A textual statistical analysis of SEFI 2014 documents

Pantelis Z. Hadjipantelis - EDG, UK
Sponsor: Joachim Schlosser - Education Marketing, DE

September 2014

We aim to provide a coherent statistical analysis of the SEFI (European Society for Engineering Education) 2014 paper submissions. The sample available consists of 133 papers; these papers are provided in .pdf, .doc(x) and .txt formats. We were also provided with an XML document having the papers' title, author(s) and abstract. Analysis is conducted in MATLAB utilizing readily available functionality. The final deliverables are:

- Correlation matrix between the most prominent document terms
- 3-D scatter and surface plot visualizing the main conceptual axes of the conference papers
- Basic hierarchical clustering of the document terms used

We currently utilize the information provided by the XML document only; as such we analyse *only* the papers' abstracts. The basic theoretical background behind the current analysis can be found in : [1, 2].
Preprocessing and dataset generation: Given the sample of documents we construct a bag-of-words to represent our data. Each word is considered an *attribute* and each document a *record* [3]. Basic preprocessing included : 1. Case-folding correction (eg. Student $\rightarrow$ student), 2. Stop-word removal (eg. *do*, *can*, etc.) and 3. Singular/plural reconstruction (eg. students $\rightarrow$ student); this 3rd step can be seen as an extremely basic *stemming* procedure [3]. In a way similar to [4] we construct a weighted term-frequency vector to represent each document. Assuming that $W = \{w_1, w_2, \ldots, w_m\}$ to be the complete vocabulary of our corpus we use an TF-IDF (Term Frequency - Inverse Document Frequency) technique to reweigh our data in terms of how important a word is to a document in the current corpus at hand. We also normalize each vector to have unit magnitude. We did not utilize the complete vocabulary of our corpus; aiming to reduce the dimensions of our dataset and based on the approach used in [5] we utilized the first 30 most frequent words (after preprocessing) in our corpus to generate our final matrix X.

$$
\begin{aligned}
X_i &= [x_{1i}, x_{2i}, \ldots, x_{mi}]^T \\
x_{ji} &= t_{ji} \log\left(\frac{n}{idf_j}\right) \\
X_i &= X_i / ||X_i||
\end{aligned}
$$

where t_{ji} is the term frequency of the word w_j in document d_i, n is the total number of documents analyzed and idf_i the number of documents containing the word w_j. Counting, string comparisons and norm normalization can be easily implemented in MATLAB using functions such as `numel`, `strcmp` and `norm`.
Dataset factorizations and document clustering: The two approaches we explore are Latent Semantic Analysis [2] (a Singular Value Decomposition (SVD) method) and document clustering based on Non-negative Matrix Factorization (NNMF) [4]. Both methods essentially project a document into a subspace defined by the basis provided by the matrix factorization technique they employ. One then uses standard clustering algorithms on the projected data to find meaningful clusters. The basic difference between the two algorithms is that SVD is not guaranteed to provide positive modes of variation while NNMF does. In that way one avoids the conceptually incoherent concept of negative document values. On the other hand the basis presented by NNMF are not guaranteed to be orthogonal, making the overall dimension reduction less efficient

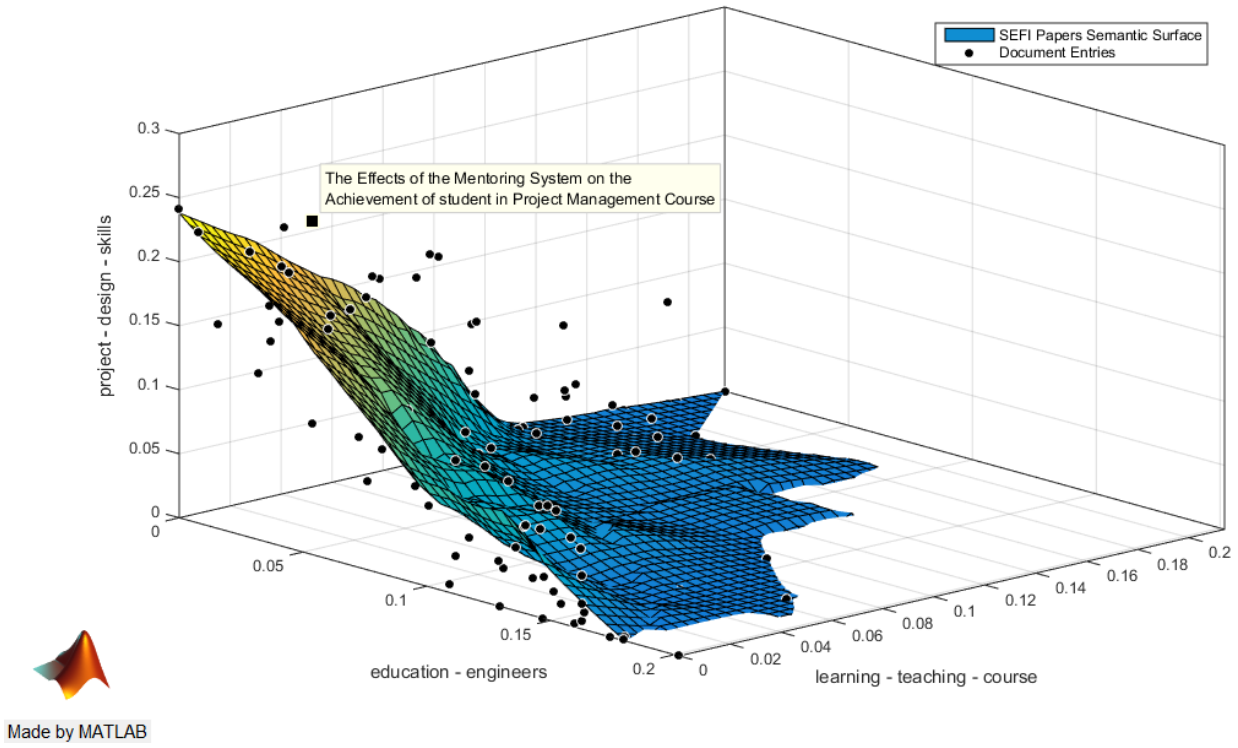

Figure 1: The semantic surface defined by the NNMF projected documents.

in terms of RSS. In short given a matrix $A_{m\times n}$ in the case of SVD one can produce a factorization:

$$A = U\Sigma V^T$$

such that U is an $m \times m$ unitary matrix, Σ is an $m \times n$ diagonal matrix of non-negative values and V an $n \times n$ unitary matrix. In the case of NNMF one can produce a factorization:

$$A = WH$$

such that both W and H are non-negative matrices of dimensions $m \times k$ and $k \times n$ respectively minimizing the functional:

$$f(W,H) = \frac{1}{2}||A - WH||_f^2$$

Computation of both the SVD as well as NNMF procedure are directly available in MATLAB using the functions such as: `svd` and `nnmf` respectively. Clustering the documents explores two approaches, a flat clustering approach and a hierarchical clustering approach. The flat clustering approach does not assume any explicit structure relating the cluster; for this we are going to use a k-means algorithm. The hierarchical clustering approach implements a hierarchical agglomerative clustering (HAC) methodology that starting from single element clusters merges them together until all elements are part of a single cluster [1]. In both clustering approaches we used a cosine distance metric as it is the most-widely used as well as the most theoretically coherent to the transformation we have conducted [2, 6, 7]. Regarding HAC we used a UPGMA algorithm to compute distances [6, 5]. Computation of k-means clustering as well as HAC using a cosine distance is immediately available in MATLAB using the functions `kmeans` and `linkage` respectively.

Final deliverables: Aside the correlation and hierarchical clustering matrices between the keywords we provide an interactive figure where clicking at the data-points returns the document title they are associated with (eg. Fig. 1). All references are available in the author's public folder.

References

[1] Manning CD, Raghavan P, Schütze H. Introduction to information retrieval. Cambridge university press Cambridge; 2008. Chapt. 16 & 17.

[2] Landauer TK, Foltz PW, Laham D. An introduction to latent semantic analysis. Discourse processes. 1998;25(2-3):259–284.

[3] Larocca Neto J, Santos AD, Kaestner CAA, Freitas AA. Document Clustering and Text Summarization. In: Proceedings of the 4th International Conference Practical Applications of Knowledge Discovery and Data Mining (PADD-2000); 2000. .

[4] Xu W, Liu X, Gong Y. Document clustering based on non-negative matrix factorization. In: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM; 2003. p. 267–273.

[5] Fung BC, Wang K, Ester M. Hierarchical document clustering using frequent itemsets. In: SDM. vol. 3. SIAM; 2003. p. 59–70.

[6] Steinbach M, Karypis G, Kumar V, et al. A comparison of document clustering techniques. In: KDD workshop on text mining. vol. 400. Boston; 2000. p. 525–526.

[7] Shahnaz F, Berry MW, Pauca VP, Plemmons RJ. Document clustering using nonnegative matrix factorization. Information Processing & Management. 2006;42(2):373–386.