Improving Newsgroup Clustering by Filtering Author-Specific Words[*]

Yuval Marom and Ingrid Zukerman
School of Computer Science and Software Engineering
Monash University, Clayton, Victoria 3800, AUSTRALIA
{yuvalm,ingrid}@csse.monash.edu.au

Introduction. This paper describes the first step in a project for topic identification in help-desk applications. In this step, we apply a clustering mechanism to identify the topics of newsgroup discussions. We have used newsgroup discussions as our testbed, as they provide a good approximation to our target application, while obviating the need for manual tagging of topics.

We have found that the postings of individuals who contribute repeatedly to a newsgroup may lead the clustering process astray, in the sense that discussions may be grouped according to their author, rather than according to their topic. To address this problem, we introduce a filtering mechanism, and evaluate it by comparing clustering performance with and without filtering.

The Filtering Mechanism. Our filtering mechanism operates in two stages. First, a 'profile' is built for each person posting to a newsgroup. This profile is a distribution of *word document frequencies*, where the document frequency of a word is the number of postings where the word is used. Next, word-usage proportions are calculated for each person. These are the word document frequencies divided by the person's total number of postings. We then filter out words that (1) have a high usage proportion, and (2) are posted by frequent contributors. For more details, see [1].

Clustering Newsgroups. We use the K-Means algorithm for clustering. This algorithm separates a dataset into k clusters based on the Euclidean distance between data points, where each data 'point' corresponds to one document (newsgroup thread). The output of the clustering process is evaluated by calculating the F-score for each cluster, and the combined F-score for all the clusters (the F-score measure reflects how many documents a cluster and a newsgroup have in common [2]). Our data representation consists of a bag-of-words with TF.IDF scoring [2]: a word-vector is made up from a chosen and fixed set of words; the vector components are determined based on how frequently each word appears in a document and how infrequently it occurs in other documents. For more details, see [1].

In order to determine the useful range of applicability of our filtering mechanism, we have evaluated clustering (and filtering) performance along the dimension of topical similarity between newsgroups. That is, we vary the level of relatedness between the newsgroups in our datasets. The least related newsgroups provide a benchmark for clustering performance, while the more related ones exemplify help-desk applications.

Results. Figure 1 shows the results obtained for three datasets with different values of k. The newsgroups in the first dataset were downloaded from the Internet. They

[*] This research was supported in part by Linkage Grant LP0347470 from the Australian Research Council and by an endowment from Hewlett Packard.

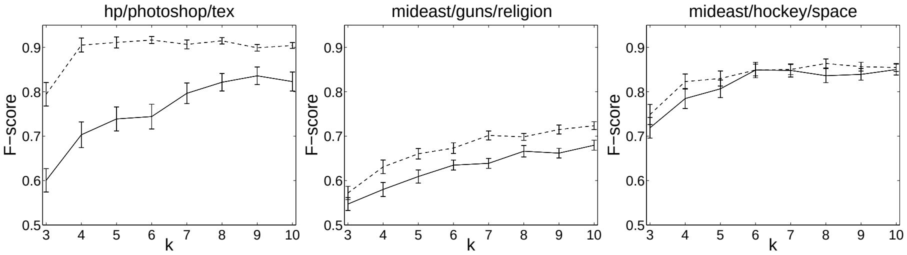

Fig. 1. Overall results for the three datasets.

are `lp.hp` (related to printing), `comp.graphics.apps.photoshop` (related to graphics), and `comp.text.tex` (related to text editing). These newsgroups are computing-related, but discuss fairly different topics. We see that for this dataset, performance is much poorer without filtering, particularly for low values of k. This suggests that author-specific words create undesirable overlaps between the clusters, which are resolved as the value of k increases because finer differences between the clusters are detected. In contrast, when filtering is used, the clustering procedure reaches its best performance with $k = 4$, where the performance is extremely good. The fact that it converges for such a low value of k suggests that there is little ‘true’ topical overlap between the newsgroups.

The second and third datasets were obtained from the “20-newsgroups” corpus (`http://people.csail.mit.edu/people/jrennie/20Newsgroups`). The second set consists of the newsgroups `talk.politics.mideast`, `talk.politics.guns`, and `talk.religion.misc`. These newsgroups discuss fairly similar topics, related to politics and religion. Because there is a large topical overlap between the newsgroups, clustering performance for this dataset is overall much poorer than for the first (and the third) dataset. As for the first dataset, the performance steadily improves as k increases, both with and without filtering. Notice also that filtering consistently improves clustering performance, which means that there are also undesirable overlaps created by author-specific words.

The third dataset is made up of the newsgroup `talk.politics.mideast`, which was also used in the second dataset, as well as `rec.sport.hockey` and `sci.space`. These newsgroups discuss very different topics, which explains why filtering has the least effect on this dataset: the documents are different enough for the clustering to perform similarly with and without filtering. That is, there are enough discriminating topical words to diminish the effect of author-specific words. Nonetheless, filtering has an effect for lower values of k, suggesting that some overlap is created by author-specific words — when enough clusters are used to account for this overlap ($k = 6$), the effect of the filtering mechanism disappears.

Conclusion. Newsgroup clustering generally benefits from a filtering mechanism that removes subjective influences of frequent contributors. The magnitude of this effect depends on the topical similarity between the newsgroups involved, and the level of granularity used in the clustering (*i.e.* the value of k).

References

1. Zukerman, I., Marom, Y.: Filtering speaker-specific words from electronic discussions. In: Proceedings of The 20th International Conference on Computational Linguistics. (2004)
2. Salton, G., McGill, M.: An Introduction to Modern Information Retrieval. McGraw Hill (1983)