Discussion

Method

I used the simple frequency counting approach as suggested, since the human curated tag list yields very good results for such a specialized tasks and is well apt to handle clustering into the cluster of way off topics. No need to reinvent the weel, if a simple method does the job very well.

For stemming and lemmatization I used the sugested ParZu tool, since it was much better than easier to use libraries like PyStemmer. Since the documentation of ParZu was poor and the tool does not seem to be structured as a python library that can easily be imported, I implemented it as a shell call. Since ParZu is slow I implemented an easy MD5 cache to avoid recomputation. ParZu does a lot of things (POS tagging, morphology analysis) that are unnecessary for this task. Looking into the code of the library and using only the necessary parts via a python import could bring dramatic speedups, but was not possible in the timeframe of this case.

To improve the naive method slightly, I lowercased all words, and checked wether the tags were contained in the stems rather than identical with them. The latter helps to identify German's famous compound nouns. Simplifying the tags of verbs to be endingless helps to further identify compound nouns from verbs like 'Versendeladebalken'.

Evaluation

The naive method has an F measure of 85% (precision: 82%, recall: 88%). It's misses are mainly due to compound nouns ('Emailadresse', 'Emailprogramm', 'Versendeladebalken', 'Adminrechte'), especially in Cluster 1 (Email). As described above this can be easily fixed with some care in choosing tags that don't appear in random other words. Other mistakes are due to the fact that Cluster 2 (Cursor) contains no verbs, adding the verb 'klicken' helps to leverage this. The only remaining error occurs in the sentence 'Mein Computer macht sich selbständig.' which is about cursors but doesn't contain any specific keywords, and thus ends up in Cluster 4 (mildly off-topic). To capture such unpredictable inputs a simple Machine Learning technique like Naive Bayes classification might help, but a bigger training set would be needed.

After the suggested improvements the algorithm has a very good F value of 96% (precision: 95%, recall: 98%) on the given training set. The problem of the algorithm mainly lies in the precision of the off-topic classifications (in the naive algoritm at 75% and 33%). Since the algorithm is supposed to be used to generate user suggestions, maybe suggesting off-topic categories should be handled with care to avoid annoying users.