Bag of words In this implementation, text documents are modeled as a collection of unordered words. We count how often each word appears in each document and store the word counts into a matrix, where each row of the matrix reprensents one document. Each column of the matrix reprensents a word from the document dictionary. Suppose we represent the set of n d documents using a matrix of word counts like this:

$$D_{1:n_d} = \begin{pmatrix} 2 & 6 & ... & 4 \\ 2 & 4 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \end{pmatrix}$$

This means that word W_1 occurs twice in document D_1. Word W_{n_w} occurs 4 times in document D_1 and not at all in document D_2.

Multinomial distribution The simplest distribution representing a text document is multinomial distribution. The probability of a document D_i is:

$$p(D_i) = \prod_{j=1}^{n_w} \mu_j^{T_{ij}}$$

Here, $_j$ denotes the probability of a particular word in the text being equal to w_j , T_ij is the count of the word in document. So the probability of document D_1 would be $p(D_1) = \mu_1^2 \cdot \mu_2^6 \cdot ... \cdot \mu_{n_w}^4 \cdot$

Mixture of Multinomial Distributions In order to do text clustering, we want to use a mixture of multinomial distributions, so that each topic has a particular multinomial distribution associated with it, and each document is a mixture of different topics. We define $p(c) = \pi_c$ as the mixture coefficient of a document containing topic c, and each topic is modeled by a multinomial distribution $p(D_i|c)$ with parameters μ_{jc} , then we can write each document as a mixture over topics as

$$p(D_i) = \sum_{c=1}^{n_c} p(D_i|c)p(c) = \sum_{c=1}^{n_c} \pi_c \prod_{j=1}^{n_w} \mu_{jc}^{T_{ij}}$$

EM for Mixture of Multinomials In order to cluster a set of documents, we need to fit this mixture model to data. In this problem, the EM algorithm can be used for fitting mixture models. This will be a simple topic model for documents. Each topic is a multinomial distribution over words (a mixture component). EM algorithm for such a topic model, which consists of iterating the following steps:

1. Expectation
   Compute the expectation of document D_i belonging to cluster c:

$$\gamma = \frac{\pi_c \prod_{j=1}^{n_w} \mu_{jc}^{T_{ij}}}{\sum_{c=1}^{n_d} \pi_c \prod_{j=1}^{n_w} \mu_{jc}^{T_{ij}}}$$

2. Maximization
   Update the mixture parameters, i.e. the probability of a word being W_j in cluster (topic) c, as well as prior probability of each cluster.

$$\mu_{jc} = \frac{\sum_{i=1}^{n_d} \gamma_{ic} T_{ij}}{\sum_{i=1}^{n_d} \sum_{l=1}^{m_w} \gamma_{ic} T_{il}}$$

$$\pi_c = \frac{1}{n_d} \sum_{i=1}^{n_d} \gamma_{ic}$$