Reading Encoded Text

KEYTON ROGERS

Historical Context

- Bits were expensive in the past. Therefore, only 7 bits were used to encode characters.
- The leading bit is left as 0, and the rest encode the character.
- As a result, only the English alphabet was encoded to begin with.
- These are the standard 128 ASCII characters (including non-printing characters).
- What about other languages?

Different Encodings

- Other languages and some symbols needed to be represented in characters as well.
- This encouraged the development of the UTF series of encoding standards.
- ASCII only used 7 bits, so UTF-8 used all 8.
- This was accomplished by changing the leading bit from 0 to 1 on UTF-8 characters.
- The ISO encoding standard is also used in some documents.
- ISO is like an international ASCII. The leading 0 bit remains, but the 127 characters were allocated to different characters in each language.
- Note that UTF-8 can handle standard ASCII characters, but not vice-versa. Also, UTF-8 is not compatible with ISO, and vice-versa.
- Multiple versions of UTF exist (UTF-16, UTF-24, UTF-32), but these standards are rarer.

Implications on Web Scraping

- Obviously, the internet has documents in multiple languages.
- What if you have to find a term in an international government’s website to find data?
- Odds are high that some if not most international webpages are encoded in some standard other than ASCII.
- The vast majority of webpages are encoded in UTF-8.
- As a result, web scrapers must be built to handle this if necessary.
- Note that Python 3.x treats all documents to be in UTF-8 by default, but this does not cover ISO.