Name: ROTHA Dapravith
ID: e20190915
Group: I5-GIC(B)

Assignment Discussion Lesson08

1) Explain LZ78 compression algorithm?

2) By using LZ78 compression algorithm, find encoder and decoder from 2 different examples? You can choose your own string.

Answer:

1). Explain LZ78 compression algorithm:

LZ78 is the lossless data compression algorithm, It has high requirements on spaces because the dictionary can occupy the whole free memory. There are two compression types of algorithms include Encoder and Decoder.

- Encoder algorithm of LZ78:

  1). Find all different symbols from an input string or a file.

  2). Build a base dictionary which contains entries and codewords.

  3). Build an output dictionary which contains entries, codewords, and outputs.

  4). Start to do loop from the first symbol.

  5). If a match is found from the base dictionary, add another symbol and compare again.

  6). If a match is not found from the base dictionary, output codeword of those symbol (form: < ?,?>).

  7). Repeat to do loop from next matching symbol until the last symbol or End Of File (EOF).

- Decoder algorithm of LZ78:

  1). Use the base dictionary from encoder.

  2). Use result of encoder as input.

  3). Build an output dictionary which contains inputs, codewords, entries, and outputs (entries must exactly the same as outputs).

  4). Start to do loop for finding outputs by using the input one by one.

  5). Repeat loop until End Of File.

2). Example 1: Find encoder and decoder of string “aabbddeggfs”

- Build a base dictionary

Entry	Codeword
a	1
b	2
d	3
e	4
g	5
f	6
s	7

- Encoder:

Entry	Codeword	Output
aa	8	<1,a>
bb	9	<2,b>
dd	10	<3,d>
egg	11	<5,g>
fs	12	<6,s>
EOF	EOF	<4, >

Thus, Encoder = {<1,a> <2,b> <3,d> <5,g> <6,s> <4, >}

- Decoder:

Input	Codeword	Entry	Output
<1,a>	8	aa	aa
<2,b>	9	bb	bb
<3,d>	10	dd	dd
<5,g>	11	gg	gg
<6,s>	12	fs	fs
<4, >	EOF	e	e

Thus, decoder of string "aabbddeggfs" is "aabbddggfse"

Example 2: Find Encoder and Decoder of string "greatgigsgreatgigs"

- Base dictionary

Entry	Codeword
g	1
r	2
e	3
a	4
t	5
i	6
s	7

- Encoder:

Entry	Codeword	Output
gr	8	<1,r>
re	9	<2,e>
eat	10	<3,a>
atg	11	<4,t>
tg	12	<5,g>
gi	13	<1,i>
ig	14	<6,g>
gs	15	<1,s>
rea	16	<9,a>
atgi	17	<11,i>
EOF	EOF	<7, >

Thus, Encoder of string “greatgigsgreatgigs” = { <1,r> <2,e> <3,a> <4,t> <5,g> <1,i> <6,g> <1,s> <9,a> <11,i> <7, > }

- Decoder:

Input	Codeword	Entry	Output
<0,g>	8	g	g
<0,r>	9	r	r
<0,e>	10	e	e
<0,a>	11	a	a
<0,t>	12	t	t
<1,i>	13	gi	gi
<1,g>	14	gg	gg
<1,s>	15	gs	gs
<2,r>	16	re	re
<3,a>	17	ea	ea
<4,t>	18	ta	ta
<1,g>	19	ge	ge
<1,i>	20	gi	gi
<1,g>	21	gg	gg
<1,s>	22	gs	gs
EOF	EOF	-	-

Thus, Decoder is greatgiggsgreatgigs