# Data mining scenarios for the discovery of subtypes and the comparison of algorithms

Colas, F.P.R.

Appendix B

# Additional Results in Text Classification

For the nearest neighbors algorithm, a number of feature space transformations is possible. The $k$ nearest neighbors classifier implemented in the `libbow` library [McC96] defines these transformations by two sets of three letters; for a detailed description of the letters, see the Table B.1. In Figures B.1, B.2 and B.3 we report histograms of the count of pairwise wins for each combination of the feature space transformations.

We remark that binary transformations (`b__`) tend to perform worse. As well, the inverse document frequency (`_t_`) does not show as crucial as we could expect given its wide use in information retrieval. Further, normalizing the scores (`__c`) did not show any improvement. The rest of the transformations seem to perform equally well to the exception of the `_tc`-transformation. Then, as a `_tc`-transformation are applied on the training set, we observe generally underperforming nearest neighbor classifiers; a possible explanation would be a software-issue while normalizing the scores of the training set. In our analyses, we avoided this type of transformations.

**Table B.1:** *The feature space transformations are defined in the* `libbow` *library by combinations of three letters that refer to the term frequency, the inverse document frequency and the normalization. Recall that $x_{ij}$ is the frequency of the word $j$ in the document $i$. This Table summarizes the different combinations.*

| | | | |
|---|---|---|---|
| | | **Term frequency (*tf*)** | |
| **n** | *none* | Raw frequencies | $tf(x_{ij}) = x_{ij}$ |
| **b** | *binary* | Binarize the raw frequencies | $tf(x_{ij}) = \begin{cases} 1 & \text{if } tf(x_{ij}) \geq 1 \\ 0 & \text{otherwise} \end{cases}$ |
| **m** | *max-norm* | Normalize $x_{ij}$ relatively to the maximum term frequency observed in a document $i$ | $tf(x_{ij}) = \frac{x_{ij}}{max_j x_{ij}}$ |
| **a** | *augmented norm* | Similar to the *max-norm* but with $\frac{1}{2}$ added | $tf(x_{ij}) = \frac{1}{2} + \frac{x_{ij}}{2max_i x_{ij}}$ |
| **l** | *log* | Logarithm of the term frequency | $tf(x_{ij}) = 1 + log(x_{ij})$ |
| | | **Inverse document frequency (*idf*)** | |
| **n** | *none* | *idf* is not used | $idf(x_{ij}) = 1$ |
| **t** | *idf* | Inverse of the frequency of the term $x_{ij}$ in the database which has $N$ documents | $idf(x_{ij}) = log\left(\frac{N}{df(x_{ij})}\right)$ |
| | | **Normalization** | |
| **n** | *none* | Normalization is not used | $\phi(x_{ij}) = tf(x_{ij})idf(x_{ij})$ |
| **c** | *cosine* | Apply a cosine normalization | $\phi(x_{ij}) = \sqrt{\frac{tf(x_{ij})idf(x_{ij})}{\Sigma_j(tf(x_{ij})idf(x_{ij}))^2}}$ |

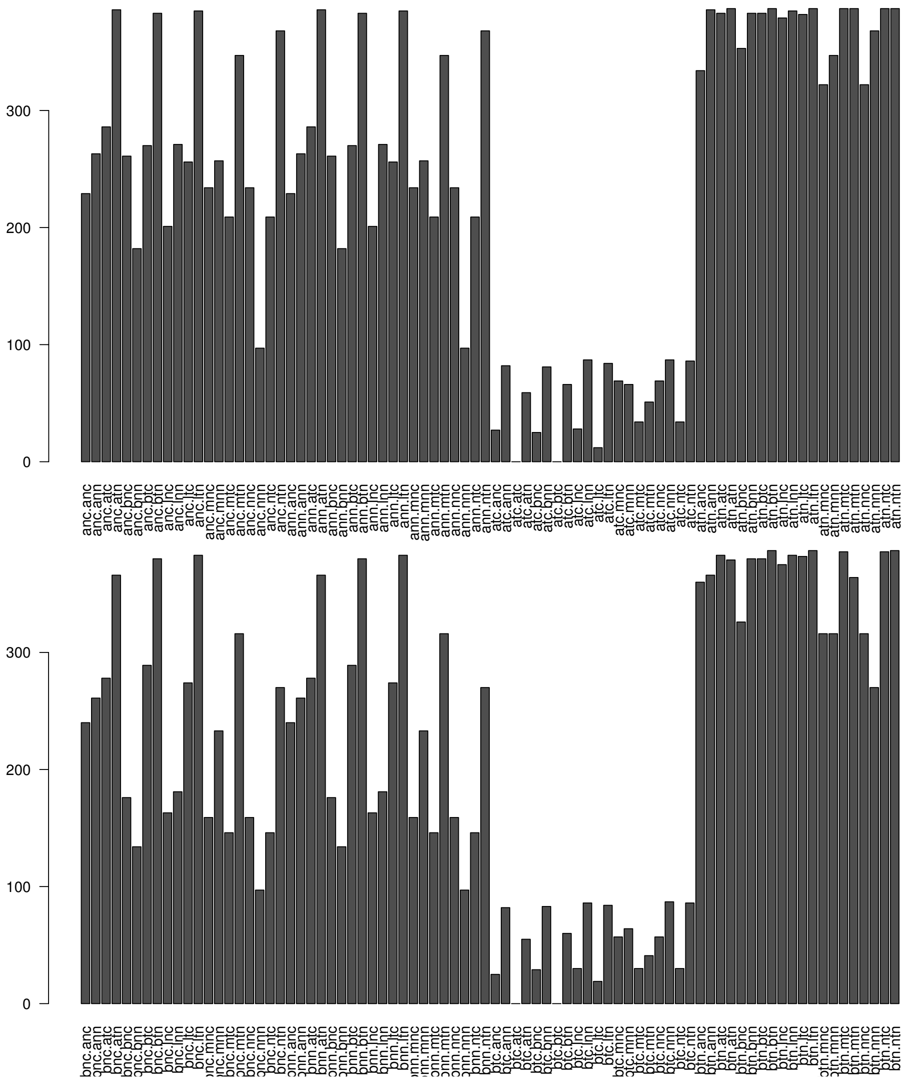


**Figure B.1:** *Counts of pairwise wins for each transformation, from* `ann.anc` *to* `btn.ntn`*.*

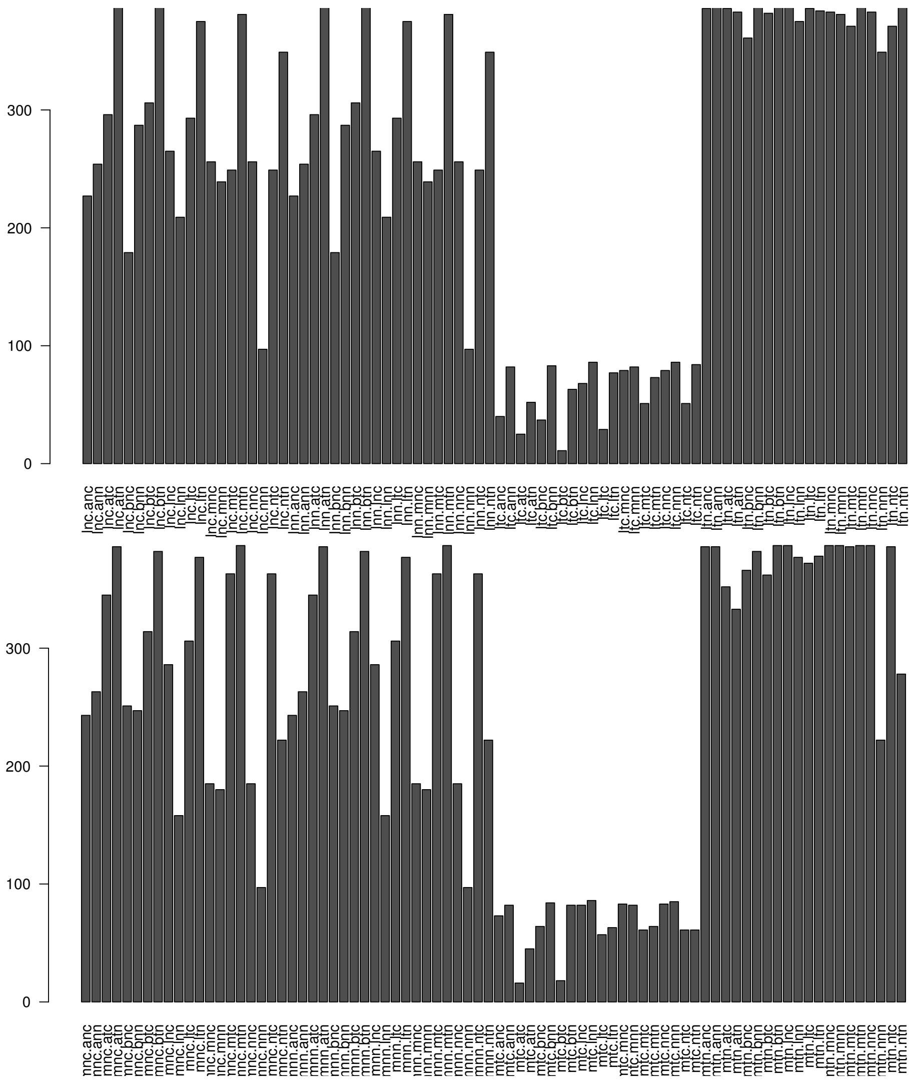


**Figure B.2:** *Counts of pairwise wins for each transformation, from* `lnc.anc` *to* `mtn.ntn`*.*

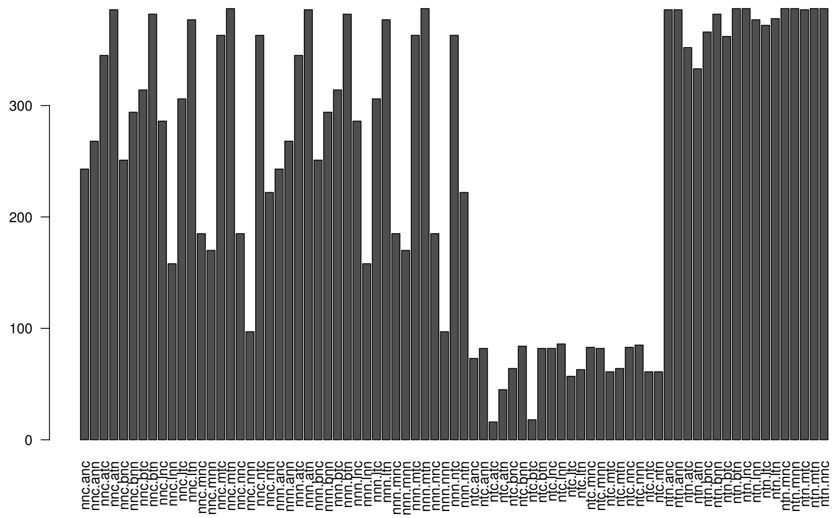


**Figure B.3:** *Counts of pairwise wins for each transformation, from* **`nnc.anc`** *to* `ntn.ntn`.