WRITEUP.pdf

Githika Annapureddy

March 2023

1 Explain what you learned

In this lab, I learned how to create a compression and decompression algorithm. I understood how the algorithm works theoretically and how to make it into code. I also got some practice creating functions for a trie and for the word ADT. I also learned how to do my own buffer management through the io.c file. I also learned how to check for endianness and switch endianness using bit operators.

2 Discuss the functionality of LZ78 compression

LZ78 Compression's input is text. It outputs a compressed version of the text via pairs.
assigning each 'new word' in a input to a different code. The code starts at 2 and goes up to MAX CODE which is UINT16 MAX in our case. A word is considered 'new' if it has not been encountered before. For example, for the input "abcababc", the new words would be
'a'
'b'
'c'
'ab'
'abc'
The codes assigned to these words would be as follows
" EMPTY CODE
'a' START CODE (2)
'b' 3
'c' 4
'ab' 5
'abc' 6
0 STOP CODE
Once these pairs are created, they are outputted as follows...
(EMPTY CODE, a)
(EMPTY CODE, b)
(EMPTY CODE, c)

(2, b)
(5, c)
(STOP CODE, 0)

The pairs encode each word by having the code represent the part of the word that has been seen before, and then the part of the word that has not been seen before. Thus, parts of the word that have been seen before are represented by one number. As the part of the word that has been seen before grows larger, the efficiency of the compression improves because a larger amount of text can be represented by one number.

LZ78 Decompression's input is the pairs created by the compression algorithm. Its output is the original text that the compression algorithm compressed.
The decompression algorithm reads in the pairs.
it adds the second item to the previously seen word which is the first item. This is a word.
then it assigns the next available code to this word.
finally, the word is outputted.
If the algorithm sees a code, such as for '(2, b)', it looks at what word it assigned to 2. This would be 'a' since 'a' was the first word it encountered and 2 is the first available code.
Thus the word for '(2, b)' would be 'ab'.
This process is repeated until all the pairs in the file have been decompressed.

3 Demonstrate how the efficiency of your compression changes with entropy. Explain the relation.

As an input has more entropy, it has more randomness. The input has fewer common characters. With greater entropy, the efficiency of the compression algorithm decreases. This is because the algorithm relies on having 'previously seen words' repeat. If they repeat, they can be replaced by a code (which is compression since the code is just one number). But, if there is more entropy, there are less repeating characters, meaning less words have been seen before and can be replaced with a code.