Domain

In addition to having characteristic ways of speaking, characters have characteristic attributes (Bender is a robot), perform characteristic actions (Bender steals a cigar) and have characteristic things done to them (Bender is caught by the police). Abstracting away from a single story, collections of actions and attributes are associated with different archetypes, each of which has its own role to play in the story (the hero saves; the villain attacks; the trickster tempts, and so on).

My goal for this project is to use natural language data from plot summaries and character bios to cluster characters into archetypes and to uncover what the most common actions and attributes of these archetypes are.

One way to approach this might be to perform LDA on a large number of TV character biographies extracted from Wikipedia, within which most of the text can be assumed to be about the character. I'd interpret each character to be a "document" and each archetype to be a "topic". For example, I might find that Bender is 80% archetype A and 20% archetype B, which I might label "trickster" and "sidekick" respectively, depending on the other characters associated with those archetypes their associated word probabilities. (This approach was inspired by this paper.)

Another approach might be to train an auto-encoder on the biographies, remove the decoder, and then cluster the characters based on that state vector using k-means, a self-organizing map, or other clustering algorithm.

Data

Character biographies are the most obvious source of data, and I propose to use as many of them as I can extract from the English language Wikipedia. These tend to be shorter than the bios on show specific Wikis, but appear to be long enough (Bender's is 4,329 words) and should be relatively easy to extract in bulk; they'll all be in the full site dump and most should be tagged with one or more sub-tags of the "Fictional character" tag.

Episode summaries are an alternative source of data, but these will need to be handled carefully because the text can't be assumed to be about any given character. With the large model loaded, `spaCy`'s parser seems to be good enough to reliably assign subjects to clauses, so it may be possible to group the sentences by their subjects and supplement the text from the character biographies. Parsing (or at least POS tags) might also be useful as a tool to filter text in character bios, or to in interpret the resulting clusters (e.g. the most frequent adjectives and actions associated with characters in cluster A).

Description	Type	Example
Character bios from Wikipedia, potentially filtered, likely with preprocessing.	Text	<ARTICLE SUBJECT> is a robot... <ARTICLE SUBJECT> was built ... <ARTICLE SUBJECT> killed <PROPER NOUN>... <ARTICLE SUBJECT> was caught by the police trying to steal...
Episode summaries from Wikipedia (maybe).	Text	

Known unknowns

- I suspect that without some level of filtering, clusters are going to want to form around genres or fictional universes rather than archetypes. This could be resisted by only including sentences where the character is the subject or object, and/or by replacing other proper names with dummy values.
- The number of clusters for to target.
- If there will any interesting clustering.
- How to judge the quality the clustering.