What is Character Encoding Auto-Detection?

WE'LL COVER THE FOLLOWING

It means taking a sequence of bytes in an unknown character encoding, and attempting to determine the encoding so you can read the text. It's like cracking a code when you don't have the decryption key.

Isn't That Impossible?

In general, yes. However, some encodings are optimized for specific languages, and languages are not random. Some character sequences pop up all the time, while other sequences make no sense. A person fluent in English who opens a newspaper and finds "txzqJv 2!dasd0a QqdKjvz" will instantly recognize that that isn't English (even though it is composed entirely of English letters). By studying lots of "typical" text, a computer algorithm can simulate this kind of fluency and make an educated guess about a text's language.

In other words, encoding detection is really language detection, combined with knowledge of which languages tend to use which character encodings.

Does Such An Algorithm Exist?

As it turns out, yes. All major browsers have character encoding auto-detection, because the web is full of pages that have no encoding information whatsoever. Mozilla Firefox contains an encoding auto-detection library which is open source. I ported the library to Python 2 and dubbed it the chardet module. This chapter will take you step-by-step through the process of porting the `chardet` module from Python 2 to Python 3.