Latent Dirichlet Allocation

Implement the latent Dirichlet allocation (LDA) model to generate a corpus from a given set of parameters. Build a function `lda_gen()` that takes four arguments:

1. `vocabulary` - list (of length V) of strings
2. `alpha` - topic distribution parameter vector, numpy array of size $(\mathrm{k},)$
3. `beta` - topic-word matrix, numpy array of size (k, V)
4. `xi` - Poisson parameter (scalar) for document size distribution

and returns:

1. `words` - list of words (strings) in a document

Note that you should draw the document length from Poisson(xi) - you could use `np.random.poisson()`.

Use the provided script to generate a corpus of documents and apply LDA parameter inference with `gensim`'s solver. **Show the inferred beta vectors and indicate how they map to the true topics above.**

Expect it to be a little noisy - it you're not sure whether the results are reasonable, ask!

You should turn in a document (`.txt`, `.md`, or `.pdf`) answering all of the **red** items above. You should also turn in Python scripts (`.py`) for *each* of the **blue** items . Unless otherwise specified, you may use only `numpy` and the standard library (the test script uses gensim, but your `lda_gen()` should not).