

Bio Cloud - A Method to Generate Sensible Word Clouds for Biological Patterns

Ayush Pareek

Student, Dept of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, India

Abstract: Text mining works widely in the field of research techniques, which allows an individual to store text and its important terms in form of electronic document (.doc, .txt). It is difficult to remember such huge amount of text; moreover the manual approach is more time taking, unreliable and accessible to that person only. Text mining techniques optimize this approach by extracting and storing this data. Computational comparison, file read, file write are done more efficiently. With the help of Bio-Cloud, we generated more semantically similar, related and significant patterns. The give, generate and get sequence modeling is adopted. Over the other available web applications, we present our application with improved stemming, relation and average case consideration. This approach do not limit the displayed number of words as all the generated sets can be traversed with the GUI, with opted size of patterns. This method is highly applicable in bioinformatics, related information retrieval from document, sentimental analysis using social websites (Twitter and Facebook), query expansion (Google) and many more.

Keywords: Word cloud, biological pattern analysis, bioinformatics, text mining.

I. INTRODUCTION

Paradigms of text mining are worthwhile in retrieving text, especially in case of enormous data analysis. They crimp a large portion of data and the relevant result is scrutinized easily [1]. The text mining applications allows more intelligible visual assessment. "Text mining" is cited as study of patterns in a given text [2]. In 1999 Hearst said that although people were familiar with text mining and this was a frequently searched phrase on popular search engine at that time but only few were working in this domain [2]. It is also said that text mining has been emerged from machine learning and statistics. Carving out niche of text and finding out the relevant similarity is the main cause of concern. The extraction of the cached similarities depends on the approach [3]. A better approach will end up with a quick and accurate result. The tag cloud in this scenario gives a quick summarization and a better way of visualization [4]. Color, size and orientation of words are the concerned attributes of a tag cloud [5].

Bioinformatics largely utilizes the domain of text mining with tag cloud representation for its ability to summarize enormous amount of biologist's data [6]. Gene2WordCloud[6] and Wordle[7] are two web applications which generates the most occurring keywords and construct a word cloud out of it. The former is superior in case of bioinformatics study due to its more intellectual approach. Wordle visualizes the most occurring words form a given text or a URL. It satisfies the color, orientation and compactness aspects in its tag clouds. However, it does not entertain the case sensitivity and the stemming relations. Thus, the repetition of words occurs irrespective of their same biological meaning. Gene2WordCloud hit a purple patch by overcoming this

issue and the term's frequency become more precise. Representing all the words in a single tag cloud mere signifies the number of instances of each word in the whole text. It does not furnish any idea about the relation of two occurring words. It does not justify with the average frequency terms either. As a matter of fact, frequently occurring words are commonly known in an analysis rather than the average occurring words. Hence more emphasis should be given to the latter for an efficient analysis.

In this paper we demonstrate the methods of extraction that are incorporated in Bio Cloud, along with the relevant tag clouds that displays Pubmed benchmark data set. These tag clouds appropriately relate the bottleneck biological terms; this makes Bio Cloud advantageous in field of bioinformatics. The approach in Bio Cloud is comprised of basic preprocessing in text mining such as stemming algorithm, stop word removal, frequency cut off on the provided free text. The user interface is developed using the PHP and the CSS web techniques. PHP is a server-side scripting language designed for web development and HTML is the standard markup language used to create web pages. This takes the input query from the user and generates certain number of patterns of given pattern length. Moreover, the generated patterns can be easily traversed using this user interface. The following discussion explains its approach.

II. PROPOSED APPROACH

The approach in Bio Cloud is comprised of basic file operations, stemming algorithm, searching and sorting techniques and some matrix manipulations on the provided

free text. Later, the PHP and the CSS techniques fetch the processed data set and represent it as a tag cloud accordingly with a GUI. The following discussion explains its approach.

A. Preprocessing

This segment refines the free text to its maximum extent and avoids discrepancies for succeeding segments. Figure 1 shows the main preprocessing steps. We took Pubmed benchmark data set as our free text for this purpose.

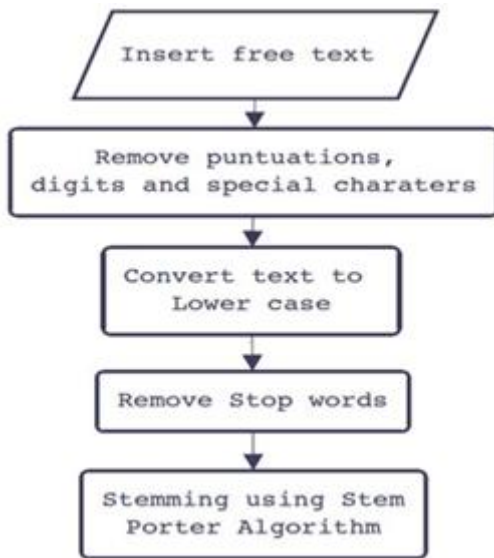


Fig.1. Preprocessing

The steps involve character by character traversal. Only lower case alphabetic values were left after the completion of these steps. This also ensures the confinement of the lower case domain. Stop words (a, an, the, etc) are removed. Stop words generally create a mess and need to be filtered out. Lastly, Stemming was done to improvise the given text. This eliminated the difference between root word and its derived words.

B. Method

After generating unique words in preprocessing part, we followed the under mentioned method (Figure 2) for the final result.

For this, matrix manipulation is performed on moderately occurring unique words and a mathematical relation is deduced (Equation 2). Based on PubMed benchmark data set we plotted frequency versus words graph (Figure 3) and drew the most relevant slope. By extrapolating we got the average frequency range (i.e. 960-25) and the corresponding average case unique words.

Now, for m such that M is a set of all the given documents and n such that N is a set of all average case unique words, we generated a frequency matrix F of size m×n. The element, f(i,j) represents weighting factor frequency of jth word in ith document for 0<i≤m and 0<j≤n. With this

frequency matrix F we calculated tf-idf(i,j) (Equation 1) [9] where tf-idf (term frequency-inverse document frequency, weighting scheme) is a matrix of same size m×n.

$$tf - idf(i, j) = f(i, j) \times \left\{ \log \left(\frac{m'}{|1+d|} \right) \right\} \quad (1)$$

where, m' is total number of given documents in dataset and d is number of documents which actually contain the jth word i.e. the count of documents having frequency of that word, greater than zero. Denominator in Equation 1 is adjusted to 1+|d| to avoid division by zero. We named this if-idf matrix as W-matrix such that W=tf-idf and w(i,j) for 0<i≤m and 0<j≤n.

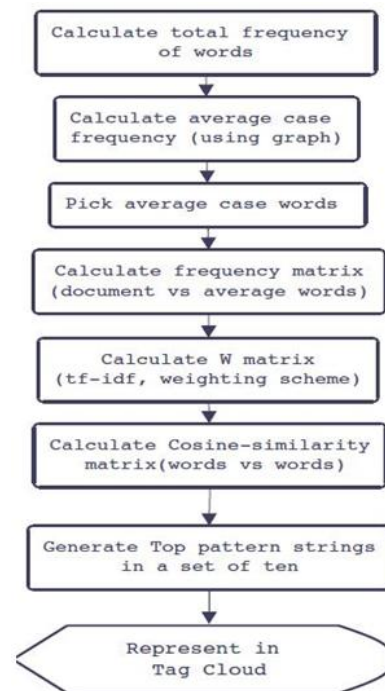


Fig.2. Algorithm

Lastly, we calculated Cosine Similarity matrix c(i,j) (Equation 2) for n words. Each c(i,j) represents cosine relation [10], between ith and jth column vectors of W-matrix hence it acquired a dimension n×n. Let 0<k≤m, we took two column vectors say, w(k,i) and w(k,j) of W-matrix and computed c(i,j) traversing all rows of same matrix with k=1 to k=m.

$$c(i, j) = \frac{\sum_{k=1}^m w(k, i) \times w(k, j)}{\sqrt{\sum_{k=1}^m (w(k, i))^2} \times \sqrt{\sum_{k=1}^m (w(k, j))^2}} \quad (2)$$

Then we combined a pairs of two c(i,j) having maximum cosine similarity value and created string patterns (maximum ten words per sting) (Algorithm 1). Finally PHP and CSS techniques were used for the generation of

tag cloud. The largest and centrally located word is the first entry of the top pattern string, following which in spiral way are the succeeding nine words. The algorithm is written below.


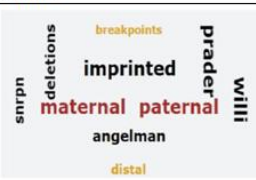
C. Algorithm 1: To create patterns from the Cosine Similarity matrix C.






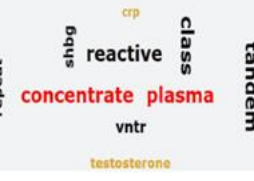

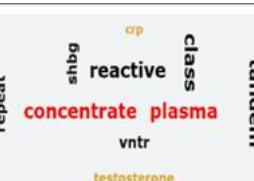

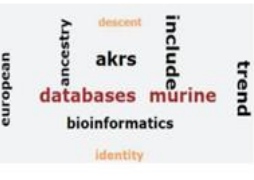
Input : Training Data Output: Cosine Similarity Matrix C, Word string set P
(maximum 10 words per string)
<ol style="list-style-type: none"> 1. Let S such that S be a set of unique words in PubMed training dataset and D such that D is a the set of all given documents 2. Create frequency matrix F 3. Do for all 4. Do for each $s_j \in S$ s.t. $j=1,2,\dots,s$ 5. Do for each $d_i \in D$ s.t. $i=1,2,\dots,d$ 6. $w_{ij} = \text{Cal_tf-idf}(F,i,j)$ 7. EndDo 8. EndDo 9. EndDo 10. Do for all 11. Do for each $s_j \in S$ s.t. $j=1,2,\dots,s-1$ 12. Do for each $s_k \in S$ s.t. $k=j+1,\dots,s$ 13. Do for each $d_i \in D$ s.t. $i=1,2,\dots,d$ 14. $c_{ij} = \text{Cal_Cosine_Sim}(W,i,j,k)$ 15. EndDo 16. $\text{max_similar} = \text{Find_Max_Cosine_Sim}()$ 17. If ($\text{stringcount} < 10$) Then 18. add s_j and s_k to p_z 19. increment stringcount by 1 20. Else 21. increment z by 1 for next string 22. EndIf 23. EndDo 24. EndDo 25. EndDo

III. EXPERIMENTAL ANALYSIS AND COMPARISONS

Based on the proposed approach of Section 2, we analyzed the text and used the average case words in the succeeding method. We created the tag cloud from the final string patterns; we analyzed our result with respect to some of the familiar online word cloud generators. Table 1 shows some Bio Cloud's tag clouds along with the comparisons of other tag clouds generator and there reference links.

TABLE I COMPARISON LIST

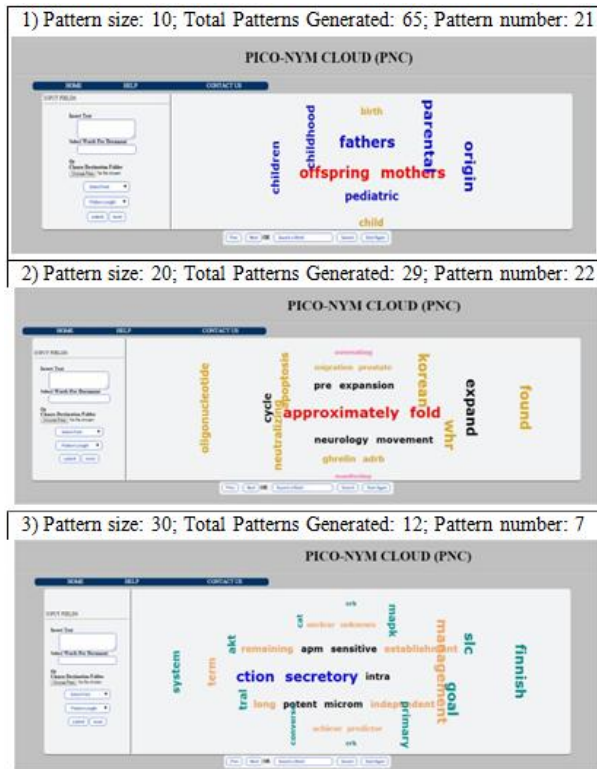
Popular Online World Cloud Generator	Tag cloud generated by Bio Cloud
 <p>1) http://tagcrowd.com/, Created By Daniel Steinback</p>	

 <p>2) http://wordsift.com/, Greg Wienties production, Stanford University © 2010</p>	
 <p>3) http://worditout.com/ © 2015 Enidea</p>	
 <p>4) http://www.jasondavies.com/wordcloud/ Copyright © Jason Davies 2014</p>	
 <p>5) http://www.wordle.net/ © 2013 Jonathan Feinberg</p>	
 <p>6) www.macvanlab.net/G2W/, Gene2WordCloud by the Ma'ayan Laboratory</p>	

The above results were generated on some portion of text (~65K words) Table 2 shows some more tag clouds created by Bio Cloud. These tag clouds are free from any punctuation, digit, special character, uppercase alphabets and general stop words. The cloud generation is possible with ten, twenty, thirty and more number of pattern lengths.

In 10-30 words patterns, it is easy to analyze rather than a cluster of words where one have carve out the niche of more than hundred words. The traversal though GUI is a user friendly approach along with display of related attributes and a jump to specific word pattern. The GUI is given below which allows a user to enter a text in the form of a text, provided the words per document are specified along with choose destination folder option. For a better look font selection, drop down menu is also provided.

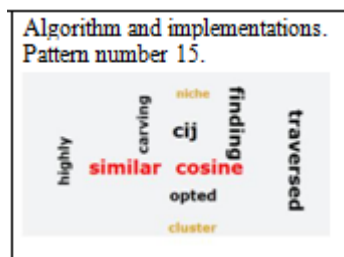
TABLE 2 TAG CLOUDS



IV. CONCLUSION

We concluded that finding a correlation improves the representation of a tag cloud. Bio Cloud would emerge as a feather in the cap of text mining and word cloud generator. Along with bioinformatics the same approach can generate word cloud for text books, novels or any free text and a reader would get the best related term without even opening his book. These patterns would then be directly used for reading, teaching, notes making and revision purposes. Finally we have given some tag clouds in Table 3; these are generated by the text of this document itself. These are self explanatory for correctness of the implemented approach of PNC.

TABLE 3 BIO CLOUD'S WORD CLOUD



REFERENCES

[1] S. M. Metev and V. P. Veiko, Laser Assisted Microtechnology, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[2] J. Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[3] S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," IEEE Electron Device Lett., vol. 20, pp. 569-571, Nov. 1999.

[4] M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gisin, "High resolution fiber distributed measurements with coherent OFDR," in Proc. ECOC'00, 2000, paper 11.3.4, p. 109.

[5] R. E. Sorace, V. S. Reinhardt, and S. A. Vaughn, "High-speed digital-to-RF converter," U.S. Patent 5 668 842, Sept. 16, 1997.

[6] (2002) The IEEE website. [Online]. Available: <http://www.ieee.org/>

[7] M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Available: <http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/>

[8] FLEXChip Signal Processor (MC68175/D), Motorola, 1996.

[9] "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.

[10] A. Karnik, "Performance of TCP congestion control with rate feedback: TCP/ABR and rate adaptive TCP/IP," M. Eng. thesis, Indian Institute of Science, Bangalore, India, Jan. 1999.

[11] J. Padhye, V. Firoiu, and D. Towsley, "A stochastic model of TCP Reno congestion avoidance and control," Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.

[12] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.11, 1997.

BIOGRAPHY



Ayush Pareek Ayush is currently pursuing an Undergraduate degree in Computer Science from The LNM Institute of Information Technology (India). He is interested in Computer Graphics, Natural Language Processing, Machine Learning and Computer Security. He has mainly worked in Summarization, Text-Classification, Lossless Compression and Deep Learning.