Name: VEN THON
ID: e20191250
Group: I5-GIC-C

Discussion08

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

Answers

1. Explain LZ77 compression algorithm:
   - Find all different symbols from an input string or a file.
   - Build a base dictionary which contains entries and codewords.
   - Build an output dictionary which contains entries, codewords, and outputs.
   - Start to do loop from the first symbol.
   - If a match is found from the base dictionary, add another symbol and compare agin.
   - If a match is not found from the base dictionary, output codeword of those symbol (from: <?,?>).
   - Repeat to do loop from next matching symbol until the last symbol or End of File (EOF).
2. By using LZ78 compressing algorithm, find encode and decode from 2 different examples.

abdecacbabbcdebedacba

❖ **Encoder:**

✓ First, we find all different characters.

There are "a, b, c, d, e".

✓ Step 2: Then build a base dictionary

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

✓ Input string: **"abdecacbabbcdebedacba".**

a	b	d	e	c	a	c	b	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

- Start from "a".
- "a" has in the base dictionary
- So, we take "ab".
- "ab" doesn't have in the dictionary.
- We add "ab" to output dictionary.
- The last codeword is 5.

- New codeword must be 6.

Entry	Codeword	Output
ab	6	<1,2>

➢ Now we are at “d”.

- “d” has in the base dictionary
- So, we take “de”.
- “de” doesn’t have in the both dictionary.
- We add “de” to output dictionary.
- The last codeword is 6.
- New codeword must be 7.

a	b	d	e	c	a	c	b	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>

➢ Now we are at “c”.

- “c” has in the base dictionary
- So, we take “ca”.
- “ca” doesn’t have in the both dictionary.
- We add “ca” to output dictionary.
- The last codeword is 7.
- New codeword must be 8.

a	b	d	e	c	a	c	b	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>

➢ Now we are at “c”.

- “c” has in the base dictionary
- So, we take “cb”.
- “cb” doesn’t have in the both dictionary.
- We add “cb” to output dictionary.
- The last codeword is 8.
- New codeword must be 9.

a	**b**	**d**	**e**	**c**	**a**	c	b	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>
cb	9	<3,2>

➢ Now we are at “a”.

- “a” has in the base dictionary
- So, we take “ab”.
- “ab” has in the base dictionary
- So, we take “abb”.
- “abb” doesn’t have in the both dictionary.
- We add “abb” to output dictionary.
- The last codeword is 9.
- New codeword must be 10.

a	**b**	**d**	**e**	**c**	**a**	**c**	**b**	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3

d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>
cb	9	<3,2>
abb	10	<6,2>

- ➢ Now we are at “c”.
- “c” has in the base dictionary
- So, we take “cd”.
- “cd” doesn’t have in the both dictionary.
- We add “cd” to output dictionary.
- The last codeword is 10.
- New codeword must be 11.

a	**b**	**d**	**e**	**c**	**a**	**c**	**b**	**a**	**b**	**b**	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>
cb	9	<3,2>
abb	10	<6,2>
cd	11	<3,4>

- ➢ Now we are at “e”.
- “e” has in the base dictionary
- So, we take “eb”.
- “eb” doesn’t have in the both dictionary.
- We add “eb” to output dictionary.
- The last codeword is 11.
- New codeword must be 12.

a	**b**	**d**	**e**	**c**	**a**	**c**	**b**	**a**	**b**	**b**	**c**	**d**	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>
cb	9	<3,2>
abb	10	<6,2>
cd	11	<3,4>
eb	12	<5,2>

➢ Now we are at “e”.

- “e” has in the base dictionary
- So, we take “ed”.
- “ed” doesn’t have in the both dictionary.
- We add “ed” to output dictionary.
- The last codeword is 12.
- New codeword must be 13.

a	b	d	e	c	a	c	b	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>
cb	9	<3,2>
abb	10	<6,2>
cd	11	<3,4>

eb	12	<5,2>
ed	13	<5,4>

➢ Now we are at “a”.

- “a” has in the base dictionary
- So, we take “ac”.
- “ac” doesn’t have in the both dictionary.
- We add “ac” to output dictionary.
- The last codeword is 13.
- New codeword must be 14.

a	b	d	e	c	a	c	b	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>
cb	9	<3,2>
abb	10	<6,2>
cd	11	<3,4>
eb	12	<5,2>
ed	13	<5,4>
ac	14	<1,3>

➢ Now we are at “b”.

- “b” has in the base dictionary
- So, we take “ba”.
- “ba” doesn’t have in the both dictionary.
- We add “ba” to output dictionary.
- The last codeword is 14.
- New codeword must be EOF.

a	b	d	e	c	a	c	b	a	b	b	c	d	e	b	e	d	a	c	b	a	EOF
start																					

Entry	Codeword
a	1
b	2
c	3

d	4
e	5

Entry	Codeword	Output
ab	6	<1,2>
de	7	<4,5>
ca	8	<3,1>
cb	9	<3,2>
abb	10	<6,2>
cd	11	<3,4>
eb	12	<5,2>
ed	13	<5,4>
ac	14	<1,3>
ba	EOF	<2,1>

✓ But, there is no next character (EOF)

Thus, Encoder = { <1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>,<2,1>}

❖ Decoder:

✓ First, use the base dictionary from encoder.
Encoder = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

Entry	Codeword
a	1
b	2
c	3
d	4
e	5

Base dictionary

✓ Next, build an output dictionary.

Input	Codeword	Entry	Output

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	<4,5>	<3,1>	<3,2>	<6,2>	<3,4>	<5,2>	<5,4>	<1,3>	<2,1>	EOF
Input										

➢ Start “<1,2>”.

- Codeword: “1” is “a” and “2” is “b”.
- Entry is “ab”. So, output is also “ab”.

- New codeword is 6.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	<4,5>	<3,1>	<3,2>	<6,2>	<3,4>	<5,2>	<5,4>	<1,3>	<2,1>	EOF
Input										

➢ Next “<4,5>”.

- Codeword: “4” is “d” and “5” is “e”.
- Entry is “de”. So, output is also “de”.
- New codeword is 7.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	<3,1>	<3,2>	<6,2>	<3,4>	<5,2>	<5,4>	<1,3>	<2,1>	EOF
Input										

➢ Next “<3,1>”.

- Codeword: “3” is “c” and “1” is “a”.
- Entry is “ca”. So, output is also “ca”.
- New codeword is 8.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de
<3,1>	8	ca	ca

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	**<3,1>**	<3,2>	<6,2>	<3,4>	<5,2>	<5,4>	<1,3>	<2,1>	EOF

Input										

➢ Next “<3,2>”.

- Codeword: “3” is “c” and “2” is “b”.
- Entry is “cb”. So, output is also “cb”.
- New codeword is 9.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de
<3,1>	8	ca	ca
<3,2>	9	cb	cb

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	**<3,1>**	**<3,2>**	<6,2>	<3,4>	<5,2>	<5,4>	<1,3>	<2,1>	EOF
Input										

➢ Next “<6,2>”.

- Codeword: “6” is “ab” and “2” is “b”.
- Entry is “abb”. So, output is also “abb”.
- New codeword is 10.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de
<3,1>	8	ca	ca
<3,2>	9	cb	cb
<6,2>	10	abb	abb

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	**<3,1>**	**<3,2>**	**<6,2>**	<3,4>	<5,2>	<5,4>	<1,3>	<2,1>	EOF
Input										

➢ Next “<3,4>”.

- Codeword: “3” is “c” and “4” is “d”.
- Entry is “cd”. So, output is also “cd”.
- New codeword is 11.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab

<4,5>	7	de	de
<3,1>	8	ca	ca
<3,2>	9	cb	cb
<6,2>	10	abb	abb
<3,4>	11	cd	cd

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	**<3,1>**	**<3,2>**	**<6,2>**	**<3,4>**	<5,2>	<5,4>	<1,3>	<2,1>	EOF
Input										

➢ Next "<5,2>".

- Codeword: "5" is "e" and "2" is "b".
- Entry is "eb". So, output is also "eb".
- New codeword is 12.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de
<3,1>	8	ca	ca
<3,2>	9	cb	cb
<6,2>	10	abb	abb
<3,4>	11	cd	cd
<5,2>	12	eb	eb

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	**<3,1>**	**<3,2>**	**<6,2>**	**<3,4>**	**<5,2>**	<5,4>	<1,3>	<2,1>	EOF
Input										

➢ Next "<5,4>".

- Codeword: "5" is "e" and "4" is "d".
- Entry is "ed". So, output is also "ed".
- New codeword is 13.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de
<3,1>	8	ca	ca
<3,2>	9	cb	cb
<6,2>	10	abb	abb

<3,4>	11	cd	cd
<5,2>	12	eb	eb
<5,4>	13	ed	ed

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	**<3,1>**	**<3,2>**	**<6,2>**	**<3,4>**	**<5,2>**	**<5,4>**	<1,3>	<2,1>	EOF
Input										

- ➢ Next “<1,3>”.
- Codeword: “1” is “a” and “3” is “c”.
- Entry is “ac”. So, output is also “ac”.
- New codeword is 14.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de
<3,1>	8	ca	ca
<3,2>	9	cb	cb
<6,2>	10	abb	abb
<3,4>	11	cd	cd
<5,2>	12	eb	eb
<5,4>	13	ed	ed
<1,3>	14	ac	ac

✓ Input string is the result of encoder.
string = {<1,2>, <4,5>, <3.1>, <3,2>, <6,2>, <3,4>, <5,2>, <5,4>, <1,3>, <2,1>, EOF}

<1,2>	**<4,5>**	**<3,1>**	**<3,2>**	**<6,2>**	**<3,4>**	**<5,2>**	**<5,4>**	**<1,3>**	<2,1>	EOF
Input										

- ➢ Next “<2,1>”.
- Codeword: “2” is “b” and “1” is “a”.
- Entry is “ba”. So, output is also “ba”.
- New codeword is 15.

Input	Codeword	Entry	Output
<1,2>	6	ab	ab
<4,5>	7	de	de
<3,1>	8	ca	ca
<3,2>	9	cb	cb
<6,2>	10	abb	abb
<3,4>	11	cd	cd
<5,2>	12	eb	eb

<5,4>	13	ed	ed
<1,3>	14	ac	ac
<2,1>	EOF	ba	ba

Thus, Decoder: abdecacbabbcdebedacba