```
    beta = np.array([
        [0.4, 0.4, 0.2, 0.0, 0.0, 0.0],
        [0.0, 0.3, 0.1, 0.0, 0.3, 0.3],
        [0.3, 0.0, 0.2, 0.3, 0.2, 0.0]
    ])
    alpha = np.array([0.2, 0.2, 0.2])
    xi = 50
    documents = [
        lda_gen(vocabulary, alpha, beta, xi)
        for _ in range(100)
    ]

    # Create a corpus from a list of texts
    dictionary = Dictionary(documents)
    corpus = [dictionary.doc2bow(text) for text in documents]
    model = LdaModel(
        corpus,
        id2word=dictionary,
        num_topics=3,
    )
    print(model.alpha)
    print(model.show_topics())


if __name__ == "__main__":
    test()
```

```
[0.33333334 0.33333334 0.33333334]
[(0, '0.330*"bass" + 0.228*"pike" + 0.201*"deep" + 0.104*"horn" + 0.082*"tuba" + 0.055*"catapult"'), (1, '0.349*"pike" + 0.2
33*"horn" + 0.202*"catapult" + 0.104*"deep" + 0.095*"bass" + 0.018*"tuba"'), (2, '0.348*"tuba" + 0.205*"bass" + 0.193*"horn"
+ 0.154*"deep" + 0.059*"pike" + 0.040*"catapult"')]
```

The word chosen from is: `"bass", "pike", "deep", "tuba", "horn", "catapult".`

From the first LdaModel output shown above, the beta vector maps generally well to the true topic, with some noise. The beta of bass is off by 0.07, the beta of pike is off by 0.178, the beta of deep is off by 0.001, the beta of horn is off by 0.104, the betas of tuba and catapult are close to 0.

The second beta vector maps well to the true topic as well. Beta of pike is the largest, which is true. Betas of catapult and deep are a bit off from 0.3. But the betas of bass and tuba are fine as they are close to 0.

The third beta vector maps the well to the true topic as well. Tuba is 0.048 off, bass is 0.1 off, horn, deep, pike, and catapult are only a bit off from their true beta values.

Overall, the beta vectors generated by the LDA model of gensim's library has some noise.