Prepping data: Tokenization

What does it do? Something so simple that it's hard to think of it as a transformation at all: breaks a text into an ordered list of words. "But isn't a text *already* an ordered list of words?" Unfortunately, not in any sense a computer can understand.

What's so great about it.

Nothing. It's just hard choices someone will criticize down the road. But it's a *necessary* set of choices to make other things possible.

Are there alternative implementations? Lots. For English, the Penn Treebank tokenization is somewhat standard; that does things like break the word "can't" into the two tokens "ca" and "n't". The big character set actors, Unicode, define their own word boundaries. Or you can just use any sequence of characters in a regular expression, which is often the easiest. That regular expression is `[A-Za-z]+` in plain English text, `\p{L}+` in Unicode.