภูมิลำเนา 15 ม.6 ต.บ้านโฮ่ง อ.บ้านโฮ่ง จ.ลำพูน 51130

ประวัติการศึกษา

-จบระดับมัธยมศึกษาจากโรงเรียนธีรกานท์บ้านโฮ่ง

-ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : wassana_ee@hotmail.com